Insertionally polymorphic human endogenous retroviruses and their potential role in cancer.

Author: Michal Izydorczyk

Director of studies: Prof Susan Brooks

Second supervisor: Dr Ravinder K Kanda

Thesis submitted in partial fulfilment of the requirements of the award of Doctor of Philosophy.

Degree awarded by Oxford Brookes University

Original submission: 01/2022; Corrected: 08/2022

Abstract

In this study a family of endogenous retroviruses, HERV-K(HML-2) was investigated. The family is the youngest endogenous retrovirus family known in humans, with some of its members being insertionally polymorphic in modern human populations. There have been numerous reports of HERV-K(HML-2) members being present in cancer samples. Here, a robust analysis of the state of HERV-K(HML-2) family in the human genome known to date is presented, along with the most complete database of known insertions, present in modern human genomes. In addition to literature data, 10 novel solo-LTR elements and 2 novel full-length truncated elements, found in recent assemblies of the Human reference genome are reported in this study. Furthermore, a bioinformatics pipeline used to detect HERV-K(HML-2) loci in genomic sequence data is presented. The pipeline is used to analyze a large dataset of 11 different cancer sequencing projects coming from The Cancer Genome Atlas, resulting in detection of 28 polymorphic insertions. 7 of these are thought to be completely novel, whereas 10 display significance in occurrence rate when comparing the cancer data to general population. A number of biochemical pathways suspected to be altered by the detected insertions were found. Nevertheless, the low frequency of occurrence for novel HERV-K(HML2) insertions detected in this study, combined with absence of novel insertions detected in cancer tissue only suggest, that the hypothesis of HERV-K(HML-2) elements being still active in modern human populations is unlikely. However, significant presence of unfixed insertions in cancer samples combined with a list of genes known to be altered in cancer and influenced by these loci suggest, that presence of these particular loci could influence cancer development and progression.

Acknowledgments

First and foremost, I am very grateful to my supervisors, Dr Ravinder Kanda and Prof. Susan Brooks for continuous support during the course of the PhD, assisting my research, their invaluable guidance on scientific writing and all the other help they provided during the past 5 years. I would also like to thank all the friends and colleagues from Oxford Brookes Biological and Medical Sciences, who have always been happy to give me guidance and created a very friendly, social atmosphere in the department. Furthermore, I am grateful for meeting a multitude of wonderful researchers from all over the world during the course of my study and having the opportunity to discuss research, especially at international conferences. I would also like to acknowledge the Oxford Brookes University and the Nigel Groome Studentship for funding my research project. Finally, I would like to thank my family, especially my parents and my sister, for supporting me from abroad and constantly encouraging to continue my scientific journey.

Abbreviation list.

1KGP – 1000 genomes project.

Ags - Self-antigens.

AIDS - Acquired immunodeficiency syndrome.

Akt / PKB – Protein kinase B.

ALL - Acute lymphoblastic leukaemia.

ALS - Amyotrophic lateral sclerosis.

AML - Acute myeloid leukaemia.

arc - Activity Regulated Cytoskeleton Associated Protein.

ARC – Advanced Research Computing.

AZC - 5-azacytidine.

BLAST – Basic local alignment search tool.

BLAT – Blast-like alignment tool.

CD48 – Cluster of Differentiation 48.

CDK6 – Cyclin-dependent kinase 6.

cDNA – Complementary DNA.

CIDP - Chronic inflammatory demyelinating polyradiculoneuropathy.

CLL - Chronic lymphocytic leukaemia.

CML - Chronic myeloid leukaemia.

c-MYC – Cellular Avian virus, **MY**elo**C**ytomatosis.

CrERVγ - Mule deer / cervid endogenous gammaretrovirus.

CRISPR/Cas9 - Clustered regularly interspaced short palindrome repeats / CRISPR associated protein 9.

CSF1R - Colony stimulating factor 1 receptor.

CTCF - CCTC binding factor.

De – Denisovan.

DM - Diabetes mellitus.

DNA - Deoxyribonucleic acid.

DNMTs - DNA methyltransferases.

DOI – Digital object identifier.

EGR1 - Early growth response protein 1.

eIF3e - Eukaryotic translation initiation factor 3 subunit E.

EMT - Epithelial–mesenchymal transition.

enJSRV – Endogenous Jaagsiekte sheep retrovirus.

env – Envelope glycoprotein.

ERG – ETS-related gene.

ERK - Extracellular signal-regulated kinase.

ERV – Endogenous retrovirus.

ERV-DC – Endogenous retrovirus – domestic cat.

ESRG – Embryonic stem cell related gene.

ETS transcription factor – E twenty-six transcription factor.

ETV1 - ETS Variant Transcription Factor 1.

FACS – Fluorescent-activated cell sorting.

fgf - Fibroblast growth factor.

FosB - FBJ murine osteosarcoma viral oncogene homolog B.

Fv1 - Murine Friend virus susceptibility – 1.

GABA - Gamma-aminobutyric acid.

gag – Group specific antigen.

GaLV – Gibbon ape leukaemia virus.

GBM - Glioblastoma multiforme.

GCT - Germ cell tumors.

GST - Glutatione-S-transferase.

HAART - Highly Active Antiretroviral Therapy.

hCG – Human chorionic gonadotropin.

HCV - Hepatitis C virus.

HEK293 - Human embryonic kidney 293.

HERV - Human endogenous retrovirus.

HGDP – Human genome diversity project.

HIV – Human immunodeficiency virus.

HML-2 – Human mouse-mammary tumor virus - like 2.

HRES - HTLV-related endogenous sequence.

HSV-1 - Herpes simplex virus 1.

HTDV - Human teratocarcinoma-derived virus.

HTLV - Human T-lymphotropic virus.

ID – Identifier.

IL-1/6 – Interleukin 1/6.

int5/aromatase – Integration site 5 / aromatase.

IV – Intravenous.

JSRV - Jaagsiekte sheep retrovirus.

KoRV - Koala retrovirus.

KRAB-ZFP - Tetrapod-specific KRAB zinc finger proteins.

LAPC - Lung alveolar proliferating cells.

LDLR – Low density lipoprotein receptor.

LGLL - Large Granular Lymphocytic Leukaemia.

LINE – Long interspersed nuclear element.

LNM – Lymph node metastasis.

LTR – Long terminal repeat.

MAPK - Mitogen-activated protein kinase.

MBRV - Melomys burtoni retrovirus.

MERV - Murine endogenous retrovirus.

MITF-M - Microphthalmia-associated transcription factor – melanocyte – specific.

MMTV – Mouse – mammary tumor virus.

MND - Motor neuron disease.

mRNA – Matrix ribonucleic acid.

MS – Multiple sclerosis.

MSRV – Multiple sclerosis associated virus.

MuLV/MLV – Murine leukaemia virus.

MuRRS/LTR-IS - Murine retrovirus related sequences/LTR-like elements.

MY/MYa – Million years / Million years ago.

NCBI – National Center of Biotechnology Information.

NCBI – National Centre of Biotechnology Information.

Ne – Neanderthal.

NF-kB - Nuclear Factor kappa-light-chain-enhancer of activated B cells.

Np9 – Nuclear protein 9.

OA – Osteoarthritis.

Oct-1 – Octamer-binding transcription factor 1.

OPA - Ovine pulmonary adenocarcinoma.

ORF - Open reading frame.

OWM – Old world monkeys.

PAMP - Pathogen associated molecular pattern.

PBMCs - Peripheral blood mononuclear cells.

PBS - Primer binding site.

PCR – Polymerase chain reaction.

PERV – Porcine endogenous retrovirus.

PGR – Progesterone receptor.

PHA – Phytohemagglutinin.

PLZF - Promyelocytic leukaemia zinc-finger protein.

p-mTOR – Phospho-mammalian target of rapamycin.

pol – Polymerase / reverse transcriptase.

pRb – Retinoblastoma protein.

pro – Proteinase.

PSA – Prostate specific antigen.

PWIDs - Persons who inject drugs.

RA - Rheumatoid arthritis.

RACE - Rapid amplification of cDNA ends.

RASGRF2 - Ras protein-specific guanine nucleotide-releasing factor 2.

RcRE- Rec Response Element.

Rec – Regulatory of expression encoded by corf.

REST - RE1 silencing transcription factor.

Rev - Regulator of expression of virion proteins.

RNA - Ribonucleic acid.

RNAi – RNA interference.

RNase H – Ribonuclease H.

Rspo – R – spondin.

RT-PCR – Reverse transcriptase – polymerase chain reaction.

SAM – Sequence Alignment/Map.

sCJD - Sporadic Creutzfeldt–Jakob disease.

shRNA – Short-hairpin RNA.

SINE - Short interspersed nuclear element.

SNP – single nucleotide polymorphism.

SOD1 – Superoxide dismutase 1.

Sp1/Sp3 – Specificity protein 1 / specific protein 3.

SRA – Sequence Read Archive.

SSAV - Simian sarcoma-associated virus.

SUMO – Small ubiquitin-like modifier.

SVA – Sine-VNTR-Alu.

TCGA - The Cancer Genome Atlas.

TDP-43 - TAR DNA binding protein 43.

THE1B - Transposon-like human element 1B.

TMPRSS2 - Transmembrane serine protease 2.

TNF-α - Tumor necrosis factor alpha.

TRIM28 - Tripartite motif-containing 28.

TRIM5α - Tripartite motif - containing protein 5, α isoform.

SAgs – Superantigens.

XRV – Exogenous retrovirus.

TSD – Terminal or target site duplications.

TZFP - Testicular zinc finger protein.

UCSC - University of California, Santa Cruz.

UV – Ultraviolet.

VNTR - Variable number tandem repeat.

WMV - Woolly monkey virus.

wnt – Wingless – type.

Contents

1. Chapter 1: Introduction.

1.1. What are ERVs?

Retroviruses are RNA viruses. The retroviral life cycle includes infecting the host cell by the viral particle (Figure 1, step 1), inserting RNA into cell's cytoplasm (Figure 1, step 2), retrotranscription of RNA into DNA (Figure 1, step 3) and integration of the DNA into the host genome (Figure 1, step 4). At that stage, the inserted material is referred to as a provirus (Nisole and Saïb, 2004). After successful integration into the host genome, the host cell transcribes viral DNA, producing viral proteins (Figure 1, step 5), and constructing mature viral molecules (Figure 1, steps 6-7). Retroviral infection of germline cells results in these genomic integrations being inherited by offspring; at this point they are called endogenous retroviruses (ERVs) (Subramanian *et al.*, 2011).

Ubiquitous to all vertebrate genomes, there are 31 recognisable families of ERVs in primates (Tristem, 2000; Belshaw *et al.*, 2005b). ERVs are classified as class I transposable elements, due to their genetic similarity to other class I transposable elements (Lander *et al.*, 2001). Members of this class, called retrotransposons, copy and paste themselves within the genome via retrotranscription – a process, that initially produces viral RNA from the DNA sequence and subsequently converts viral RNA into DNA, which is then inserted into the host genome at a new position (Finnegan, 1989).

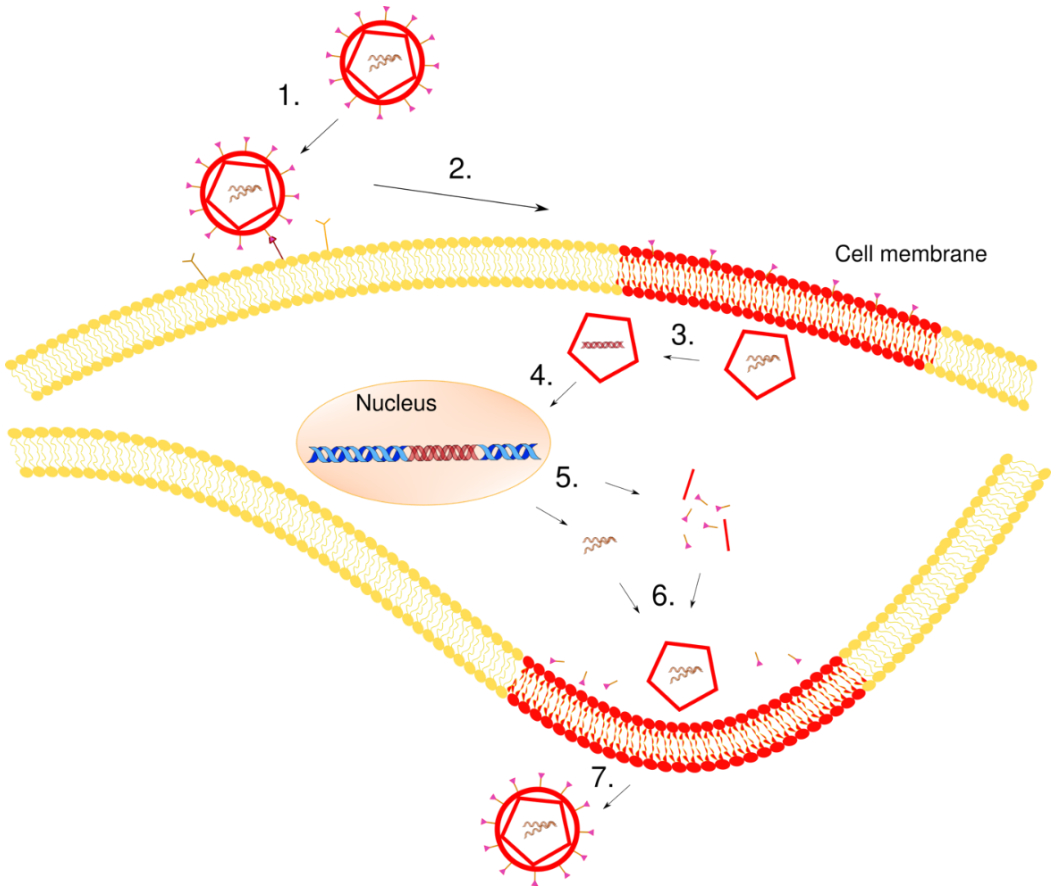

Figure 1 Schematic representation of the retroviral life cycle. For detailed explanation, please refer to the text.

Many of the ERV insertions found within the genomes of jawed vertebrates, such as humans (Lander *et al.*, 2001), pigs (Huh *et al.*, 2007), koalas (Tarlinton, Meers and Young, 2006), sheep (York *et al.*, 1991), deer (Elleder *et al.*, 2012) or mouse (Dudley and Risser, 1984) are thought to be ancient insertions. These are remnants of past infection by exogenous retroviruses that no longer exist, that became fixed in the genomes of their hosts by stochastic processes such as genetic drift (Ishida *et al.*, 2015). There is also some evidence for horizontal transmission of certain infectious retroviruses between different mammalian species, particularly rodents and true carnivore species, such as cats, that host ERV-DC proviral insertions, closely related to mouse and rat endogenous retroviruses (Anai *et al.*, 2012). Over the course of evolutionary time, vertebrates obtained hundreds of thousands of ERV loci, which proliferated through virtually all of the known vertebrate species' genomes (Coffin, Hughes and Varmus, 1997). The majority of these elements have lost their ability to replicate and infect other cells due to the accumulation of mutations, partial deletions, or recombinational deletions. Particularly, a recombination event between the two similar long terminal repeat sequences (LTRs) that flank the viral genes can cause deletion of the internal proviral components, resulting in a solo LTR structure; solo LTRs are found to be 10-100 times more common than their full length counterparts (Tristem, 2000; Belshaw *et al.*, 2005b). However, the LTRs still contain promoters and regulatory regions that can affect the transcription of nearby genes. As the sequences of ERVs are highly similar within each family, it has also been suggested that this could drive non-allelic homologous recombination between different proviral loci within the hosts genome (Campbell *et al.*, 2014). Such recombinations could cause duplications, deletions or rearrangements of larger portions of the genome, as well as disrupt expression of existing genes by facilitating recombination between regions in immediate vicinity of the insertions (Eichler, 2001).

1.2. ERV structure.

Figure 2 Schematic representation of a full-length ERV structure. The full-length virus typically spans 7-12kb and consists of 4 genes: *gag*, *pro*, *pol* and *env*, flanked by two long terminal repeats (LTRs). The entire provirus is flanked by two Terminal site duplications (TSDs), typically 6-8bp sequences, identical at the time of integration (Medstrand and Blomberg, 1993).

As illustrated in Figure 2, a full-length ERV typically spans 7-12kb. There are also instances of truncated proviruses within the human genome (Shin *et al.*, 2013). The genetic structure of a typical full-length virus usually consists of 4 genes*: gag, pro, pol* and *env* flanked by LTRs (Medstrand and Blomberg, 1993). The *gag* gene produces a group-specific antigen. The primary role of this

polyprotein is to form the main part of the viral capsid, and it is also involved in viral particle assembly, virion release and coupling with cell membrane during infection (Kraus *et al.*, 2011). The *pro* gene encodes proteinase that cleaves the Gag polyprotein. The *pol* gene encodes polymerase / reverse transcriptase, which performs reverse transcription from RNA to DNA during the integration process. It also encodes RNase H and integrase that facilitate the event of integration into the host DNA (Löwer *et al.*, 1995). The *env* gene encodes the envelope (Env) glycoprotein, which is secreted into the outer layer of the viral capsid and facilitates binding and entry into host cell (Turner *et al.*, 2001). The internal genes are flanked by two long terminal repeats (LTRs), which harbour regulatory sequences, that attach promoters to control expression of flanking genes - approximately 20% of transcription factor binding sites in humans and mice are thought to be LTR-derived (Sundaram *et al.*, 2014). The two flanking LTR sequences are identical at the time of virus integration and can accumulate mutations post-integration. The differences between the LTRs can be used to approximate insertion ages within the genomes, assuming the standard mutation rate of LTRs as 0.24-0.45% per million years (Subramanian *et al.*, 2011). Due to high sequence identity, the LTR sequences flanking a single provirus may also recombine and produce a solo LTR structure, which deletes the internal region; solo LTR structures are 10 – 100 times more common than the full-length proviral counterparts (Belshaw *et al.*, 2004). During the process of ERV integration into the host genome, an identical region of 4-8bp is created on both the 5' and 3' ends of the provirus. The integrase enzyme creates a staggered cut at the insertion target site producing a sticky end, which is filled by the polymerase enzyme, creating this repeated sequence at both ends of the virus (Figure 3). These sequences are called terminal or target site duplications (TSDs) and are unique to each insertion (as insertion of ERVs is random) (Mizuuchi, 1984). Two different insertions with identical TSDs may be a result of a duplication event, or if the flanking sequences don't match the provirus may have undergone recombination with another ERV locus elsewhere in the genome (Kahyo *et al.*, 2017).

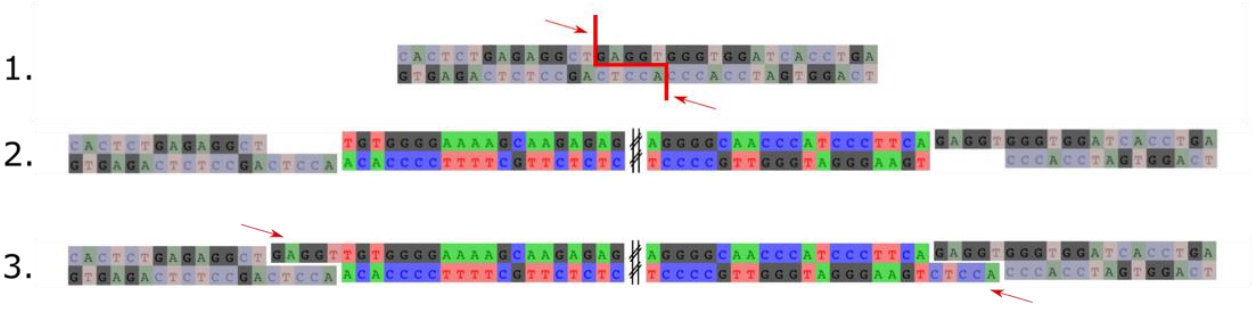

Figure 3 Terminal-site duplication (TSD) formation process. The insertion site in the reference genome is cut by the integrase enzyme (step 1, arrows indicate the excision point), which creates a short chain (usually 4-6bp) of overhanging bases. The element is integrated into the genome (Step 2, the element is in colour) and DNA repair mechanism replaces missing bases at the 5' and the 3' ends of the element, creating a duplicated sequence, known as TSD (step 3).

1.3. ERV classification.

Exogenous and endogenous retroviruses classification have arisen independently, which makes them difficult to categorize. There are examples of ERVs, like the mouse mammary tumor virus (MMTV) or the Jaagsiekte sheep retrovirus (JSRV), that occur both in exogenous and endogenous form (Cohen and Varmus, 1979; York *et al.*, 1991), but most ERVs are the remnants of past retroviral infections (Boeke and Stoye, 1997). Such viruses no longer have direct exogenous counterparts, which makes classification challenging. The International Committee on the Taxonomy of Viruses (ICTV) classifies retroviruses into 7 retroviral families (example in brackets), based on the phylogenetic relationships of genome sequences (Coffin *et al.*, 2021):

- Gammaretrovirus (Murine leukaemia virus).
- Epsilonretrovirus (Walleye dermal sarcoma virus).
- Alpharetrovirus (Avian leukosis virus).
- Betaretrovirus (Mouse mammary tumor virus).
- Deltaretrovirus (Bovine leukaemia virus).
- Lentivirus (Human immunodeficiency virus 1).
- Spumavirus (Chimpanzee foamy virus).

ERV classification is largely based on phylogenetic relationships and sequence similarity with known exogenous retroviruses and loosely divided into three classes:

- Class I - gammaretrovirus-like proviruses.
- Class II - betaretrovirus-like proviruses.
- Class III - spumaretrovirus-like proviruses.

However, since that classification, families of lentiviral and spumaretroviral proviruses were characterized, which do not fit into the previously proposed types (Katzourakis *et al.*, 2007). Individual ERV lineages are classified as families, which together with three main ERV classes contrasts with specific taxonomic terms of ‘family’ and ‘class’ and should not be used interchangeably. Unfortunately, further naming of individual families and its members is not standardized and very often problematic, due to different research groups assigning various names and subclasses to specific ERV loci, very often confusing the same proviral loci. An example would be human endogenous retrovirus – K (HERV-K), originally called by Callahan *et al.* (1982) human mouse mammary tumor virus-like 2 (HML-2), due to it being recognized in a human DNA sample by southern blotting using mouse mammary tumor virus probe. Later, different authors started to

refer to that group as HERV-K10, HTDV/HERV-K, HERV-K(HML-2), HERV-K or HERVK (Blomberg *et al.*, 2009). To this date, there is no standardized system of classifying and naming ERV loci. Three proposed recent examples are discussed below, although the naming used in the literature is still highly variable and confusing.

A transposable element classification system was recently proposed by Wicker *et al.* (2007), where all transposable elements are divided into two classes: retrotransposons (class I) and DNA transposons (class II). These further subdivide into subclass 1 that transposes via cut-and-paste of two DNA strands and subclass 2, replicated without double-stranded cleavages. ERVs are a superfamily within the LTR subclass of class I transposable elements. It has been proposed, that in order to belong to a superfamily, an element must follow a "80-80-80" rule: segments of at least 80bp must be used in comparisons, there must be at least 80% of identical nucleotides within an alignment and the stretches of identity must encompass at least 80% of aligned sequences.

The most widely used ERV classification system is based on RepBase annotation (Kapitonov and Jurka, 2008). RepBase is a database of eukaryotic transposable elements, used in conjunction with Repeatmasker tool to detect and annotate repetitive genomic sequences, including ERVs. The convention used in RepBase is available through major tools, like the University of California, Santa Cruz (UCSC) genome browser and Ensembl database. It classifies ERVs as Type 2 transposable elements (retrotransposons), further specifying them to be LTR retrotransposons. ERVs are subdivided into ERV1, ERV2 and ERV3 superfamilies, based on the length of their TSDs: 4bp, 6bp or 5bp, respectively. ERV1 primarily overlaps with gammaretroviruses, ERV2 with Betaretroviruses and ERV3 is similar to spumaretroviruses. Repbase LTR classification is based on the degree of identity: less than 90% of identical bases classifies a subfamily, less than 75% a new family.

Gifford *et al.* (2018) has discussed the topic again recently and proposed a schematic of ERV classification based on the phylogenetic studies between the insertions. They largely follow the classification into III separate classes, but propose novel nomenclature, standardizing unique IDs for all ERVs among different species by naming them as: "Category-Taxonomic group. Numeric ID-Species ID", e.g. ERV-K(HML.2).113-Hsa. The first segment defines the element as endogenous retrovirus, the second identifies the unique ERV subgroup and ID within that subgroup and the final gives a species identifier.

Nevertheless, continuous improvements in sequencing technologies followed by constant accumulation and analysis of sequences of different vertebrates introduces a great variety in classifications of ERVs. Detailed categorization of ERV families among organisms evolves with growing body of data. To date, there is no single ubiquitous system of ERV classification and accurate study of these proviral sequences depends on careful analysis of existing literature and organization of data according to criteria specific to particular group of ERVs.

1.4. ERV adaptation.

In vertebrate genomes, most ERVs are ancient insertions that are fixed in the genomes of their hosts. They are vertically transmitted from parent to offspring, suggesting that the majority of these loci are neutral, with little detrimental impact on the host genome (Boeke and Stoye, 1997). However, in some instances ERV insertions have been co-opted by the host genome and repurposed to serve another function in a variety of tissue (Seifarth *et al.*, 2005).

1.4.i. Co-option of an *env* gene for placental formation.

Syncytin is an essential gene required for placenta formation in all placental mammals (Mi *et al.*, 2000). The acquisition of this gene was not a single evolutionary event. Env proteins from various ERVs appear to have been adopted during the course of evolution as independent events and can be observed in rodents (*Syncytin-A*, *-B*, *-Mar1*), lagomorphs (*Syncytin-Ory1*), apes (*Syncytin-1,-2*), ruminants (*Syncytin-Rum1, Fematrin-1*) and carnivores (*Syncytin-Car1*) (Lavialle *et al.*, 2013). *Syncytin-1* and *Syncytin-2* proteins have been encoded by *env* sequence of Human Endogenous Retrovirus - W (HERV-W) and Human Endogenous Retrovirus - FRD (HERV-FRD) respectively (Grandi *et al.*, 2016). These sequences existed for at least 30 and 45 million years in the primate lineage, with *env* gene being highly conserved among species (Turner *et al.*, 2001).

1.4.ii. Co-option of a *gag* gene in the brain.

The *arc* gene is important in neuronal tissue and it has been found to originate from transposable elements (Ashley *et al.*, 2018). Long lasting information storage in neurons, as well as synaptic plasticity was found to be tightly associated with Arc expression in neuronal tissue – the *arc* gene contains a fragment of *gag* gene believed to be adopted from the *Ty3/gypsy* retrotransposon family (Zhang *et al.*, 2015). It has been recently shown, that Arc can self-assemble into virus-like capsids while expressed in bacteria culture and encapsulate mRNA, forming extracellular vesicles (Lizarraga-Valderrama and Sheridan, 2021). Such vesicles can interact with neurons in a way resembling an exogenous retrovirus infecting target cells, inducing self-expression in affected neurons and spreading further into the tissue (Pastuzyn *et al.*, 2018).

1.4.iii. ERVs in immunity.

ERVs can also be adopted by many organisms to sustain resistance against exogenous retroviral infections, and there are numerous examples of such adoptions in the literature (Aswad and Katzourakis, 2012; Kanda, Tristem and Coulson, 2013). A few specific examples are given below.

Murine Friend virus susceptibility-1 (Fv1) genetic sequence is approximately 40% identical to murine endogenous retrovirus - L (MERV-L) *gag* gene. The product is functionally similar to capsid-binding restriction factor TRIM5α (Yap *et al.*, 2004), which upon retroviral infection, recognizes motifs in retrovirus capsid proteins. Moreover, it inhibits the uncoating of retroviral nucleic acid,

preventing reverse transcription and transport of the viral genome to the cell's nucleus. However, the precise mechanism of Fv1 action is still unknown, especially the method of selecting and binding retroviral capsids.

Endogenous mouse mammary tumor virus superantigens (MMTV SAgs) are transmembrane glycoproteins, used by exogenous MMTV to promote proliferation of lymphocytes in the host. They increase infection rate by transferring virus from the gut to mammary glands (Golovkina *et al.*, 1992). Expression of the same *sag* gene in conjunction with corresponding major histocompatibility complex molecule endogenously results in apoptosis of T-cells susceptible to MMTV infection in mice. This leads to immunity against exogenous MMTV infection by blocking its transportation to the mammary gland upon contact with infected milk (Perzova *et al.*, 2017) .

Similarly, presence of various ERV *env* genes can provide superinfection resistance, provided the exogenous virus uses the same receptor to enter the target cells. Endogenously expressed Env protein binds the receptors on cells surface, blocking further attachment of exogenous virions. Multiple species including chickens, sheep and mice use a number of different ERVs to provide that protection (Weiss, 2013).

1.5. ERVs in different organisms.

Different vertebrate species have acquired endogenous retroviruses during their evolutionary history. As discussed previously, many of these ERV insertions will be ancient integrations, with the exogenous form of the virus no longer present. However, there are a number of examples (in a number of different species) where we do have the exogenous and the endogenous form of the virus, present today. These ERVs are thought to be active and allow the possibility of researching real-time genome invasion by endogenous retroviruses, as well as interaction with their exogenous counterparts.

1.5.i. Mouse mammary tumor virus (MMTV).

There are 21 distinct LTR endogenous retroviral families detected in the genomes of laboratory mice in total. Nine of these are considered still active within these rodents and been present in their evolutionary lineage for at least 70 million years (McCarthy and McDonald, 2004). Detection of mammary cancer-related virus in mice dates back to 1936, where Bittner discovered a cancer inducing factor. That factor was spreading from mother to offspring in laboratory mice, and was later found to be transmitted via milk (Bittner, 1936). MMTV has been demonstrated to cause mammary cancer (Stewart, Pattengale and Leder, 1984) and, less often, T-cell lymphomas in mice (Dudley and Risser, 1984). Lately, MMTV - like gene sequences were found in various organisms (Szabo *et al.*, 2005), including humans and other primates and there is strong evidence towards its causal role in human breast cancer (Mazzanti *et al.*, 2015; Lawson and Glenn, 2017; Lessi *et al.*,

2020). MMTV is a class II ERV, resembling the betaretrovirus XRV genus (Bittner, 1936). The significance of this virus is characterized by close evolutionary relationship with the human ERV, HERV-K(HML-2) subfamily. HML-2 stands for human mouse mammary tumor virus-like 2, due to it being recognized in a human DNA sample by Southern blotting using mouse mammary tumor virus probe (Callahan *et al.*, 1982). Callahan and Smith (2008) used inverse PCR to analyse MMTV integrations in MMTV-infected mice that developed mammary tumors. They confirmed 5 cases of MMTV integration into the intron of a known oncogene, *eIF3e,* in the opposite transcriptional orientation of the gene. That created a cryptic transcription termination site and caused truncation of the mRNA product, making it oncogenic. Similarly, in 9 cases MMTV inserted within the location of another known oncogene, *Notch4* (Sugaya *et al.*, 1994), in the same orientation as the gene. The result of that mutation was transcription of a *Notch4* mRNA, which started at the 3' MMTV LTR and changed the expression of the transmembrane domain of *Notch4*, resulting in a constitutively activated form of Notch4. These examples clearly indicate that integration sites of MMTV can directly influence cellular signalling pathways involved in cancer development via insertional mutagenesis. It can result in both premature transcription termination (resulting in a truncated product or mutation), or an extension of the transcript sequence (resulting in a constitutively activated receptor) (Callahan and Smith, 2008). In the case of constitutive Notch activation, the result is enhanced cell proliferation and other vital cellular processes. MMTV can also insert within the promoter regions of numerous known oncogenes, like *wnt1/wnt10b* (Nusse *et al.*, 1984), *Fgf3* (Dickson *et al.*, 1984), *Fgf10* (Theodorou *et al.*, 2004), *Fgf8* (MacArthur, Shankar and Shackleford, 1995), *int-5/aromatase* (Durgam and Tekmal, 1994), *Notch1* (Diévart, Beaulieu and Jolicoeur, 1999), *Wnt1/Wnt3* (Schroeder, Troyer and Lee, 2000), *eIF3e-p48* (Marchetti *et al.*, 1995), *Rspo2* (Lowther *et al.*, 2005) and *Rspo3* (Gattelli *et al.*, 2006). The clear involvement of MMTV in transcription rate and functionality of known oncogenes reported in the literature illustrates its importance in development of mammary tumors, and leukaemia in mice.

1.5.ii. Murine leukaemia virus (MuLV).

Murine leukaemia virus is a positive sense, single-stranded gammaretrovirus, that induces leukaemia in mice (Moloney, 1960). The infection mechanism is very similar to other retroviruses and MuLV has been used as a model of viral infection in humans (Largaespada, 2000). The mouse genome also harbours multiple endogenous copies of MuLV. Some endogenous copies of MuLV are replication competent and capable of inducing neoplasia, but most of them are defective (Chattopadhyay *et al.*, 1980). Recombination between exogenous MuLV with its proviral counterpart is thought to contribute towards the development of malignant tumors with short latency periods (Stoye and Coffin, 1987). Mice species contain a number of highly similar proviruses, distinct for every species, which suggests that endogenization and amplification occurs in bursts

(Boeke and Stoye, 1997). In fact, it has been demonstrated, that MuLV infected cell lines produce viral particles, clearly visible in electron microscopy. These particles contain all genetic information of the virus and their infectivity remains high *in vitro* (Houzet *et al.*, 2006). An example of MuLV oncogenesis is the activity of Moloney murine leukaemia virus, which leads to T lymphomas in virtually 100% of infected animals within 3 months. This is a well characterized example of ERV-induced oncogenesis; initially, the virus causes spleen enlargement, which leads to increased production of immature B and T lymphocytes, which in turn causes 4-10-fold increase in myeloid and erythroid stem cells. These hyperplastic cells migrate to thymus and infect it with MuLV, that promotes expression of various protooncogenes in thymocytes due to its LTR activity, resulting in thymoma formation (Davis *et al.*, 1987).

1.5.iii. Gibbon ape leukaemia virus (GaLV).

The Gibbon ape leukaemia virus is a gammaretrovirus is an oncogenic virus first discovered in gibbons from Bangkok in late 1960s (Theilen *et al.*, 1971). The sequence of this virus is most closely related to Melomys burtoni retrovirus (MBRV) isolated from a rodent in Papua New Guinea (Alfano *et al.*, 2016). GaLV has been associated with malignant lymphoma in a white-handed gibbon (Johnsen *et al.*, 1971). There are seven GaLV families known so far: GALV-SEATO (Kawakami and Buckley, 1974), GALV-SF (Kawakami, Kollias and Holmberg, 1980), GALV-H (Krakower *et al.*, 1978), GALV-BR (Todaro *et al.*, 1975), GALV-mar (genbank sequence U20589.1), GALV-X (Burtonboy *et al.*, 1993) and a Simian sarcoma-associated virus / woolly monkey virus (SSAV/WMV) (Theilen *et al.*, 1971). GALV-mar and GALV-X are *in vitro* observed, whereas others have been found in animals suffering from sarcoma neoplasms. SSAV was found in a new world Woolly monkey and it is a defective virus, which has most likely been transferred from a GaLV-positive gibbon (Theilen *et al.*, 1971). GaLV is thought to originate from Murine leukaemia-like viruses (MuLV) (Lieber *et al.*, 1975); the relationship was established by serological and low-resolution DNA sequence analysis methods (sequence identity is only about 55% for *env* gene and 68-69% for *pol)*. There is also a high degree of sequence similiarity between GaLV and koala retrovirus (KoRV). Initially, phylogenetic analysis of the KoRV sequences grouped it closely with GaLV, excluding PERV and MuLV/MLV groups. This was surprising, considering both geographical and taxonomic distance between gibbons and koalas and the fact that both of these viruses are considered still active (Hanger *et al.*, 2000). Lack of similar viruses in wombats, or other marsupials closely related to koalas suggested relatively recent integration via cross-species transmission (Herniou *et al.*, 1998). Lately GaLVs were found to be most similar to MBRV. *Myleomys burtoni* occurs only in Papua New Guinea and Australia, yet the widespread of GaLV and its variants suggests, that there is another form of a related retrovirus in vertebrate hosts yet undiscovered, that shares high degree of similarity with both. The possible reservoir for such virus would be bats, which could be responsible for spreading the virus around

various Australasian species. However, the quite high degree of identity between GaLV and MuLV makes this hypothesis less likely, since two inter-species transmissions between rodents and bats and, subsequently, bats and gibbons would very likely cause a higher rate of evolution (Cui, Tachedjian and Wang, 2015).

1.5.iv. Koala endogenous retroviruses (KoRV).

Koala endogenous retrovirus (KoRV) is a relatively young retrovirus, which is currently colonizing the koala genome. Phylogenetic analysis of KoRV indicates high similarity to the Gibbon ape leukaemia virus (GALV), suggesting cross-species transmission (possibly via rodent intermediate), and indicating that KoRV invaded the koala genome within the last century (Alfano *et al.*, 2016). That makes the KoRV the least mutated, and its ongoing activity makes it a good model to study the process of viral endogenization, as we have the endogenous and exogenous forms currently existing. All KoRV insertions detected in the koala genome are full-length and replication-competent, and very closely related to Gibbon ape leukaemia virus (GaLV). Some insertions are not present in all individuals, which suggests recent activity of this particular virus within the modern Koala population, resulting in unfixed loci within the Koala genome (Quigley *et al.*, 2018). In fact, none of the KoRV loci found by Tarlinton, Meers and Young (2006) were fixed in the koala population. This suggests that the endogenizing retroviruses are active within koala populations presently, a finding further confirmed by the presence of these proviruses in Koalas germ cells. Moreover, the prevalence of KoRV in different populations varies greatly; along the east coast of Australia, in the north the prevalence of detected KoRV insertions is greater than the south, with no insertions being detected in koalas from Kangaroo island. The population study was expanded by Ávila-Arcos *et al.* (2013), who performed PCR analysis on museum koala skin specimens (collected from the late 19th century and across the 20th century). They discovered that 3 out of the 18 tested samples did not contain any KoRV sequences, whereas 5 of the remaining contained full-length KoRV inserts, and 10 contained at least one internal viral gene. All of the 15 KoRV-positive samples came from northern Australia. It indicates that the virus spread among Koala populations is an ongoing process, slowly expanding from northern to southern populations of Koalas. KoRV is associated with chlamydia infections and formation of leukaemia among koalas. Exogenous KoRV exists with up to ten envelope subtypes (KoRV-A – KoRV-J), with KoRV-B being the variant most associated with cancer development. Xu *et al.* (2013) studied 13 koalas from the Los Angeles Zoo and found an association between KoRV and lymphoma in 6 of the individuals, which carried the KoRV-B subtype. Since the original discovery of KoRV, a number of additional unfixed loci have been described (Zheng *et al.*, 2020).

1.5.v. Porcine endogenous retroviruses (PERVs).

PERVs are gammaretroviruses that can be found in genomes of all known pig strains. There is a great deal of interest in PERV activity, particularly with regards to the potential for cross-species transmission during xenotransplantation from pigs to humans. PERVs have a typical retrovirus structure (see section 1.2), differing mostly in the envelope gene sequence in respect to receptor-binding domain, dividing the PERVs into 3 classes; PERV-A, -B, and -C (Takeuchi *et al.*, 1998). PERV LTRs show a high degree of similarity with murine retrovirus related sequences/LTR-like elements (MuRRS/LTR-IS) found in mouse genome, suggesting that the source of initial infection could originate from small rodents (Huh *et al.*, 2007). The evolutionary origins of known PERVs have been recently reviewed by Chen *et al.* (2018), who used *in silico* analysis to screen 14 pig genomes available in GenBank. They identified 185 PERV sequences with at least a single LTR, including 65 full-length PERVs. Most pig breeds contained 2-10 full-length PERVs, however the majority of the sequences were mutated. TSD analysis revealed, that 14 out of 65 proviruses exhibited dissimilar 5' and 3' TSDs, which did not group together in a maximum-likelihood phylogenetic analysis, suggestive of recombination. PERV-A and PERV-B emerged in the same time period (approximately 6.6 MYa and 6.4 MYa, respectively), whereas PERV-C is a much younger group of insertions (3.4 – 4.4 MYa). PERVs have been reported to be able to infect human cell lines and express viral particles, as well as PERV *gag* and *pol* genes (Kono *et al.*, 2021); however a separate study showed that patients receiving islet cells from pigs did not suffer from PERV infections (Wynyard *et al.*, 2014). Given the contradictory results, the risks of cross-species transmission of PERVs to humans during xenotransplantation is still unclear.

1.5.vi. Jaagsiekte sheep retrovirus (JSRV).

JSRV is a betaretrovirus, related to mouse mammary tumor virus (MMTV) (York *et al.*, 1991). Its genomic organisation is typical for a betaretrovirus, with *gag*, *pro*, *pol* and *env* genes flanked by two LTRs, spanning 7.5 kb in total. It also contains a putative protein, which functionality is unknown, termed Orf-x, located in the 3' end of *pol* gene. Exogenous JSRV was found to be the cause of ovine pulmonary adenocarcinoma (OPA), a sheep lung cancer (Palmarini *et al.*, 1999). It shares some aspects with human lung adenocarcinomas, which makes it a good model for studying the disease. Importantly, the virus was demonstrated to induce adenocarcinoma *in vivo* in mice and healthy sheep by infecting and transforming lung proliferating type 2 pneumocytes / lung alveolar proliferating cells (LAPC) (Murgia *et al.*, 2011). JSRV can infect other types of cells, but it only causes tumor development in that specific type of lung tissue. Other pathogenic agents, like lungworms, Maedi-Visna virus or bacteria can make the sheep vulnerable to JSRV infection and cancer development via sustained lung inflammation, which induces LAPC proliferation (Dawson *et al.*, 1990). The unique oncogenic abilities of the virus are largely dependent on the *orf-x* gene, which is

a dominant oncogene. *Orf-x* expression leads to rapid malignant transformation *in vitro*, however not many animals infected naturally develop tumors; this is most likely due to insufficient LAPC tissue in most sheep (Palmarini *et al.*, 1999).

JSRV can also be found within the sheep genome, where it is referred to as endogenous Jaagsiekte sheep retrovirus (enJSRV). The endogenous counterparts share 85%-89% sequence identity between *gag* and *env* genes with exogenous JSRV. The sheep genome contains at least 27 proviral insertions of enJSRV, many of which are unfixed in the sheep populations, suggesting a relatively recent time for integration of enJSRV. The youngest proviral insertion is enJSRV-26, inserted less than 200 years ago (Arnaud *et al.*, 2007). EnJSRV is expressed in many sheep tissues, like cervix, uterus, epithelia, and thymus (Murgia *et al.*, 2011). There has been some evidence to suggest that enJSRVs may provide some protection against exogenous JSRV (Miller, 2003; Arnaud, Murcia and Palmarini, 2007). *Env* expression can interfere with receptors used by exogenous counterparts, such as Hyal2, which is also used for entry by exogenous JSRV and sheep Enzootic nasal tumor virus (Miller, 2003). Another mechanism is JLR or "JSRV late restriction", an effect of R21W mutation present in endogenous *gag* gene of enJS56A1 provirus, which unmutated sequence is conserved in many betaretroviruses. Such mutated Gag molecules form chimeric multimers with exogenous viral Gag, that are quickly degraded in proteasome (Arnaud, Murcia and Palmarini, 2007). The mutation has been amplified in domestic sheep and enJSRV-20 also harbours such mutation, most likely due to homologous recombination, which indicates positive selection of this particular variant (Arnaud *et al.*, 2007). The relatively low number of proviral endogenous insertions in the general sheep population, with many of them being unfixed, and very closely related to the exogenous counterparts, suggests that the endogenization must have happened relatively recently. Therefore, JSRVs can be a good model to study active endogenization processes and interaction between exogenous and endogenous counterparts.

1.5.vii. Mule deer / cervid endogenous gammaretroviruses (CrERVγ).

Wild mule deer have been experiencing continuous infections of a relatively novel gammaretrovirus that appears in an endogenous form. Elleder *et al.* (2012) found seven CrERVγ insertions in the genomes of mule deer (n=10) and a single specimen of white-tailed deer. All CrERVγ-in1 integrations were diploid, suggesting that CrERVγ-in1 is fixed in the mule deer genome, however it was not found in the white-tailed deer. Estimated integration times, based on sequence differences between 5' and 3' LTRs show a range between 0.47 to 1 MYa for the CrERVγ-in1 provirus that harbours two differences between the LTRs. The insertions CrERVγ-in2-CrERVγ-in6 were found in separate subsets of individuals, whereas CrERVγ-in7 was detected at extremely low frequencies (4 out of 200 samples). Researchers were unable to estimate the age of CrERVγ-in2-CrERVγ-in7 due to lack of mutations between 5' and 3' LTR sequences - CrERVγ-in7 was sequenced and contained

identical 449bp LTRs, as well as intact internal genes, suggesting these are recent integrations. Kamath *et al.* (2014) evaluated 357 samples from 13 deer populations, detected all 7 previously found inserts and 7 novel ones. Overall, all of the tested proviruses were unfixed across different deer species. They found evidence of recombination between this different insertions in a phylogenetic analysis. Additionally, Kamath *et al.* (2014) found several low-frequency insertions (CrERV-in6, CrERV-in10 and CrERV-in11), providing further evidence for a relatively recent integration time for these insertions. Greenberg and Bourc'his (2019) analysed methylation pattern among the deer insertions and found, that the majority of CpG sites in the LTR regions of the proviruses were highly methylated (88% to 90%), in line with silencing mechanisms (genome defence) of transposable elements observed in many other species.

1.6. What are HERVs?

Endogenous retroviruses that are predominantly found in humans are referred to as **h**uman **e**ndogenous **r**etro**v**iruses (HERVs). In the human genome, there are approximately 30 different families of HERVs, which make up approximately 8 % of our genome, representing approximately 500,000 elements (Figure 4). HERVs have been active for at least last 50 million years and are present in other primate species, like gorillas and chimpanzees (Lander *et al.*, 2001). HERVs have been proliferating through the primate genomes for over 30 million years (Katzourakis, Rambaut and Pybus, 2005). Over the course of evolution, they have lost their ability to replicate due to accumulating random mutations and truncations. However, some insertions, especially the younger ones have retained coding potential for HERV genes (Turner *et al.*, 2001).

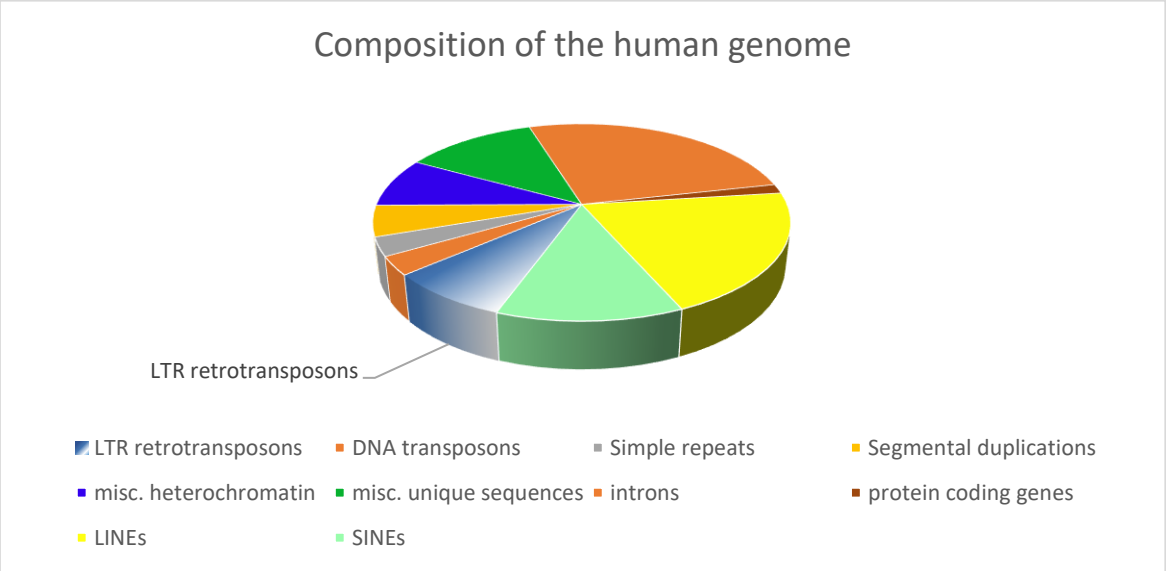

Figure 4 Composition of the human genome. Based on data from Lander *et al.*, 2001.

1.7. HERV classification.

HERVs are divided into three different classes, based on similarity between their *pol* gene sequences. Initially, Callahan *et al.* (1982) proposed two classes, class I similar to gammaretroviruses and epsilonretroviruses, including murine leukaemia virus and class II, similar to betaretroviruses and deltaretroviruses. More recently, Bénit *et al.* (1999) identified HERV-L group of endogenous retrovirus-like elements in humans, which was soon found to be related to spumaviruses by Herniou *et al.* (1998) and termed class III. The letter, following the HERV designation refers to the amino acid associating with the primer binding site (PBS) used to begin reverse transcription and designates separate families. According to Tristem *et al.* (2000), there are at least 31 recognizable families of HERVs, that can be found in the human genome (see table 1). Most of the literature reviewed relies in its research on the nomenclature proposed by Tristem *et al.* (2000) but classification of HERVs is an ongoing process as the sequencing technology advances and new insertions are being described. To date, there have been different proposals to change and reorganize the HERV nomenclature (Blomberg *et al.*, 2009), but none of them have been widely adopted in the scientific community.

Table 1 31 recognizable families of human endogenous retroviruses. The right columns indicate the amino acid (AA) associated with the Primer Binding Site (PBS). Based on Tristem *et al.* (2000).

HERV family	PBS AA	HERV family	PBS AA	HERV family	PBS AA	HERV family	PBS AA
HERV-A	Alanine	HERV-R type (b)	Arginine	HERV-H	Histidine	HERV(AC096774)	-
HERV-L	Leucine	HERV-E	Glutamic Acid	HERV-F type (b)	Phenyloalanine	HERV-R type (c)	Arginine
HERV-S	Serine	HERV-K (HML-2) type (b)	Lysine	HERV-F	Phenyloalanine	HERV(AC018462)	-
HERV-U2	-	HERV(AC069462)	-	HERV(XA)	-	HERV-K(HML-2)	Lysine
HERV-U3	-	HERV-FDR	-	HERV-F type (c)	Phenyloalanine	HERV-K(HML-5)	Lysine
HERV-T	Threonine	ERV9	-	HERV-ADP	-	HERV-K(HML-6)	Lysine
RRHERV-I	Isoleucine	HERV-W	Tryptophan	HERV-I	Isoleucine	HERV-K(HML-9)	Lysine
HERV-R	Arginine	HERV-P	Proline	HERV-L type (b)	Leucine		

29 of HERV families are present in all of the great apes, as well as humans and the integration dates are estimated 25-30 MY for most of the families, with HERV-FDR being exceptionally old, dated 53-87 MY (Tristem, 2000).

There are two families of HERVs that are very interesting from functional and disease-association point of view, namely the HERV-W and HERV-K families, due to their reported involvement in various biochemical processes in the host organism. HERV-K family members are the youngest among all HERVs and, although some insertions found in humans are found in genomes of many of our ancestors (including new and old-world monkeys), there are human specific insertions among

the HERV-K family (Medstrand and Mager, 1998). Its evolutionary origin is discussed further in section 1.10. HERV-W integrations are estimated to have entered the primate lineage 43-20 MYa (Grandi *et al.*, 2018), with few integrations fixing in orangutans (<20MYa) and gorillas (<17MYa) (Voisset *et al.*, 1999). One provirus on chromosome 12q13.3 appears to be inserted approximately 7MYa and human-specific, however the accumulation of mutations and lack of LTR sequences makes that estimation fairly uncertain (Grandi *et al.*, 2016).

1.8. HERV-W in health and disease.

HERV-W has been mostly studied in context of their adaptation as syncytins, discussed in section 1.4.i, which are derived directly from the Env protein of HERV-W origin. It has also been reported to be transcriptionally active in many healthy tissues, including brain, breast, muscle, spleen, lungs, digestive, cardiovascular and reproductive systems (Li and Karlsson, 2016). Unfortunately, these studies are transcriptome based and there have only been a few reports, that linked specific HERV-W loci and tissue expression (Li *et al.*, 2019). HERV-W has also been implicated in a number of diseases. Several types of cancer, including breast (Bjerregaard *et al.*, 2006), brain (Yi, Kim and Kim, 2004), colon (Stauffer *et al.*, 2004), kidney (Schön *et al.*, 2001), lung (Yi, Kim and Kim, 2004), liver (Kim, Ahn and Kim, 2008), prostate (Yi, Kim and Kim, 2004), ovarian (Hu *et al.*, 2006), oesophageal (Yi, Kim and Kim, 2004), skin (Yi, Kim and Kim, 2004) and T-cell (Yi, Kim and Kim, 2004) have shown overexpression of HERV-W elements. Additionally, there was a report of reduced HERV-W promoter methylation in ovarian cancer for some loci (Menendez, Benigno and McDonald, 2004). This suggests, that despite fixed nature of the HERV-W insertions in the human genome, alterations in expression control mechanisms of the proviral loci occur in cancer cases and can enhance their promoter activity. However, the results are inconclusive since there are reports for breast, colon, liver, stomach and uterus cancers (Kim, Ahn and Kim, 2008), as well as HERV-W expression in breast, colon, placenta and kidney cancers against normal tissue that show no significant differences in HERV-W transcript levels (Stauffer *et al.*, 2004).

HERV-W expression is observed in autoimmune diseases, like multiple sclerosis (MS) (Perron *et al.*, 1989). In general, these diseases are caused by the loss of tolerance for self-antigens (Ags). That results in an immune reaction of the organism against its own tissue and chronic, widespread inflammation, which destroys organs. Despite fixation of HERV-W family in the human genome, overexpression of HERV-W components induces innate and adaptive immunity, observed in patients with autoimmune disorders (Volkman and Stetson, 2014). It is speculated, that HERV RNA may be recognized as pathogen associated molecular patterns (PAMPs) and trigger auto-antibody production (Wolff *et al.*, 2017). In some types of MS patients, a putative multiple sclerosis-related retrovirus (MSRV) has been detected (Perron *et al.*, 1997), which is recently supposed to be a single HERV-W transcript, or a combination of multiple transcripts (Grandi *et al.*, 2016). The RNA of this

HERV-W/MSRV *pol* transcript was found using RT-PCR in brain (Johnston *et al.*, 2001), B cells (Perron *et al.*, 1997), as well as cerebrospinal fluid and plasma (Dolei *et al.*, 2002). Unfortunately, the transcripts have not been associated with specific loci (Christensen, 2005). *Env* transcript was also found in brains (Antony *et al.*, 2004) and peripheral blood mononuclear cells (PBMCs) (Perron *et al.*, 2012) of MS patients. Expression of HERV-W/MSRV is suspected of triggering abnormal immune response, especially by overproduction of cytokines, like interleukins 1 and 6 (IL-1, IL-6) and tumor necrosis factor α, that cause demyelination in MS patients. Pro-inflammatory cytokines were found to be activated in Vβ16 T-lymphocytes as a result of MSRV *env* expression (Rolland *et al.*, 2005).

HERV-W is also contributing to other autoimmune related disorders, like rheumatoid arthritis (RA), osteoarthritis (OA) or chronic inflammatory demyelinating polyradiculoneuropathy (CIDP). Gaudin *et al.* (2000) found HERV-W/MSRV RNA to be overexpressed in 50% of RA patients. Bendiksen *et al.* (2014) found expression of HERV-W *env* gene in OA patients to be active in 88% of tested cartilage tissue, in comparison to 38% in control tissue. Faucad *et al.* (2016) found similar overexpression of HERV-W/MSRV RNA in PBMCs of 50% of tested CIDP patients and reported HERV-W/MSRV RNA Env protein presence in 5 out of 7 tested nerve lesions.

Neurological disorders, like Motor Neuron Disease (MND), sporadic Creutzfeldt–Jakob disease (sCJD) and schizophrenia also associate with elevated HERV-W expression. *Env* transcripts can be found in MND patient biopsies and accompanies upregulation of oxidative stress-response *SOD1* gene (Oluwole *et al.*, 2007). According to Jeong *et al.* (2010), significant upregulation of HERV-W *pol* transcripts compared to healthy controls can be detected in almost all sCJD cases. Karlsson *et al.* (2001) detected HERV-W/MSRV pol mRNA in cerebrospinal fluid of 29% acute onset schizophrenia patients and 5% subjects in later stages, compared to none in corresponding controls. Huang *et al.* (2011) confirmed this, detecting HERV-W *env* in 36% of tested recent-onset schizophrenia patients. They found out, that the overexpression of HERV-W *env* causes production of dopamine receptor D3 and brain-derived neurotrophic factor in U251 human glioma cells, both factors strongly associated with schizophrenia. HERV-W LTR was reported by Hegyi (2013) to be located in regulatory region of GABA receptor B1 gene, a receptor downregulated in schizophrenia.

HERV-W is also contributing to exogenous viral infections. In AIDS patients suffering from dementia HERV-W RNA is overexpressed in brain tissue (Johnston *et al.*, 2001). HIV Tat transactivator protein can trigger expression of HERV-W *env* in astrocytes (van Horssen *et al.*, 2016). Herpes simplex virus 1 (HSV-1) can transactivate expression of HERV-W Gag and Env proteins *in vitro* (Ruprecht *et al.*, 2006). In HeLa cells, the HSV-1 IE1 protein activates HERV-W LTR, most likely by modulation of the Oct-1 transcription factor (Lee *et al.*, 2003). Epstein-barr virus Env glycoprotein can trigger expression of HERV-W in PBMCs of MS patients as well as controls or U87-MG astrocyte cells. The activation pathway possibly alters NF-kB signalling (Mameli *et al.*, 2012).

In addition to manifestation of the proviruses in various diseases, another very important fact is polymorphism occurring among these families, both sequential (HERV-W and HERV-K) and insertional (HERV-K). HERV-W Xq22.3 (ERVWE2) insertion has recently been found to be polymorphic by Garcia-Montojo *et al.* (2014). They found elevated expression of HERV-W in tested MS patients comparing to controls and a possible link to the HERV-W copy on chromosome Xq22.3, which was amplified using PCR and sequenced to find rs6622139 (T/C), rs6622140 (G/A) and rs1290413 (G/A) mutations in MS patients, out of which rs6622139 (T/C) significantly associated with sociability and severity of MS as well as increase in ERVWE2.

1.9. HERV-K insertional polymorphism.

The interesting aspect of HERV-K family related to incorporation is the observation that some of the members of the family are insertionally polymorphic among the population. This means that some people have certain viral insertions incorporated into their genome and some do not (Belshaw *et al.*, 2005). Viruses occupying insertionally polymorphic loci have infected the human population relatively recently - the most recent HERV-K family integrations are HERV-K113 and HERV-K115. Analysis of HERV-K113 sequence shows that there are no mutations between its LTRs, which should occur every 200,000-450,000 years. This can be calculated according to the fact the mutation rate of the endogenous retroviruses is 2.3×10^{-8} to 5×10^{-8} substitutions per site per year and the length of LTR is 969bp. Additionally K113 and K115 insertions possess intact open reading frames (ORFs) for all internal genes, which also suggests their younger age. The lack of accumulated recombinations and mutations is due to less time passing since infection (Turner *et al.*, 2001; Burmeister *et al.*, 2004). However, further study among 156 HIV-1+ subjects from United States revealed the presence of three single nucleotide polymorphism sites in the K113 5′ LTR and four in the K115 5′ LTR. In the study the insertion dates have been estimated between 800,000 and 1.1 million years ago (Jha *et al.*, 2009). Presence of intact HERV-K integrations in the human genome, which are seemingly inactive suggests that, unless the virus suddenly lost its ability to infect humans, there should be active, infectious insertions in the human population today, that occur in a very low rate and were not observed yet (Turner *et al.*, 2001).

Furthermore, insertionally polymorphic proviruses may appear in the human genome in 4 different states: (1) a full-length complete provirus; (2) truncated proviral sequence, where only some viral genes/gene fragments are present in the investigated locus; (3) solo LTR and (4) a preinsertion site, which consists of a single copy of the DNA sequence repeated as TSD at the known location of viral insertion (but no virus itself) (Subramanian *et al.*, 2011).

1.10. HERV-K classification.

Since the HERV-K family is the youngest of all HERV families and it's the only one displaying insertional polymorphism (discussed in section 1.9), there has been a great interest in the scientific world to study its functionality and activity in diseases. The particular interest is focused on the contribution of polymorphic insertions, which could selectively influence the development of certain diseases in individuals harbouring certain insertions, absent from the general population.

The HERV-K family associates' lysine with the PBS and divides further into 11 families: HML-1 – HML-11, based on phylogenetic relations (Callahan *et al.*, 1982). The most recently active HERV family, is the HERV-K(HML-2) group which has been infecting the primate lineage for over 30 million years (Subramanian *et al.*, 2011). The HERV-K(HML-2) group is the only endogenous retrovirus subfamily known to have members encoding all, seemingly intact proteins, both enzymatic and structural (Turner *et al.*, 2001). Some copies of this particular retroviral family (HERV-K(HML-2)) are human specific (suggesting that they inserted into the genome after the divergence of humans and chimpanzee, approximately 5-7 million years ago). Some of these human specific loci are also insertionally polymorphic, suggesting an integration time within the last 800,000 years ago (Kanda, Tristem and Coulson, 2013).

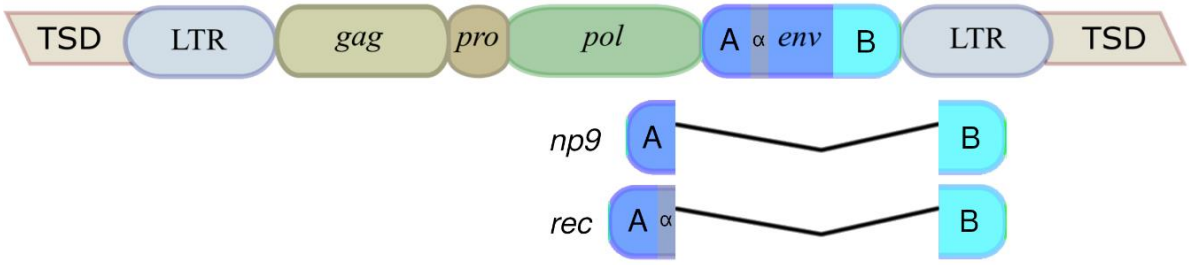

Figure 5 The alternatively spliced variants of *env* – the *np9* transcript (type 1 viruses) and the *rec* transcript (type 1 viruses). The *rec* transcript contains a 292-bp segment (α), absent in the *np9* variant, in addition to other fragments of *env* (A and B).

As mentioned previously, the "K" represents the use of the lysine tRNA residue to prime reverse transcription, whereas "HML-2" (**h**uman endogenous **m**ouse mammary tumor virus **l**ike-**2**) points out the similarity to murine betaretrovirus MMTV (Löwer, Löwer and Kurth, 1996). *Env* can undergo alternative splicing into *np9* (type 1 viruses) or *rec* (type 2 viruses) transcripts (see Figure 5), according to the presence or absence of a 292-bp deletion (Löwer *et al.*, 1995). Phylogenetic analysis of HERV-K(HML-2) LTR sequences revealed further sub-classification of the HML-2 family into LTR5Hs, LTR5A and LTR5B groups of viruses. LTR5A and LTR5B have integrated into the mammalian lineage earlier than LTR5Hs subgroup, and type 1 proviruses belong exclusively to the LTR5Hs group (Buzdin *et al.*, 2003). Furthermore, the youngest members of HERV-K(HML-2) family appear to have inserted into the human genome approximately 0.67–1.8 MYa, after the divergence of chimpanzees and humans 5-7 MYa. Therefore they are regarded as human specific (Subramanian

et al. 2011) (Figure 6). Some of these human specific insertions are insertionally polymorphic (presence/absence of polymorphism in different individuals), suggesting relatively recent activity. As of 2020, 36 such polymorphic insertions have been found, and abnormal expression of HERV-K(HML-2), including some of these polymorphic insertions, was observed in diseases, like multiple sclerosis, HIV or cancer (Wildschutte *et al.* 2016).

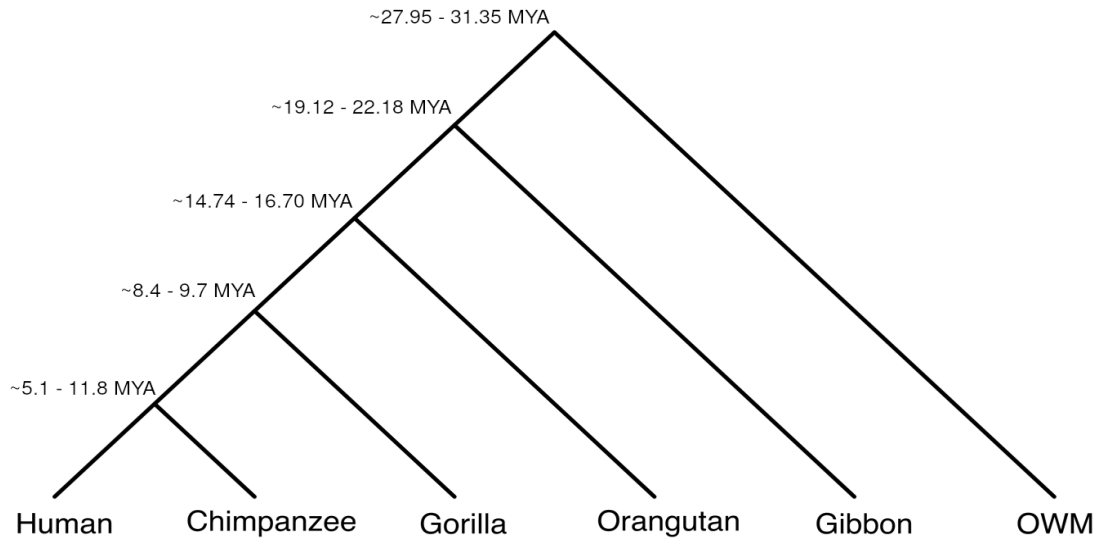

Figure 6 Phylogeny of primate evolution, dates represent approximate time of divergence. (Divergence times taken from timetree.org (Kumar *et al.*, 2017))

Furthermore, Agoni *et al.* (2012) sequenced and analysed Denisova and Neanderthal genomes for novel HERV-K(HML-2) insertions. They found 12 HERV-K insertions in Denisovan genome and 3 insertions in Neanderthal genome that have been missing from the human reference genome in the corresponding locations. Subsequently, Marchi *et al.* (2013) confirmed presence of 7 of these insertions in modern humans, by bioinformatics analysis of 21 Cancer Genome Atlas patients genomes, as well as 46 cancer patients of the WGS500 whole genome. Shortly after, Lee *et al.* (2014) performed similar bioinformatics analysis on the most recent Neanderthal and Denisovan genome assemblies at the time (Meyer *et al.*, 2012), which displayed much better coverage than source data analysed by Agoni *et al.* in 2012 (30-52 vs. 1.3-1.9 fold). They identified 6 additional HERV-K insertions, not present in modern human reference genome hg38 and 3 insertions previously reported by Marchi *et al.* (2013). 15 of insertions found in ancient hominids were identified by Wildschutte (2016) in modern human populations. Since both Denisovans and Neanderthals diverged 1.3-0.78 MYa and 0.8-0.3 MYa (Kumar *et al.*, 2017) from modern humans respectively, presence of these unfixed insertions both in extinct subspecies of archaic hominids and modern human populations at low frequency suggest, that insertion dates of these proviral loci is relatively recent and most likely have occurred not long before split from the last common ancestor. However, there are reports of evidence of interbreeding between Neanderthals and

Denisovans and also evidence of interbreeding between both these archaic groups and modern humans (Rogers, Harris and Achenbach, 2020). Therefore, it is also possible that the unfixed proviruses observed in modern populations may have proliferated in the genomes of the Denisovans and Neanderthals via these events, or have been transferred to modern humans from them.

HERV-K(HML-2) has also proliferated through genomes of other modern primates post human divergence. Gorillas have been recently reported to harbour HERV-K(HML-2) elements, that are distinct from humans. Holloway *et al.* (2019) analysed the latest gorilla reference genome assembly (gorGor5), as well as 79 high-coverage great ape genomes coming from the Great Ape Genome Project (Prado-Martinez *et al.*, 2013) using BLAST. Additionally, they prepared whole-genome DNA from three individuals and amplified HERV-K(HML-2) LTR sequences, which they then sequenced. They identified 150 gorilla-specific HERV-K(HML-2) insertions, including 31 full-length proviruses. Forty-two of these were absent from the gorilla reference assembly and 47 were identified as insertionally polymorphic. Phylogenetic estimations suggested that these insertions proliferated through the gorilla genome approximately 200,000 years ago, much later than the split between gorillas, chimpanzees and humans (Kumar *et al.*, 2017) (Figure 6). The phylogenetic analysis performed by Holloway *et al.* (2019) confirms this, showing all gorilla-specific insertions clustering in one clade, separate from the human and chimpanzee insertions. Additionally, branch lengths in the gorilla clade are much shorter than in humans and chimpanzees. A large number of proviruses harbour identical 5' and 3' LTR sequences, many of which are also identical among multiple insertions, showing a very low divergence and supporting a very recent, common evolutionary origin. Moreover, one of the detected proviruses has been found to retain intact ORFs for all retroviral genes, suggesting a possibility of replication competence.

1.11. HERV-K control mechanisms.

One of the mechanisms for controlling HERV-K expression is DNA methylation, which typically represses gene transcription. As Lavie *et al.* (2005) reviewed, most of ERVs are inactivated by early methylation during the embryogenesis. The vast majority of the methyl groups are added to cytosine residues. In mammals, this process is almost exclusively limited to CpG dinucleotides - in fact, over 75% of mammalian CpG sequences were found to be methylated (Jabbari and Bernardi, 2004). HERV-K LTRs expression has been found to be efficiently reduced by CpG-mediated silencing; raising methylation levels among CpG sites in U3 regulatory regions of LTRs correlate with reduced transcriptional activity of the affected HERV-K(HML-2) loci (Lavie *et al.*, 2005).

Two types of methylation play a key role in HERV regulation – histone methylation and methylation of CpG islands, located in LTR regions that control HERV expression. In mammals, CpG methylation is maintained by a family of DNA methyltransferases (DNMTs), which keeps HERV LTRs methylated

across most normal tissues. *De novo* DNA methylation primarily concerns cytosine residues and involves DNMT3A and DNMT3B enzymes, which work together with DNMT3L cofactor (Bestor *et al.*, 1988; Okano, Xie and Li, 1998; Aapola *et al.*, 2000). These reactions happen mostly in embryonic state, whereas DNMT1 methyltransferase maintains methylation state in adult tissue. These processes are a part of complex interactions between DNA binding proteins, particularly RE1 silencing transcription factor (REST) (Ballas *et al.*, 2005) or CCTC binding factor (CTCF) (Hughes *et al.*, 2012). Additionally, chromatin density is controlled by histone methylation and acetylation. Strict control over ERV methylation is particularly important to control ERV expression levels. Although specific pathways targeting ERVs are not defined, there is some evidence for the aforementioned DNMT3A/B/L and G9a methyltransferases being directly involved in proviral methylation (Leung *et al.*, 2011). Alteration of that process has been proven to influence expression of neighbouring genes, most likely due to high activity of promoter sequences in LTR regions of ERVs (Macfarlan *et al.*, 2012). The specific methylation reaction is mediated by tetrapod-specific KRAB zinc finger proteins (KRAB-ZFP) (Bellefroid *et al.*, 1991), along with TRIM28 cofactor, which binds to a 39-nucleotide element overlapping the primer binding site of HERV-K. This silencing method is tightly associated with LTR-retrotransposons and seems to have co-evolved with them; KRAB-ZFP genes emerge in cells abundant in LTR integrations (Thomas and Schneider, 2011). On the other hand, Turelli *et al.* (2014), found this mechanism to be completely absent in the embryonic kidney 293T cell line, suggesting that HERV-K plays an essential role in stem cells.

These control mechanisms might be reversed. It can happen by disruption of the cell cycle during cancer formation, resulting in uncontrolled expression and lowering the silencing mechanisms. It promotes uncontrolled expression by activation of HERV-derived LTRs that can take control of adjacent genes (Roulois *et al.*, 2015). The silencing is limited in embryonic and pluripotent stem cells, where HERVs become active due to epigenetic reset - most of the genome becomes demethylated and subsequently undergoes remethylation. As a result, high HERV-K(HML-2) activity in healthy stem and embryo cells upregulates transcription of nearby genes until the eight-cell stage (Grow *et al.*, 2015). HERV-K(HML-2) expression is also upregulated by Sp1 and Sp3 zinc finger proteins, expressed during oxidative stress and involved in control of many housekeeping genes (Diem *et al.*, 2012).

1.12. External factors influencing HERV-K.

Many external factors may have a role in HERV-K expression. Ultraviolet radiation, known to damage DNA and being a significant mutagen in melanoma development (Schanab *et al.*, 2011), was found to increase expression of HERV-K transcripts, as well as spliced Rec and Np9 proteins. Irradiation with 30 mJ/cm^2 of UVB was found to be enough to induce overexpression of HERV-K *pol* transcript in primary epidermal keratinocytes (Hohenadl *et al.*, 1999). Moreover, ultraviolet UVC

radiation between 10-30 mJ/cm^2 was sufficient to increase HERV-K Rec and Np9 expression significantly in normal human epidermal melanocytes. The expression did not change for MEWO melanoma cell lines, which suggests a role in melanoma development from healthy cells (Reiche, Pauli and Ellerbrok, 2010).

Different hormones, including progesterone, other androgens and estrogen influence prostate (Royuela *et al.*, 2001; Grindstad *et al.*, 2018) and breast cancer (Tian *et al.*, 2018) - particularly, some prostate cancer tissue overexpress progesterone A and progesterone B receptors. Presence of these receptors correlates with poor prognosis for prostate cancer patients; Grindstad *et al.* (2018) analysed prostatectomy specimens from 535 prostate cancer patients and found progesterone A (PGRA) and B receptors (PGRB) expression via tissue microarray. Immunohistochemistry located the PGRA stromal tissue and PGRB both in stromal and epithelial tissue. Multivariate and univariate analyses revealed strong association of presence of progesterone receptors and negative prognosis for prostate cancer patients exhibiting the receptors. Golan *et al.* (2008) have found T47D breast cancer cell lines treated with progesterone to express 5- to 10-fold more HERV-K *env* transcripts compared to untreated T47D breast cancer cells via RT-PCR. Treating T47D cells for 48-hours with β-estradiol, followed by 48-hours of progesterone increased expression of HERV-K reverse transcriptase in breast cancer cell lines 5-10 times in comparison to untreated cells, displayed with fluorescent antibody staining and confocal microscopy. Estradiol is also able to activate HERV-K expression, just like progesterone, by binding into progesterone-response element located within the long terminal repeat of the HERV-K (Nguyen *et al.*, 2019). Consistently, Wang-Johanning *et al.* (2003), showed, that T47D breast cancer cells treated with 10nM of estradiol increase HERV-K *env* transcript expression 6-fold.

Peripheral blood mononuclear cells were analysed to find if gene expression profiles in prostate cancer patients and healthy controls correlate with smoking status. RT-PCR of *gag* sequences isolated from men without cancer and prostate cancer patients shows that in both groups the RNA of HERV-K is elevated. RNA levels were significantly higher in cancer cases though (about 10 times). Current smokers displayed more than 3 times increased *gag* mRNA concentration, compared to people, who had never smoked (Wallace *et al.*, 2014).

Lemaître *et al.* (2017) have studied the association between expression of HERV-K Env protein and the epithelial–mesenchymal transition (EMT) pathway. EMT is a process in which epithelial cells loose polarity and cell-to-cell adhesion, becoming invasive mesenchymal stem cells, which are multipotent and can differentiate into several other types of cells (Kalluri and Weinberg, 2009). HERV-K is associated with induction of several EMT transcription factors, such as ETS Variant Transcription Factor 4, ETS Variant Transcription Factor 5 and Early Growth Response 1 factors which are downstream effectors of the MAPK ERK1/2. These are also associated with cellular transformation and shifting the cell phenotype towards more mesenchymal. A number of genes

are expressed by HERV-K Env transfection, including the aforementioned transcription factors, leading to activation of the ERK1/2 pathway (Zhou *et al.*, 2016), specifically in melanoma cell lines including SKMel28, WM3526, WM3682, as well as MCF10A breast cell lines. Several other studies link HERV-K expression with ERK1/2 pathway activation in different cancer types, including melanoma, pancreatic cancer (Li *et al.*, 2017) or hepatocellular carcinoma (Ma *et al.*, 2016). Some studies suggest, though, that the overexpression of HERV-K in certain cancers is a result of ERK pathway alteration, as opposed to ERK being activated by HERV-K (Li *et al.*, 2010).

1.13. Recombinational activity of HERV-K.

Non-allelic homologous recombination may lead to genomic rearrangements, which might be especially dangerous in diseases like cancer, where aberrant HERV activity is implicated (discussed in detail in section 1.16). Due to sequence similarity, especially in the LTR regions, HERV-K insertions are suspected to undergo recombination, which has been reported previously (Kamp *et al.*, 2000; Hughes and Coffin, 2001). Recombination between HERV-K insertions could result in activation of adjacent genes due to LTR promoter activity. This could be caused by gene conversion or mutations and influence expression of other sequences (Hughes and Coffin, 2005). Such recombination events may damage existing genes if the recombining sequence includes up- or downstream DNA, which in turn translocate in the genome. On the other hand, it could cause frameshifts due to extra start codons originating from recombining viral sequences (Young, Stoye and Kassiotis, 2013). Other possibilities include sequential duplications of certain HERV-K(HML-2) elements, which are observed throughout the human genome. These may influence chromatin density and expression of other genes in regions rich in duplications. Such duplications occurring in random pattern between chromosomes may also influence expression of other genes just like recombination (Blekhman, Oshlack and Gilad, 2009). By inserting HERV-K(HML-2) sequences inside other genes or adjacent to them such duplications could disturb their expression regulatory mechanisms (Fuentes, Swigut and Wysocka, 2018). Hughes and Coffin (2005) studied 15 HERV-K elements and found that 5 of them displayed evidence of gene conversion or ectopic recombination events by clustering with different elements or elements in different primates in their phylogenetic analysis.

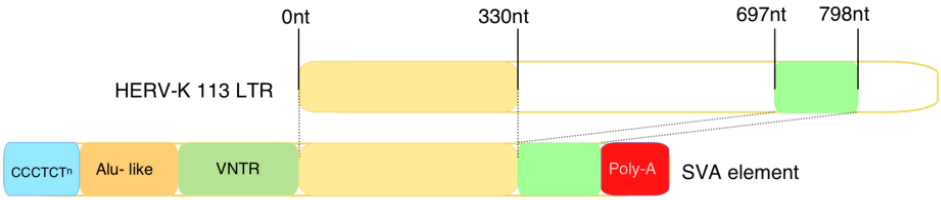

Figure 7 Structure of an SVA element compared to HERV-K(HML-2) LTR. Two regions are shared between SVA and HERV-K 113 LTR. The yellow region represents a sequence of 0-330nt, the green region represents a sequence of 697-798nt in HERV-K 113. VNTR - variable number tandem repeat, a short nucleotide sequence, organized as a tandem repeat, Alu-like - region made up of two antisense Alu fragments, separated by an intervening region, CCCTCTn - variable number of a simple CCCTCT hexamer copies and a poly-A - tail consisting of multiple, repeated adenosine monophosphates (Shen *et al.*, 1994).

The typical structure of HERV-K(HML-2) LTR element described here resembles parts of structure of another transposable element present in humans, Sine-VNTR-Alu, or SVA (Ostertag *et al.*, 2003) (see Figure 7). In fact, SVA elements share two regions with HERV-K(HML-2): 1-330nt and (approximately) 697-798nt, with regards to canonical HERV-K(HML-2) LTR (accession number AY037928; 1-968bp) (Ono, Kawakami and Takezawa, 1987; Shen *et al.*, 1994). These regions of homology between the two families of transposable elements could serve as possible sources of homologous recombination between HERV-K(HML-2) and SVAs. This is especially important due to the number of SVA elements present in the human genome being up to an order of magnitude higher than HERV-K(HML-2) elements (Lander *et al.*, 2001).

1.14. Master gene hypothesis.

As observed in other families of transposable elements, particularly SINEs, the existing subfamilies of transposable elements undergo sequential replacement, suggesting that only some of the elements can create copies, effectively becoming master genes in their ability to replicate within the genome (Johnson and Brookfield, 2006). Furthermore, an older *Alu* subfamily was found to be able to replace members of younger families via gene conversion events. The relatively young Sb2 *Alu* element at LDLR locus displays highly distinctive sequence when comparing to the corresponding sequences from other primates, including chimpanzee, orangutan and gorilla. Phylogenetic analysis showed that it has undergone gene conversion with an older, truncated *Alu* repeat, suggested to be a member of a master gene family (Kass, Batzer and Deininger, 1995). Since HERV-K(HML-2) was found to display evidence of ectopic gene conversion (as discussed in 1.13), insertional polymorphism in modern populations of humans (see section 1.9) and its role in various diseases (see sections 1.15-1.16), it would be interesting if there were any evidence for such master genes and their role within HERV-K(HML-2) insertional and recombinational activity.

1.15. HERV-K in various diseases.

HERV-K(HML-2) role in disease has focused a great deal of interest. The fact that it is the youngest of all known human endogenous retroviruses makes it the least mutated one, possibly harbouring most of the functionality of its exogenous ancestors (Turner *et al.*, 2001). Insertional polymorphism is a very interesting aspect from the disease point of view, since it implies that functionality provided by polymorphic proviruses might manifest itself in certain diseases (Meyer *et al.*, 2017). Additionally, since viral functionality is generally unfavourable to the host organism, it is highly possible that any new proviral insertions would disturb normal genetic profiles and in turn contribute to disease development. Even if such insertion was completely dormant, the epigenetic mechanisms controlling it may still be altered by external factors and in turn activate viral expression in the host organism (Hurst and Magiorkinis, 2017). Indeed, a role for HERV-K(HML-2)

was suggested in a variety of diseases, including amyotrophic lateral sclerosis (ALS) (Douville *et al.*, 2011), rheumatoid arthritis (RA) (Freimanis *et al.*, 2010), HIV/AIDS (Contreras-Galindo *et al.*, 2013) or diabetes mellitus (DM) (Marguerat *et al.*, 2004). Recently, there has been a very strong report linking presence of certain HERV-K(HML-2) insertion and susceptibility to addiction, which was confirmed with proposing a biochemical process behind the phenomenon (Karamitros *et al.*, 2018). Below a short review of these diseases is presented, along with selected work regarding HERV-K(HML-2) role in each of them.

1.15.i. Amyotrophic lateral sclerosis (ALS).

A hypothetical involvement of HERV-K in ALS is based on viral proteins observed to be overexpressed in ALS patients, including *pol* and *env* products that are presumably neurotoxic. Douville *et al.* (2011) has found increased expression of HERV-K(HML-2) *pol* transcripts in a brain tissue samples from 20 ALS patients. Cloning and sequencing the particular mRNA revealed that the transcripts came mostly from 3q27.2 (20% samples) and 7q34 (85% of samples) proviral loci. Immunostaining revealed the presence of HERV-K(HML-2) reverse transcriptase in prefrontal and motor cortex tissue samples in 10 out of 13 tested specimens. The *pol* transcript was also strongly positively correlated with the expression of TAR DNA binding protein 43 (TDP-43) transcripts. TDP-43 is a protein involved in many pathways, including stabilizing and processing mRNA, cell cycle regulation and apoptosis (Vogt *et al.*, 2018). TDP-43 can form insoluble aggregates in ALS patients if it becomes hyperphosphorylated and abnormally ubiquitylated (Van Deerlin *et al.*, 2008). This shows that specific HERV-K insertions can be directly associated in a process, that influences ALS progression and presumably these proviral loci could be targets for therapeutic and diagnostic applications.

1.15.ii. Rheumatoid arthritis (RA).

RA patients have been found to overexpress Rec and Np9 transcripts in synovial cells (Ehlhardt *et al.*, 2006). Significant correlation was observed between presence of one of the youngest, polymorphic insertions, HERV-K113, and RA among patients in the Polish population using RT-PCR, compared to healthy controls (Krzysztalowska-Wawrzyniak *et al.*, 2011). Freimanis *et al.* (2010) demonstrated that increased levels of HERV-K(HML-2) in synovial fibroblasts correlated with presence of proinflammatory cytokines (TNF-α and IL-6) in comparison to control groups. They have also confirmed that Epstein–Barr virus proteins can significantly increase HERV-K(HML-2) *gag* expression in in synovial fibroblasts and correlate with inflammation levels. Unfortunately, the nature of RA and complexity of the mechanisms linking the immune system, endogenous and exogenous retroviruses make it difficult to assess the molecular dependencies between all the factors.

1.15.iii. Human immunodeficiency virus / Acquired immunodeficiency syndrome (HIV/AIDS).

Since HERVs and HIV are both retroviruses, controlled by LTRs and sharing basic gene structure, like both encoding Gag, Pro, Pol and Env proteins, there is a possibility of interaction between the two. Indeed, there are reports, like work of Young *et al.* (2018), that show upregulation of particular HERV-K(HML-2) proviruses : 6q25.1, 8q24.3, and 19q13.42, on average between 3- and 5-fold in HIV-1-infected CD4+ T cells, comparing to HIV-1 free control, whereas 12q24.33 HERV-K insertion was repressed in infected cells. O'Carroll *et al.* (2020) have demonstrated recently that HIV-1 Rev protein is able to mediate the nuclear export of an element present in HERV-K mRNA called the Rec Response Element (RcRE), just like the Rec protein produced from the HERV-K(HML-2) transcript (Gray *et al.*, 2019), and increase HERV-K expression levels. Rev turns out to be a functional analogue of Rec and the two can complement their function, boosting viral expression. The mutual dependency between HERV-K and HIV can be used in HIV progression diagnosis, as demonstrated by Contreras-Galindo *et al.* (2007). The authors tested influence of HIV-suppressive HAART in four patients responding positively to the therapy and six, where HAART was failing to help. In the group responding positively, the titres for HERV-K remained undetectable, whereas high titres preceded HIV-1 rebounds in patients where HAART was less effective. Therefore, levels of HERV-K viral load may be used as a hint towards HAART success rate and possible HIV-1 treatment.

1.15.iv. Diabetes mellitus (DM).

A particular HERV-K(HML-2) insertion has been implicated to play a role in DM. The provirus is HERV-K18(1q23.3) and it is located within intron 1 of CD48 on human chromosome 1q, which encodes T-cell superantigen (SAg). T-cells with HERV-K18 SAg reactive T-cell receptor Vβ7 chains were previously confirmed in pancreas, spleen (Conrad *et al.*, 1994; Somoza *et al.*, 1994) and in circulation (Luppi *et al.*, 2000) at early stages of type 1 DM. HERV-K18 was confirmed to be associated both with type 1 and type 2 diabetes. Marguerat *et al.* (2004) studied genomes of 754 families with both parents and at least two offspring affected by type 1 DM, with data obtained from Bain *et al.* (1990) and Lernmark *et al.* (1997) sequencing projects (Bain, Todd and Barnett, 1990; Lernmark *et al.*, 1997). They genotyped five SNPs within the HERV-K18 sequence and 9 SNPs within the adjacent CD48 locus. A significant association between type 1 DM individuals and nt8146, nt8594, and nt8460 SNPs has been confirmed and they hypothesized a possible influence of these SNPs on expression of adjacent genes, particularly CD48 (Marguerat *et al.*, 2004). Dickerson *et al.* (2008) observed similar association between HERV-K18 polymorphisms and type 2 DM in patients who additionally suffered from schizophrenia. They used RT-PCR with fluorescent labelling to amplify and detect SNPs in HERV-K18 region. Individuals with schizophrenia and diabetes shown statistically significant association with a haplotype defined by 2 polymorphisms:

7086 T/T and 8146 T/T in the envelope region of HERV-K18. These findings confirm that a specific HERV-K(HML-2) is associated with DM occurrence in different, independent populations, although detailed genetic and biochemical analysis of interaction between the provirus and DM patient's organism needs to be further assessed.

1.15.v. Addiction.

There is little known about HERV activity playing role in substance abuse. However, a recent study by Karamitros *et al.* (2018) shows a correlation between presence of a certain HERV-K(HML-2) integration (namely *RASGRF2-int*, also called De6/Ne1/K10) located in the middle of RASGRF2 gene (implicated in various substance abuse disorders (Fasano *et al.*, 2009; Schumann *et al.*, 2011)) and intravenous (IV) drug abuse. The authors tested two, independent human populations of 102 Greece-based HIV-positive persons who inject drugs (PWIDs) and 100 HIV-positive individuals who contracted HIV via other routes. 14 PWIDs were positive for *RASGRF2-int* insertion vs. 6 non-PWIDs, that suggests twofold higher frequency and was confirmed to be significantly correlated with IV drug use. To confirm the findings and ascertain independence from other factors such as ethnicity and origin, a similar test was performed on a cohort of chronic hepatitis C virus (HCV) infected United Kingdom individuals who used drugs intravenously. The control consisted of HCV-positive individuals who got infected through bleeding disorders. 34 out of 100 PWIDs were *RASGRF2-int* positive in relation to 8 out of 84 controls, again proven to be statistically significant. Inserting *RASGRF2* into HEK293 cell line using CRISPR/Cas9 produced *RASGRF2-201* and *RASGRF2-206* transcripts that lead to enhancement of dopaminergic activity (as shown for substance abuse disorders, especially alcoholism (Stacey *et al.*, 2012; Easton *et al.*, 2014)) and result in increased addiction potential. Upregulation of HERV-K(HML-2) insertion associated with *RASGRF2* was suggested to be activating the *RASGRF2* expression via chromatin remodelling. This shows a clear correlation between proviral HERV-K(HML-2) insertion and susceptibility to drug use, suggesting the possible biochemical pathway that can influence the differences in drug tolerance and susceptibility to dependency between individuals harbouring an insertion and those who do not.

1.16. HERV-K in cancer.

1.16.i. HERV-K oncogenic properties.

An increased amount of HERV-K(HML-2) transcription has been reported in association with a number of different types of cancers (reviewed in sections 1.6.ii-1.6.vii), suggesting that activity of HERV-K(HML-2) might contribute to oncogenesis. Due to hyper- and hypomethylation HERV-K proviruses may be over- or underexpressed in cancer cells. HERV LTR sequences are targets of many DNA binding proteins, such as p53 (Beyer *et al.*, 2011), and could drive expression of genes located nearby. For example, proto-oncogene stimulating factor CSF1R, activated by THE1B retrotransposon LTR is known to be aberrantly activated in Hodgkin's lymphoma cells that are resistant to apoptosis (Lamprecht *et al.*, 2010). Although no HERVs have been demonstrated to be infectious *in vivo,* one of the possible oncogenic mechanisms regarding HERV activity is insertional mutagenesis. In fact, several types of cancer cell lines, like melanoma (Muster *et al.*, 2003) and teratocarcinoma (Bieda, Hoffmann and Boller, 2001), have been observed to produce retrovirus-like particles. These particles, despite being seemingly intact, have never been reported to infect neighboring cells. Even the most recent insertions, HERV-K113 and HERV-K115 have been found to produce retroviral particles *in vitro*, yet none of them were infectious (Boller *et al.*, 2008). HERV insertions might cause alternative splicing of existing proteins. Since some members of the HERV-K family have inserted into the human genome after the human – chimp split, it is possible that these insertional and recombinational proprieties contribute to genetic differences between humans and chimpanzees (Khodosevich, Lebedev and Sverdlov, 2002). HERV-K provirus also facilitates oncogenic fusions, for example fusions between members of the ETS gene family, such as fused ERG and ETV1, which is a known oncogene observed in prostate cancer (Helgeson *et al.*, 2008). Another way of direct influence of HERVs on cancer are native, tumor-promoting HERV transcripts. The envelope glycoprotein transmembrane subunit of HERV-K(HML-2) contains an immunosuppressive domain that is expressed on the surface of the cell. This could be actively counteracting the immune response to cancer cells, a characteristic also adopted by syncytin-2 which facilitates fusion between placenta cells using the Env transmembrane subunit (Blaise *et al.*, 2003).

Another oncogenic factor of HERV-K family is the presence of spliced *rec* and *np9* transcripts, coming from the *env* gene. Rec protein is highly similar to HIV Rev protein, that is aiding the transport of transcripts from the nucleus (Dewannieux, Blaise and Heidmann, 2005). Np9 is an alternatively spliced variant, sharing only 14 base pairs with Rec. Both of these proteins are observed in elevated levels in many malignant tissue types (Tavakolian, Goudarzi and Faghihloo, 2019), including melanoma, germ cell tumors, leukaemia and blood lymphocytes. The elevated levels are present in cancer cells but not healthy cells, such as melanocytes. Presence of Rec and

Np9 was observed to disturb various biochemical pathways within cells that contribute to progression of different types of cancer. Both of the proteins are interacting with promyelocytic leukaemia zinc-finger protein (PLZF), which acts as transcriptional repressor of c-MYC proto-oncogene (Denne *et al.*, 2007). Rec also binds testicular zinc finger protein (TZFP) and small glutamine-rich tetratricopeptide repeat protein, which cause androgen repression (Kaufmann *et al.*, 2010). Np9 interacts with Numb/Notch signaling cascade by binding to Numb protein X, dysregulating it in several types of cancer (Armbruester *et al.*, 2004).

HERV-K activity has been extensively studied in the following five types of cancer: breast cancer, brain cancer, prostate cancer, melanoma and leukaemia. These are discussed below and followed by reports of HERV involvement in other cancers. These include pancreatic cancer, recently discovered to express polymorphic HERV-K113/K115 proviruses and a unique 794 bp HERV-K-derived spliced transcript (Li *et al.*, 2017).

1.16.ii. Breast cancer.

Human endogenous retroviruses (HERVs) display high resemblance to mouse mammary tumor virus, which is known to cause mammary cancer in mice (Burmeister *et al.*, 2004). HERV transcripts have been found both in healthy and malignant breast tissue before, as illustrated by a cDNA sequencing study, conducted by Flockerzi *et al.* (2008) on 49 tumor samples and corresponding non-tumor controls. Transcription of the HERV-K(HML-2) elements was found to be upregulated in malignant cells. In particular, locus c1_B (HERV-K102) was overexpressed both in malignant and normal tissue, whereas locus c3_C (K121) was significantly overexpressed in normal tissue compared to malignant samples. This suggests that a specific pattern of HERV-K expression is important in breast cancer, rather than simple measurement of total HERV-K expression rate or presence of a particular transcript.

Johanning *et al.* (2017), performed a large study of 512 breast cancer patients based on bioinformatics analysis of data from The Cancer Genome Atlas and Cancer Genomics Hub to assess expression of particular HERV-K insertions. The authors hypothesized that specific HERV-K expression patterns could also be a potential therapeutic and diagnostic target in basal breast cancer. They found a 1.7-fold raise of total HERV-K RNA expression in basal breast cancer subtype compared other subtypes. Moreover, the upregulated proviruses included insertionally polymorphic loci (specifically HERV-K108 that has functional *env* gene, HERVK109 that has functional *gag* gene (Dewannieux, Blaise and Heidmann, 2005), HERV-K113 and HERV-K115, which are the youngest genomic integrations and insertionally polymorphic (Turner *et al.*, 2001)), suggesting, that presence of these loci might be involved in development of basal subtype of breast cancer.

Johanning *et al.* (2017) were also able to correlate the overexpression of HERV-K with overexpression of cyclin kinase CDK6, that controls cell proliferation (Matushansky, Radparvar and Skoultchi, 2003), cell cycle progression inhibitor E2F5 (Wyllie, 2002) and retinoblastoma pRb protein, a key tumor suppressor (Murphree and Benedict, 1984), found to be over-phosphorylated at sites S807/S811. Deregulation of these proteins is often observed in cancers. The findings suggest that HERV-K expression plays a role in Rb phosphorylation, that inhibits expression of genes controlled by E2F transcription factors and influences a number of different signaling factors involved in the cell cycle control. These factors include E-cadherin, β-catenin and p-mTOR-s2448, which were found to be positively correlated with HERV-K in basal breast cancer patients. In particular, β-catenin concentrations were significantly upregulated in the cytosol and nuclei of basal invasive breast cancers, which was associated with HERV-K overexpression (Johanning *et al.*, 2017). The research done by Johanning *et al.* (2017) links HERV-K(HML-2) with the more aggressive basal breast cancer phenotype. Furthermore, it suggests, that proviral activation is associated with changes in the regulation of molecular pathways, leading to alteration of the cell cycle observed to be malfunctioning in many types of cancer. HERV-K involvement in Rb protein phosphorylation in breast cancer cells shows a correlation between HERV-K(HML-2) expression and tumor suppression.

Wang-Johanning *et al*. (2003) showed that expression of HERV-K was largely dependent upon hormonal levels *in vitro* due to estrogen-driven promoters, particularly increased by β-estradiol and progesterone treatment. Reverse transcription and sequencing done by Johanning *et al.* (2017) for clinical samples show that HERV-K108 and HERV-K109 loci show the highest levels of transcription, in addition to HERV-K10, HERV-K102, HERV-K103, HERV-K104 and HERV-K107 being overexpressed. Additionally, two of the youngest, insertionally polymorphic HERV-K(HML-2) members, HERV-K113 and HERV-K115, were found to be overexpressed in the studied cell line samples. Overexpression was directly dependent on the hormone dosage and upon increasing hormonal levels HERV-K transcription increased 4-10 fold (Wang-Johanning *et al.*, 2003). This expands the Johanning *et al.* (2017) research, showing that HERV-K expression is modulated by hormonal levels and plays a role in various breast cancer subtypes.

More analysis of breast cancer cell lines shows that HERV-K activity changes cell morphology and promotes epithelial-mesenchymal transition (EMT). Proviral expression, proposed previously to be affecting biomolecular pathways that control the cell cycle, was tested by Lemaître *et al.* (2017) in breast cell lines. The authors used overexpressed HERV-K(HML-2) Env protein in MCF10A cell line. As a result, a change in the morphology was observed for the tested cell line. The cells started becoming spindle-shaped and elongated: changed their original organization pattern, becoming more dispersed, displaying migration and invasive proprieties. On a molecular level, the cells showed an increase in expression of the mesenchymal markers: fibronectin (Boucaut *et al.*, 1984)

and N-cadherin (Takeichi, 1988). EMT-associated transcription factors Snai1 and 2 expression also increased (Cano *et al.*, 2000), along with a decrease in E-cadherin (Takeichi, 1988), which are molecular changes typical for cells undergoing epithelial-mesenchymal transition. Furthermore, evidence for this can be found on the generic level, where HERV-K overexpression upregulated 7 transcription factors, including EGR1 (Baron *et al.*, 2006), ETV4, ETV5 (Oh, Shin and Janknecht, 2012) and FosB (Ting *et al.*, 2019) which are directly associated with EMT. The ERK1/2 proteins were found to be overphosphorylated, therefore the overexpression of the abovementioned factors is very likely due to activation of MAPK pathway (Pearson *et al.*, 2001).

The correlation between HERV-K(HML-2) expression and progression of breast cancer into more malignant forms has been confirmed by Zhao *et al.* (2011), in a study of 40 Chinese women who were suffering from the basal subtype of this disease. Compared to 70% of breast cancer samples harboring the HERV-K *env* mRNA, Zhao *et al.* (2011) observed the same RNA only in 20% of adjacent healthy tissue and no HERV-K *env* mRNA in control samples of 40 healthy women. Complete absence of the viral transcriptome in non-tumor tissue samples strongly suggests association of HERV-K(HML-2) *env* expression and occurrence of breast cancer. Quantitative measurement of the viral mRNA significantly correlated with disease progression and tumor size: in 82.9% of patients that displayed high viral expression, the tumor size exceeded 3 cm. 72.7% of tested patients with highly elevated *env* expression suffered from recurrence and metastasis of their disease. EMT evidence observed *in vitro* directly relates to rising expression of HERV-K(HML-2) in patients with later stages of the breast cancer, where only 25% of stage I patients display high load of viral expression. High expression of HERV-K(HML-2) was observed in 58% of patients in stage II and 80% of stage III breast cancer patients.

Studies performed both *in vitro* on cell lines and clinical samples clearly show that HERV-K(HML-2) correlates with development of breast cancer. On the phenotypic level, the patient data show that raising HERV-K(HML-2) transcript presence is observed with progression of the disease stage and marks metastasis of breast malignancies. On the molecular level, these processes are confirmed by observations of transcription factors controlling expression of proteins involved in cell cycle control and processes, like the MAPK signaling pathway, that is involved in cell proliferation as well as molecular metabolism (Pearson *et al.*, 2001). HERV-K(HML-2) expression coincides with alteration of cell signaling proteins levels and as a result of complex cellular processes, completely healthy cells can acquire proliferative, malignant proprieties. Effects of such processes like acceleration of malignant transformation and tumor cells becoming metastatic are observed in patients with increasing expression of HERV-K(HML-2) viral proteins.

1.16.iii. Melanoma.

HERV-K expression has been found to be one of the key requirements in the transition of melanoma cells into more malignant phenotypes. Just like in breast cancer, analysis of HERV-K(HML-2) expression in melanoma shows that proviral transcripts are present in cancer cells. Moreover, melanoma cells can both undergo transformation towards metastasis and evade the immune system with involvement of HERV-K(HML-2) expression and downregulation of melanoma differentiation antigens (Maeurer *et al.*, 1996). Serafino *et al*. (2009) has put TVM-A12 melanoma cell lines to starvation conditions in a low-serum medium, and confirmed formation of metastatic cells via FACS. RNA interference used to downregulate HERV-K *pol* and *env* expression completely diminished the effects of low-serum, the cells did not display any evidence of metastatic proprieties. According to Schmitt *et al.* (2013), HERV-K(HML-2) loci that become overexpressed under stressful (UV-irradiation) conditions in melanoma cell lines include ERVK-1, ERVK-7 and ERVK-5 in 29, 32 and 53% in tested melanoma cell lines (RNA3, LNM2 and RNA3 respectively) and ERVK-14, overexpressed in up to 92% of tested melanoma lines, specifically the WM3734a cell line. Direct silencing of HERV-K(HML-2) was displayed to prevent progression of malignant cells into invasive and metastatic forms. It shows that HERV-K(HML-2) expression must be involved in molecular processes causing the cells to undergo transformation into more malignant phenotypes. Melanoma cell lines were even found to produce non-infective HERV-K(HML-2) viral particles (Muster *et al.*, 2003). However, the precise molecular mechanisms were not studied in detail.

Katoh *et al.* (2011) have studied activation of HERV-K insertions by a melanoma oncogene, MITF-M. Analysis of HERV-K LTR sequences revealed that MITF-M can bind to three motifs in addition to TATA box. HEK293 cell lines transfected with the MITF-M expression vector significantly induced *env* and *rec* mRNA synthesis, whereas melanoma cell lines produced MITF-M factor itself. Semiquantitative RT-PCR of three melanoma cell lines, G361, SM-MEL-28 and MeWo, showed expression of HERV-K *gag* (1:4:2.7 fold in the tested lines, respectively), *env* (1:3:1.4) and *rec* (1:2:1.5) transcripts. Normal human melanocytes showed very faint signals for the three genes. Although the MITF-M factor is present in all melanocytes, it's the post-translational modifications including Ser73 phosphorylation, altered in malignant cells, that determines its affinity to HERV-K(HML-2) promoters (Hemesath *et al.*, 1998). Among different cancers, only melanoma and teratocarcinoma tumors display both spliced *env* and *rec* transcripts (Löwer *et al.*, 1993). Büscher *et al.* (2005) has confirmed presence of HERV-K transcripts in 34 patients melanoma metastases using RT-PCR, but found spliced *env* and *rec* mRNA only in 45% of the biopsies tested. The authors have reported antibodies against HERV-K(HML-2) in melanoma patients for the first time. Out of 60 melanoma patient serum samples, 13 reacted strongly with the recombinant HERV-K transmembrane protein in Western blot. They have also tested viral expression in SK-MEL-28 cell lines, where they observed formation of viral-like particles, unable to infect new cells; similar to

previous report from Muster *et al.* (2003). This shows that HERV-K(HML-2) is not only expressed in the cancer tissue samples, but can produce proteins and even assemble viral-like particles in the cancer cells. These particles are defective, most likely due to mutations in the HERV-K(HML-2) sequences present in the genome.

Moreover, Argaw-Denboba *et al.* (2017) has found, that TVM-A12 melanoma cell lines undergo phenotype switching after placement in stem-cell medium, shifting from anchorage-independent aggregates with limited proliferation capabilities in serum-free medium. Gradual addition of Fetal bovine serum to the growth medium caused the cells to progressively turn into adherent phenotype and become spindle-shaped. The morphology changes were followed by changes in expression profiles, including increased levels of HERV-K(HML-2) *env* transcripts. Additionally, the cells started displaying migration and proliferation abilities. These are all characteristics of EMT (see section 1.12), which was completely diminished upon silencing of HERV-K(HML-2) expression via RNAi. It was suggested, that other external factors, especially UV can result in similar HERV-K(HML-2) upregulation and, consequently, EMT (Hohenadl *et al.*, 1999).

Overall, the presence of HERV-K(HML-2) in melanoma is well documented and there is evidence of possibly the strongest prevalence of proviruses on different aspects of melanoma. It is the only type of cancer where both spliced *rec* and *env* transcripts were described to occur simultaneously (Büscher *et al.*, 2005). The literature also provides a clear correlation between expression of HERV-K and EMT characteristics in melanoma cells, which were reversible upon silencing of HERV-K transcription. Moreover, expression of the proviruses and viral-like particles can be directly observed in melanoma cell lines. All of the above evidence shows that HERV-K(HML-2) is a good candidate to study the development of melanoma.

1.16.iv. Prostate cancer.

Overexpression of HERV-K(HML-2) transcripts in prostate cancer cell lines have been well documented by Agoni *et al.* (2013). RT-PCR and subsequent sequencing identified HERV-K102 and HERV-K118 expression in DU145 cells, with additional HERV-K50F loci active in PC3 and VCap cells. Spliced products of these specific HERV-K loci were detected, along with many solo LTRs proven to be independent of full-length insertions by RT-PCR. This shows that in multiple different prostate cancer cell lines, only those three insertions are transcriptionally active. K108 was especially prevalent, since its sequence retained full-length ORFs for all viral proteins and it undergoes splicing at the canonical *env* site (Mayer *et al.*, 1999). Agoni *et al.* (2013) used a quantitative RT-PCR to show that the solo-LTR transcripts represent over 100-fold higher levels compared to full-length proviruses and, as Subramanian *et al.* (2011) showed, generally reflect the ratio between the amounts of known solo-LTRs in the reference human genome and the full-length elements (for detailed study, see sections 1.17 and 3.1). Findings show again that in prostate cancer cell lines, like

in breast cancer cell lines (see section 1.16.ii) or melanoma (section 1.16.iii), it is important to observe the specific pattern of HERV-K(HML-2) expression and identify loci that are transcriptionally active, rather than study the transcription levels as a whole to get the most accurate picture of HERV-K(HML-2) activity.

More recently, other researchers have been trying to use HERV-K transcripts as prostate cancer biomarkers. An interesting study was performed by Wallace *et al.* (2014), who hypothesized that the HERV-K transcripts can be used for early tumor detection, based on earlier research on the topic (Goering, Ribarska and Schulz, 2011; Reis *et al.*, 2013). They tested a large cohort of 294 prostate cancer patients with RT-PCR, 142 of which were of African-American origin and 152 were European-American; 135 were healthy men, 75 African-American and 60 European-American. The results were that HERV-K *gag* mRNA can be found in PBMCs of both prostate cancer patients and samples from healthy men, but the transcript levels are significantly higher in men with prostate cancer. Using multivariate regression analysis, they were able to associate an above median *gag* expression in PBMC within the tested cohort with 6-fold increased odds of being prostate cancer positive, whereas men with the highest quartile of expression were >12-fold more likely to be prostate cancer positive. Additionally, the statistical significance grew with subjects' age. Although smoking status did not correlate with HERV-K expression, it increased the strength of association between HERV-K *gag* expression and disease with growing pack-years of tobacco exposure. This contrasts with previous research done by Gabriel *et al.* (2010), who strongly suggested smoking and its metabolites regulate transcription of various HERVs, including HERV-K; however, they studied HERV-K(HML-6) family. Interestingly, all of the aforementioned effects described by Wallace *et al.* (2014) were greatly pronounced in the African-American population, since the baseline levels of HERV-K *gag* were up to 10-fold higher compared to European-American population, both for cancer and non-cancer cases. Based on that research, HERV-K(HML-2) mRNA levels in PBMCs can be used as a reliable biomarker for prostate cancer detection. It could be especially useful in complementing or replacing widely used PSA levels as diagnostic marker, since it is known that PSA level testing sensitivity decreases with age, whereas HERV-K *gag* levels are actually more significantly associated with disease status for older men (Martin, Starks and Ambs, 2013).

Cancer progression in many prostate cancer cases is associated with hypomethylation of transposable elements, such as LINE-1 (Estécio *et al.*, 2007). In some prostate cancer cell lines and tissues, HERV-K elements become hypomethylated and involved in the possibility of cancer-specific genomic translocations. Tomlins *et al.* (2007) have identified a particular fragment of HERV-K(HML-2) locus 22q11.23, which undergoes fusions with members of the ETS family of genes, namely ERG, ETV1 and ETV4 in many cases of prostate cancers. Rapid amplification of cDNA ends (RACE) experiments, that reverse transcribed HERV-K(HML-2) mRNA and amplified resulting cDNA with PCR revealed that the recombination reaction involves HERV-K 22q11.23 insertion, which

interchanges its two first exons with exons 1-4 of ERV1 gene. This causes aberrant expression of the HERV-K-ETV1 hybrid, probably due to insertion of promoter and other regulatory sequences from the proviral locus. The resulting rearrangement has been proven in a mouse model to induce neoplastic phenotype changes in the prostate, supporting the oncogenic role of such recombination. Chromosomal proximity of the ETS genes and TMPRSS2 androgen receptor, which also undergoes partial fusion, can cause overexpression of these genes upon androgen stimulation. DNA methylation of that particular fusion was studied by Goering, Ribarska and Schulz (2011), who found frequent overexpression of HERV-K 22q11.23 using qRT-PCR in several prostate cancer cell lines, including 22Rv1, LNCaP, MDA PCa 2b and NCCIT. The relative expression of the provirus could be observed in all tested prostate cancer cell lines – interestingly MDA PCa 2b HERV-K(HML-2) RNA levels were about 100-fold higher than in other cell lines. Similarly, HERVK17 expression was elevated in 22Rv1, LNCaP and MDA PCa 2b, again approximately 5-10-fold higher in the latter that in other cell line types. Cell lines lacking androgen receptors (PC-3, DU-145) did not exhibit overexpression of HERV-K. These particular insertions were of interest, because of previous reports of them being overexpressed in LNCaP prostate cancer lines (Tomlins *et al.*, 2007; Hermans *et al.*, 2008). The prostate cancer cell lines found to overexpress both HERV-K loci have also had lower methylation values in a pyrosequencing experiment of bisulfite-treated DNA using proprietary assays. The methylation values of repetitive elements were consistently 10-fold lower for cell lines overexpressing either HERVK17 or HERV-K 22q11.23 – positive cell lines.

To sum up, prostate cancer is a malignancy that exhibits specific HERV-K(HML-2) overexpression, and the presence of certain provirus at 22q11.23 locus was found to be a source of recombination within transformed cells (Goering, Ribarska and Schulz, 2011). The high correlation between expression of certain HERV-K(HML-2) transcripts can provide an alternative diagnostic method, complimentary to PSA level monitoring, especially that it was found to be more precise and sensitive to prostate cancer occurrence. Furthermore, expression patterns of different proviral insertions vary between non-malignant prostate cells and malignant counterparts, suggesting that specific insertions might be associated with tumor development and progression, as well as severity of the disease. This provides a potential targeted therapy opportunity, where silencing expression from specific HERV-K(HML-2) loci could be beneficial for reducing the rate of malignant transformation, however specific studies on such novel therapies have not been yet conducted.

1.16.v. Leukaemia / Lymphoma.

In order to investigate the presence of HERV-K(HML-2) transcripts and protein products in blood cancer, Contreras-Galindo *et al.* (2008) performed RT-PCR studies on plasma samples taken from patients with diffuse large B-cell lymphoma as well as Hodgkin lymphoma. Amplification of viral *env* gene confirmed an elevated presence of HERV-K(HML-2) in lymphomas. Moreover, it showed that type 1 HERV-K(HML-2) is present in Hodgkin and non-Hodgkin lymphomas, whereas type 2 proviruses are present only in non-Hodgkin lymphoma. Sequencing of the RNA revealed that the transcripts come from at least 16 different type 1 polymorphic loci and 18 type 2 polymorphic loci. Presence of all, correctly processed viral proteins and reverse transcriptase suggested, that there are mature viral particles emerging from the samples (Simpson *et al.*, 1996). Successful treatment of the disease, where the lymphoma went into remission, was associated with HERV-K(HML-2) titers being reduced to undetectable levels, suggesting a strong correlation between malignancy and expression of polymorphic insertions. The research clearly shows strong association between HERV-K(HML-2) expression and development of lymphoma, characterizing its subtypes with different proviral loci. Total expression arrest following lymphoma remission strongly suggests a correlation between the disease and viral expression.

Depil *et al.* (2002) have shown some more precise evidence for the role of HERV-K(HML-2) in blood cancer. They confirmed expression of HERV-K10 *gag* via RT-PCR in PBMCs of 3 chronic lymphocytic leukaemia (CLL) patients, 2 acute myeloid leukaemia (AML) patients, 2 chronic myeloid leukaemia (CML) and 1 acute lymphoblastic leukaemia (ALL) patient, as well as 8 normal PBMC samples. However, the copy number in cancer patients were 5- to 8-fold higher in cancer samples, showing significant proviral overexpression in cancer cases. To test if the overexpression was reproducible in PBMCs *in vitro*, the authors induced cell division using phytohemagglutinin (PHA), gamma irradiation at 10 or 25 Gy rate and a demethylating substance, 5-azacytidine (AZC), to test if the expression is induced as a stress response (Urnovitz and Murphy, 1996). PHA did not induce any changes in HERV-K(HML-2) expression on healthy PBMCs, whereas other agents increased HERV-K expression by 2.4-fold, significantly less than in cancerous samples. Stressing agents, such as AZC can alter the patterns of CpG islands methylation. Gene silencing via CpG methylation is one of the major mechanisms the cells use to control HERV-K(HML-2) expression (as discussed in section 1.11). On the other hand, gamma-irradiation can induce general DNA mutational damage, especially frequent double-strand brakeage that can result in recombinations (Sikpi *et al.*, 1992). Transposable elements, such as HERV-K(HML-2) sequences, due to their high degree of similarity and relative abundance in the genome could be centers of such recombinations.

In the case of the *env* gene functional variants, especially the *np9,* there is more information about its role in cancer cells. The gene was found to be overexpressed in leukaemia (Fischer *et al.*, 2014) and has been proven to activate a number of signalling pathways in leukaemia cases, which gives

insight into the importance of HERV-K(HML-2) in cancers. Malignant hematopoietic cell lines, namely four CML (K562, K562/adr, KCL-22, KCL-22M), four AML (KG-1, HL-60, NB4, Kasumi-1), four ALL (Jurkat, Molt-4, H9, Raji) were tested by Chen *et al.* (2013) to asses the role of Np9 in cell growth of common leukaemia types. In order to do that, the authors incubated the cell lines with a cocktail of viral inducers to enhance transcription of putative retroviruses (5'-iodo-2' deoxyurine (IdUR), 5'-azacytidine, phytohaemagglutinin and phorbol myristate acetate). Four specific cell lines (K562, Kasumi-1, NB4 and Jurkat) were characterized with high levels of Np9 expression in western blotting. In these lines, markers for different cellular pathways were also overexpressed: K562 cells displayed expression of Notch1, ERK, myc/Akt and β-catenin markers, the Jurkat cell line was found to co-express Notch1, c-myc/Akt and β-catenin markers. Control cells from normal donors did not display overactivation of any pathways. Alterations in these pathways, especially β-catenin (Gandillet *et al.*, 2011), ERK (Steelman *et al.*, 2011), Akt (Sykes *et al.*, 2011) and Notch1 (Tatarek *et al.*, 2011) were found to promote malignant growth of leukaemia cells *in vitro* previously. To validate this, the Np9-negative Raji cell line was transfected with a lentivirus, inducing the *np9* expression. This upregulated β-catenin, phospho-ERK, c-MYC, cleaved Notch1 and increased cell proliferation rate. Silencing *np9* with shRNA in Jurkat leukaemia cells significantly reduced levels of the abovementioned markers. This complements the proviral overexpression observations, providing evidence for a very strong correlation between specific pathways that control cell cycle, proliferation and signalling functions. The pathways have been reported to be altered in numerous cases of cancer, including leukaemia, which is confirmed by the data presented by Chen *et al.* (2013). Their research associates HERV-K(HML-2) expression with many molecular mechanisms, that contribute to cancer development and progression.

The role of Np9 in leukaemia particularly, and its influence on various pathways, can be partly attributed to its interaction with Promyelocytic Leukaemia Zinc Finger Protein (PLZF). A number of techniques, including marking with fluorescent proteins, *in vitro* coimmunoprecipitation and reporter gene assays were used by Denne *et al.* (2007) to illustrate these interactions. The authors added glutatione-S-transferase (GST), glutatione-S-transferase fused with Np9 (GST-Np9), PLZF and 5' PLZF fragment (245/543 amino acids) in various combinations, labeled with 35S-translabeled methionine-cysteine onto GST antibody linked to Sepharose beads. The results were displayed on gel elecrophoresis and showed bands that corresponded to PLZF-Np9 complex. Moreover, expression of these proteins in fusion with fluorescent labels inside teratocarcinoma Tera-1 cell line enabled colocalization of the complex *in vitro* around the nucleus. Transfection of 293T cells with PLZF-responsive fragment in front of a luciferase gene showed that upon PZLF and Np9 overexpression the luciferase was suppresed, whereas PZLF alone produced a significant fluorescent signal. That showed the potential of Np9 interferering with PLZF function as a transcriptional repressor. Use of a U937 lymphoma cell line, harboring a PLZF gene under the

control of a Tet-responsive promoter enabled that the quantification of Rec protein expression, and showed that it is strictly associated with PLZF levels, and that PLZF directly represses c-Myc (McConnell *et al.*, 2003). The interaction between HERV-K(HML-2) *env* splicing product, Np9, illustrates the complexity of influences between the proviral expression and various biochemical pathways in cancer. The Np9 protein is able to interact with transcription factors. These include PLZF, that influences a number of important signalling pathways, including Notch1 (Tatarek *et al.*, 2011), c-myc/Akt (Sykes *et al.*, 2011) and β-catenin (Gandillet *et al.*, 2011), resulting in alterations in molecule levels, that transport signal in these pathways. Due to accumulation of such factors and cell cycle or metabolic changes, the healthy cells progress towards malignant transformation.

In leukaemia, HERV-K sequences can also contribute to disease development by interaction with other, similar retroviral sequences which transactivate HERV-K(HML-2) LTRs, subsequently influencing a range of other genes. Adult T-cell leukaemia is caused by human T-lymphotropic virus (HTLV), a deltaretrovirus similar to HIV (Kanki, Hopper and Essex, 1987). It occurs in exogenous and endogenous form, the latter is called HTLV-related endogenous sequence, or HRES-1. Like HIV, it attacks CD4+ T-cells, but rather than destroying them, it enables uncontrolled proliferation. HERV-K10 Env protein sequence is highly similar to human T cell lymphoma/leukaemia viruses 1 and 2 (HTLV-1 and 2), gp21 Env protein and HTLV p24 Gag in a specific, 8 amino acid sequence, that is recognized by the antibodies for HTLV proteins. HRES was also found to influence various neurodegenerative diseases (Gessain *et al.*, 1985; Kaplan *et al.*, 1990; Hjelle *et al.*, 1992). Perzova *et al.* (2015) analysed HRES from 100 random human samples, 53 large granular lymphocytic leukaemia patients, 16 HTLV patients with myelopathy (HTLVm), 58 HTLV patients who did not have myelopathy (HTLVn) and 83 multiple sclerosis patients. They sequenced proviral DNA from 4 HTLV patients and 4 HTLVn patients and found no differences in sequences for *gag* gene between HRES and HTLV. The authors used enzyme immunoassay to assess immune response against HRES-1 and found that whitin the HTLVm group, 13/16 patients were seropositive (81%), significantly more than HTLVn (27/58 ; 47%). For Large Granular Lymphocytic Leukaemia (LGLL) patients (7/53 ; 13%) or MS patients (1 / 83 ; 1%) the numbers were also significantly lower. Both HTLVn and HTLVm groups displayed seropositivity against peptides shared by HERV-K10 and HRES-1 sequences at much higher levels than the healthy population samples or LGLL or MS patient samples. Toufaily *et al*. (2011) have found, that HTLV-1 Tax protein transactivates HERV-K LTR controlled transcription. Therefore, it is possible that due to HTLV presence, the HERV-K LTR is subsequently activated and further activates other genes. The authors have used Jurkat, a human T lymphocyte line, co-transfected with the different pHERV-LTR-Luc constructs and Tax expression vector. Tax overexpression induced 4-5-fold increase in expression of different HERV-W loci, 7-fold of HERV-H and approximately 3-fold increase of HERV-K transcripts. The interaction between HERVs and Tax could contribute to development HTLV-related neurological diseases, such as myleopathy in leukaemia patients.

To sum up the involvement of HERV-K(HML-2) in leukaemia, research suggests this type of cancer could be significantly influenced by the presence of HERV-K(HML-2). Leukaemia cases are characterized by a specific sets of HERV-K being expressed and those have been proven to influence further transcription factors directly, like the Np9 protein associating with PZLF (Denne *et al.*, 2007). These factors could potentially disrupt a number of processes involved in the cell cycle, such as Notch1 (Tatarek *et al.*, 2011) or Akt (Sykes *et al.*, 2011) signalling. A variety of research shows expression of certain HERV-K(HML-2) insertions associating with specific leukaemia types, which could be used as biomarker in the future diagnostic methods. The interactions between HERV-K(HML-2) proviruses and a variety of other factors important in cancer, including HTLV retroviruses (Toufaily *et al.*, 2011) highlight the importance of HERV in a multitude of processes, very often acting as transcription initiators that activate many different secondary pathways.

1.16.vi. Other cancers.

There are a limited number of studies that indicate HERV-K(HML-2) association with other cancers, highlighted below.

Brain cancer. There is limited information of HERV-K(HML-2) activity in brain cancer cell lines, especially glioblastoma multiforme (GBM) and other astrocytic brain tumors. Flockerzi *et al.* (2002) has done a genome-wide analysis of HERV-K(HML-2) expression on brain cancer cell lines of common-type meningioma grade I, meningioma grade II, atypical meningioma grade I, atypical meningioma grade III and glioblastoma multiforme. They tested 23 different post-mortem brain tissue specimens. The RNA was isolated and analyzed via RT-PCR and cDNA sequencing. On average 6.5 viral loci was overexpressed per tumor sample, while 7.4 was expressed for non-cancerous cell lines. 8 loci were found for cancer-free, schizophrenia patient tissue and 6.8 for cancer-free, bipolar syndrome patient sample.

Testicular cancer. Proviral expression is associated with testicular tumors, including seminomas and mixed germ cell tumors (GCT). Kleiman *et al.* (2004) used an immunofluorescent assay to detect antibodies against HERV-K Gag and Env in GCT samples. In a tested group of 310 GCT patients 209 (67%) were positive either for HERV-K Gag or Env. Moreover, serological response against viral antigens were highest in seminoma samples (77%) and mixed GCT with elements of seminoma (81%), which exceeded the expression of classical GCT markers like human chorionic gonadotropin (hCG). In patients undergoing cisplatin chemotherapy that went into remission, 94/145 displayed at least 2-fold decrease in HERV-K Gag/Env antibody titers. 42 patients did not show any response to treatment, which was accompanied by stable or increased antibody levels for HERV-K Gag/Env. In 9 patients, the antibody levels did not correlate with disease progression.

Lung cancer. HERV-K Env protein has been recently suggested to be an effective lung cancer biomarker. Zare *et al.* (2017) have indicated that ERVK-21 derived *env* mRNA can be detected at 10-fold higher rate in blood of patients diagnosed with lung cancer, including adenocarcinoma, squamous cell carcinoma and small-cell lung cancer, compared to healthy controls. They also analyzed the presence of HERV-P, -H and -R in blood, which were overexpressed at approximately 5-fold higher rate than in healthy controls.

1.16.vii. State of HERV-K(HML-2) in cancer research.

The presented examples provide evidence for HERV-K(HML-2) expression being altered in different types of diseases, including cancer (see sections 1.16.i-1.16.vi). HERV-K(HML-2) are directly interacting with cellular pathways involved in cancer progression; further the fact that there is a group of insertionally polymorphic HERV-K(HML-2) viruses suggests a hypothesis that HERV-K(HML-2) might be directly influencing cancer. Presence of particular polymorphic insertions might make certain individuals that carry them more susceptible to cancer, if these insertions are involved in development of the disease. Recombinational capacity of HERV-K(HML-2) and presence of full-length proviruses suggest that even though the virus is not infectious it might contribute to recombinations, especially in cancer cells. Viral sequence similarity may also cause transposition of longer stretches of DNA during recombination events, that could destabilize the entire genome and influence many different pathways due to structural reorganization of the nucleic acid. In malignant tissue, activated HERV-K LTRs might influence nearby oncogene expression by recruiting additional transcription factors. On the other hand, overexpression of HERV-K(HML-2) could serve as a cancer development marker and if it is directly influencing oncogene expression, then there is a possibility of treating cancer by repressing HERV-K(HML-2). Therefore, a better understanding of HERV-K(HML-2) role in cancer, particularly focused on polymorphic integrations in specific cancer cases might provide vital new insight and possibly influence cancer diagnostics or treatment.

1.17. State of HERV-K(HML-2) in the human genome.

The study by Subramanian *et al.* (2011) was conducted on the GRCh37/hg19, February 2009 (Church *et al.*, 2011) build of the human reference genome (and unplaced contigs) and identified 89 proviruses and 947 solo LTRs. Subramanian *et al.* (2011) have collated a large amount of information about HERV-K(HML-2) elements in the reference genome known at the time of writing, as well as provided a first comprehensive database of known elements and its corresponding locations in the reference genome. Using sequence of one of the youngest HERV-K(HML-2) members, HERV-K113, which appears to have its LTR sequence free from mutations (Turner *et al.*, 2001), as query Subramanian *et al.* (2011) ran BLAT to mine the GRCh37/hg19 human reference genome for the existing proviruses.

Furthermore, using the K113 internal gene sequence, Subramanian *et al.* (2011) looked for HERV-K(HML-2) internal genes adjacent to detected LTRs, to map all of the full-length or partial proviruses. Solo LTR structures were identified using UCSC Genome Browser RepeatMasker algorithm, which compared the sequences in the reference genome with all HML-2 LTRs in the RepBase repetitive element database. In total, the authors detected 87 full-length insertions present in the reference genome and 946 solo-LTRs. Additionally, they have reported two youngest and least mutated polymorphic elements not present in the reference genome (HERV-K113 and

HERV-K115), previously described by Turner *et al.* (2001). Based on the LTR sequence, the authors have provided updated classification of the HERV-K(HML-2) family into two types of proviruses, according to Löwer *et al*. (1993) classification. The viruses were subdivided into LTR5A, LTR5B and LTR5Hs phylogenetic subgroups (Macfarlane and Simmonds, 2004). For each subgroup, Subramanian *et al.* (2011) performed phylogenetic analysis and proposed updated classification, correcting a few inconsistencies in assignment proposed in the RepBase. To confirm the correct assignment of detected elements into HML-2 subfamily of HERV-K, the authors also performed phylogenetic analysis based on sequences of internal genes. Using method previously described by Johnson and Coffin (1999), based on comparing the number of mutations between 5' and 3' LTRs in the same insertion and assuming fixed mutation rate in the genome, the authors provided estimate age of all detected proviruses.

However, since the study by Subramanian *et al.* in 2011, many other HERV-K(HML-2) insertions have been identified from numerous other studies, that are not present in the reference genome. Additionally, the reference human genome has also been updated (current build GRCh38.p13), and the co-ordinates provided Subramanian *et al.* (2011) do not lift over accurately to the new genome build. To allow comparison to the numerous human genomic datasets that are recently emerging, this database of HERV-K insertions needed updating.

Overall, these studies raise some interesting questions with respect to the impact of HERV-K(HML-2) on the human genome. Recent advances in sequencing technologies enable more rigorous analyses of the state of the HERV-K(HML-2) family, specifically:

- Identification and characterization of the baseline state of known insertions in recent reference genomes (NCBI36, GRCh37 and GRCh38).
- To ascertain relationships between different HERV-K(HML-2) loci, and their potential impact on genome evolution, specifically with respect to:
    i. Segmental duplications
    ii. Recombination between different HERV-K loci. A recombination event would be indicated by two loci sharing non-matching TSDs in a cross-pattern (see figure 8).

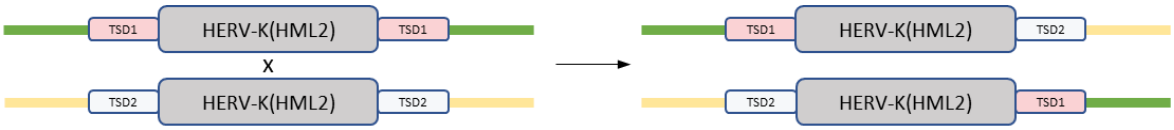

Figure 8 A representation of HERV-K(HML-2) insertions, that share their TSD sequences in a cross-pattern, which would suggest a recombination event happened between them.

1.18. Summary.

The data presented in the literature regarding HERV-K(HML-2) clearly provides evidence for possible role of HERV-K(HML-2) family of proviruses in various diseases, including different types of cancer (reviewed in 1.16). However, many of the studies report a variety of information, especially regarding unfixed, polymorphic insertions found in different populations (Wildschutte *et al.*, 2016; Xue *et al.*, 2020). Moreover, the latest overall report on HERV-K(HML-2) insertions present in the reference genome, performed by Subramanian *et al.* (2011), was done on GRCh37 version of the human reference genome, which has been updated to GRCh38.p13 (October 2019) as of now. In addition to the work done by Subramanian *et al.* (2011), a number of other reports on HERV-K(HML-2) in the human genome were published since. Recent studies include work done by Zahn *et al.* (2015), who have identified two particular HERV-K loci that have proliferated through the centromeric and pericentromeric regions of the human genome since the original 2011 Subramanian paper. Sequencing of archaic hominids (Prüfer *et al.*, 2014; Meyer *et al.*, 2012) has also revealed several other insertionally polymorphic loci, some of which have been confirmed in modern day humans (Lee *et al.*, 2014; Marchi *et al.*, 2013). Lately, advances in sequence technologies provided massive improvements in build quality of the latest version of the human genome. Therefore, it is timely to revisit and collate the information on HERV-K(HML-2) in the human reference genome and modern populations. Further, it will serve as a baseline for studying the state of HERV-K(HML-2) in cancer, especially regarding presence of polymorphic HERV-K(HML-2) insertions and their influence on the disease.

Project aims

- Review the current HERV-K(HML-2) literature, with particular focus on its relation to various types of cancer.
- Identify and characterize the baseline state of known insertions in recent reference genomes (NCBI36, GRCh37 and GRCh38).
- Ascertain similarities and potential relationships between existing insertions to determine the impact on genome evolution.
- Identify novel HERV-K(HML-2) insertions in cancer samples and establish their relations between them and elements known previously, especially regarding segmental duplications.
- Detect if non-reference HERV-K(HML-2) insertions that are present both in cancer data samples and corresponding controls or cancer samples only.
- Explore the possibility of any known or novel insertion directly influencing cancer progression.

Objectives

- Collate information found on HERV-K(HML-2) insertions in the literature.
- Catalogue TSD sequences for HERV-K insertions present in the reference genome, as well as ascertain locations across NCBI36, GRCh37 and GRCh38 human reference genome versions.
- Compare TSD sequences between insertions to estimate the levels of recombination within the HERV-K(HML-2) family.
- Analyse cancer samples, focusing on unmapped contigs, which might contain data on possible novel insertions related to particular cancer.
- Assess polymorphic states within novel insertions, especially if the insertion is present in full-length or solo LTR form.
- Perform TSD and statistical analysis on novel insertions to establish relation between them and previously described ones.

2. Chapter 2: Methodology.

2.1. Introduction.

The methodology presented in subsequent paragraphs describes the procedures undertaken during the study. Initial approach to the literature review, described in section 2.2, revealed a lot of studies and information about HERV-K(HML-2), which needed to be systematically categorized. There were some discrepancies among the different studies regarding some of the known insertions (as explained in section 2.3.i), so the analysis started with identification of all of the HERV-K(HML-2) proviral elements present in the reference genome. The literature data was used as reference to extract all known sequences of HERV-K(HML-2) elements present in the human reference genome and obtained sequences were realigned to ascertain accurate genomic locations (for details, please see sections 2.3.i-2.3.ii). In order to ascertain relations between reported insertions, immediate flanking sequences were used to detect duplicated sequences. Furthermore, TSDs were reported for all insertions and similarity analysis was used to estimate the possibility of recombination between different HERV-K(HML-2) elements (section 2.3.iv-2.3.v). This has provided an exhaustive dataset of the HERV-K(HML-2) state in the human genome, which served as a baseline for the cancer analysis.

The second part of the project focused on analysing a number of cancer genomic datasets, coming from The Cancer Genome Atlas (TCGA). As discussed in section 2.4.ii, the technical difficulties regarding sequencing of transposable elements made analysis of known insertions unreliable in the cancer data. Therefore, analysis focused on extracting unmapped genomic reads and detecting non-reference insertions present in a variety of different cancers. Detected reads were aligned against HERV-K113 reference genome and results indicating presence of an insertion were reported. Accurate genomic locations were produced using sequences flanking the detected inserts, which mapped to the human reference genome. In order to describe the possible relation to cancer, the vicinity of detected locations for non-reference proviral insertions were analysed for presence of genes active in cancer, using various functional genomic databases. The genes were then assigned into molecular pathways that they are active in, focusing on the ones known to be disrupted in cancer. This provided *in silico* evidence for the possibility of detected insertions being associated with particular cancers, which would require conformation in further studies regarding HERV-K(HML-2) activity in cancer on a molecular level.

2.2. Chapter 1 - Literature review.

The methodology used in the analysis is summarised in Figure 12 and further explained below. In order to assess the impact of HERV-K(HML-2) insertions in cancer genomes, it is first necessary to ascertain the impact of HERV-K(HML-2) insertions on non-cancer genomes. Information on HERV-K(HML-2) insertions existing in Hg37 and Hg38 human reference genome (Lander *et al.*, 2001) was collected from the literature. Searching for publications on the topic of HERV-K and cancer was conducted by using NCBI databases – PubMed and PubMed Central (Bethesda(MD), 1988). The main search terms used to research the literature were:

HERV*K* AND reference*

HERV*K* AND human*

HERV*K* AND cancer*

HERV*K* AND review* AND cancer*

HERV*K* AND polymorph*

Specific terms were also used in the search when referring to individual publications, insertions or types of cancer, for example a search for publications referring to HERV-K research in melanoma:

HERV*K* AND melano*

The search was also expanded by citation chaining (reference mining), using relevant bibliographic references and following the provided DOI numbers to obtain manuscripts, or searching for specific article titles using PubMed/PubMed Central as described above.

The majority of the publications are from 2001-present, with few older references where specific information was looked up in source articles that first mention it.

2.3. Chapter 3.1 – Reference genome analysis.

2.3.i. Querying the reference genome.

The information obtained from the literature served as a baseline for creating the database of known HERV-K(HML-2) insertions. The majority of insertions were obtained from Subramanian *et al.* (2011). However, the co-ordinates provided by Subramanian *et al.* (2011) were found to be inaccurate (see chapter 1.17 for explanation), hence the first step was to obtain accurate co-ordinates. For these known insertions, 20bp fragments of the 5' and 3' ends of the virus along with 20bp of flanking sequence from each end (40bp total from each end of the insertion) were extracted from GRCh38/Hg38. The extraction was performed using the *extractfromref_optimal.py* python script (appendix 1.i), and sequences were aligned manually using Geneious software using HERV-K113 (AY037928) sequence as reference.

Some insertions did not display similarity within the ends of the HERV sequence, suggesting that they may be truncated. For these insertions, 10kb of flanking sequence was extracted and using BLAST, 5' and 3' ends of these particular HERV insertions were examined, using HERV-K113 (AY037928) sequence as reference (program: blastn, evalue=0.05, word_size=11). This allowed to:

1. identify accurate start/end co-ordinates for each HERV-K(HML-2) insertion (necessary for the following TSD analysis).
2. identify exactly which insertions were truncated.
3. Identify which insertions were solo-LTRs and which insertions were full-length elements.
4. locate the detected sequences in GRCh36/Hg18, GRCh37/Hg19, GRCh37/Hg38 versions of the human reference genome (using BLAT software) and record their precise coordinates (necessary for analysis of cancer genome sequences later).

Subramanian *et al.* (2011) used strict criteria for categorizing HERV-K(HML-2) insertions as full-length proviruses. Their minimum criterion for a provirus were:

1. the presence of an LTR and a "hit" matching > 50% of the length of a full gene, OR
2. two proximal genes with > 50% hits and no LTR.

For the remaining elements, Subramanian *et al.* (2011) used an arbitrary cut-off of 750 bp of LTR sequence to designate solo-LTRs. The issue with this approach was that many truncated proviruses that did not meet their criteria of a FL provirus were designated as solo LTR's. For the purposes of this study, TSD surrounding each FL and solo LTR insertion was identified. As illustrated in figure 9 below, some of the truncated proviruses that Subramanian *et al.* (2011) classified as solo LTRs would have given incorrect information with respect to their TSD.

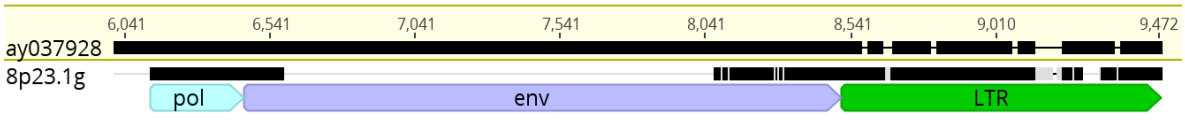

Figure 9 Example of solo-8p23.1g insertion, designated as solo LTR (marked green), which contained fragments of internal HERV-K(HML-2) sequence in the immediate vicinity (*pol* and *env* genes, blue and violet respectively).

In order to correctly identify the TSDs, it was necessary to reclassify truncated full-length elements, that did not meet Subramanian *et al.* (2011) criteria for full-length elements. For the purposes of this study, loci were classified as full-length (or rather, truncated full length elements) if they contained 20bp of sequence of any internal gene in the immediate 500bp flanking sequence.

2.3.ii. Identifying TSDs.

As discussed in section 1.2, every full-length or solo LTR insertion is flanked by identical 4-6bp sequences on both 5' and 3' ends, which are created during the insertion process by duplicating 4-6bp of the insertion site. Briefly, the integrase enzyme creates staggered cut at the target site producing sticky end, which is filled by the polymerase enzyme, creating 4-6bp of repeated sequence at both ends of the virus. These sequences are called terminal site duplications (TSDs) and can provide information on recombination/duplication of viral sequences in the genome, as one can suspect that two different insertions with identical TSDs may be a result of a duplication, or if the flanking sequences don't match the provirus has probably underwent recombination in the past (Kahyo *et al.*, 2017). The idea of investigating HERV-K(HML-2) insertions by comparing TSDs has been undertaken previously, for example by Contreras-Galindo *et al.* (2013), where they focused on one particular locus (K111), located within the centromeric regions of human chromosomes. However, this approach can also be used on a genome-wide scale to investigate recombination between different HERV-K(HML-2) loci. Briefly, there are two approaches:

1. Multiple loci that share identical TSDs may have proliferated through the genome by segmental duplication. By comparing the flanking sequences of HERV-K(HML-2) insertions that share identical TSDs, we can investigate this question.
2. The TSDs are identical at the point of integration and therefore loci, that do not display identical TSDs must have undergone recombination with another copy in the genome.

Using the 40bp alignments from the 5' and 3' ends created in 2.3.i, the TSD presence on each side of the insertion was determined (Figure 10). All insertions were categorised into five groups:

- inserts with matching TSDs and complete 5' and 3' end of the virus.
- inserts with complete 5' and 3' end of the virus but with no matching TSDs.
- 5' truncated insertions.
- 3' truncated insertions.
- insertions truncated on both ends.

For the 5' and 3' truncated insertions and insertions with non-matching TSDs on each end it was not possible to define the TSD. In these instances, TSD search focused on 4-10bp of sequence flanking the recognizable end(s). The bases were compared to TSDs confirmed for other insertions so it is plausible to say that such insertion had a TSD matching a less mutated insertion (at the time it was full-length) and its existence can be attributed to recombination activity of HERV-K. To account for possible mutations a single nucleotide difference between the 5' and the 3' flanking sequences was considered (*tsd_wooble.py* python script; Figure 13, step 2; appendix 1.ii). Similar function was implemented in the last step of the analysis (*tsd.py* python script; Figure 13, step 4; appendix 1.iii), that accounted for such mutation between TSDs/flanking sequences between compared pairs of insertions. In every comparison, only a single mutation was allowed.

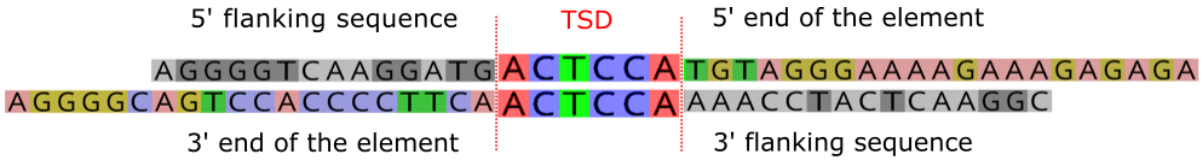

Figure 10 Fragments of flanking sequence and the Terminal Site Duplication (TSD) of HML-2.LTR4

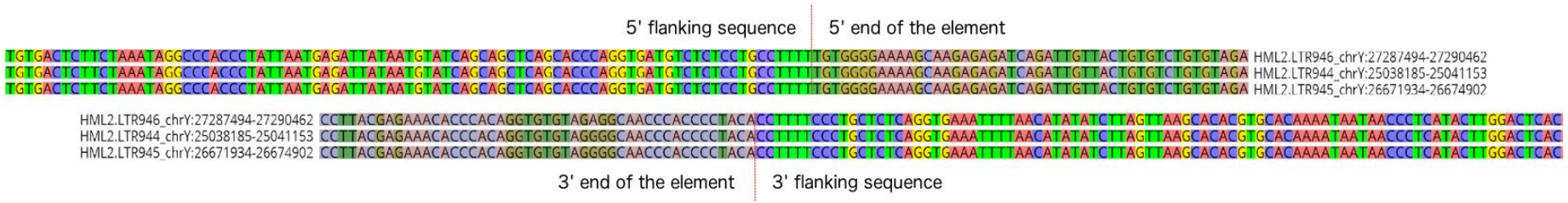

Figure 11 Segmentally duplicated sequences of HML-2.LTR944, HML-2.LTR945, HML-2.LTR946. Extended flanking sequences are nearly identical.

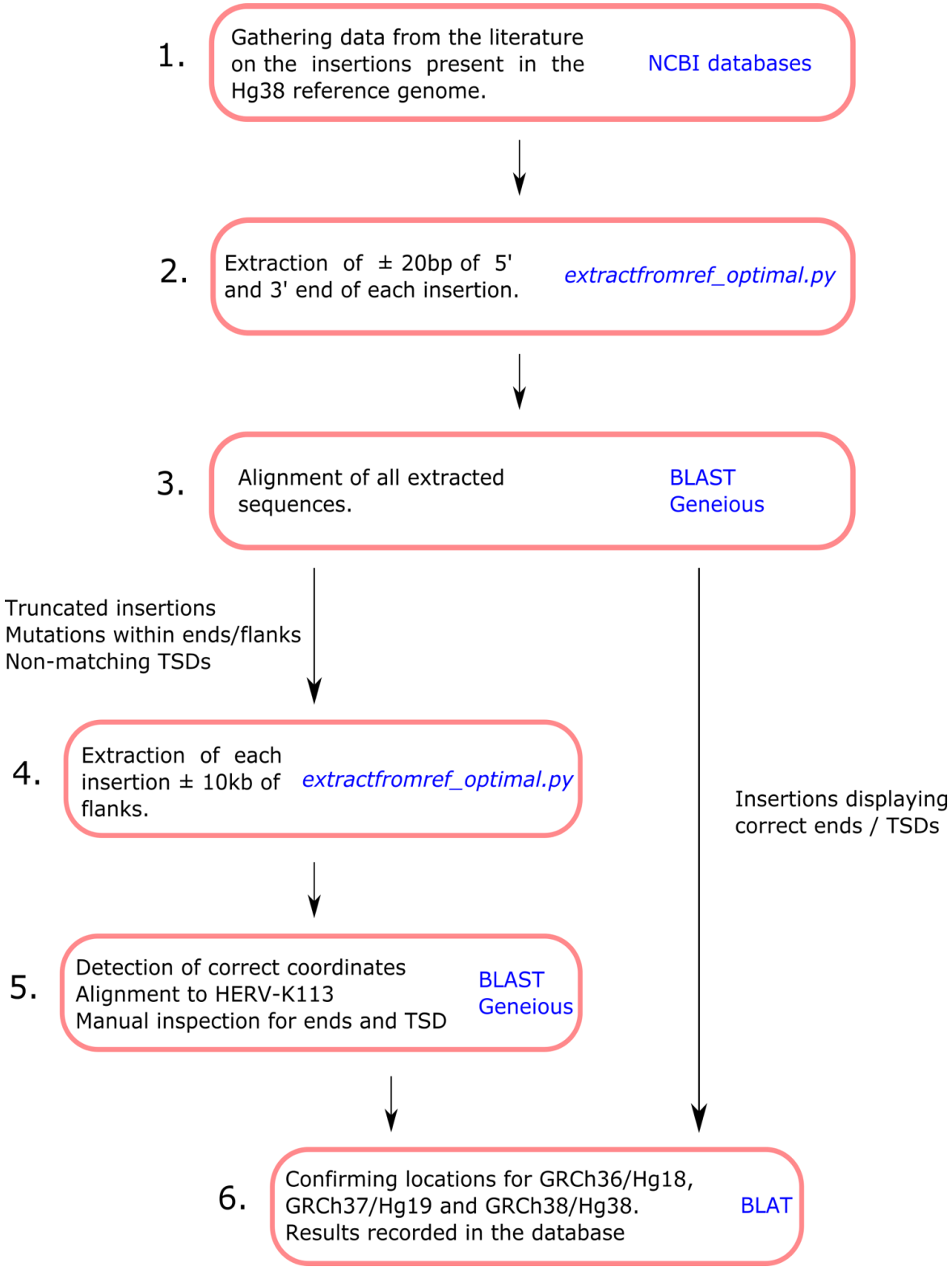

Figure 12 Reference Genome querying process. Explanation of each step can be found in the main body of text. *Blue text represents names of scripts and tools used in each step.

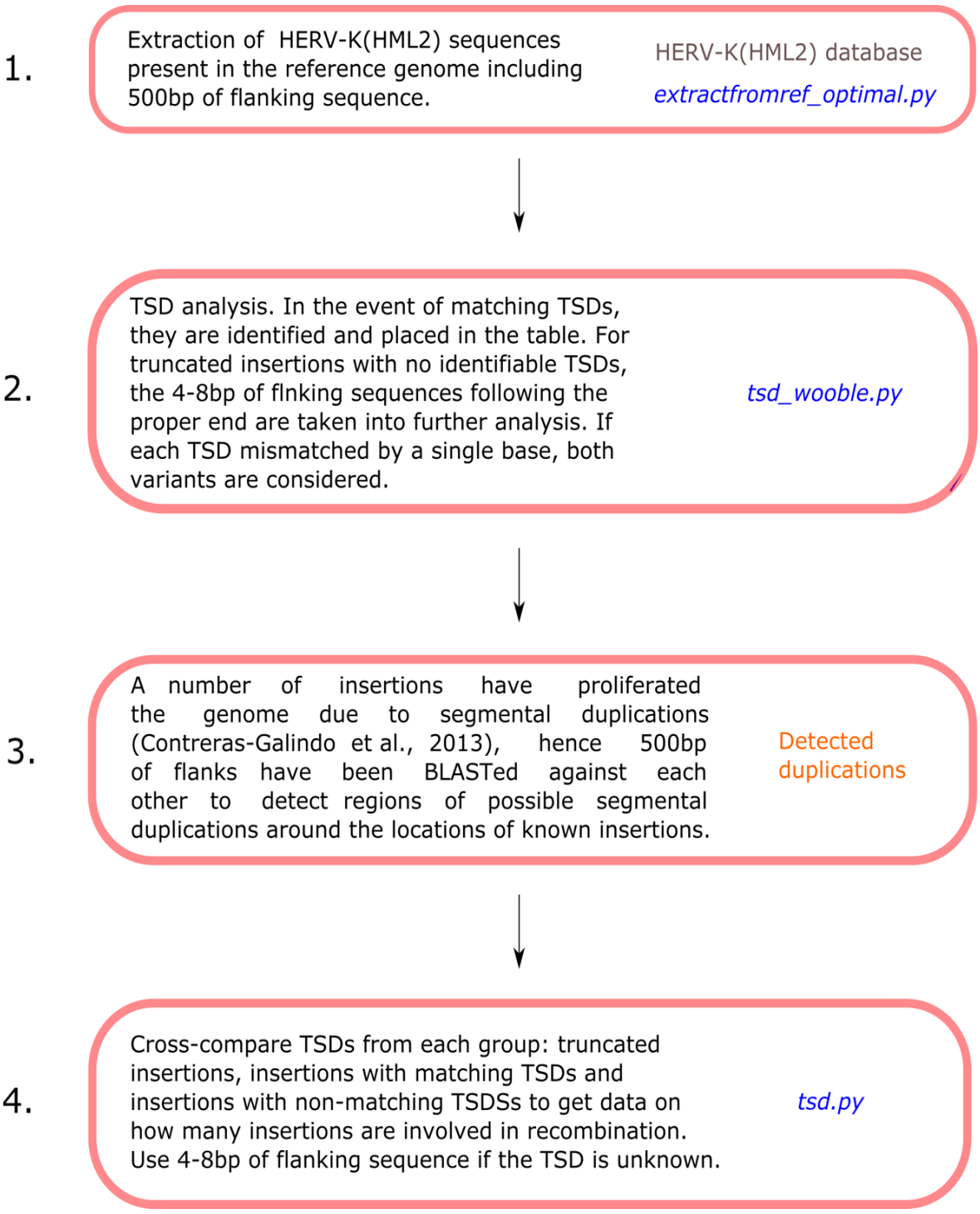

Figure 13 Step-by-step TSD analysis process. Explanation of each step can be found in the main body of text. *Blue text represents names of scripts and tools used in each step. **Orange text indicates a list of duplications detected according to methodology described in chapter 2.3.iii.

2.3.iii. Detection of segmental duplications - methodology.

A number of HERV-K(HML-2) insertions have been identified in the human reference genome to be the result of segmental duplications that have occurred in the genome. A study by Contreras-Galindo *et al.* (2013) identified a specific insertion that had proliferated through the centromeric regions of the human genome as a result of segmental duplications. However, the extent to which HERV-K(HML-2) insertions have spread through the remainder of the human genome in this manner (via segmental duplications) is unknown. To ascertain the contribution of segmental duplications to the proliferation of HERV-K(HML-2) insertions through the human genome, the flanking sequences adjacent to each HERV-K(HML-2) insertion were examined in order to identify segmentally duplicated loci.

For each HERV-K(HML-2) insertion, 500bp of flanking sequence was extracted from the 5' end and the 3' end of each insertion. All of the 5' flanks were compared to each other using BLAST to identify segmental duplications; a BLAST database was created from the 5' flank file, and each sequence in this file was searched for in this database (see table 2 for parameters used). The same search was performed on the 3' flanks.

Table 2 Parameters used in the BLAST search to identify segmentally duplicated regions.

Blast command used.	makeblastdb -in flank.fasta -dbtype nucl blastn -word_size 11 -evalue 0.001 -db flank.fasta -query flank.fasta -outfmt 6 -max_target_seqs 500000 -out output.txt
Short description for the used parameters.	-word_size 11: default for blastn according to online manual -evalue 0.001: gathers results that provide significant alignments -outfmt 6: tabular format with results used in analysis -max_target_seqs 500000: parameter set to ensure all the significant alignments are taken into consideration.

Examined sequences included 500bp of flanking sequence (Figure 14) on each side of the virus (Figure 13, step 3). The viral flanking sequences as well as the virus sequences have been divided into separate *fasta* files using *splitends.py* (appendix 1.iv) Python script. The output was filtered to remove duplicate entries (Figure 14, step 1, *cleanup.py,* appendix 1.v). The results coming from the same query-subject sequence pairs (for example, fragmented BLAST hits along the 500bp) were grouped together (*group.py*, Figure 14, step 2; appendix 1.vi). In some instances, the BLAST result identified similarities within the flanking sequences due to the presence of other transposable elements in the human genome (e.g. *Alu*, SINE or LINE elements), which were not informative to the question of larger segmental duplications. For this reason, essential criteria for identification of a "true" segmental duplication were specified as the presence of identical TSDs within the two sequences being compared (allowing for 1bp mutation). Therefore, the output was selected for results, that overlapped the 10bp range immediately adjacent to the target insertion location using

*selector.*py script (Figure 14, step 3; appendix 1.vii). The results displaying similar TSDs/immediate flanking sequences were selected using *comparetsd_segdupv2.py* script (Figure 14, step 4; appendix 1.viii). Separate 5' and 3' duplications have been cross-referenced using *connectends.py* script (Figure 14, step 5; appendix 1.ix). Subsequently, *getduplist.py* produced a list of such segmental duplication groups automatically (Figure 14, step 6; appendix 1.x). The remaining BLAST results revealed that many of the query-subject pairs indicated sequence similarity only on a single flank of particular HERV-K(HML-2) locus. To extract such results, *getoneside.py* script was used (Figure 14, step 7; appendix 1.xi) and further manual investigation revealed a small number of duplications, that were not picked up automatically by BLAST. These included some of the truncated insertions, where TSD analysis was not possible and some insertions displaying point mutations within their immediate flanking sequences.

2.3.iv. Detecting recombination events – TSD comparison methodology.

Insertions were divided into those that have matching TSD sequence following 5' and 3' ends, truncated loci and loci with mismatching flanking sequences. During TSD recognition, a single base-pair mismatch between compared TSDs has been allowed to account for possible random mutation during the insertion process and both variants of sequences have been considered during the analysis. To process every insertion and find the possibility of mutation a *tsd_wooble.py* Python script was used (Figure 13, step 2 appendix 1.ii). TSD comparison was ran against known TSDs collected from individual insertions that had distinct, matching 4-8bp flanking sequences on each end. For each of the other group, TSD was searched for on a recognisable end, which includes untruncated 5' or 3' end of the virus to avoid the possibility of matching real TSD with random sequences flanking the truncated insertion. The 4-8bp were also compared against each other for 5' and/or 3' truncated viruses to see possible recombination events that may have led to recombination of the virus and/or its region in the genome before the truncation occurred. Individual insertions that paired to any other insertion in the genome was counted into possible scenarios of not exhibiting any activity, being involved in a segmental duplication or being involved into recombination, which was done using *tsd.py* Python script (Figure 13, step 4; appendix 1.iii). Individual insertions may be involved in recombination and segmental duplications at the same time, so the results were carefully categorised and made sure that the inserts are not double counted. Finally, resulting element pairs were inspected for combinations, where single, longest exact TSD match could be selected out of all detected possibilities. In case of no exact matches, single longest match was permitted, that allowed for one, point mutation within the corresponding TSDs.

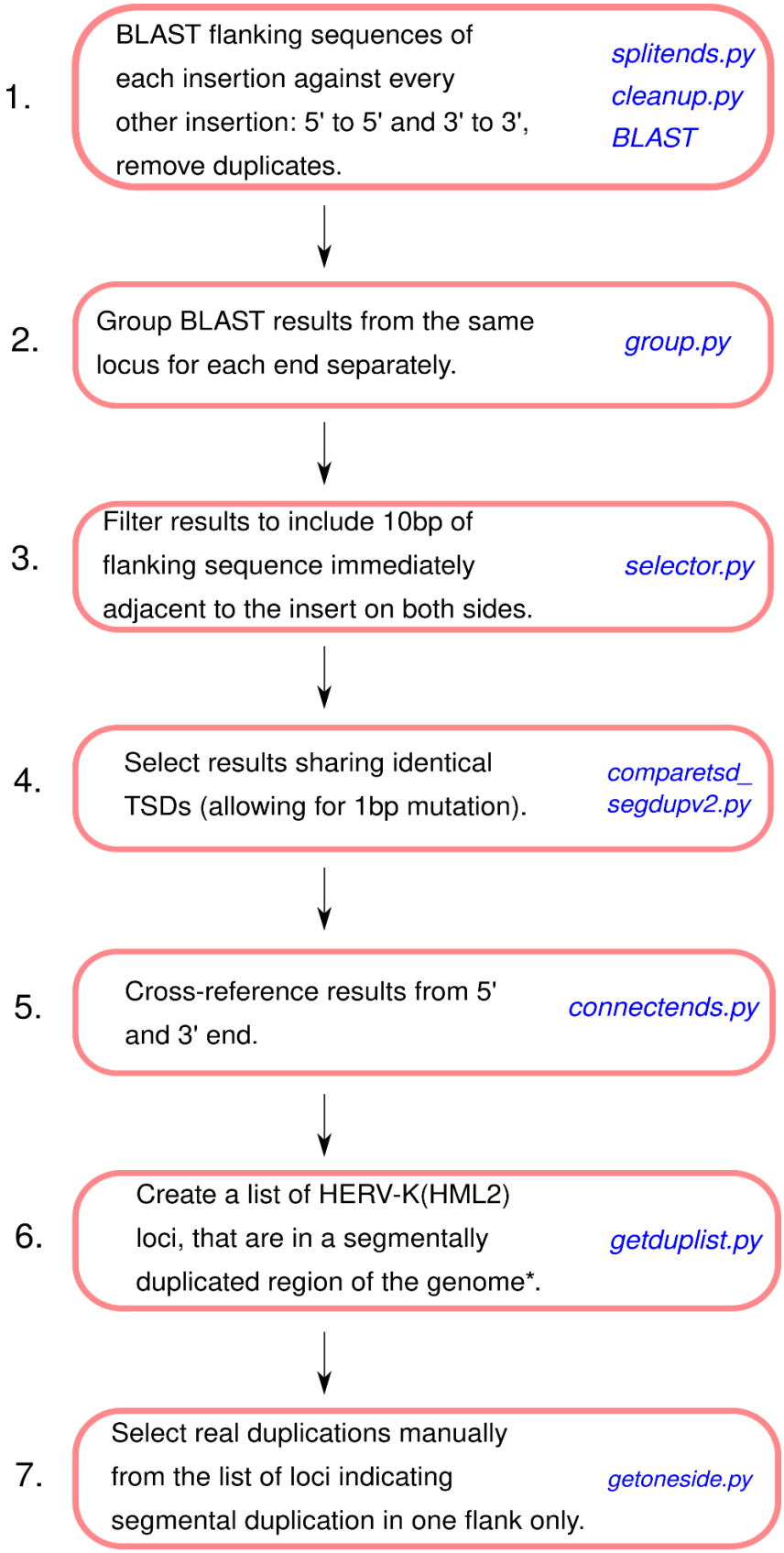

Figure 14 Step by step segmental duplication analysis process. *5' flanking sequence of different loci are identical and the 3' flanking sequence of different loci are identical. **blue text indicates script names.

2.3.v. The rate of TSD exchange in the reference genome - statistical analysis.

In order to assess the distribution of TSD exchange events, especially with regard to the master gene hypothesis (see section 1.14), the distribution of insertions selected in section 3.1.iii was analysed using Kolmogorov-Smirnov Goodness-of-Fit Test against the distribution of all known insertions in the reference genome. The elements selected in section 3.1.iii only included pairs of non-matching TSDs with matching TSDs (see section 3.1.iii for details). In such combinations we can suspect the matching TSD-insertion to be the donor and the non-matching TSD-insertion to be the acceptor element in the recombination/gene conversion event. The Kolmogorov-Smirnov D-statistic was calculated as the maximum vertical distance between the empirical cumulative distribution of observed donor insertions and the cumulative distribution of all HERV-K(HML-2) loci (n=1118):

$$D_{1118} = max(F(x) - G(x))$$

Where F(x) is the distribution of insertions selected as possibly recombining / undergoing gene conversions and G(x) is the distribution of all HERV-K(HML-2) elements in the reference genome. "x" represents chromosomes.

The critical value for the Kolmogorov's D statistic was calculated according to the formula:

$$D_{crit} = KSINV(\alpha, n, 1118)$$

Where KSINV is the inverse of the Kolmogorov distribution at α=0.05, for n selected insertions and 1118 total identified HERV-K(HML-2) elements. The KSINV is a Microsoft Excel function, contained within Real Statistics Resource Pack (https://www.real-statistics.com/).

2.3.vi. Finding the rate of HERV-K(HML-2) insertion in humans.

The average insertion rate of HERV-K(HML-2) was calculated, following the method used by Belshaw *et al.* (2005). Briefly, the number of human-specific insertions was divided by the average number of human generations since the divergence of chimpanzees and humans 6.7 million years ago (Kumar *et al.*, 2017), assuming a human generation time of 20 years (Chen and Li, 2001; Belshaw *et al.*, 2005). The number of human-specific insertions was determined by searching for sequences of HERV-K(HML-2) elements, including 500bp of flanking sequences extracted from the human reference genome (see section 2.3.i for details), using UCSC BLAT software, in the chimpanzee reference genome, PanTro6.

2.4. Chapter 3.2 – Cancer data analysis.

2.4.i. Cancer data selection.

In light of the HERV-K(HML-2) activity in cancer reviewed in section 1.16, a number of cancers were selected for data analysis, as listed in table 3. The cancer data were taken from The Cancer Genome Atlas project (*TCGA*, 2021), which collates sequences of over 20,000 primary cancer samples from 33 cancer types, including 10 rare cancers. The program includes genomic and transcriptomic sequence data, that has been collected from over 11,000 patients for 12 years between 2005-2018. All of the samples in the TCGA project are selected based on the poor prognosis of a particular case, importance of the cancer type in populations, availability of patient consent and paired normal tissue samples. All meet technical criteria for tissue size and amounts of tumor cells. These are specifically defined for different types of cancer, for example tissue containing more than 50% of necrosis were determined inadequate. The samples included were at least 200 mg in weight, with no less than 80% of tumor nuclei, which was further reduced to 60% as the sequencing technology improved. Samples were characterized by DNA sequencing, gene expression profiling, copy number variation profiling, SNP analysis, methylation profiling and exon sequencing. Approximately 10% of the samples had undergone whole genome sequencing. The samples selected for the analysis in this project have all had whole-genome DNA sequenced, with corresponding controls available.

The samples selected for the study included breast, melanoma and prostate (2 projects) cancers, which have been previously studied widely regarding HERV-K(HML-2) presence in the literature (reviewed in sections 1.16.ii-1.16.iv). Samples of other malignant tumors have been also selected for analysis from the data stored in TCGA, including pancreas, liver, cervix, small bowel cancer and myeloma. This enabled a study of HERV-K(HML-2) prevalence in other cancers, less reviewed in the literature regarding HERV-K(HML-2). Finally, a study of mixed samples (see Table 3) was included to increase the variety of analysed data.

The researched projects were mainly consisting of cancer single-cell NGS data, together with control samples coming from adjacent healthy tissue/blood from the same patient. As mentioned in Table 3, most of the analysed projects contain data collected in this fashion. Only “Breakpoint detection using long insert whole genome sequencing”, “Sequence Analysis of Mutations and Translocations Across Breast Cancer Subtypes” and “Melanoma Genome Sequencing Project” contained data organized in a different pattern. The “Breakpoint detection using long insert whole genome sequencing” contains sequencing data from 3 patients, each had a tumor sample, adjacent healthy tissue sample and a blood sample sequenced. “Sequence Analysis of Mutations and Translocations Across Breast Cancer Subtypes” analyses 10 subjects, where one of the subjects had 11 tumor samples and 11 adjacent control samples sequenced. The metadata for each of the analysed projects is summarized below. “Melanoma Genome Sequencing Project” contained 25

samples of matched melanoma tumors and controls from normal cells of the same individuals, but only 8 of the analysed tumor samples contained non-reference sequence data, which was suitable for this study

Table 3 Samples selected for analysis from The Cancer Genome Atlas.

Project	Cancer(s)	No of samples (tumor / control)
Characterization of complex chromosomal aberrations in primary prostate cancer genomes (Berger *et al.*, 2011)	Prostate	7/7
Melanoma Genome Sequencing Project (Berger *et al.*, 2012)	Melanoma	8/25
Prostate Cancer Genome Sequencing Project (Armenia *et al.*, 2018)	Prostate	53/53
Genome-Wide Characterization of Pancreatic Adenocarcinoma Patients Using Next Generation Sequencing (Liang *et al.*, 2012)	Pancreas	1/1
Breakpoint detection using long insert whole genome sequencing (Liang *et al.*, 2014)	Liver	30/30
Small Bowel Neuroendocrine Tumors (Carcinoid Tumors) (Francis *et al.*, 2013)	Small bowel	12/12
A pilot study using next generation sequencing in advanced cancers: feasibility and challenges (Weiss *et al.*, 2013)	Pancreas, lung, kidney, ureterus bronchus, basal cell cancer	9/9
Genomic Sequencing of Cervical Cancers (dbGaP accession:phs000600.v1.p1)	Cervix	14/14
Genomics of Hepatocellular Carcinoma (Griffith *et al.*, 2016)	Liver	30/30
Sequence Analysis of Mutations and Translocations Across Breast Cancer Subtypes (Banerji *et al.*, 2012)	Breast	20/20
Towards a Genomic Understanding of Myeloma (dbGaP accession: phs000348.v1.p1)	Myeloma	25/25

Selected project metadata is presented below.

- Characterization of complex chromosomal aberrations in primary prostate cancer genomes

All of the 7 tested individuals were male. The age ranged from 57 to 69. All of the individuals suffered from a primary prostate cancer, with 4 samples characterized at stage T2c, 1 sample at stage T3a and 2 samples at stage T3b.

- Melanoma Genome Sequencing Project

The project tested genomic data from 122 subjects. Out of these, only 25 male patient samples were available as full genomic sequences. 25 control samples included non-reference (unmapped) genomic data, suitable for non-reference HERV-K(HML-2) analysis. Only 8 of the available matched tumor genomic sequencing data contained suitable unmapped reads and were analysed in this study. The analysed data encompassed 3 groups, two of which contained T4b prostate stage patient data, whereas the last one focused on T3 NOS patients.

- Prostate Cancer Genome Sequencing Project

No relevant metadata is available.

- Genome-Wide Characterization of Pancreatic Adenocarcinoma Patients Using Next Generation Sequencing.

The tested project sequenced genomes of cells from 4 patients, but contained only one full genome sequence of pancreatic adenocarcinoma, the mean age in the group was 62 years and all the tested patients were of Caucasian origin, suffering from stage IV pancreatic adenocarcinoma.

- Breakpoint detection using long insert whole genome sequencing

No relevant metadata is available.

- Small Intestine Neuroendocrine Tumors (Carcinoid Tumors)

There were 12 patient's full genome sequencing data available. The carcinoid tumor patients belonged to 4 groups or mean ages: 56.12, 53.6, 55.39, 57.12. The first 3 groups were males, the last group were females.

- A Pilot Study Using Next Generation Sequencing in Advanced Cancers: Feasibility and Challenges

The tested project defined 15 patients, but contained 9 full genome sequences of liver or pancreas adenocarcinoma patients. The mean age in the group was 48.89 years and all the tested patients were of Caucasian origin.

- Genomic Sequencing of Cervical Cancers

All of the tested patients were female, from Norway, with the mean age of 47.84 years. The patients suffered from cervical squamous cell carcinoma.

- Genomics of Hepatocellular Carcinoma

All of the tested patients were white females, suffering from hepatocellular carcinoma.

- Sequence Analysis of Mutations and Translocations Across Breast Cancer Subtypes

The tested patients were categorized into 3 groups: two of Mexican origin, with mean ages of 51.06 and 54.21 years and one of Vietnamese origin, with mean age of 47.67 years. All of the patients suffered from breast cancer, of infiltrating ductal carcinoma / luminal A subtype, stage II.

- Towards a Genomic Understanding of Myeloma

Metadata unavailable.

2.4.ii. Querying TCGA sequencing data samples.

Genetic alignments of cancer patients and healthy cell controls corresponding to each cancer sample extracted from the same individual from The Cancer Genome Project. Exact details on selected cancers can be found in the "Cancer data selection", section 2.4.i. The data is deposited in Sequence Read Archive maintained by the National Center for Biotechnology Information (NCBI). Data was processed using NCBI Sequence Read Archive (SRA) toolkit version 2.10.8. A part of that toolkit, prefetch program, was used to download the data into University of Oxford Advanced Research Computing (ARC) cluster for further processing (Figure 16, step 1), using the following command:

Table 4 Download parameters.

prefetch --max-size 500G <accession>
Where: --max-size – determines maximum size of downloaded file. <accession> accession number of the downloaded sample.

Downloaded files contained human medical data, therefore the source was encrypted. To decrypt it and convert into conventional, Sequence Alignment/Map (SAM) format, in order to further process it, sam-dump tool of the SRA toolkit was used. The tool was used to extract location containing known insertions (command 1, Table 5), as well as pairs of reads, where one is mapped and the other is not (command 2, Table 5). Such partially-mapped reads represent data that does not map to the reference genome and contains potential novel insertions. The unmapped data was filtered using GNU grep tool, provided by the Unix environment.

Table 5 Data extraction parameters.

(1) sam-dump –ngc <key> chrX:start-end <filename> (2) sam-dump –ngc <key> --unaligned --aligned-region chrX <filename> | grep $'[*]\t0\t0\t[*]'
Where: --ngc – loads encryption key for patient data decryption. --unaligned – extracts unaligned reads in addition to aligned ones. --aligned-region <chromosome:location> - extracts reads from within specified region. $'[*]\t0\t0\t[*]' – grep filter representing string that contains two zeros, flanked and separated by a single tab character. Such string is a distinctive property of paired-end reads saved in SAM format, in which one of the reads is mapped to the reference genome and the other is not.

Extracted loci coming from the known insertions have been further analysed. Data was separated into individual loci using *isolate.py* script (appendix 1.xii, Figure 16, step 2). Some of the insertions appeared to be missing from the source data. Therefore, the alignment coverage of these insertions has been investigated using *coverage3.py* script (appendix 1.xiii), that displayed coverage values at -30bp - +30bp range of each known insertion. Many of the insertions have been found to lack any coverage and it was confirmed manually, across all the available projects (Figure 15). An attempt to realign previously unaligned reads, which mate pairs come from known insertion loci did not improve the coverage significantly. It was concluded, that this is a result of low mapping quality between transposable elements in the analysed cancer data, which made characterization of known insertions unreliable. Therefore, the research focused on partially unmapped reads to find novel insertions.

Partially mapped reads have been investigated for presence of HERV-K113 (accession number AY037928) LTR (1-100bp and 869-969bp) using BLAST (Figure 16, step 3). The database used contained 100bp of 5' end and 100bp of 3' end of the HERV-K113 LTR, because read length for each of the analysed project was approximately 100bp and only such ends would be detectable for novel insertions. The SAM formatted source data was converted into FASTA using *sam2fasta.py* script (appendix 1.xiv). Initially, the reads have been aligned using BLAST with parameters described in command 1 below, particularly using word size = 11.

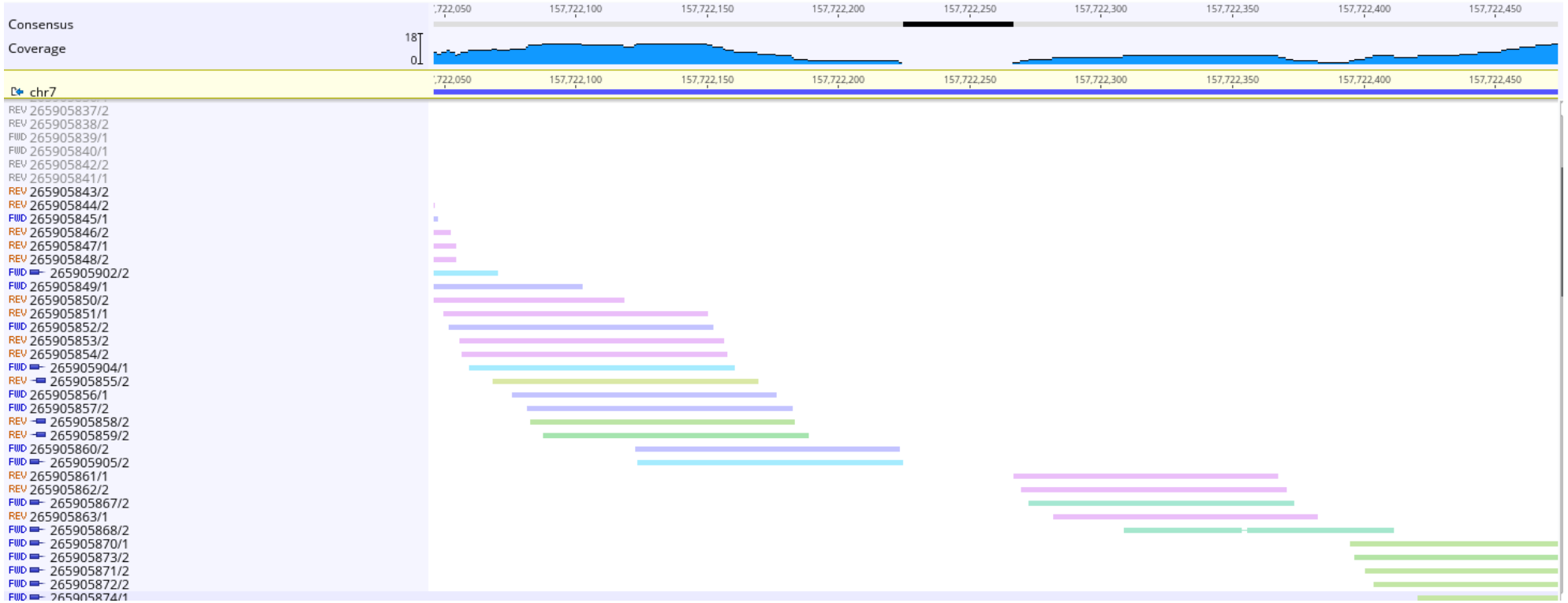

Figure 15 Example coverage at the junction of LTR403. Notice the lack of reads representing the junction between flank and LTR start position and low coverage of internal sequence compared to the flank. The reads are aligned to chromosome 7 of the NCBI36/Hg18 human reference genome (yellow bar on the top). The missing coverage is represented by the black bar and lack of signal in the blue "Coverage" section, above the reference sequence. The individual reads are visible in the alignment and represented by numeric identifiers on the left (FWD – forward alignment, REV – reverse alignment). The colours of the reads represent paired distance, where different shades of colours represent observed distances between paired reads (green – expected distance, blue – over expected distance, yellow – under expected distance, pink/violet – unpaired reverse/forward read).

Table 6 BLAST analysis parameters.

(1) blastn -num_threads 2 -db ltrends.fasta -query <fasta> -evalue 0.001 -word_size 11 -max_target_seqs 500000 -outfmt 6 (2) blastn -num_threads 2 -db ltrends.fasta -query <fasta> -evalue 0.001 -word_size 16 -max_target_seqs 500000 -outfmt 6
Where: -num_threads – processor cores used in the analysis. -db – blast database name. -query – query file name. -evalue – expect value, number of random results expected from a source query of a given size. -word_size – minimal match to count the result as positive. -max_target_seqs – parameter ensuring all the positive results are retained. -outfmt 6 - tab formatted results.

The search yielded a great number of random results, which have been removed manually at the end of the investigation. To reduce that, word size was increased to 16 (command 2 above), which was found to retain valid results, while removing many of the random hits from the analysis. Reads have been selected based on BLAST results, that was 30bp or longer and extracted into fasta files, using *extractunmapped.py* script (Figure 16, step 4; appendix 1.xv).

Produced reads have been sorted by chromosomal location using *sort.py* script (appendix 1.xvi). Sorted data was further separated into reads coming from 300bp wide contigs, using *contigs.py* script (Figure 16, step 4; appendix 1.xvii). Resulting read contigs were divided into contigs coming from non-polymorphic known reads, reads coming from known loci, which are polymorphic and present in the reference genome and HERV-K reads coming from locations that are not present in reference genome. This was done using *removeknown.py* script (appendix 1.xviii). Because of the aforementioned coverage problems, many of the possibly novel loci were incomplete, so the resulting files were filtered using *53pw.py* script (appendix 1.xix), to extract reads that contained fragments from both 5' and 3' ends of HERV-K113 LTR. That step also filtered out any SVA element (see section 1.13), since SVAs do not contain last 100bp of HERV-K 113 3' end (Figure 16, step 5). The resulting reads were loaded into Geneious software and aligned to HERV-K113 reference genome (AY0379278). Alignments were inspected manually and mapped against human reference genome to pinpoint locations of novel insertions (Figure 16, step 6).

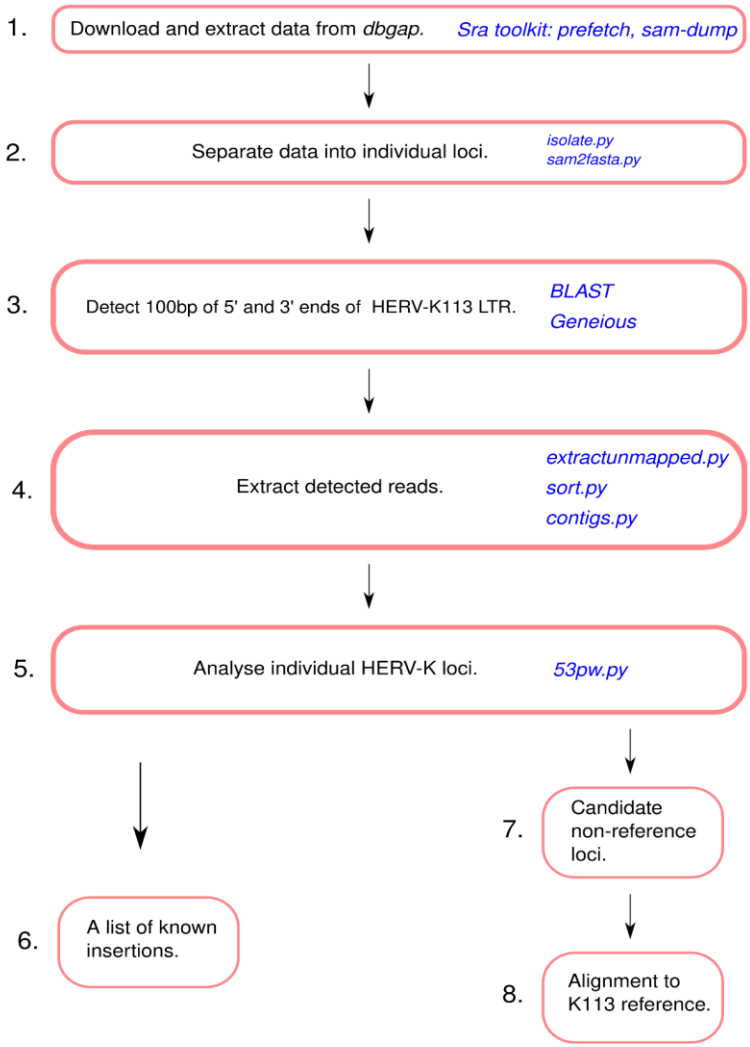

Figure 16 Step by step analysis of novel HERV-K(HML-2) insertions in cancer genomes.

2.4.iii. Database search.

Locations of detected insertions were grouped and ranges of +/-250kb around each insertion were inserted into UCSC (University of California Santa Cruz) genome table browser, both for Grch37 and Grch38 reference genome versions. Resulting output was extracted and gene names in the selected ranges were listed, which produced 218 different genes in the vicinity of the located insertions. These gene names were used in W*ebgesalt* and S*tring-db* interaction databases as queries to obtain a list of genes that are reported to interact with query genes. Resulting summaries for both databases contained a list of 443 genes in total, including genes in the influence range of the HERV-K insertion, and genes in their interaction networks. All of the identified gene names were collated and inserted into *OpenTargetsPlatform* database as query, to download a database of literature references related to these genes in a tabular format. The query produced 280,179 hits. These literature references contained metadata, including disease names if studies they referred to focused on particular diseases. These records were screened for cancer studies by looking into

disease keywords in the metadata and searching for known cancer names. The names of all diseases listed in that sample was extracted using *identifycancer.py* (appendix 1.xx) from the database. The names of known cancers were then subsequently selected and the entries in the database were filtered by these cancer names, using script *filtercancer.py* (appendix 1.xxi), which yielded 100,247 references to cancer studies involving the proteins of interest. The entries were further filtered for those containing *reactome* pathway database references. The *reactome* IDs used with *reactome* python API (*getdata.py* script, appendix 1.xxii) were used to download information about the pathways the detected genes are involved in.

2.4.iv. Non-reference insertions in cancer - statistical analysis.

In order to study statistical significance of novel insertions found in cancer, the frequency of occurrence for each insertion was calculated and compared to frequencies obtained from general population data presented in Wildschutte et al. (2016). Reads from detected cancer insertions have been aligned to HERV-K113 LTR reference genome (AY037928) using Geneious software and resulting alignments were saved in *.sam* format. A *gentable.py* python script (appendix 1.xxiii) was used to obtain frequency of occurrence of each insertion among the tested cancer population. The script counted the occurrence of both 5' and 3' ends of a particular insertion within every individual in the tested cancer populations. Further, it provided a frequency of occurrence of these insertions in the tested cancer population, by counting how many subjects harboured reads suggesting presence of HERV-K(HML-2) insertion at a particular locus. Because of aforementioned coverage problems (Figure 15), the exact allele frequency measurement by counting the individual reads in each alignment file would be unreliable. The frequency calculated was based on either presence or absence of reads representing insertions in particular individuals. Then the individuals were assumed to all be heterozygotes to capture the extreme values of allele counts for the most likely scenario, where majority of the individuals bearing novel insertions are heterozygous in regard to those insertions. The assumed viral insertion allele frequencies were calculated according to formula:

$$f_{ins} * \frac{n}{n * 2}$$

The assumed preinsertion site allele frequencies were calculated according to formula:

$$f_{ins} * \frac{n}{n * 2} + (1 - f_{ins}) * \frac{n * 2}{n * 2}$$

Where f_{ins} is the frequency of HERV-K(HML-2) locus obtained from particular cancer sample, n represents the number of individuals in the tested population and n*2 represents the number of genomes in a diploid population.

Source data from Wildschutte *et al.* (2016) has been obtained from *Dataset_S02* (supplementary information, (Wildschutte *et al.*, 2016)). To obtain frequency of occurrence for each insertion *calculatefreq.py* python script (appendix 1.xxiv) was used. The script counted all of the homozygotes, heterozygotes and individuals with missing data within 1000 genomes project and Human Genome Diversity Project samples, according to the nomenclature Wildschutte *et al.* (2016) used in their source data. Based on allele frequency and sample count, allele counts for pre-insertion sites and proviral insertions were calculated for each of the normal human populations studied in Wildschutte *et al.* (2016). The allele counts were analysed using Fisher's Exact Test to analyse the significance of insertion frequencies among the normal human populations, using *canceranalysis.R* script (appendix 1.xxv), that compared each of the population within itself using Fishers Exact test of significance. The allele counts for each insertion have been added up to obtain counts representing the Human Genome Diversity Project and 1000 Genomes Project, studied in Wildschutte *et al.* (2016). To study the significance of frequency of occurrence of polymorphic HERV-K(HML-2), a Fisher's Exact Test was used, using *fishetestall.R* script (appendix 1.xxvi). Results were considered significant if P_{corr}<1.1E-004, following the universal statistical threshold (P<0.05) and applying Bonferroni correction (Bonferroni, 1936) for multiple testing, by dividing target P by the number of performed tests on reference populations (n_t=450):

$$P_{corr} = \frac{P}{n_t}$$

3. Chapter 3: Results.

3.1. State of HERV-K insertion in the reference human genome.

3.1.i. HERV-K(HML-2) database; TSD analysis.

Information collected from the literature and confirmed in recent versions of the human reference genome (GRCh36/Hg18, GRCh37/Hg19 and GRCh38/Hg38) revealed a total of 1133 HERV-K(HML-2) insertions. Additionally, details of sequences (flanking sequence of each locus, 5' and 3' end of each locus, TSDs) and references for all of the reported insertions were recorded. The most complete HERV-K(HML-2) database of information collected in this study is available online at:

https://docs.google.com/spreadsheets/d/1VhELw8Qq4rI4ezHhE8VvratJfIIInXMU/edit?usp=sharing&ouid=111879202149729360013&rtpof=true&sd=true

1119 insertions are well defined elements, which precise locations are known. One of the reported insertions is a fusion of two full-length HERV-K proviruses that share their TSD (7p22.1a/7p22.1b, see Table 7), therefore in the TSD analysis in sections 3.1.i onwards, the total number of insertions is 1118. Among the previously reported loci (Subramanian *et al.*, 2011), two loci were found to be duplicates and have been removed from the latest build (GRCh38/Hg38) of the reference genome. HML-2.LTR466 and HML-2.LTR485 are thought to be the same as HML-2.LTR465 and HML-2.LTR480/481 respectively. These two loci were omitted from further analysis. Three previously described solo-LTRs (HML-2.LTR189, HML-2.LTR569 and HML-2.LTR863) are in fact parts of full-length elements, present in the reference genome (3q21.2, 11q12.3 and 20q11.22 respectively); hence the three solo-LTRs were omitted from further analysis. Table 7 below summarizes the findings of Subramanian *et al.* (2011), and compares it to results provided in this study, as well as other studies reported to date. Overall, this study provides information on 10 newly discovered solo-LTR elements in the recent human reference genomes, as well as additional 25 solo-LTRs reported in various recent studies. Moreover, this study reports 19 full-length elements not mentioned in Subramanian *et al.* (2011), two of which were previously reported in the literature (references provided online in the HERV-K(HML-2) database). Therefore, data presented here represents the most comprehensive collection and analysis of the data on HERV-K(HML-2) family with regards to the human reference genome to date.

Table 7 Summary of reported HERV-K(HML-2) insertions

	Number of insertions presented by Subramanian *et al. (2011)* .	Insertions reported in the literature to date. References provided online in the HERV-K(HML-2) database.	Insertions reported in this study. References provided online in the HERV-K(HML-2) database.
Solo LTRs found in the reference genome	946	971	981
Full length insertions found in reference genome (including truncated loci).	87	89	106
Insertions not present in reference genome	2	46	46

Table 8 summarizes the data collected on full-length insertions, table 9 provides similar summary on truncated insertions, table 10 include solo-LTR elements reported in this study, that were not reported by Subramanian *et al*. (2011). Remaining solo-LTR elements are listed in appendix 2.

TSD analysis was conducted as described in methodology. Briefly, out of 1118 insertions identified, 929 showed matching TSDs; the remainder either had non-matching TSDs (80 insertions) or were truncated.

Table 8 Summary of full-length insertions reported in this study

Chromosome	Locus	Name	GRCh38 / Hg38 position	Column1	Type	Reference	Status
1	1p36.21b	K76 K6; K(OLDAL023753)	13206972	13216513	LTR5B	(Reus et al.; 2001)	AGCGTA/AGTGTA
1	1p36.21c	K6; K76	13352736	13362257	LTR5B	(Reus et al.; 2001)	AGTGTA
1	1p31.1	K116 K4; ERVK-1	75377086	75383458	LTR5_Hs	(Hughes and Coffin; 2001)	ATGGAA
1	1q21.3	K102	150632808	150635885	LTR5_Hs	(Subramanian et al.; 2011)	CTCAGC
1	1q22	K102 K50a; ERVK-7; K(C1b)	155626666	155635845	LTR5_Hs	(Barbulescu et al.; 1999)	GGGATG
1	1q23.3	K110 K18;K(C1a);ERVK-18	160690785	160700016	LTR5_Hs	(Barbulescu et al.; 1999)	TGAGAC
1	1q24.1	K12	166605366	166611021	LTR5B	(Romano et al.; 2006)	ACATGC
3	3p25.3	K11 ERVK-2	9847662	9854552	LTR5_Hs	(Hughes and Coffin; 2001)	-
3	3q12.3	K(II); ERVK-5; K118	101691893	101701015	LTR5_Hs	(Costas et al.; 2001)	GAGGT
3	3q13.2	K106 K(C3);K68;ERVK-3	113024277	113033435	LTR5_Hs	(Barbulescu et al.; 1999)	GGCTGG
3	3q21.2	K121 ERVK-4; K(I), c3_C	125890459	125899596	LTR5_Hs	(Sugimoto et al.;2001)	GGCCC
3	3q27.2	K117 ERVK-11	185562548	185571727	LTR5_Hs	(Hughes and Coffin; 2001)	GGTACA
4	4p16.3b	K77	3977324	3986912	LTR5A	(Romano et al.; 2006)	ATTTG
4	4p16.1a	K17b	9121786	9131367	LTR5A	(Romano et al.; 2006)	ATTTG
4	4p16.1b	K50c	9657956	9667550	LTR5A	(Macfarlane and Simmonds; 2004)	ATTTG
4	4q32.3	K5 ERVK-12	164995688	165002916	LTR5_Hs	(Hughes and Coffin; 2001)	TTTT
4	4q35.2	-	190106259	190113550	LTR5A	(Subramanian et al.; 2011)	AACGT/AACAT
5	5p13.3	K104 K50d	30486653	30496098	LTR5_Hs	(Barbulescu et al.; 1999)	CAGAAC
5	5p12	-	46000057	46009900	LTR5_Hs	(Subramanian et al.; 2011)	CTCCC
5	5q33.2	K18b ERVK11	154635953	154644657	LTR5_Hs	(Romano et al.; 2006)	ATTACT
5	5q33.3	K107 K10; K(C5); ERVK-10	156657706	156666885	LTR5B	(Ono et al.; 1987)	ACTGC
6	6p21.1	K(OLDAL035587) KOLD35587	42893671	42903629	LTR5B	(Reus et al.; 2001)	AAACT/AAATT
6	6p11.2	K23	60654987	60660975	LTR5A	(Romano et al.; 2006)	CTTGT
6	6q14.1	K109 K(C6); ERVK-9	77716945	77726366	LTR5_Hs	(Barbulescu et al.; 1999)	ATATGC
7	7p22.1a	K108L K (HML.2-HOM); K(C7); ERVK-6	4582426	4591897	LTR5_Hs	(Barbulescu et al.; 1999)	GGTTTC
7	7p22.1b	K108 K108R; ERVK-6	4590930	4600400	LTR5_Hs	(Barbulescu et al.; 1999)	GGTTTC
8	8p23.1a	K115 ERVK-8	7497875	7507337	LTR5_Hs	(Turner et al.; 2001)	CCTTT
8	8p23.1b	K27	8197178	8206699	LTR5A	(Hughes and Coffin; 2001)	ATTTG
8	8p23.1c	-	12216461	12225988	LTR5A	(Hughes and Coffin; 2001)	ATTTG
8	8p23.1d	KOLOLD130352	12458983	12468498	LTR5A	(Hughes and Coffin; 2001)	ATTTG
8	8p22	-	17907211	17916624	HML-11	(Subramanian et al.; 2011)	AGGTT
8	8q11.1	K70 K43	46264028	46272039	LTR5A	(Romano et al.; 2006)	-
8	8q24.3a	K127	139459906	139463016	LTR5_Hs	(Subramanian et al.; 2011)	TTTCT
8	8q24.3c	-	144860789	144860789	unknown	(Wildschutte et al.; 2016)	ACATGT
9	9q34.11	K31	128850236	128857457	LTR5A	(Hughes and Coffin; 2001)	TTCAG/CTCAG
9	9q34.3	K30	136780314	136789776	LTR5B	(Hughes and Coffin; 2001)	-
10	10p14	K33 K(C11a); ERVK-16	6824179	6833641	LTR5_Hs	(Costas et al.; 2001)	TCATTC
11	11p15.4	K7	3447426	3456979	LTR5A	(Romano et al.; 2006)	ATTTG/ATTTTG
11	11q12.3	K(OLDAC004127)	62368491	62383091	LTR5B	(Reus et al.; 2001)	-
11	11q22.1	K118 K(C11c); K36; ERVK-25	101695063	101704528	LTR5_Hs	(Costas et al.; 2001)	TTGTG
11	11q23.3	K37 K(C11b); ERVK-20	118721015	118730174	LTR5_Hs	(Subramanian et al.; 2011)	AGCCT
12	12p11.1	K50e	34619620	34629285	LTRHs	(Romano et al.; 2006)	-
12	12q14.1	K119 K(C12); K41; ERVK-21	58327459	58336915	LTR5_Hs	(Costas et al.; 2001)	TTGGTA
12	12q24.11	K129	110570038	110571543	LTR5_Hs	(Medstrand; Mager; 2004)	GTATT
12	12q24.33	K42	133090536	133096478	LTR5A	(Romano et al.; 2006)	TAAAT
19	19p12a	K52	20276591	20286706	LTR5B	(Hughes and Coffin; 2001)	-
19	19p12b	K113 ERVK26 De1	21658734	21658734	unknown	(Turner et al.; 2001)	CTCTAT
19	19p12d	ERVK27	22231577	22231577	unknown	(Lee et al.; 2012)	TTTT/TCTT
19	19p12e	De11 AluSq	22274442	22274442	unknown	(Agoni et al.; 2012)	AAGCAAAAC
19	19q11	K132 ERVK-19	27637590	27646476	LTR5_Hs	(Subramanian et al.; 2011)	AGGTAT
19	19q13.12b	K50f KOLD12309; K(OLDAC012309)	37106647	37116164	LTR5_Hs	(Reus et al.; 2001)	GGTCTT/GATCTT
20	20q11.22	K59 K(OLDAL136419)	34126942	34136578	LTR5B	(Subramanian et al.; 2011)	AAAA
22	22q11.21	K101 K(C22); ERVK-24	18938674	18947848	LTR5_Hs	(Barbulescu et al.; 1999)	ACCCAG
X	Xq21.33	De9	94351604	94351604	unknown	(Agoni et al.; 2012)	ATAAT
X	Xq28b	K63	154608423	154615762	LTR5B	(Subramanian et al.; 2011)	TCCAGC/TCCACC

Table 9 Summary of truncated full-length insertions reported in this study.

Chromosome	Locus	Name	GRCh38 / Hg38 position		Type	Reference	Status
1	1p36.21a	-	12780117	12785935	LTR5B	(Subramanian et al.; 2011)	3' TRUNCATED
1	1p35.1	solo-1p35.1	33063516	33065110	LTR5B	(Xue et al.; 2020)	3' TRUNCATED
1	1p34.3	-	36488984	36491127	LTR5_Hs	(Subramanian et al.; 2011)	5' TRUNCATED
1	1q32.2	-	207635112	207639289	LTR5_Hs	(Subramanian et al.; 2011)	5' TRUNCATED
1	1q43	-	238762295	238764403	LTR5B	(Subramanian et al.; 2011)	5' TRUNCATED
2	2q21.1	K120	129961965	129965077	LTR5_Hs	(Subramanian et al.; 2011); (Shin et al.; 2013)	5' TRUNCATED
2	-	LTRW3	186520908	186522372	LTR5B	wysocka et al.; 2018	5' AND 3' TRUNCATED
3	3p12.3a	solo-3p12.3	75536830	75538747	LTR5A	(Xue et al.; 2020)	5' TRUNCATED
3	3p12.3	K(II)	75551324	75559999	LTR5A	(Subramanian et al.; 2011)	5' TRUNCATED
3	3q21.2a	solo-3q21.2	125799251	125801116	LTR5A	(Xue et al.; 2020)	5' TRUNCATED
3	3q24	K122 ;ERVK-13	148563690	148567632	LTR5_Hs	(Subramanian et al.; 2011)	5' TRUNCATED
3	3q26.1	K123	171237865	171238015	unknown	(Shin et al.; 2013)	5' AND 3' TRUNCATED
4	4p16.3a	-	241201	245672	LTR5B	(Subramanian et al.; 2011)	5 AND 3' TRUNCATED
4	4p16.3	solo-4p16.3	4073397	4075313	LTR5A	(Xue et al.; 2020)	5' TRUNCATED
4	4p16.1c	solo-4p16.1c	9034417	9036346	LTR5A	(Xue et al.; 2020)	5' TRUNCATED
4	4p16.1d	solo-4p16.1d	9567314	9569224	LTR5A	(Xue et al.; 2020)	5' TRUNCATED
4	4q13.2	-	68597991	68603505	LTR5A	(Subramanian et al.; 2011)	5' TRUNCATED
4	4q32.1	K124	160658786	160661290	LTR5_Hs	(Subramanian et al.; 2011); (Shin et al.; 2013)	3' TRUNCATED
5	-	LTRW5	93456674	93458200	LTR5B	wysocka et al.; 2018	5' AND 3' TRUNCATED
6	-	LTRW7	27826044	27826682	LTR5B	(wysocka et al.; 2018)	5' AND 3' TRUNCATED
6	6p22.1	K69 K(OLDAL121932); K20	28682591	28692958	LTR5_Hs	(Subramanian et al.; 2011)	5' TRUNCATED
6	6q13	K125	73333259	73333400	unknown	(Shin et al.; 2013)	5 ' AND 3' TRUNCATED
6	6q25.1	-	150859609	150862438	LTR5B	(Subramanian et al.; 2011)	5' TRUNCATED
7	7q11.21	-	66004684	66007804	LTR5B	(Subramanian et al.; 2011)	5' TRUNCATED
7	7q22.2	K126 ERVK-14	104747922	104752822	LTR5_Hs	(Subramanian et al.; 2011)	3' TRUNCATED
7	7q34	ERVK-15 K(OLDAC004979)	141751126	141756139	LTR5_Hs	(Reus et al.; 2001)	3' TRUNCATED
8	8p23.1f	solo-8p23.1f	7126987	7128897	LTR5A	(Xue et al.; 2020)	5' TRUNCATED
8	-	LTRM7	7185491	7187404	LTR5A	This study	5' TRUNCATED
8	8p23.1g	solo-8p23.1g	8100098	8102011	LTR5A	(Xue et al.; 2020)	5' TRUNCATED
8	14	LTRM15	12565087	12567003	LTR5A	This study	5' TRUNCATED
8	8p23.1e	solo-8p23.1e	12623206	12625104	LTR5A	(Xue et al.; 2020)	5' TRUNCATED
8	8q24.3b	K29	145019967	145032060	LTR5A	(Hughes and Coffin; 2001)	5' TRUNCATED
10	10q24.2	K128 c10_B; ERVK-17	99820812	99827988	LTR5_Hs	(Macfarlane and Simmonds; 2004)	5' TRUNCATED
11	11p15.4a	solo-11p15.4a	3544471	3546375	LTR5A	(Xue et al.; 2020)	5' TRUNCATED
11	11q12.1	-	58999977	59005723	LTR5_Hs	(Subramanian et al.; 2011)	5' TRUNCATED
12	12q23.3	solo-12q23.3	107826191	107827421	LTR5B	(Xue et al.; 2020)	3' TRUNCATED
14	14q11.2	K71 K(OLDAL136419);	24011391	24015762	LTR5A	(Reus et al.; 2001)	5 ' AND 3' TRUNCATED
14	14q32.33	K(OLDAC004034)	105673313	105676203	LTR5A	(Romano et al.; 2006)	5' TRUNCATED
15	15q25.2	-	84160268	84164397	LTR5B	(Subramanian et al.; 2011)	3' TRUNCATED
16	16p13.3	K(OLDAC004034)	2926159	2928722	LTR5B	(Subramanian et al.; 2011)	3' TRUNCATED
16	-	LTRW15	34412057	34414806	LTR5B	wysocka et al.; 2018	5' TRUNCATED
16	16p11.2	K130	34997024	34999771	LTR5_Hs	(Subramanian et al.; 2011); (Shin et al.; 2013)	5' TRUNCATED
17	17p13.2	K131	6175597	6175597	unknown	(Shin et al.; 2013)	5' AND 3' TRUNCATED
17	17p13.1	-	8056939	8063347	HML-11	(Subramanian et al.; 2011)	5 AND 3' TRUNCATED
19	19p13.3	ERVK-22	385099	387637	LTR5_Hs	(Subramanian et al.; 2011)	5' TRUNCATED
19	19p12c	K51	22575018	22581759	LTR5_Hs	(Subramanian et al.; 2011)	5' TRUNCATED
19	19q13.12a	-	35572311	35576532	LTR5_Hs	(Subramanian et al.; 2011)	5' AND 3' TRUNCATED
19	19q13.41	-	52740440	52754485	LTR5B	(Subramanian et al.; 2011)	5' TRUNCATED
19	19q13.42	LTR13	53359091	53364791	LTR5B	(Subramanian et al.; 2011)	5' TRUNCATED
20	20p11.21b	*solo-20p11.21b*	23693935	23695327	LTR5B	(Xue et al.; 2020)	3' TRUNCATED
20	20p11.21a	*solo-20p11.21a*	23755863	23757247	LTR5B	(Xue et al.; 2020)	3' TRUNCATED
21	21q21.1	K133 K60; ERVK-23	18561341	18569644	LTR5_Hs	(Subramanian et al.; 2011)	3' TRUNCATED
22	22q11.23	K(OLDAP000345); KOLD345	23537740	23546899	LTR5B	(Subramanian et al.; 2011)	3' TRUNCATED
X	Xq11.1	-	62740079	62742584	LTR5B	(Subramanian et al.; 2011)	5 AND 3' TRUNCATED
X	Xq12	-	66464290	66466340	LTR5A	(Subramanian et al.; 2011)	5' AND 3' TRUNCATED
X	Xq28a	K63	154588662	154591382	LTR5B	(Subramanian et al.; 2011)	3' TRUNCATED
Y	Yp11.2	-	6958400	6965431	LTR5A	(Subramanian et al.; 2011)	5' TRUNCATED
Y	Yq11.23a	-	24250775	24254888	LTR5B	(Subramanian et al.; 2011)	3' TRUNCATED
Y	Yq11.23b	-	25415255	25419369	LTR5B	(Subramanian et al.; 2011)	3' TRUNCATED

Table 10 Summary of solo-LTR insertions reported in this study, absent from Subramanian et al. (2011).

Chromosome	Locus	Name	GRCh38 / Hg38 position		Type	Reference	Status
1	1p31.1c	AluSz	79326944	79326944	unknown	(Wildschutte et al.; 2016)	ACCAT
1	1p21.1	-	105473253	1054732532	unknown	(Marchi et al.; 2014)	CAAAAT
1	1p13.2	De5;K1; ERVK1 L1 (L1PA6)	111259970	111259970	unknown	(Agoni et al.; 2012)	CTTTTT
1	-	LTRM9	144555550	144556489	LTR5_Hs	This study	-
1	-	LTRW2	146381151	146382090	LTR5B	(wysocka et al.; 2018)	-
1	1q41	ERVK_2;K2 L1 (L1MDa)	223404967	223404967	unknown	(Lee et al.; 2012)	AAATAC
2	-	LTRW4	208199555	208200169	LTR5B	(wysocka et al.; 2018)	3' TRUNCATED
3	3q11.2	L1 (L1PA10)	95224645	95224645	unknown	(Wildschutte et al.; 2016)	GTATA
3	3q22.1	N3q22.1	130447719	130447719	unknown	(Xue et al.; 2020)	UNKNOWN
4	4p16c	K6 ERVK6	9601616	9601616	unknown	(Lee et al.; 2012)	GAACT
4	4p16d	L2 L2b; SLCA29 intron 5/6	9979981	9979981	unknown	(Wildschutte et al.; 2016)	AGAAG
4	4q26	De13	117182108	117182108	unknown	(Lee et al.; 2014)	UNKNOWN
5	5p15.32	ERV1 (LTR1C)	4537491	4537491	unknown	(Wildschutte et al.; 2016)	GAAAG
5	5q12.3	Ne7;K12 L1 L1M6	65092618	65092618	unknown	(Lee et al.; 2014)	GACGTG
5	5q14.1	De6; Ne1;K10 RASGRF2 intron 17	81146447	81146447	unknown	(Agoni et al.; 2012)	GACCAG
6	6p22.3	AluSx	16004632	16004632	unknown	(Wildschutte et al.; 2016)	GCCAC
6	6p22.2	solo-6p22.2	25999231	26000197	LTR5B	(Xue et al.; 2020)	ACCTCC/ACCTAC
6	6p21.32	N6p21.32	32675687	32675687	LTR5_Hs	(Xue et al.; 2020)	UNKNOWN
6	6p21.32	-	32680263	32680264	unknown	(MARCHI et al.; 2014)	ACTCC
6	-	LTRW8	73805312	73805991	LTR5B	(wysocka et al.; 2018)	3' TRUNCATED
6	6q26	De2; ERVK12	160849872	160849872	unknown	(Agoni et al.; 2012)	TACGCC
7	7p14.3	solo-7p14.3	30718584	30719589	LTR5B	(Xue et al.; 2020)	-
7	7q36.3	-	158980699	158980699	unknown	(Wildschutte et al.; 2016)	CAGAGG
8	8p12	solo-8p12	30118525	30119387	LTR5B	(Xue et al.; 2020)	GGAGAC/GGAGAT
8	8q21.11	Ne4	74432836	74432836	unknown	(Lee et al.; 2014)	TCAAT
10	10q24.1	solo-10q24.1	97416846	97418118	LTR5B	(Xue et al.; 2020)	GTCCAG
10	10q24.2a	ERVK17, De12	99256365	99256365	unknown	(Agoni et al.; 2012)	CATCT
10	10q26.3	INPP5A intron 2	132630508	132630508	unknown	(Wildschutte et al.; 2016)	TTCAC
11	11p15.4b	solo-11p15.4b	4459435	4461476	LTR5B	(Xue et al.; 2020)	TTTAGT/ATTAGT
11	11q12.2a	De4; ERVK18; K18	60682417	60682417	unknown	(Agoni et al.; 2012)	CATTTT
11	11q12.2	solo-11q12.2	60713186	60714479	LTR5B	(Xue et al.; 2020)	TCTGAA
11	-	LTRW11	71767205	71767904	LTR5_Hs	(wysocka et al.; 2018)	GAGGG
12	-	LTRM18	10394625	10395183	unknown	This study	AGTGA
12	12q12	ERVK20; HERV-K-Ne6	43919858	43919858	unknown	(Lee et al.; 2014)	GTGGT
12	12q24.31	K21 ;ERVK21	123581930	123581930	unknown	(Lee et al.; 2012)	ATGAAC/ACGAAC
12	12q24.32	-	127153657	127153657	unknown	(Kidd JM et al.; 2008)	CAAGTA
13	18	LTRW13	18250955	18251919	LTR5B	(wysocka et al.; 2018)	GGTCCC
13	13q31.3	Ne2; ERVK22	90090929	90090929	unknown	(Agoni et al.; 2012)	AGTAGT
14	-	LTRW14	25792775	25793313	LTR5_Hs	(wysocka et al.; 2018)	5' TRUNCATED
15	15q13.1	ERVK23	28184959	28184959	unknown	(Wildschutte et.al; 2016)	ATAGAT
15	15q21.2	N15q21.2	51358631	51358631	unknown	(Xue et al.; 2020)	UNKNOWN
15	15q22.2	ERVK24, K24	63082395	63082395	unknown	(Lee et al.; 2012)	TTTTTC
16	-	LTRM11	32230109	32230840	LTR5B	This study	5' TRUNCATED
16	16q23.1	solo-16q23.1	75815005	75816296	LTR5B	(Xue et al.; 2020)	AGGAT
18	18q12.1	DeNe2	30138227	30138227	unknown	(Lee et al.; 2014)	CAAGTA
19	19p13.3a	DeNe1	2890621	2890621	unknown	(Lee et al.; 2014)	CTCCC
19	19p12	solo-19p12	23277028	23278544	LTR5B	(Xue et al.; 2020)	AGCAT/AGCAC
19	19q11	N19q11	27707680	27707680	unknown	(Xue et al.; 2020)	UNKNOWN
19	19q12	De3/ERVK28 K28	29364874	29364874	unknown	(Agoni et al.; 2012)	TTACCA
19	-	LTRW17	52024482	52024993	LTR5B	(wysocka et al.; 2018)	3' TRUNCATED
19	-	LTRW18	52483847	52484545	LTR5_Hs	(wysocka et al.; 2018)	5' TRUNCATED
19	19q13.41	Ne3	52825075	52825075	unknown	(Agoni et al.; 2012)	UNKNOWN
19	19q13.41a	solo-19q13.41	52932615	52934255	LTR5B	(Xue et al.; 2020)	ATGCAT
19	19q13.43	Ne5	57485571	57485571	unknown	(Lee et al.; 2014)	GTCTA
20	-	LTRW19	4034112	4034982	LTR5B	(wysocka et al.; 2018)	3' TRUNCATED
20	20p12.1	HERV-K-De14/ERVK30 HERV-K-De14/ERVK30	12421743	12421743	unknown	(Hughes and Coffin; 2001)	AGTGG
20	22	LTRM1	29316282	29317182	LTR5A	This study	CTCAC
20	30	LTRM2	29440303	29441256	LTR5B	This study	-
20	31	LTRW22	30849917	30850867	LTR5B	(wysocka et al.; 2018)	ATATAG/ATATAC
21	32	LTRW23	6538815	6539774	LTR5A	(wysocka et al.; 2018)	AATCC
21	21q22.11	N21q22.11	32451912	32451912	unknown	(Xue et al.; 2020)	UNKNOWN
22	31	LTRM5	11323908	11324867	LTR5B	(Xue et al. 2020) / This study	ATATAG
22	31	LTRM6	11555908	11556858	LTR5B	(Xue et al. 2020) / This study	ATATAG/ATATAC
22		LTRM19	11608468	11609492	LTR5A	(Xue et al. 2020) / This study	CTCAC
22	22q11.23b	De7;K16 ERVL-MalR (MLT1C); ERVK16; previously mapped to 9q34.11a	23510352	23510352	unknown	(Wildschutte et al.; 2016)	AAACG
22	-	LTRW24	23536062	23537250	LTR5B	(wysocka et al.; 2018)	GGCTAG/GGGTAG
chrUn_KI270749v1	-	DeNe3	63347	63347	unknown	(Lee et al.; 2014)	UNKNOWN
X	-	LTRM17	987530	988060	LTR5B	This study	5' TRUNCATED
X	-	LTRM8	1170893	1171940	LTR5A	This study	CTGAGG
X	-	LTRW25	108376867	108377393	LTR5B	(wysocka et al.; 2018)	5' TRUNCATED
Y	Yq11.223	solo-Yq11.223	23880366	23881090	LTR5B	(Xue et al.; 2020)	GAAAT
Y	Yq11.23	solo-Yq11.23	25789017	25789741	LTR5B	(Xue et al.; 2020)	GAAAT

3.1.ii. Segmental duplications.

The analyses of flanking sequences identified a high degree of similarity for 199 of the HERV-K(HML-2) insertions in the human genome (see table 11). Forty-six distinct segmentally duplicated groups were identifiable. These groups form tandem duplications, where duplicated loci are located in a continuous fashion within one chromosome, duplications across different chromosomes and mixed groups, where solo LTRs appear to have proliferated through the genome as segmental duplications of full-length elements. Recognition of all insertion within groups of segmental duplications and their description is important in light of analysis of potential HERV-K(HML-2) recombinational activity (see 3.1.iii), since the paired TSDs within these groups needs to be excluded from further analysis – pairing of the similar TSDs within groups would produce false-positive TSD exchange signals. Detailed list can be found in appendix 3.

Table 11 Groupings of 199 HERV-K(HML-2) insertions, that proliferated through the genome via segmental duplications. For an extended version, please see appendix 3.

Duplication no.	Member loci	Status / TSD
1	15q25.2, Yq11.23a, Yq11.23b	5' AND 3' TRUNCATED
2	16p11.2, LTRW15	5' TRUNCATED
3	20p11.21a, 20p11.21b	3' TRUNCATED
4	HML2.LTR205, HML2.LTR381, LTRW19	3' TRUNCATED
5	1p36.21b, 1p36.21c	AGCGTA/AGTGTA
6	HML2.LTR50, HML2.LTR52, HML2.LTR55, HML2.LTR60, HML2.LTR62, HML2.LTR63	CTGAAC
7	HML2.LTR54,HML2.LTR61	FL, NON-MATCHING TSD
8	LTRM9, LTRW2	FL, NON-MATCHING TSD
9	HML2.LTR527, HML2.LTR529, HML2.LTR530	AATGAT/GATGAT
10	11p15.4, 4p16.1b, 8p23.1b, 8p23.1c, 8p23.1d, HML2.LTR195, HML2.LTR579, HML2.LTR601, 4p16.1a, 4p16.3b	ATTTG/ATTTTG
11	HML2.LTR544,HML2.LTR367,HML2.LTR193,HML2.LTR172,HML2.LTR581,HML2.LTR187,HML2.LTR577,HML2.LTR218,HML2.LTR583,HML2.LTR366,HML2.LTR222,HML2.LTR227,HML2.LTR219,HML2.LTR545,HML2.LTR576,HML2.LTR388,HML2.LTR224,HML2.LTR192,HML2.LTR418,HML2.LTR405,HML2.LTR223,HML2.LTR230,HML2.LTR226,HML2.LTR431,HML2.LTR435,HML2.LTR221,HML2.LTR419,HML2.LTR584,HML2.LTR406,HML2.LTR432,HML2.LTR174,HML2.LTR603,HML2.LTR188,HML2.LTR228,HML2.LTR602,HML2.LTR585	AATGAG/AGTGAA/AGCGAG/AGTGAG
12	HML2.LTR555,HML2.LTR556	GTTCA
13	HML2.LTR578,HML2.LTR194,HML2.LTR173,HML2.LTR368,HML2.LTR220	ATCAA/ATCAG/GTCAG
14	11p15.4a,3p12.3a,8p23.1g,4p16.1d,4p16.1c,8p23.1f,LTRM15,LTRM7,4p16.3,8p23.1e,3q21.2a	5' TRUNCATED
15	HML2.LTR600,HML2.LTR428,HML2.LTR412,HML2.LTR434,HML2.LTR433,HML2.LTR225,HML2.LTR421,HML2.LTR426,HML2.LTR425,HML2.LTR424,HML2.LTR423,HML2.LTR422,HML2.LTR411,HML2.LTR410,HML2.LTR409,HML2.LTR408,HML2.LTR407,HML2.LTR429,HML2.LTR420,HML2.LTR417,HML2.LTR413,HML2.LTR430,HML2.LTR416,HML2.LTR415,HML2.LTR414,HML2.LTR427	CTTGGA/CGTGGA
16	HML2.LTR634,HML2.LTR874	GATTTC
17	HML2.LTR637,HML2.LTR638	AGGCT
18	LTRW13,HML2.LTR888,HML2.LTR887,HML2.LTR890,HML2.LTR886,HML2.LTR891,HML2.LTR894,HML2.LTR859	GGTCCC
19	HML2.LTR649,HML2.LTR131,HML2.LTR482,HML2.LTR486,HML2.LTR873,HML2.LTR680,HML2.LTR746	TATTT
20	HML2.LTR650,HML2.LTR884,HML2.LTR651,HML2.LTR130	CTGTT/GTGTT
21	HML2.LTR656,HML2.LTR657	GCTATA
22	HML2.LTR707,LTRM19	CTCAC
23	HML2.LTR722,HML2.LTR728,HML2.LTR725	GGTGGG
24	HML2.LTR735,HML2.LTR738	CCACAC
25	HML2.LTR743,HML2.LTR509	FL, NON-MATCHING TSD
26	HML2.LTR748,HML2.LTR234	CCTGTG
27	HML2.LTR826,HML2.LTR830	CTGGAT
28	HML2.LTR116,HML2.LTR479,HML2.LTR478,HML2.LTR480,HML2.LTR481	TGTGG
29	HML2.LTR124,HML2.LTR125	TCGTGG
30	LTRM2,HML2.LTR175	FL, NON-MATCHING TSD
31	LTRW22,LTRM6,HML2.LTR936,LTRM5	ATATAG/ATATAC
32	HML2.LTR882,LTRW23	AATCC
33	5q33.3,HML2.LTR310,HML2.LTR354	ACTGC/TCTGC/TCTAC
34	HML2.LTR384,HML2.LTR386	CAATCC/TAATCC
35	HML2.LTR477,HML2.LTR494	TAGAA/CAGAA
36	HML2.LTR484,HML2.LTR487	TCCACC
37	HML2.LTR896,HML2.LTR928	FL, NON-MATCHING TSD
38	HML2.LTR897,HML2.LTR929	CTCTG/CACTG
39	HML2.LTR898,HML2.LTR930	GCCAGC/GCCAAC
40	HML2.LTR911,HML2.LTR931	CCTAC
41	HML2.LTR913,HML2.LTR932	TCTATG
42	HML2.LTR944,HML2.LTR946,HML2.LTR945	CCTTTT
43	Yq11.223,Yq11.23	GAAAT
44	HML2.LTR810,HML2.LTR785	GGCTG
45	HML2.LTR721,HML2.LTR729,HML2.LTR868	CTTTGG/CTTTTG/CTTTGA
46	4q35.2,HML2.LTR213	AACGT/AACAT

Figure 17 shows HERV-K insertions formed by segmental duplications across the human genome. Furthermore, Figure 18a shows the number of HERV-K insertions formed by segmental duplications on each chromosome. It is evident that most HERV-K(HML-2) elements that have undergone segmental duplications are located on chromosome 8. However, according to She *et al.* (2004), only 2.04% of chromosome 8 contains segmental duplication, in contrast to 4.90% for the entire human genome. Therefore, it is surprising, that HERV-K(HML-2) elements mobilized via segmental duplications are concentrated on chromosome 8. Subramanian *et al.* (2011) also observed that chromosome 8 is unusually provirus-rich, but they did not analyse them in terms of duplications and speculated that chromosome 8 might be preferential for maintaining proviruses.

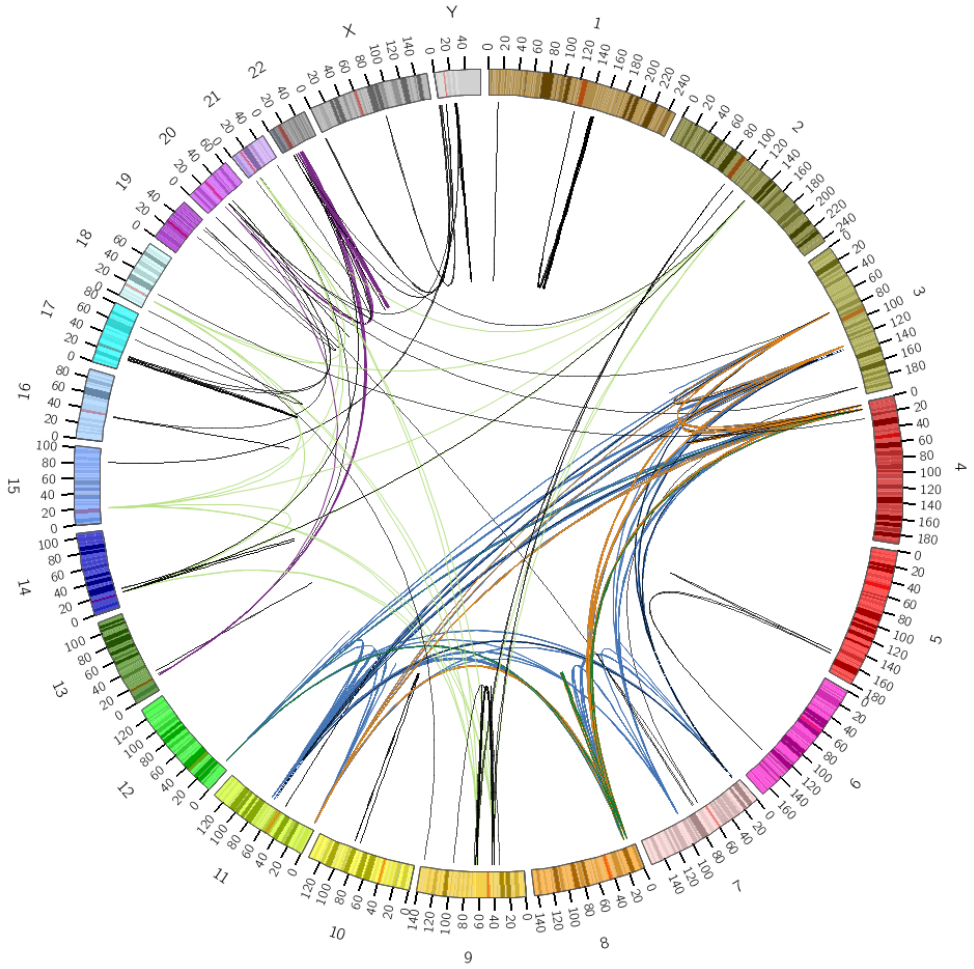

Figure 17 Groupings of segmentally duplicated HERV-K(HML-2) insertions across the human genome. Outer symbols represent chromosomes, whereas inner numbers approximate locations. Insertions that belong to 46 segmental duplication groups are connected within their group with coloured lines.

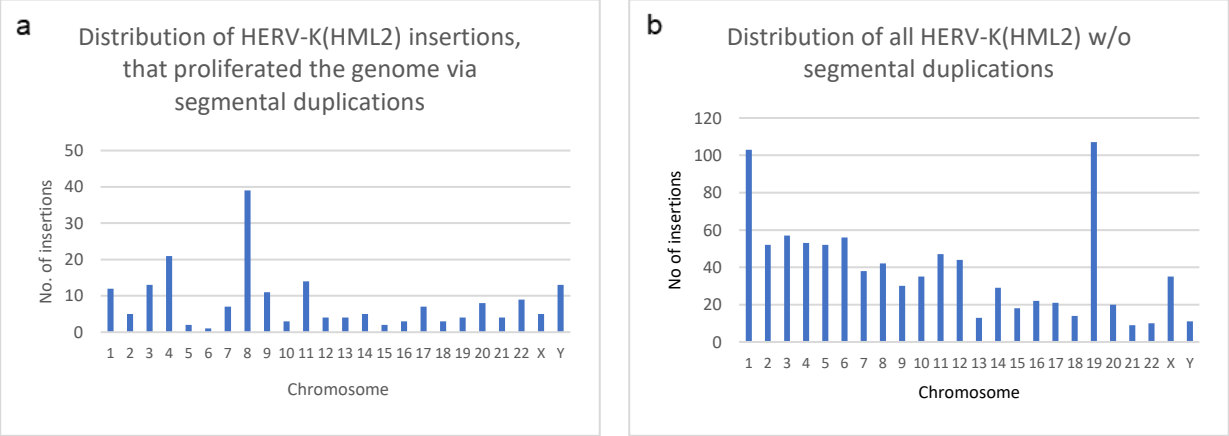

Figure 18 a) Distribution of 199 (46 segmental duplication groups) HERV-K insertions formed by segmental duplications in human reference genome Hg38. b) Distribution of the remaining HERV-K(HML-2) loci across the human reference genome Hg38.

3.1.iii. Recombinational activity between chromosomes.

In order to analyse possible recombinational activity (see section 1.13), in which HERV-K(HML-2) is involved, similarities between TSD insertions were explored. Table 12 shows the *all possible combinations of TSDs* between different HERV-K(HML-2) loci in the genome, that could result in the observed HERV-K(HML-2) loci with non-matching TSDs. These numbers give an estimation of the number of TSD exchange events, that have occurred over the course of evolution within the HERV-K(HML-2) family of endogenous retroviruses (for detailed methodology, see section 2.3.ii-2.3.v). The results have been categorised according to the state (full-length or truncated)**.** Figure 19 illustrates all of the possible variants of TSD match compared in this analysis. Insertions exhibiting identical TSDs on both ends in different loci, along with high degree of similarity in extended flanking sequence, were categorised as segmental duplications with matching TSDs (Table 12, A2). Insertions with non-matching TSDs were also observed in segmental duplications, by comparing the degree of identity for the extended flanks (Table 12, B2). Some of the 5' and 3' truncated insertions are also observed in segmental duplications (Table 12, C2 and D2 respectively). All of the above segmental duplications were presented in more detail in chapter 3.1.ii. Insertions that had identical TSDs on both sides and shared at least one TSD with another locus were marked as involved in TSD exchange (Table 12, A3). Presence of these insertions within at least some of the observed pairs of TSDs suggest TSD exchange events, where the element exhibiting identical TSDs is the donor of the TSD. For each of the loci with non-matching TSDs (table 12, 3B), a TSD match can be observed to another locus (whether that be truncated or a full-length element). This indicates, that all of the reported elements with non-matching TSDs are in fact involved in TSD exchanges with other insertions reported in Table 12, most likely elements exhibiting identical TSDs. For the truncated insertions the exact TSD sequence cannot be established (only one end of the LTR can be observed). However, by comparing the immediate flanking sequence (10bp) of the available end of the

truncated insertion, with all known TSDs and extending flanking sequences of other truncated insertions, possible TSD matches with another insertion were explored. These elements can also be considered donors of TSD exchange events, however their exact state cannot be determined, since there is a chance that they exhibited non-matching TSD sequences before mutations that caused the truncation. For the 5' truncated loci these are represented in table 12, C3 and for the 3' truncated insertions these are represented in table 12, D3. The remainder of the HERV-K(HML-2) insertions that display truly unique TSDs (only one occurrence in the genome) are counted in row 1 of table 11. Unique TSDs are the most expected group of insertions, since there are no known mechanisms, that would bias towards formation of certain TSD sequences. Insertions truncated on both the 5' and 3' ends did not give enough information to conduct analysis due to no observable TSDs (table 12, E1).

Table 12 HERV-K(HML-2) loci potentially involved in TSD exchange in the human reference genome hg38.

	Matching TSDs (A)	Non-matching TSDs (B)	5' Truncated insertions (C)	3' truncated insertions (D)	5'+3' truncated insertions (E)
Insertions not involved in any activity (1)	496	0	0	0	12
Insertions involved in segmental duplications (2)	167	11	13	8	0
Insertions involved in TSD exchange (3)	266	69	41	35	0
Total = 1118	929	80	54	43	12

The first row represents categories of insertions: insertions that are not truncated and have matching Terminal Site Duplications (TSDs) (A), insertions that are not truncated but the 5' TSD and the 3' TSD don't match (B), insertions that are truncated on the 5' end (C), insertions that are truncated on the 3' end (D), insertions truncated both on 5' and 3' ends (E). Subsequent rows represent insertions that have not been involved in any activity (1), insertions found to be involved in segmental duplications (2) and insertions involved in TSD exchange (3).

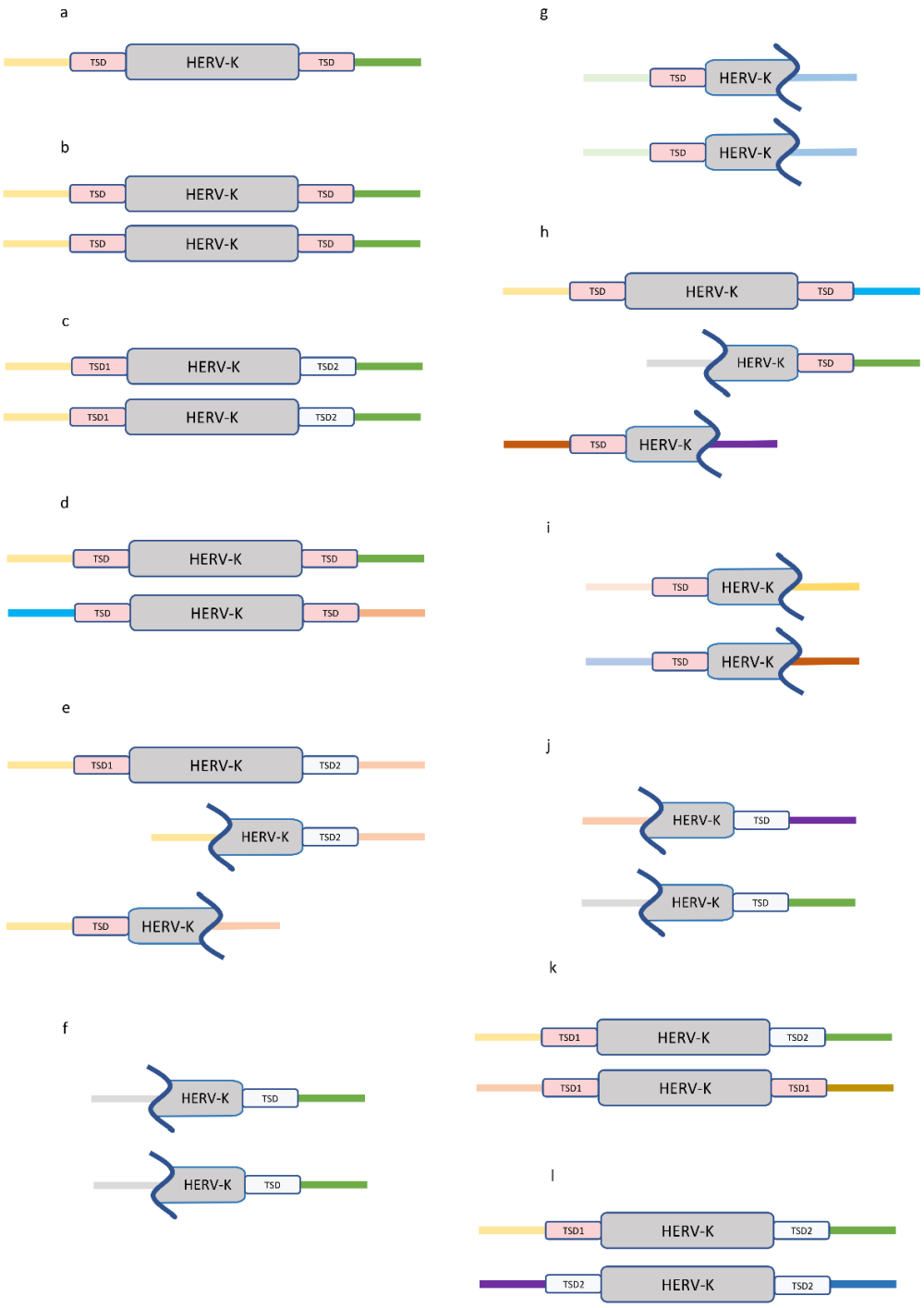

Figure 19 Examples of variants of matching TSDs between different categories of insertions. a) Insertions not involved in any activity, with matching TSDs. b) Two insertions with matching TSDs in a segmental duplication, matching TSDs and extended flanking sequences c) Two insertions with non-matching TSDs in a segmental duplication, matching extended flanking sequences. d) Two insertions with matching TSDs but extended flanking sequences don't match. e) 5' and 3' truncated insertions involved in a segmental duplication with a full-length insertion, with different TSDs. f) Two 5' truncated elements involved in a segmental duplication. g) Two 3' truncated elements involved in a segmental duplication. h) 5' and 3' truncated insertions involved in a TSD exchange with a full-length insertion, with matching TSDs on both ends. i) 3' truncated insertions with matching TSD sequences on their untruncated end. j) 5' truncated insertions with matching TSD sequences on their untruncated end. k) 5' end of an insertion with non-matching TSDs, matching to a TSD sequence of an insertion with matching TSDs. l) 3' end of an insertion with non-matching TSDs, matching to a TSD sequence of an insertion with matching TSDs.

If we consider a pair of loci, where one locus has identical TSDs and the second locus has non-matching TSDs, but shares one TSD with the first, we can define those loci as “donors” and “acceptors” of TSD sequences in exchange events (see figure 20). Thirty-one pairs of insertions representing loci with matching TSDs (donors of recognized TSD sequences) and non-matching TSDs (acceptors of TSDs) have shown evidence of such events occurring. For the remaining 38 loci with non-matching TSDs (table 12, B3 indicates 69 loci, 69 – 31 = 38), it was not possible to determine the source of TSD mismatch.

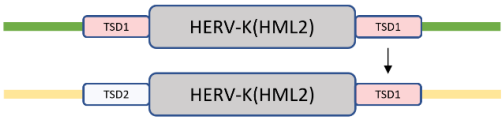

Figure 20 TSD exchange event, where one of the insertions with a well-defined TSD sequence (donor, top) shares that TSD with another insertion (acceptor, bottom), resulting in a non-matching TSD in the acceptor.

Table 13 The results of a two-sample Kolmogorov-Smirnov Test for the distribution of selected pairs of donor/acceptor elements indicating possible TSD exchange events, tested against the distribution of all known HERV-K(HML-2) elements in the human genome.

Donor elements		Acceptor elements	
α	0.05	α	0.05
D-stat	0.1228	D-stat	0.0707
D-crit	0.3325	D-crit	0.3325
Is significant?	**NO**	Is significant?	**NO**

The distributions of donors seem to be concentrated in chromosome 1, whereas acceptor elements appear to be enriched in chromosome 19. This allocation however correlates with the distribution of all of HERV-K(HML-2) elements (see figure 21) and the distributions have not found to be significantly different from each other via two-sample Kolmogorov-Smirnov Test (see table 13).

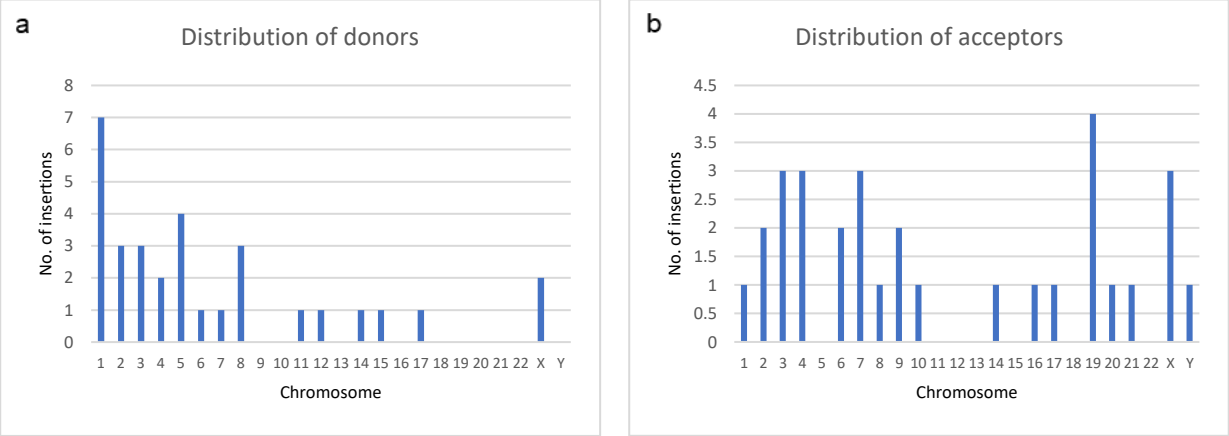

Figure 21 Distribution of donor (a) and acceptor (b) HERV-K(HML-2) elements, selected in possible TSD exchange events

3.1.iv. HERV-K(HML-2) insertion rate in humans.

The insertion rate was calculated according to section 2.3.vi. The human-specific insertion count did not include segmentally duplicated elements, since these were found to be parts of larger duplicated segments of the genome, therefore duplication events have most likely not occurred due to HERV-K(HML-2) activity, but external factors. Overall 353 elements have been found to be human-specific, out of which 248 did not form part of a segmental duplication. Therefore, the estimated insertion rate of HERV-K(HML-2) in humans is:

$$\mu = \frac{248}{6.7 * 10^6/20} \approx 7.4 * 10^{-4}$$

3.1.v. Summary of the state of HERV-K(HML-2) in the reference genome.

In the abovementioned part of the study, a comprehensive review of the state of HERV-K(HML-2) was performed. All of the information about known HERV-K(HML-2) loci to date was collated, and additionally 10 completely new solo LTRs and 2 truncated full-length elements are reported. Further, numerous elements mentioned in this section were studied and reclassified to create an accurate database of known HERV-K(HML-2) proviruses – the full database (see section 3.1.i).

199 proviruses were observed to proliferate through the genome via segmental duplications, showing an unexpected concentration on chromosome 8. Only 2.04% of chromosome 8 consist of segmental duplications, comparing to 4.9% of the whole genome (She *et al.*, 2004), whereas chromosome 8 contains the largest number of segmentally duplicated HERV-K(HML-2) loci, making it unusually rich in proviruses. TSD exchange can be observed for the studied family of proviruses and pairs of TSD donors and acceptors were identified. Their distribution across the human genome shows no evidence of Master gene (see chapter 1.14) for the HERV-K(HML-2) family of endogenous retroviruses (Johnson and Brookfield, 2006).

3.2. HERV-K(HML-2) in cancer.

Analysis of 11 projects that collate cancer data from The Cancer Genome Atlas (TCGA) uncovered 28 non-reference HERV-K(HML-2) insertions, 21 out of which have previously been found in the literature and 7, at the time of the analysis, was found to be novel HERV-K(HML-2) insertions. 20 of the detected polymorphic HERV-K(HML-2) loci were found in several different projects and 8 appear to be specific to one type of cancer type/project, as detailed in table 14.

Detected loci display recognisable TSDs and, apart from two inserts, do not contain any mutations within the TSD sequences. Completely novel insertions do not share their TSDs with any of the previously known ones. Analysis of TSDs and extended flanks of detected insertions suggest that HERV-K-c5 and HERV-K-c16 appear have proliferated through the genome via segmental duplication. The extended flanks of these insertions are almost identical but due to sequencing technology limitations, which is 100-120bp per read for the analysed data, it is impossible to precisely determine the nature of these inserts.

Allele frequencies for 21 of the 28 insertions detected in cancer genomes (previously reported by Wildschutte *et al.* (2016)), were calculated from the data from the 1000 Genomes Project (1KGP) and the Human Genome Diversity Project (HGDP) (see 2.4.iv for details). The P-values of allele frequency distributions of these 21 non-reference insertions provided by Wildschutte *et al.* (2016) in general populations using Fishers Exact test are presented in table 15. Significance of each of the tested frequency (p-value < 0.05) ascertain the usefulness of these frequencies in comparison with cancer populations. The results of that statistical analysis, again using Fishers Exact test, is presented in table 16. The table marks insertions, that display difference in frequency of occurrence within the tested cancer populations (top row) for tumor (T) and non-tumor (N) control samples, comparing the occurrence with frequencies observed in 1KGP and HGDP. The final result includes 10 polymorphic loci, that display significantly different frequencies of occurrence in cancer, comparing to normal populations, after applying the Bonferroni correction for multiple testing (Bonferroni, 1936).

Table 14 A list of nonreference HERV-K(HML-2) loci detected in various cancer projects. References point to publications, that reported the loci for the first time. 7 of the detected locations are novel at the time of the PhD study and specific to the tested population of cancers.

Name	Project	Reference ver	Chromosome	Orientation	Breakpoint	TSD	Reference
HERV-K-c1	neuroendocrine	NCBI36	1	-	105817397	CAAAAT	(Marchi et al.; 2014)
HERV-K-c2	prostate, cervix , hepatocellular cellular , melanoma, myeloma , neuroendocrine , breast , advanced cancers, breakpoint	GRCh37	1	-	111802591	CTTTTT	(Agoni et al.; 2012)
HERV-K-c3	myeloma, prostate	NCBI36	1	+	221644932	AAATAC	(Lee et al.; 2012)
HERV-K-c4	prostate	GRCh37	3	-	164057183	TCTTGG	**This study**
HERV-K-c5	prostate, myeloma , neuroendocrine, breast	NCBI36	3	+	199330196	ATATAG	**This study**
HERV-K-c6	myeloma , pancreas, cervix , neuroendocrine , hepatocellular , melanoma, breast, prostate, advanced cancers, breakpoint	NCBI36	4	-	9212337	GAACT	(Lee et al.; 2012)
HERV-K-c7	prostate, hepatocellular , myeloma, breast	GRCh37	5	+	64388445	GACGTG	(Lee et al.; 2014)
HERV-K-c8	cervix , hepatocellular cellular , melanoma, myeloma, prostate	GRCh37	5	-	80442199	GACCAG	(Agoni et al.; 2012)
HERV-K-c9	breast	GRCh37	5	-	144596332	CATGAG	**This study**
HERV-K-c10	melanoma, myeloma, prostate	GRCh37	6	+	16004863	GCCAC	(Wildschutte et al.; 2016)
HERV-K-c11	prostate, breast, cervix, melanoma, myeloma, neuroendocrine, hepatocellular , breakpoint , advanced cancers	GRCh37	6	+	32643464	CATCCT	(Xue et al.; 2020)
HERV-K-c12	prostate, breast, hepatocellular cellular , melanoma, myeloma, cervix, Advanced_cancers	GRCh37	6	+	32648040	ACTTC	(MARCHI et al.; 2014)
HERV-K-c13	cerivx, hepatocellular cellular , melanoma, myeloma, neuroendocrine , breast, prostate, advanced cancers, breakpoint	NCBI36	6	+	161190894	TACGCC	(Wildschutte et al.; 2016)
HERV-K-c14	breast, cervix, melanoma, neuro, myeloma, neuroendocrine , hepatocellular cellular, prostate , breakpoint	GRCh37	9	+	132205207	AAACG	(Agoni et al.; 2012)
HERV-K-c15	breast, myeloma, prostate	GRCh37	11	-	60499889	CATTTT	(Agoni et al.; 2012)
HERV-K-c16	prostate, melanoma, myeloma , breakpoint	NCBI36	12	+	34216577	ATATAG	**This study**
HERV-K-c17	prostate, melanoma, breast,cervix, hepatocellular, myeloma, neuroendocrine , advanced cancers	GRCh37	12	+	44313661	GTGGT	(Lee et al.; 2014)
HERV-K-c18	cervix, hepatocellular cellular, melanoma, myeloma, neuroendocrine , prostate, advanced cancers, breast	GRCh37	12	-	124066476	ACGAAC	(Lee et al.; 2012)
HERV-K-c19	prostate, cervix, hepatocellular , myeloma, neuroendocrine , advanced cancers	GRCh37	13	-	90743182	AGTAGT	(Agoni et al.; 2012)
HERV-K-c20	prostate, myeloma, neuroendocrine , breast, cervix, hepatocellular, pancreas , melanoma , advanced cancers, breakpoint	GRCh37	15	-	63374593	TTTTTC	(Lee et al.; 2012)
HERV-K-c21	prostate	GRCh37	17	-	28891180	GGTGGC/GGTGGG	**This study**
HERV-K-c22	prostate, cervix, hepatocellular , myeloma, neuroendocrine , breast, advanced cancers, breakpoint	GRCh37	19	-	21841535	CTCTAT	(Turner et al.; 2001)
HERV-K-c23	prostate	GRCh37	19	+	22466363	GCTCA	(Agoni et al.; 2012)
HERV-K-c24	prostate, hepatocellular , neuroendocrine , breast, melanoma, myeloma, cervix, Advanced_cancers, Breakpoint	GRCh37	19	-	29855780	TTACCA/TCACCA	(Agoni et al.; 2012)
HERV-K-c25	hepatocellular	GRCh37	20	+	12402391	AGTGG	(Hughes and Coffin; 2001)
HERV-K-c26	hepatocellular	GRCh37	X	-	93606607	ATAAT	(Agoni et al.; 2012)
HERV-K-c27	prostate	GRCh37	X	-	120213910	CCAGTA	**This study**
HERV-K-c28	prostate, myeloma	NCBI36	Y	+	10566775	CTATGT	**This study**

Table 15 P-values of distributions of non-reference insertions studied in Wildschutte *et al.* (2016).

Insertion	P	Insertion	P
HERV-K-c1	1.00E-06	HERV-K-c15	1.00E-06
HERV-K-c2	8.40E-05	HERV-K-c17	1.00E-06
HERV-K-c3	1.00E-06	HERV-K-c18	1.00E-06
HERV-K-c6	1.00E-06	HERV-K-c19	1.00E-06
HERV-K-c7	1.00E-06	HERV-K-c20	1.00E-06
HERV-K-c8	1.00E-06	HERV-K-c22	1.00E-06
HERV-K-c10	0.04096	HERV-K-c23	1.00E-06
HERV-K-c12	1.00E-06	HERV-K-c24	1.00E-06
HERV-K-c13	1.00E-06	HERV-K-c25	1.00E-06
HERV-K-c14	1.00E-06	HERV-K-c26	1.00E-06

The +-250kb vicinity of the location of each detected insertion was studied to detect genes, that might be influenced by HERV-K(HML-2) promoter activity. As stated in methodology (see section 2.4.iii for details), the +-250kb locations in vicinity of detected insertions was used in UCSC genome browser, to obtain a list of genes in that location. These genes were then used in Stringdb and Webgestalt gene/protein interaction databases to obtain a list of elements interacting with these genes. The literature database OpenTargetsPlatform was queried with those gene names and outputted a database of references involving genes used as query. Each entry contained a large amount of metadata, including diseases described in the particular study involving the queried gene and references to Reactome pathway database. Using both of these proprieties the list was narrowed down to studies referencing specific pathways, with regard to queried gene and cancer. The pathway names were extracted, collated and analysed using various literature references to select and describe metabolic processes known to be active/altered in cancer and found to be possibly influenced by the detected HERV-K(HML-2) insertions. A summary of pathways detected for found genes, especially emphasizing pathways important in various cancers is reported in table 17 below.

Table 16 P values computed for occurrence of nonreference HERV-K(HML-2) loci in the cancer genomes, compared to data derived from Wildschutte et al. (2016) for Human Genome Diversity Project (HGDP) and 1000 Genomes Project (1KGP). The green colour indicates cases, where both results are statistically significant ($P < 0.05$, after applying Bonferroni correction (Bonferroni, 1936) for multiple testing: $P < 1.1E-04$, values in red), yellow shows cases, where difference is significant only against one of the normal populations and no colouring shows results that do not appear to be statistically significant.

	Breast (40)				Cervix (28)				Hepatocellular (60)				Melanoma (32)				Myeloma(50)			
	T(20)		N(20)		T(14)		N(14)		T (30)		N (30)		T(8)		N(25)		T(25)		N(25)	
	HGDP	1KGP	HGDP	1KGP	HGDP	1KGP	HGDP	1KGP	HGDP	1KGP	HGDP	1KGP	HGDP	1KGP	HGDP	1KGP	HGDP	1KGP	HGDP	1KGP
HERV-K-c1																				
HERV-K-c2	1.9040E-01	3.8720E-02	1.3090E-01	1.6750E-02	2.3680E-03	1.8350E-04	2.3680E-03	1.8350E-04	3.3210E-02	1.5940E-03	3.3210E-02	1.5940E-03	6.0120E-01	4.5450E-01	2.2400E-01	3.3160E-02	5.3410E-02	2.6740E-03	1.2140E-01	1.5570E-02
HERV-K-c3																	1.0000E+00	4.4240E-01	1.0000E+00	4.4240E-01
HERV-K-c6	8.8430E-03	9.3110E-04	3.9650E-02	8.3470E-03	5.5320E-02	1.9680E-02	5.5320E-02	1.9680E-02	7.3940E-02	1.1220E-02	3.6810E-02	5.0750E-03	5.8660E-01	4.4170E-01	2.3080E-01	1.0940E-01	1.1990E-01	2.9040E-02	1.6810E-01	5.8490E-02
HERV-K-c7	2.7400E-01	7.2130E-01	2.7400E-01	7.2130E-01					4.5710E-02	1.0000E+00	4.5710E-02	1.0000E+00					3.2050E-01	5.1660E-01	3.2050E-01	5.1660E-01
HERV-K-c8					1.0000E+00	5.3420E-01	1.5860E-01	3.9430E-02	2.8650E-01	2.3720E-02	7.0410E-01	2.2090E-01			6.8100E-01	1.5310E-01	6.8100E-01	1.5310E-01	6.8100E-01	1.5310E-01
HERV-K-c10													1.3110E-01	4.1780E-02	1.0130E-01	8.5140E-03	1.0000E+00	1.0000E+00	3.2050E-01	1.2240E-01
HERV-K-c12	4.9970E-02	4.3950E-03	7.3230E-03	2.0860E-04	2.6760E-02	2.8990E-03	6.4520E-03	5.7990E-04	1.5720E-04	1.0680E-07	1.5720E-04	1.0680E-07	1.3780E-01	6.5260E-02	2.3130E-02	5.6060E-04	3.2810E-01	7.5520E-02	6.9850E-01	2.3810E-01
HERV-K-c13	7.8670E-06	4.2560E-05	7.8670E-06	4.2560E-05	1.5980E-07	1.0210E-06	2.5270E-08	1.3920E-07	2.5000E-09	7.1130E-10	1.9090E-08	2.1260E-08	2.0080E-03	1.8500E-02	8.4840E-07	3.3230E-06	2.8470E-07	9.4120E-07	8.4840E-07	3.3230E-06
HERV-K-c14	8.4590E-01	2.2000E-16	6.9640E-01	2.2000E-16	1.9330E-01	2.2000E-16	3.7630E-01	2.2000E-16	4.6530E-02	2.2000E-16	6.7680E-02	2.2000E-16	2.6090E-01	2.2000E-16	7.9410E-02	2.2000E-16	2.1390E-01	2.2000E-16	1.5530E-01	2.2000E-16
HERV-K-c15	3.1060E-01	3.7160E-02	1.8810E-01	1.0120E-02													4.3770E-01	7.2820E-01	4.3770E-01	7.2820E-01
HERV-K-c17	5.9670E-02	1.9770E-01	3.3080E-02	9.5200E-02	8.2990E-03	4.3260E-02	3.2150E-02	1.2050E-01	4.6770E-03	2.2150E-02	4.6770E-03	2.2150E-02	5.8110E-03	1.7590E-02	4.0430E-04	1.3040E-03	1.0160E-02	4.5680E-02	1.0160E-02	4.5680E-02
HERV-K-c18	4.2260E-02	1.0750E-11	4.2260E-02	1.0750E-11	1.0000E+00	6.8370E-05	7.6290E-01	8.8740E-06	3.2530E-01	7.9590E-13	3.2530E-01	7.9590E-13	1.0000E+00	1.6530E-03	6.1000E-01	4.5370E-10	6.1000E-01	4.5370E-10	6.1000E-01	4.5370E-10
HERV-K-c19					3.5050E-01	2.9910E-01	5.5240E-01	7.9080E-01	7.7420E-02	6.9210E-02	1.4430E-01	1.5020E-01					1.9280E-01	2.3130E-01	1.9280E-01	2.3130E-01
HERV-K-c20	1.6250E-05	7.6550E-08	1.7310E-04	2.0130E-06	1.1620E-03	5.0370E-05	2.6190E-03	2.2660E-04	4.9160E-05	2.8280E-08	1.0930E-06	4.8830E-11	1.0420E-03	2.3400E-04	1.6520E-05	2.3220E-08	1.4110E-04	4.9420E-07	1.4110E-04	4.9420E-07
HERV-K-c22	7.8040E-02	5.3320E-01	4.2290E-02	2.2010E-01	3.9500E-02	2.1220E-01	8.5220E-03	4.6930E-02	3.5630E-04	5.9440E-03	9.7750E-05	1.0920E-03					2.0320E-02	1.9230E-01	2.0320E-02	1.9230E-01
HERV-K-c23																				
HERV-K-c24	1.9730E-01	2.6790E-01	9.0410E-02	8.0250E-02	2.9540E-01	4.4760E-01	8.9320E-02	1.2640E-01	5.1580E-03	3.6280E-03	5.1580E-03	3.6280E-03	1.1230E-01	1.3930E-01	2.2700E-01	3.1620E-01	8.1980E-03	4.0920E-03	8.1980E-03	4.0920E-03
HERV-K-c25									4.2370E-01	1.0000E+00	4.2370E-01	1.0000E+00								
HERV-K-c26									1.0000E+00	6.1490E-01	1.0000E+00	6.1490E-01								

	Neuroendocrine (24)				Pancreas (2)				Prostate (120)				ADVC (18)				Breakpoint (12)			
	T(12)		N(12)		T(1)		N(1)		T(60)		N(60)		T(9)		N(9)		T(6)		N(6)	
	HGDP	1KGP	HGDP	1KGP	HGDP	1KGP	HGDP	1KGP	HGDP	1KGP	HGDP	1KGP	HGDP	1KGP	HGDP	1KGP	HGDP	1KGP	HGDP	1KGP
HERV-K-c1	1.0000E+00	1.0000E+00	1.0000E+00	1.0000E+00																
HERV-K-c2	8.2240E-01	5.4130E-01	3.6330E-01	1.5170E-01					3.2140E-03	2.4760E-07	8.0790E-03	1.8130E-06	2.0310E-01	9.6330E-02	4.4360E-01	2.3790E-01	5.4600E-01	3.8770E-01	2.2780E-01	1.4600E-01
HERV-K-c3									1.8900E-01	1.0000E+00	1.8900E-01	1.0000E+00								
HERV-K-c6	3.6830E-01	2.9490E-01	1.1280E-01	5.8410E-02	1.0000E+00	1.0000E+00	1.0000E+00	1.0000E+00	1.0860E-01	7.8470E-03	8.2450E-02	4.4010E-03	1.2190E-01	5.6060E-02	1.2190E-01	5.6060E-02	2.9170E-02	1.4710E-02	2.3230E-01	2.3380E-01
HERV-K-c7									7.6680E-03	3.8900E-01	3.0960E-02	8.3230E-01								
HERV-K-c8									7.0870E-01	1.0000E+00	7.0870E-01	1.0000E+00								
HERV-K-c10									1.0000E+00	2.7150E-01	1.0000E+00	2.7150E-01								
HERV-K-c12									2.2490E-06	4.5040E-14	2.5660E-11	2.2000E-16	8.6240E-02	2.5940E-02	2.0060E-02	5.3230E-03				
HERV-K-c13	5.1160E-04	4.4340E-03	5.1160E-04	4.4340E-03					3.5720E-11	9.1970E-15	3.5720E-11	9.1970E-15	9.0020E-06	7.1560E-05	5.8030E-05	4.4280E-04	4.3520E-03	2.0230E-02	1.9150E-02	9.5650E-02
HERV-K-c14	4.7880E-01	2.2000E-16	2.4660E-01	2.2000E-16					1.0140E-01	2.2000E-16	7.6630E-02	2.2000E-16					7.4920E-01	1.1610E-08	7.4920E-01	1.1610E-08
HERV-K-c15									2.7310E-02	4.3760E-02	2.7310E-02	4.3760E-02								
HERV-K-c17	2.2770E-02	9.1650E-02	2.2770E-02	9.1650E-02					9.5700E-05	1.2710E-04	3.9030E-05	4.8500E-05	2.3510E-02	9.3500E-02	2.3510E-02	9.3500E-02				
HERV-K-c18	7.5540E-01	6.5200E-04	3.0280E-01	7.0410E-07					1.0000E+00	2.2000E-16	1.0000E+00	2.2000E-16	7.2890E-01	3.3060E-03	1.0000E+00	4.8140E-04				
HERV-K-c19	7.5720E-01	1.0000E+00	1.8170E-01	1.6060E-01					1.6860E-03	4.4300E-05	4.0470E-04	1.1150E-05	7.3480E-01	5.0640E-01	1.0000E+00	7.4900E-01				
HERV-K-c20	3.0720E-03	4.1790E-04	9.2080E-06	2.8740E-07	4.5310E-01	4.2880E-01	4.5310E-01	4.2880E-01	2.1370E-06	1.2840E-13	5.9560E-07	7.0630E-15	1.5670E-03	1.8730E-04	1.5670E-03	1.8730E-04	5.9070E-03	2.3970E-03	3.7460E-02	1.2590E-02
HERV-K-c22	1.8820E-01	6.0350E-01	1.8820E-01	6.0350E-01					2.0670E-06	1.4560E-05	4.3500E-06	4.9520E-05	6.9460E-02	2.2750E-01	6.9460E-02	2.2750E-01	2.8960E-01	7.0560E-01	2.8960E-01	7.0560E-01
HERV-K-c23									1.2520E-02	1.6990E-08	7.0790E-03	1.7930E-09								
HERV-K-c24	4.9720E-01	6.8100E-01	6.5740E-01	1.0000E+00					5.7130E-02	3.2420E-02	7.7790E-02	5.0900E-02	1.9540E-02	2.9280E-02	7.0600E-02	9.3370E-02	7.6910E-01	1.0000E+00	7.6910E-01	1.0000E+00
HERV-K-c25																				
HERV-K-c26																				

Table 17 A list of detected genes influencing molecular pathways reported to possibly be influenced by studied HERV-K(HML-2) elements in cancer.

Insert	Gene in 250kb range	orientation insert	Insert location	orientation gene	5' genomic location of gene	3' genomic location of gene	Distance	Pathways	Gene interacting
HERV-K-c11	HLA-DQB1	+	32643464	-	32627244	32636160	7304	RAS / FGFR / AKT / SOS / STAT / ERK / ERK	HLA-DRA
HERV-K-c11	HLA-DQA1	+	32643464	+	32595956	32614839	28625	Wnt	HLA-DQA2
HERV-K-c11	HLA-DRB1	+	32643464	-	32546546	32557625	85839	Translation	PSMB8
HERV-K-c11	HLA-DRB6	+	32643464	-	32520490	32527799	115665	transcription	PSMB8
HERV-K-c11	HLA-DRB5	+	32643464	-	32485120	32498064	145400	Wnt	HLA-DRA
HERV-K-c11	HLA-DRA	+	32643464	+	32407619	32412823	230641	NOTCH1	HLA-B
HERV-K-c12	HLA-DQB1	+	32648040	-	32627244	32636160	11880	transport	BTN3A3
HERV-K-c12	HLA-DQB1	+	32648040	-	32627244	32636160	11880	RAS / FGFR / AKT / SOS / STAT / ERK / RAF;Protein productio	CCR6
HERV-K-c12	HLA-DQB1	+	32648040	-	32627244	32636160	11880	NOTCH1	HLA-DPB1
HERV-K-c12	HLA-DQB1	+	32648040	-	32627244	32636160	11880	RAS / FGFR / AKT / SOS / STAT / ERK / ERK;carbohydrate pro	HLA-DRA
HERV-K-c12	HLA-DQB1	+	32648040	-	32627244	32636160	11880	RAS / FGFR / AKT / SOS / STAT / ERK / ERK;carbohydrate pro	HLA-DRB1
HERV-K-c12	HLA-DQB1	+	32648040	-	32627244	32636160	11880	Protein production	UBE2L6
HERV-K-c12	HLA-DQA1	+	32648040	+	32595956	32614839	33201	NOTCH1	HLA-DPB1
HERV-K-c12	HLA-DQA1	+	32648040	+	32595956	32614839	33201	RAS / FGFR / AKT / SOS / STAT / ERK / ERK;carbohydrate pro	HLA-DQA2
HERV-K-c12	HLA-DQA1	+	32648040	+	32595956	32614839	33201	RAS / FGFR / AKT / SOS / STAT / ERK / ERK;carbohydrate pro	HLA-DRA
HERV-K-c12	HLA-DQA1	+	32648040	+	32595956	32614839	33201	RAS / FGFR / AKT / SOS / STAT / ERK / ERK;carbohydrate pro	HLA-DRB1
HERV-K-c12	HLA-DRB1	+	32648040	-	32546546	32557625	90415	transport	BTN3A3
HERV-K-c12	HLA-DRB1	+	32648040	-	32546546	32557625	90415	RAS / FGFR / AKT / SOS / STAT / ERK / RAF;Protein productio	CCR6
HERV-K-c12	HLA-DRB1	+	32648040	-	32546546	32557625	90415	Immune response;NOTCH1	HLA-B
HERV-K-c12	HLA-DRB1	+	32648040	-	32546546	32557625	90415	NOTCH1	HLA-DPB1
HERV-K-c12	HLA-DRB1	+	32648040	-	32546546	32557625	90415	RAS / FGFR / AKT / SOS / STAT / ERK / ERK;carbohydrate pro	HLA-DRA
HERV-K-c12	HLA-DRB1	+	32648040	-	32546546	32557625	90415	RAS / FGFR / AKT / SOS / STAT / ERK / ERK;carbohydrate pro	HLA-DRB1
HERV-K-c12	HLA-DRB1	+	32648040	-	32546546	32557625	90415	NOTCH1	IRF1
HERV-K-c12	HLA-DRB1	+	32648040	-	32546546	32557625	90415	RAS / FGFR / AKT / SOS / STAT / ERK;IgE / FCERI;apoptosis;I	PSMB8
HERV-K-c12	HLA-DRB6	+	32648040	-	32520490	32527799	120241	mitosis	C1QA
HERV-K-c12	HLA-DRB6	+	32648040	-	32520490	32527799	120241	Immune response;NOTCH1	HLA-B
HERV-K-c12	HLA-DRB6	+	32648040	-	32520490	32527799	120241	NOTCH1	HLA-DPB1
HERV-K-c12	HLA-DRB6	+	32648040	-	32520490	32527799	120241	RAS / FGFR / AKT / SOS / STAT / ERK / ERK;beta catenin;cart	HLA-DQA2
HERV-K-c12	HLA-DRB6	+	32648040	-	32520490	32527799	120241	RAS / FGFR / AKT / SOS / STAT / ERK / ERK;carbohydrate pro	HLA-DRA
HERV-K-c12	HLA-DRB6	+	32648040	-	32520490	32527799	120241	RAS / FGFR / AKT / SOS / STAT / ERK / ERK;carbohydrate pro	HLA-DRB1
HERV-K-c12	HLA-DRB6	+	32648040	-	32520490	32527799	120241	RAS / FGFR / AKT / SOS / STAT / ERK;IgE / FCERI;apoptosis;I	PSMB8
HERV-K-c12	HLA-DRB5	+	32648040	-	32485120	32498064	149976	RAS / FGFR / AKT / SOS / STAT / ERK / RAF;Protein productio	CCR6
HERV-K-c12	HLA-DRB5	+	32648040	-	32485120	32498064	149976	Immune response;NOTCH1	HLA-B
HERV-K-c12	HLA-DRB5	+	32648040	-	32485120	32498064	149976	NOTCH1	HLA-DPB1
HERV-K-c12	HLA-DRB5	+	32648040	-	32485120	32498064	149976	RAS / FGFR / AKT / SOS / STAT / ERK / ERK;carbohydrate pro	HLA-DRA
HERV-K-c12	HLA-DRB5	+	32648040	-	32485120	32498064	149976	RAS / FGFR / AKT / SOS / STAT / ERK / ERK;beta catenin;cart	HLA-DRB1
HERV-K-c12	HLA-DRA	+	32648040	+	32407619	32412823	235217	transport	BTN3A3
HERV-K-c12	HLA-DRA	+	32648040	+	32407619	32412823	235217	Immune response	HLA-B
HERV-K-c12	HLA-DRA	+	32648040	+	32407619	32412823	235217	NOTCH1	HLA-DPB1
HERV-K-c12	HLA-DRA	+	32648040	+	32407619	32412823	235217	RAS / FGFR / AKT / SOS / STAT / ERK / ERK;carbohydrate pro	HLA-DQA2
HERV-K-c12	HLA-DRA	+	32648040	+	32407619	32412823	235217	RAS / FGFR / AKT / SOS / STAT / ERK / ERK;carbohydrate pro	HLA-DRA
HERV-K-c12	HLA-DRA	+	32648040	+	32407619	32412823	235217	RAS / FGFR / AKT / SOS / STAT / ERK / ERK;carbohydrate pro	HLA-DRB1
HERV-K-c12	HLA-DRA	+	32648040	+	32407619	32412823	235217	RAS / FGFR / AKT / SOS / STAT / ERK;IgE / FCERI;apoptosis;I	PSMB8
HERV-K-c13	PLG	+	161190894	+	161123270	161174347	16547	RAS / FGFR / AKT / SOS / STAT / ERK / ERK;Hedgehog;RAS /	PLG
HERV-K-c13	LPA	+	161190894	-	160952515	161087407	103487	Hedgehog;RAS / FGFR / AKT / SOS / STAT / ERK / RAF;Immur	PLG
HERV-K-c14	C9orf106	+	132205207	+	132083295	132087184	118023	DNA processing	ASB6
HERV-K-c17	TWF1	+	44313661	-	44187526	44200178	113483	transport	CHPT1
HERV-K-c17	TWF1	+	44313661	-	44187526	44200178	113483	RAS / FGFR / AKT / SOS / STAT / ERK;Hedgehog;Kinase / ATK	TMED2
HERV-K-c18	RILPL1	-	124066476	-	123955925	124018265	48211	RAS / FGFR / AKT / SOS / STAT / ERK;Hedgehog;Kinase / ATK	TMED2
HERV-K-c18	KMT5A	-	124066476	+	123868320	123893905	172571	Nerotransmitters;RAS / FGFR / AKT / SOS / STAT / ERK;RAS /	KMT5A
HERV-K-c18	SBNO1	-	124066476	-	123773656	123849390	217086	beta catenin	ATP6V0A2
HERV-K-c2	CHI3L2	-	111802591	+	111743393	111786062	16529	Immune response;EFGR	EPHX2
HERV-K-c2	CHI3L2	-	111802591	+	111743393	111786062	16529	histone methylation;RAS / FGFR / AKT / SOS / STAT / ERK;Fibr	TUBA3C
HERV-K-c2	DENND2D	-	111802591	-	111729796	111747157	55434	transport	CEPT1
HERV-K-c2	CEPT1	-	111802591	+	111682249	111727724	74867	transport	CEPT1
HERV-K-c2	DRAM2	-	111802591	-	111659955	111682838	119753	transport	CEPT1
HERV-K-c2	DRAM2	-	111802591	-	111659955	111682838	119753	Ethanol processing;transcription	OVGP1
HERV-K-c20	TPM1	-	63374593	+	63334831	63364114	10479	Cell cycle	LMOD1
HERV-K-c20	TPM1	-	63374593	+	63334831	63364114	10479	Translation;Lipid processing;DNA repair;Transport / blood ;DNA	SDC2
HERV-K-c20	TPM1	-	63374593	+	63334831	63364114	10479	Cell cycle	TPM1
HERV-K-c20	TLN2	-	63374593	+	62682725	63136830	237763	Chaperones	RSU1
HERV-K-c20	TLN2	-	63374593	+	62682725	63136830	237763	caspase /diablo;RAS / FGFR / AKT / SOS / STAT / ERK;Amino	TLN1
HERV-K-c21	GOSR1	-	28891180	+	28804380	28854610	36570	Lipid processing	CLTCL1
HERV-K-c22	ZNF429	-	21841535	+	21679484	21739072	102463	PIP	ZNF43
HERV-K-c22	ZNF429	-	21841535	+	21679484	21739072	102463	PIP	ZNF626
HERV-K-c22	ZNF493	-	21841535	+	21579921	21610375	231160	PIP	ZNF43
HERV-K-c22	ZNF493	-	21841535	+	21579921	21610375	231160	PIP	ZNF486
HERV-K-c23	ZNF676	+	22466363	-	22361893	22379753	86610	PIP	ZNF43
HERV-K-c23	ZNF257	+	22466363	+	22235254	22274282	192081	Nerotransmitters;Neurotransmission;Hedgehog	ZDHHC2
HERV-K-c23	ZNF257	+	22466363	+	22235254	22274282	192081	PIP	ZNF43
HERV-K-c23	ZNF257	+	22466363	+	22235254	22274282	192081	PIP	ZNF486
HERV-K-c23	ZNF257	+	22466363	+	22235254	22274282	192081	PIP	ZNF92
HERV-K-c3	SUSD4	+	223578309	-	223394161	223537544	40765	ubiquitination	CAPN8
HERV-K-c5	LRCH3	+	197845799	+	197518097	197615307	230492	SUMOylation	RIT1
HERV-K-c8	RASGRF2	-	80442199	+	80256491	80525975	83776	RAS / FGFR / AKT / SOS / STAT / ERK;apoptosis;Amino acid p	RASGRF2

4. Chapter 4: Discussion.

4.1. Database of HERV-K(HML-2) elements in the human genome.

Endogenous retroviruses constitute a substantial portion of all mammalian genomes. In the majority of cases, many of these genomic insertions are relics of past retroviral infections that serve little function (with exception to those rare instances of co-option of various genes which benefit the host – see section 1.4). The human genome is one of the best characterized genomes, enabling us to look at endogenous retroviral dynamics in great detail. The most recently active retroviruses in the human genome belong to the HERV-K(HML-2) family of endogenous retroviruses, with a considerable number of human specific insertions (inserted after the divergence of chimpanzees and humans) with some of these loci also being insertionally polymorphic in humans (suggesting an integration time after the emergence of modern humans – section 1.9). This particular family of endogenous retroviruses has also been implicated in many human diseases, with a number of studies reporting increased expression of HERV-K(HML-2) in various transcriptomic datasets.

The purpose of this thesis was to investigate the role of HERV-K(HML-2) endogenous retrovirus family in cancer genomes. As explained in detail in chapter 1, the aims of the project can be summarized as follows:

a) Review the state of HERV-K(HML-2) literature with regard to cancer.
b) Identify known insertion in recent reference genome and characterize potential relationships between them.
c) Identify novel HERV-K(HML-2) elements in cancer samples as well as previously known elements.
d) Investigate the possible roles of HERV-K(HML-2) elements in cancer.

In order to perform an in-depth review of the state of knowledge about HERV-K(HML-2) the latest literature on HERV-K(HML-2) was investigated. Baseline information about reference HERV-K(HML-2) insertions was obtained from a comprehensive reference genome study, performed by Subramanian *et al.* (2011). More recent research, done by other groups, including Wildschutte *et al.* (2016) and Shin *et al.* (2013) has revealed that there is much more data available on HERV-K(HML-2), particularly when considering non-reference or truncated elements. These studies have also underlined the importance of terminal site duplications, discussed in detail in section 1.2, and showed some inconsistencies among the published data. Therefore, after collecting all the available information from the latest literature, summary of which can be found in chapter 1, the state of the HERV-K(HML-2) elements in the reference genome was studied (for details, refer to chapters 2.3 and 3.1). This has organized the baseline data on HERV-K(HML-2), including correct proviral sequences, locations of each insertion among recent human reference genome versions (NCBI36,

GRCh37/Hg19, GRCh38/Hg38) as well as much more metadata, all included in a HERV-K(HML-2) database (refer to section 3.1.i for a link to online version). All of the data was compiled in order to create a background for studying the role of HERV-K(HML-2) in cancer genomes. However, given that the vast majority of information was scattered across a number of different studies, a lot of the data was checked and had to be corrected, notably the locations of all known HERV-K(HML-2) insertions and their sequences. Moreover, the process led to discovery of 12 insertions, which were unknown previously. Additionally, the value of HERV-K(HML-2) insertion rate was calculated, based on HERV-K(HML-2) elements detected in chimpanzee reference genome, PanTro6. The result ($7.4 * 10^{-4}$, see section 3.1.iv) is approximately double the value provided by Belshaw et al. (2005) ($3.8 * 10^{-4}$), which is due to a much more exhaustive source data, including all known HERV-K(HML-2) insertions known to date. Since 2005, the genome sequencing technologies have advanced and numerous new reference genomes have been published (please refer to chapter 1.17 for details). Therefore, the first part of this study can be an excellent base for any research focused on HERV-K(HML-2) family and its role in different processes. Furthermore, the fact that all of the insertions have been extracted from the reference genome, analyzed using a consistent methodology and well-established tools like BLAST (refer to section 2.3 for details) creates a comprehensive and reproducible database of information about the HERV-K(HML-2) family. The methodology presented could be directly reapplied onto many other types of sequencing data. This creates an organized and expandable summary of information about known HERV-K(HML-2) elements, which is otherwise severely inconsistent in the recently published literature.

4.2. Recombinational potential of HERV-K(HML-2).

Recombinational activity of HERV-K(HML-2) (reviewed in 1.13) was explored by analysing TSD sequences for reported insertions (please refer to 2.3.i-2.3.iv for detailed methodology and 3.1.iii for results, especially table 12). It indicated that 411 HERV-K insertions are *potentially involved* in TSD exchange. This is especially prominent in the "non-matching TSDs" group, (table 12, 3B, section 3.1.iii), which immediate flanking sequences are not similar despite retaining correct 5' and 3' LTR sequences (compared to canonical HERV-K113 (accession number: AY037928) reference). The presence of insertions, that display non-matching TSDs but intact 5' and 3' ends suggested, that such TSDs have arisen due to recombination (Figure 8, section 1.17). This has initially indicated a possibility of retroelement-mediated ectopic recombination/gene conversion events, which could further contribute to cancer development. Such idea has been explored using phylogenetic analysis to by Hughes and Coffin (2005). However, their approach was based on a limited knowledge on HERV-K(HML-2) from 2005 and focused only on 15 elements, 6 of which were suggested to undergo recombination/gene conversion. Based on assumption that human genome contained approximately 3900 HERVs they concluded that approximately 1050 elements undergone such

recombinations, which would make 27%. Analysis presented here showed that no pairs of loci exhibiting non-matching TSDs show matches on opposite ends simultaneously (5' TSD on insertion 1 matching 3' TSD on insertion 2 and 3' TSD on insertion 1 matching 5' TSD on insertion 2 – Figure 8, section 1.17). This indicates that actual homologous recombination process between such loci is not occurring, contrary to findings of Hughes and Coffin (2005). For the insertions that do not display recognizable TSDs, there must have been TSD exchange with another locus (as explained in section 3.1.iii), however the exact molecular details of such mechanism remain elusive. All of these elements (excluding elements involved in segmental duplications), have been found to potentially share their immediate flanking sequence with other elements, that had well-defined TSDs. Therefore, it suggests that majority of these HERV-K(HML-2) insertions are indeed very likely to be involved in TSD exchange events. Furthermore, over 28% of elements display recognizable TSDs display similar evidence for sharing such TSD with another insertion(s). More detailed analysis focused only on pairs of insertions showing non-matching TSDs (TSD acceptors) and insertions with well-defined TSDs (TSD donors). 31 such pairs of insertions were recognized as potential TSD donors/acceptors (see section 3.1.iii). However, the observed relation between distributions of donors/acceptors (showed in figure 21 a and b respectively, section 3.1.iii) and general distribution of HERV-K(HML-2) (figure 18 b, section 3.1.iii) shows that there is no recognizable source of these events or Master gene (see 1.14), as suggested by Johnson and Brookfield (2006), regarding the HERV-K(HML-2) family of proviruses. Moreover, the data provided by TSD analysis suggest, that the sequence exchange does not extend beyond the TSD, therefore even if it is generated via gene conversion/ectopic recombination, there's no explicit evidence for such events to actually occur between the selected HERV-K(HML-2) elements.

The findings of TSD comparison analysis reflect with the state of elements bearing non-matching TSD in the reference genomes of other primates. These include the chimpanzee, gorilla, gibbon and bonobo, where vast majority of HERV-K(HML-2) insertions contain identical or nearly identical, non-matching TSD sequences as found in humans (detailed results of this part of research can be found in the online database provided in section 3.1.i). This would suggest, that either the mismatching TSDs are a result of random mutations and the recombination/gene-conversion events driven by HERV-K(HML-2) do not occur at all or occur very rarely. Alternatively, one could hypothesize that observed differences between TSDs are indeed results of ectopic recombination. However, presence of identical sequences in other primates related to humans would place all of these events in common ancestors that predate the human/orangutan split (approximately 15.20 MYa (Kumar *et al.*, 2017)). Nevertheless, all that leads to a conclusion that HERV-K(HML-2) does not contribute significantly to recombination events in the human genome.

4.3. Segmentally duplicated HERV-K(HML-2) loci.

Analysis of nearby flanking sequences performed on reported HERV-K(HML-2) elements revealed that approximately 17% of total elements (199/1161) proliferated through the genome via segmental duplications. The contribution of segmental duplications with regards to HERV-K(HML-2) has never been studied to such detail in the literature. Previous studies report two insertions (K111 and K222) that have been found to be segmentally duplicated in the centromeric region of various chromosomes (Contreras-Galindo *et al.*, 2013). These viruses do not seem to be related despite they're both located in the centromere. Moreover, comparison of TSDs and immediate flanking sequences support the idea that these integrations must have happened separately during the course of evolution, and then undergone segmental duplication in the centromeric regions. Phylogenetic analysis revealed that K222 virus has been truncated on the 5' end very early after the integration and subsequent mutations have rendered the virus inactive due to premature stop codons. The similarity of centromere sequences might have caused duplication of K111 and K222 proviruses due to homologous recombination or gene conversion (Zahn *et al.*, 2015). There are also segmentally duplicated insertions on chromosome X and Y (HML-2.LTR897-HML-2.LTR929, HML-2.LTR898-HML-2.LTR930), that are found in the pseudoautosomal regions, identical on X and Y chromosomes, which allows the genes inside the pseudoautosomal regions to recombine during meiosis and they are not inherited in a sex-linked fashion (Helena Mangs and Morris, 2007).

The general distribution of segmental duplications in the human genome is significantly different to the distribution of segmentally duplicated HERV-K(HML-2). Chromosome 8 shows the largest amount of segmentally duplicated insertions - 39/80 elements (49%, whereas only 2.04% of chromosome 8 consist of segmental duplications (She *et al.*, 2004)) , which contributes to its high proviral content. These observations are important for future studies, especially those that would focus on distribution patterns and interaction between the HERV-K(HML-2) family of transposable elements.

4.4. Perspectives for HERV-K(HML-2) evolution studies.

Future work on the evolutionary dynamics and state of HERV-K(HML-2) family should focus on determining the state of insertions in genomes of other primates, particularly chimpanzees, orangutans, gorillas and bonobos. Testing these great apes in the first instance could provide a better estimation of the evolutionary history of HERV-K(HML-2) elements in the genomes of great apes, since the oldest elements integrated into the human lineage over 40 million years ago (Subramanian *et al.*, 2011). Previous analysis of genomic sequences for a variety of great ape species, such as work done by Prado-Martinez *et al.* (2013) show that nowadays there is a large quantity of chimpanzee sequencing data available. However, it was also observed that HERV-

K(HML-2) has most likely stopped proliferating through the genomes of chimpanzees but it is much more actively circulating the gorilla lineage. Recent study of the gorilla genome shows 150 gorilla-specific HERV-K(HML-2) insertions (31 of which contained two LTRs) in the gorilla genomes, some of them dating as early as <100,000 years (Holloway *et al.*, 2019).

Data about evolutionary history of great apes could be combined with studying the state of HERV-K(HML-2) family of transposable elements in these species to obtain a more accurate evolutionary history and dynamics of this family of retroelements. It would be especially interesting to see the state of insertions that are truncated or exhibit non-matching TSDs in humans. If genomes of other primates showed equivalent insertions in their original state, it could be used to approximate the moment in evolutionary history when changes to the sequences of these altered HERV-K(HML-2) loci happened and ascertain their original state. On the other hand, if the elements found were to exhibit a high degree of similarity to loci found in modern humans, that would suggest such loci are very old. Any activity that leads to mutations resulting in truncations of full-length elements or alterations in their TSDs would have then occurred a long time ago. This would support the idea that although HERV-K(HML-2) is still considered polymorphic today, its activity in modern populations is negligible.

4.5. Presence of HERV-K(HML-2) in cancer genomes.

The database assembled in the initial part of this thesis provided a foundation for studying genomes of cancer patients, coming from The Cancer Genome Atlas project. A bioinformatics pipeline was developed for that purpose and provided a reproducible framework for detection of non-reference HERV-K(HML-2) insertions, as well as other transposable elements (for exact methodology, refer to sections 2.4.ii-2.4.iv). It is applicable to virtually all short-read genomic data and could easily be adjusted to analyse long-read genomic data. Methodology presented here could provide a consistent platform of characterizing HERV-K(HML-2) across a multitude types of data, as well as other types of similar transposable elements.

The analysis of cancer genomes encompassed 14 different types of cancer and a total of 435 genomes (209 cancer genomes, and 226 control genomes) from The Cancer Genome Atlas project (see section 2.4.i for details). The work focused on finding novel HERV-K(HML-2) elements in the tested samples (see section 3.2). Presented approach was based on BLAST analysis and assembly of detected reads into nonreference HERV-K(HML-2) loci with regard to presence of correct TSD and consistent coverage of both ends for each nonreference element (see section 2.4.ii for details). It has proven to be effective, overall detecting 28 HERV-K(HML-2) loci, 7 of which contained novel data, not reported anywhere before. TSD comparison between novel insertions found in cancer and HERV-K(HML-2) loci known to date did not produce any relevant matches. Since it is assumed that novel insertions entered the human populations relatively recently and the TSD sequences are

mostly unmutated, the author did not consider the possibility of point mutations within the TSD comparisons. Analysis of TSDs and extended flanks of detected insertions suggest that HERV-K-c5 and HERV-K-c16 appear have proliferated through the genome via segmental duplication. However, that result could be an artefact of the short-read technology. The presence of both insertions within the same projects (prostate and myeloma cancers, table 14, section 3.2) and separately in individual projects (listed in table 14, section 3.2), suggests that it is possible that these insertions are in fact separate occurrence. It is not likely, that this was typical segmental duplication, due to the fact, that the similar preinsertion sites are already present in the reference genome. Moreover, the molecular mechanisms of such event observed are not clear and cannot be determined by analysing data studied in this project.

4.6. HERV-K(HML-2) activity and potential influence on cancer.

There are two possible scenarios for novel HERV-K(HML-2) insertions in cancer samples:

a) The elements would be found in both cancer and control samples.
b) The elements would appear only in cancer samples.

In all of the analyzed samples, only scenario a) could have been observed. This suggests, that HERV-K(HML-2) do not undergo insertion/recombination due to malignant activity of cancer cells, but rather presence of certain insertions could be associated with development of cancer. In order to investigate this hypothesis further, a statistical analysis was performed (see sections 2.4.iv and 3.2) and indeed, for a number of insertions the observed allele frequencies show significant difference in cancer samples compared to general population. 21 of the 28 insertions detected in cancer genomes had been previously described by Wildschutte *et al.* (2016), with allele frequencies, calculated from the data from the 1000 Genomes Project and the Human Genome Diversity Project. The author compared observed allele frequencies of these particular loci from the cancer datasets, to those calculated by Wildschutte *et al.* (2016) in the general population, using Fishers Exact test (Monte Carlo method, with 10^6 replicates). For further information, please refer to section 3.2. As the complexity of the calculations for the performed Fishers exact test exceeded the capabilities of modern computers, the calculation of p-values was computed by Monte Carlo simulation, with 10^6 of replicates (see *canceranalysis.Rmd* for details, appendix 1.xxv). However, the estimated p values were 4 orders of magnitude lower than widely considered threshold of 5% for every compared insertion except HERV-K-c10, that was closer to the upper limit, but still significant. Therefore, to compare the occurrence of the detected loci in cancer with the general population, the counts for each insertion studied in Wildschutte *et al.* (2016) were summed up and used as a representation of occurrence of studied loci in 1000 Genomes Project and Human Genome Diversity Project. The occurrence of nonreference insertions in cancer genomes has been compared to its occurrence

calculated from cumulative data presented in Wildschutte *et al.* (2016) using the same method of Fishers exact test. However, no simulation needed to be performed since each insertion in each cancer was compared to the cumulative figure representing the 1000 Genomes Project or Human Genome Diversity Project. The results of this analysis are presented in table 16, section 3.2.

This analysis shows, that although a group of unfixed insertions was previously detected in normal human populations, its frequency of occurrence in cancer samples is significantly higher from the frequency in healthy individuals. In fact, 10 of analysed insertions show evidence for occurring significantly different in some cancers when compared to normal population, with 5 out of that being significant against both 1000 Genomes Project and Human Genome Diversity Project normal population sampled (see table 16, section 3.2). This indicates that although these loci were previously detected by other authors, their occurrence in the cancer samples is different to normal populations, which means they may play an additional role in these cancers. Although their exact role is not clear, and processes proposed further are just possible factors that may contribute to cancer development, the statistical analysis shows a correlation between the insertions selected in table 16 (section 3.2) and occurrence of particular cancers. This itself provides a strong evidence toward a relationship between presence of HERV-K(HML-2) polymorphic insertions and carcinogenesis. In order to better understand the relations, it would be beneficial to obtain a sample of normal population and perform the entire analysis according to the methodology presented for cancer samples, rather than relying on occurrence provided by external authors. This would eliminate possible biases between differences on various stages of the provirus detection methodology as well as calculation of the final frequencies. Unfortunately, it was not possible to perform such analysis during the course of this PhD project due to time and funding constraints, so the comparison to previously published research was used as reference to general, cancer-free population. To further examine the problem, it would also be important to confirm the presence of these insertions via PCR and possibly resequence the source genomic samples as a starting point, both of which go beyond this project, due to time and resource limitations.

The observed polymorphic HERV-K(HML-2) element allele frequencies were significantly associated with occurrence of cancers, such as breast, prostate, melanoma, myeloma, cervix and hepatocellular, compared to data from 1000 Genomes Project and Human Genome Diversity Project (Wildschutte *et al.*, 2016). Other studies further suggest direct influence of HERV-K(HML-2) on these types of cancer. Examples include association of increased breast cancer progression rate and decrease of overall survival with increased HERV-K(HML-2) *env* expression (Zhao *et al.*, 2011). HERV-K(HML-2) transcripts were also associated with prostate cancer stage and metastasis, which suggests its significance as a biomarker for the disease (Rhyu *et al.*, 2014). A correlation between presence of HERV-K(HML-2) transcripts and MEK-ERK pathway alteration in melanoma (Li *et al.*,

2010) or correlation between liver cirrhosis, tumor stage and survival rate in hepatocellular carcinoma were also reported previously (Ma *et al.*, 2016). Many more examples of observed associations between HERV-K(HML-2) and different types of cancer were reviewed in chapter 1.16.

4.7. Influence of polymorphic HERV-K(HML-2) loci on oncogenesis.

To investigate these findings further, regions flanking locations of polymorphic insertions observed to be enriched in cancer samples were studied for genes associated with tumors, which produced a large database of factors potentially influenced by HERV-K(HML-2). These were found to be involved in crucial molecular processes, which are often observed to be altered in cancer, providing some more direct evidence for the possible role of HERV-K(HML-2) elements found in the studied data. It ties into the findings in literature mentioned above, as well as in the chapter 1.16, where different cellular processes are discussed with regards to HERV-K(HML-2) and cancer.

Many of the molecular pathways detected in the analysis and widely implicated in cancer are directly involved in mechanisms like the control of the cell cycle via cyclins, DNA synthesis or cellular signalling (see table 17, section 3.2). Particularly, MAPK signalling pathway is of interest, because it governs many different processes inside the cell in response to stress, growth signalling etc. (Pierce, Luttrell and Lefkowitz, 2001). Due to extracellular signals it activates cascades of kinases, which can influence cell metabolism, control transcription (via transcription factor effectors), cell growth, differentiation and development, as well as trigger apoptosis (Guo *et al.*, 2020). MAPK signalling can also transactivate a variety of different signalling pathways observed in this study, such as Notch1, which is an evolutionarily conserved signalling pathway controlling interactions between adjacent cells (Venkatesh *et al.*, 2018). Similarly, Wnt signalling pathway was observed to be possibly influenced by HERV-K insertions in this study, which also controls gene transcription and expression, particularly during embryonic development (Zhan, Rindtorff and Boutros, 2017). The Wnt pathway controls body axis pattern, migration and proliferation of the cells and cell specification, all of which can be altered during carcinogenesis via disrupting the Wnt pathway (Tai *et al.*, 2015). On the other hand, it was found that the studied transposable elements can influence genes involved in apoptosis, SUMOylation and ubiquitination. All of these processes are involved in programmed cell death, apoptosis being the actual process (Han *et al.*, 2018). SUMOylation can often alter proprieties of target proteins by conjugating them with Small Ubiquitin-like Modifiers (SUMO), which can affect further protein complex formation and signalling target proteins, important in regulation of DNA damage repair, immune responses, carcinogenesis or cell cycle progression (Zhao *et al.*, 2020). Ubiquitination is a similar process, but it mainly marks its targets for degradation in the proteasome. Furthermore, the attachment of ubiquitin molecules can affect activity of the target protein, alter its location within the cell or facilitate interactions between affected proteins (Sun, Liu and Yang, 2020). Unfortunately, it is beyond the capacity of this project and the provided data to further

observe the role of HERV-K(HML-2) in any of these molecular processes listed. The literature suggests (Fuentes, Swigut and Wysocka, 2018), that HERV-K(HML-2) LTRs found in those particular locations influence the expression of listed genes and therefore their presence can alter signalling within those pathways. It was recently demonstrated by Karamitros *et al.* (2018) that a particular insertion of HERV-K(HML-2) can directly influence a known gene related to addictive behaviour, which had phenotypic effects on a wider population of people abusing drugs (reviewed in section 1.15.v). The authors were able to determine the gene influenced, and propose the specific mechanism that is affected within the cell and manifests itself in higher tendency to drug abuse with people with a certain insertion. This is a strong evidence, that HERV-K(HML-2) can be directly involved in different molecular processes, including the ones presented here. In order to fully explore this, wider research should be conducted, involving testing for the presence of particular insertions in cell lines and studies of signalling changes within proposed pathways. Because of the complexity of cellular signalling mechanisms, such experiments would have to be carefully constructed and monitor a wide variety of factors, including the levels of expression of particular insertions, the levels of possibly affected genes or metabolites involved in the processes. It would involve a group of interdisciplinary researchers just to fully study one of the pathways presented here in context of the influence of particular HERV-K insertion on it. Nevertheless, such research requires much more work and to do similar experiments for a sample as large as one studied in this thesis would greatly surpass the capacity of a PhD project. Therefore, the literature analysis of found insertions and genes they may influence found here could be a good basis for future research of HERV-K(HML-2) role in cancer.

4.8. Summary and future prospects.

The findings presented here strongly suggest a role of non-reference HERV-K(HML-2) elements in particular cancers studies, especially that these were found to possibly influence a multitude cellular processes involved in cancer development and progression. For future research, it would be quite important to confirm the *in silico* findings. It would be important to observe the detected novel and non-reference insertions in an *in vitro* cell study and then look into the proposed cellular pathways that are important in cancer. By measuring levels of expression of genes found to be possibly influenced by HERV-K(HML-2) and, ideally, comparing them to a control group, some more direct proof of HERV-K(HML-2) role could be obtained. Furthermore, levels of protein products interacting with each other in the described pathways could be studied and the range of cellular signalling within these could be assessed. That would provide even more evidence for the specific influence of particular HERV-K(HML-2) elements to processes directly affecting cancer development and progression.

In the future, such analysis should be also applied to, ideally, long-read sequencing data coming from modern human populations, like the 1000 Genomes Project or Human Genome Diversity Project, similarly to what was done by Wildschutte *et al.* (2016). However, the analysis should focus on long-read sequencing. This could enable to encompass novel polymorphic HERV-K(HML-2) insertions found in low frequencies in cancer samples in this study. Additionally, it would be possible to detect and accurately map other HERV-K(HML-2) insertions that occur in regions of the genome that are difficult to assemble, such as parts of segmental duplications or centromeres. Incorporating long-read sequencing techniques and using HERV-K(HML-2) detection methodology presented in this study could produce data on the most interesting, polymorphic HERV-K(HML-2) elements, suspected of being still active nowadays. Many of the studies, including the data coming from the Cancer Genome Atlas project analysed in this study is based on short-read Illumina sequencing, which exhibits low error rate but covers only 50-300bp per read. Such data is insufficient to accurately map regions of the genome displaying high degrees of similarity, like segmental duplications and therefore it would be crucial to apply accurate long-read sequencing data in future HERV-K(HML-2) studies. Even though the limitations described above, the methodology applied in this study was sufficient to detect presence of novel HERV-K(HML-2) insertions in cancer samples. However, the findings could be greatly expanded if the analysed data was based on accurate long-read assemblies. Additionally, long-reads would contain internal regions of the detected HERV-K(HML-2) elements. That information would enable the author to correctly categorize particular non-reference loci with regard to its retroviral family or perform a robust phylogenetic analysis of these sequences.

Additionally, there have been recent reports, such as work done by Thomas, Perron and Feschotte (2018), that found other HERV families – HERV-W with insertionally polymorphic loci. Moreover, they found approximately 69 HERV-H dimorphisms, one of which was found to contribute to expression of *ESRG,* an early marker of human cell reprogramming into induced pluripotent stem cells. The authors used a similar pipeline based on BLAST search to analyse their data. The pipeline assembled in this study could effectively be applied to other types of retroelements with little adjustments of parameters, depending on source data used. It would be interesting to test the presented approach on data encompassing other HERV proviral families, such as HERV-W and HERV-H confirmed to show polymorphic alleles in modern populations. The results could then be compared to findings about the HERV-K(HML-2) family and a more complete picture of evolutionary dynamics and genetic interactions, including different families of endogenous retroviruses could be investigated.

The unfixed HERV-K(HML-2) insertions detected in this study were found to be present both in cancer and control cases, which shows that individuals were carriers of the particular insertion in their somatic cells and the presence of these insertions were not attributed only to cancer cells. Although expression control mechanisms such as CpG methylation and histone acetylation are significantly altered in cancer, no novel insertions of HERV-K(HML-2) were detected in the studied cancer samples only. Additionally, a few novel insertions were observed in very low frequencies, which all suggests that the hypothesis of HERV-K(HML-2) being still active in modern populations recently is quite unlikely. Therefore, in order to study the dynamics of integrations of endogenous retroviruses, it would be beneficial to focus on families proven to be active today, such as KoRV, observed both in endogenous and exogenous forms in recent samples (Tarlinton, Meers and Young, 2006; Hayward *et al.*, 2020). As mentioned above, the author believes that methods presented here could be reapplied with little modifications to other endogenous retroviruses. Researching an active ERV family, such as KoRV, could provide invaluable information about the dynamics of active insertions within populations, as well as the state of young, unmutated insertions in the context of their possible contribution to genomic recombination and insertional polymorphism. Collected data could be used to describe the details of insertion process, as well as interaction of newly inserted retroelements with other loci present in the genome. Research like that could perhaps be used to better explain the effects observed for different families of endogenous retroviruses, such as the TSD exchange quantified in this study. It could also help to describe the mechanisms of truncation for various HERV-K(HML-2) loci present in the human reference genome, as well as genomes of other organisms.

5. References.

Aapola, U. *et al.* (2000) 'Isolation and initial characterization of a novel zinc finger gene, DNMT3L, on 21q22.3, related to the cytosine-5-methyltransferase 3 gene family', *Genomics,* 65(3), pp. 293-8. Available at: https://doi.org/10.1006/geno.2000.6168.

Agoni, L. *et al.* (2012) 'Neandertal and Denisovan retroviruses', *Curr Biol,* 22(11), pp. R437-8. Available at: https://doi.org/10.1016/j.cub.2012.04.049.

Agoni, L., Guha, C. and Lenz, J. (2013) 'Detection of Human Endogenous Retrovirus K (HERV-K) Transcripts in Human Prostate Cancer Cell Lines', *Front Oncol,* 3, pp. 180. Available at: https://doi.org/10.3389/fonc.2013.00180.

Alfano, N. *et al.* (2016) 'Endogenous Gibbon Ape Leukemia Virus Identified in a Rodent (Melomys burtoni subsp.) from Wallacea (Indonesia)', *J Virol,* 90(18), pp. 8169-80. Available at: https://doi.org/10.1128/JVI.00723-16.

Anai, Y. *et al.* (2012) 'Infectious endogenous retroviruses in cats and emergence of recombinant viruses', *J Virol,* 86(16), pp. 8634-44. Available at: https://doi.org/10.1128/JVI.00280-12.

Antony, J. M. *et al.* (2004) 'Human endogenous retrovirus glycoprotein-mediated induction of redox reactants causes oligodendrocyte death and demyelination', *Nat Neurosci,* 7(10), pp. 1088-95. Available at: https://doi.org/10.1038/nn1319.

Argaw-Denboba, A. *et al.* (2017) 'HERV-K activation is strictly required to sustain CD133+ melanoma cells with stemness features', *J Exp Clin Cancer Res,* 36(1), pp. 20. Available at: https://doi.org/10.1186/s13046-016-0485-x.

Armbruester, V. *et al.* (2004) 'Np9 protein of human endogenous retrovirus K interacts with ligand of numb protein X', *J Virol,* 78(19), pp. 10310-9. Available at: https://doi.org/10.1128/JVI.78.19.10310-10319.2004.

Armenia, J. *et al.* (2018) 'The long tail of oncogenic drivers in prostate cancer', *Nat Genet,* 50(5), pp. 645-651. Available at: https://doi.org/10.1038/s41588-018-0078-z.

Arnaud, F. *et al.* (2007) 'A paradigm for virus-host coevolution: sequential counter-adaptations between endogenous and exogenous retroviruses', *PLoS Pathog,* 3(11), pp. e170. Available at: https://doi.org/10.1371/journal.ppat.0030170.

Arnaud, F., Murcia, P. R. and Palmarini, M. (2007) 'Mechanisms of late restriction induced by an endogenous retrovirus', *J Virol,* 81(20), pp. 11441-51. Available at: https://doi.org/10.1128/JVI.01214-07.

Ashley, J. *et al.* (2018) 'Retrovirus-like Gag Protein Arc1 Binds RNA and Traffics across Synaptic Boutons', *Cell,* 172(1-2), pp. 262-274.e11. Available at: https://doi.org/10.1016/j.cell.2017.12.022.

Aswad, A. and Katzourakis, A. (2012) 'Paleovirology and virally derived immunity', *Trends Ecol Evol,* 27(11), pp. 627-36. Available at: https://doi.org/10.1016/j.tree.2012.07.007.

Bain, S. C., Todd, J. A. and Barnett, A. H. (1990) 'The British Diabetic Association--Warren repository', *Autoimmunity,* 7(2-3), pp. 83-5. Available at: https://doi.org/10.3109/08916939008993380.

Ballas, N. *et al.* (2005) 'REST and its corepressors mediate plasticity of neuronal gene chromatin throughout neurogenesis', *Cell,* 121(4), pp. 645-657. Available at: https://doi.org/10.1016/j.cell.2005.03.013.

Banerji, S. *et al.* (2012) 'Sequence analysis of mutations and translocations across breast cancer subtypes', *Nature,* 486(7403), pp. 405-9. Available at: https://doi.org/10.1038/nature11154.

Baron, V. *et al.* (2006) 'The transcription factor Egr1 is a direct regulator of multiple tumor suppressors including TGFbeta1, PTEN, p53, and fibronectin', *Cancer Gene Ther,* 13(2), pp. 115-24. Available at: https://doi.org/10.1038/sj.cgt.7700896.

Bellefroid, E. J. *et al.* (1991) 'The evolutionarily conserved Krüppel-associated box domain defines a subfamily of eukaryotic multifingered proteins', *Proc Natl Acad Sci U S A,* 88(9), pp. 3608-12. Available at: https://doi.org/10.1073/pnas.88.9.3608.

Belshaw, R. *et al.* (2005) 'Genomewide screening reveals high levels of insertional polymorphism in the human endogenous retrovirus family HERV-K(HML2): implications for present-day

activity', *J Virol,* 79(19), pp. 12507-14. Available at: https://doi.org/10.1128/JVI.79.19.12507-12514.2005.
Belshaw, R. *et al.* (2004) 'Long-term reinfection of the human genome by endogenous retroviruses', *Proc Natl Acad Sci U S A,* 101(14), pp. 4894-9. Available at: https://doi.org/10.1073/pnas.0307800101.
Bendiksen, S. *et al.* (2014) 'Human endogenous retrovirus W activity in cartilage of osteoarthritis patients', *Biomed Res Int,* 2014, pp. 698609. Available at: https://doi.org/10.1155/2014/698609.
Berger, M. F. *et al.* (2012) 'Melanoma genome sequencing reveals frequent PREX2 mutations', *Nature,* 485(7399), pp. 502-6. Available at: https://doi.org/10.1038/nature11071.
Berger, M. F. *et al.* (2011) 'The genomic complexity of primary human prostate cancer', *Nature,* 470(7333), pp. 214-20. Available at: https://doi.org/10.1038/nature09744.
Bestor, T. *et al.* (1988) 'Cloning and sequencing of a cDNA encoding DNA methyltransferase of mouse cells. The carboxyl-terminal domain of the mammalian enzymes is related to bacterial restriction methyltransferases', *J Mol Biol,* 203(4), pp. 971-83. Available at: https://doi.org/10.1016/0022-2836(88)90122-2.
Bethesda(MD) (1988) *National Center for Biotechnology Information (NCBI).* Bethesda (MD): National Library of Medicine (US), National Center for Biotechnology Information Available at: https://www.ncbi.nlm.nih.gov/.
Beyer, U. *et al.* (2011) 'Endogenous retrovirus drives hitherto unknown proapoptotic p63 isoforms in the male germ line of humans and great apes', *Proc Natl Acad Sci U S A,* 108(9), pp. 3624-9. Available at: https://doi.org/10.1073/pnas.1016201108.
Bieda, K., Hoffmann, A. and Boller, K. (2001) 'Phenotypic heterogeneity of human endogenous retrovirus particles produced by teratocarcinoma cell lines', *J Gen Virol,* 82(Pt 3), pp. 591-596. Available at: https://doi.org/10.1099/0022-1317-82-3-591.
Bittner, J. J. (1936) 'SOME POSSIBLE EFFECTS OF NURSING ON THE MAMMARY GLAND TUMOR INCIDENCE IN MICE', *Science,* 84(2172), pp. 162. Available at: https://doi.org/10.1126/science.84.2172.162.
Bjerregaard, B. *et al.* (2006) 'Syncytin is involved in breast cancer-endothelial cell fusions', *Cell Mol Life Sci,* 63(16), pp. 1906-11. Available at: https://doi.org/10.1007/s00018-006-6201-9.
Blaise, S. *et al.* (2003) 'Genomewide screening for fusogenic human endogenous retrovirus envelopes identifies syncytin 2, a gene conserved on primate evolution', *Proc Natl Acad Sci U S A,* 100(22), pp. 13013-8. Available at: https://doi.org/10.1073/pnas.2132646100.
Blekhman, R., Oshlack, A. and Gilad, Y. (2009) 'Segmental duplications contribute to gene expression differences between humans and chimpanzees', *Genetics,* 182(2), pp. 627-30. Available at: https://doi.org/10.1534/genetics.108.099960.
Blomberg, J. *et al.* (2009) 'Classification and nomenclature of endogenous retroviral sequences (ERVs): problems and recommendations', *Gene,* 448(2), pp. 115-23. Available at: https://doi.org/10.1016/j.gene.2009.06.007.
Boeke, J. D. and Stoye, J. P. 1997. Retrotransposons, Endogenous Retroviruses, and the Evolution of Retroelements. Cold Spring Harbor Laboratory Press, Cold Spring Harbor (NY).
Boller, K. *et al.* (2008) 'Human endogenous retrovirus HERV-K113 is capable of producing intact viral particles', *J Gen Virol,* 89(Pt 2), pp. 567-572. Available at: https://doi.org/10.1099/vir.0.83534-0.
Bonferroni, C. E. (1936) *Teoria statistica delle classi e calcolo delle probabilità.* Seeber. Pubblicazioni del R. Istituto superiore di scienze economiche e commerciali di Firenze.
Boucaut, J. C. *et al.* (1984) 'Prevention of gastrulation but not neurulation by antibodies to fibronectin in amphibian embryos', *Nature,* 307(5949), pp. 364-367. Available at: https://doi.org/10.1038/307364a0.
Burmeister, T. *et al.* (2004) 'Insertional polymorphisms of endogenous HERV-K113 and HERV-K115 retroviruses in breast cancer patients and age-matched controls', *AIDS Res Hum Retroviruses,* 20(11), pp. 1223-9. Available at: https://doi.org/10.1089/aid.2004.20.1223.

Burtonboy, G. *et al.* (1993) 'Isolation of a C-type retrovirus from an HIV infected cell line', *Arch Virol,* 130(3-4), pp. 289-300. Available at: https://doi.org/10.1007/BF01309661.

Bénit, L. *et al.* (1999) 'ERV-L elements: a family of endogenous retrovirus-like elements active throughout the evolution of mammals', *J Virol,* 73(4), pp. 3301-8. Available at: https://doi.org/10.1128/JVI.73.4.3301-3308.1999.

Büscher, K. *et al.* (2005) 'Expression of human endogenous retrovirus K in melanomas and melanoma cell lines', *Cancer Res,* 65(10), pp. 4172-80. Available at: https://doi.org/10.1158/0008-5472.CAN-04-2983.

Callahan, R. *et al.* (1982) 'Detection and cloning of human DNA sequences related to the mouse mammary tumor virus genome', *Proc Natl Acad Sci U S A,* 79(18), pp. 5503-7. Available at: https://doi.org/10.1073/pnas.79.18.5503.

Callahan, R. and Smith, G. H. (2008) 'Common integration sites for MMTV in viral induced mouse mammary tumors', *J Mammary Gland Biol Neoplasia,* 13(3), pp. 309-21. Available at: https://doi.org/10.1007/s10911-008-9092-6.

Campbell, I. M. *et al.* (2014) 'Human endogenous retroviral elements promote genome instability via non-allelic homologous recombination', *BMC Biol,* 12, pp. 74. Available at: https://doi.org/10.1186/s12915-014-0074-4.

Cano, A. *et al.* (2000) 'The transcription factor snail controls epithelial-mesenchymal transitions by repressing E-cadherin expression', *Nat Cell Biol,* 2(2), pp. 76-83. Available at: https://doi.org/10.1038/35000025.

Chattopadhyay, S. K. *et al.* (1980) 'Structure of endogenous murine leukemia virus DNA in mouse genomes', *Proc Natl Acad Sci U S A,* 77(10), pp. 5774-8. Available at: https://doi.org/10.1073/pnas.77.10.5774.

Chen, F. C. and Li, W. H. (2001) 'Genomic divergences between humans and other hominoids and the effective population size of the common ancestor of humans and chimpanzees', *Am J Hum Genet,* 68(2), pp. 444-56. Available at: https://doi.org/10.1086/318206.

Chen, T. *et al.* (2013) 'The viral oncogene Np9 acts as a critical molecular switch for co-activating β-catenin, ERK, Akt and Notch1 and promoting the growth of human leukemia stem/progenitor cells', *Leukemia,* 27(7), pp. 1469-78. Available at: https://doi.org/10.1038/leu.2013.8.

Chen, Y. *et al.* (2018) 'Ancient origin and complex evolution of porcine endogenous retroviruses', *bioRxiv*, pp. 431858. Available at: https://doi.org/10.1101/431858.

Christensen, T. (2005) 'Association of human endogenous retroviruses with multiple sclerosis and possible interactions with herpes viruses', *Rev Med Virol,* 15(3), pp. 179-211. Available at: https://doi.org/10.1002/rmv.465.

Church, D. M. *et al.* (2011) 'Modernizing reference genome assemblies', *PLoS Biol,* 9(7), pp. e1001091. Available at: https://doi.org/10.1371/journal.pbio.1001091.

Coffin, J. *et al.* (2021) 'ICTV Virus Taxonomy Profile:' *J Gen Virol,* 102(12). Available at: https://doi.org/10.1099/jgv.0.001712.

Coffin, J. M., Hughes, S. H. and Varmus, H. E. (1997) 'Retroviruses'.

Cohen, J. C. and Varmus, H. E. (1979) 'Endogenous mammary tumour virus DNA varies among wild mice and segregates during inbreeding', *Nature,* 278(5703), pp. 418-23. Available at: https://doi.org/10.1038/278418a0.

Conrad, B. *et al.* (1994) 'Evidence for superantigen involvement in insulin-dependent diabetes mellitus aetiology', *Nature,* 371(6495), pp. 351-5. Available at: https://doi.org/10.1038/371351a0.

Contreras-Galindo, R. *et al.* (2007) 'Comparative longitudinal studies of HERV-K and HIV-1 RNA titers in HIV-1-infected patients receiving successful versus unsuccessful highly active antiretroviral therapy', *AIDS Res Hum Retroviruses,* 23(9), pp. 1083-6. Available at: https://doi.org/10.1089/aid.2007.0054.

Contreras-Galindo, R. *et al.* (2013) 'HIV infection reveals widespread expansion of novel centromeric human endogenous retroviruses', *Genome Res,* 23(9), pp. 1505-13. Available at: https://doi.org/10.1101/gr.144303.112.

Contreras-Galindo, R. *et al.* (2008) 'Human endogenous retrovirus K (HML-2) elements in the plasma of people with lymphoma and breast cancer', *J Virol,* 82(19), pp. 9329-36. Available at: https://doi.org/10.1128/JVI.00646-08.

Cui, J., Tachedjian, G. and Wang, L. F. (2015) 'Bats and Rodents Shape Mammalian Retroviral Phylogeny', *Sci Rep,* 5, pp. 16561. Available at: https://doi.org/10.1038/srep16561.

Davis, B. R. *et al.* (1987) 'Characterization of a preleukemic state induced by Moloney murine leukemia virus: evidence for two infection events during leukemogenesis', *Proc Natl Acad Sci U S A,* 84(14), pp. 4875-9. Available at: https://doi.org/10.1073/pnas.84.14.4875.

Denne, M. *et al.* (2007) 'Physical and functional interactions of human endogenous retrovirus proteins Np9 and rec with the promyelocytic leukemia zinc finger protein', *J Virol,* 81(11), pp. 5607-16. Available at: https://doi.org/10.1128/JVI.02771-06.

Depil, S. *et al.* (2002) 'Expression of a human endogenous retrovirus, HERV-K, in the blood cells of leukemia patients', *Leukemia,* 16(2), pp. 254-9. Available at: https://doi.org/10.1038/sj.leu.2402355.

Dewannieux, M., Blaise, S. and Heidmann, T. (2005) 'Identification of a functional envelope protein from the HERV-K family of human endogenous retroviruses', *J Virol,* 79(24), pp. 15573-7. Available at: https://doi.org/10.1128/JVI.79.24.15573-15577.2005.

Dickerson, F. *et al.* (2008) 'Polymorphisms in human endogenous retrovirus K-18 and risk of type 2 diabetes in individuals with schizophrenia', *Schizophr Res,* 104(1-3), pp. 121-6. Available at: https://doi.org/10.1016/j.schres.2008.05.005.

Dickson, C. *et al.* (1984) 'Tumorigenesis by mouse mammary tumor virus: proviral activation of a cellular gene in the common integration region int-2', *Cell,* 37(2), pp. 529-36. Available at: https://doi.org/10.1016/0092-8674(84)90383-0.

Diem, O. *et al.* (2012) 'Influence of antipsychotic drugs on human endogenous retrovirus (HERV) transcription in brain cells', *PLoS One,* 7(1), pp. e30054. Available at: https://doi.org/10.1371/journal.pone.0030054.

Diévart, A., Beaulieu, N. and Jolicoeur, P. (1999) 'Involvement of Notch1 in the development of mouse mammary tumors', *Oncogene,* 18(44), pp. 5973-81. Available at: https://doi.org/10.1038/sj.onc.1202991.

Dolei, A. *et al.* (2002) 'Multiple sclerosis-associated retrovirus (MSRV) in Sardinian MS patients', *Neurology,* 58(3), pp. 471-3. Available at: https://doi.org/10.1212/wnl.58.3.471.

Douville, R. *et al.* (2011) 'Identification of active loci of a human endogenous retrovirus in neurons of patients with amyotrophic lateral sclerosis', *Ann Neurol,* 69(1), pp. 141-51. Available at: https://doi.org/10.1002/ana.22149.

Dudley, J. and Risser, R. (1984) 'Amplification and novel locations of endogenous mouse mammary tumor virus genomes in mouse T-cell lymphomas', *J Virol,* 49(1), pp. 92-101. Available at: https://doi.org/10.1128/JVI.49.1.92-101.1984.

Durgam, V. R. and Tekmal, R. R. (1994) 'The nature and expression of int-5, a novel MMTV integration locus gene in carcinogen-induced mammary tumors', *Cancer Lett,* 87(2), pp. 179-86. Available at: https://doi.org/10.1016/0304-3835(94)90220-8.

Easton, A. C. *et al.* (2014) 'Rasgrf2 controls noradrenergic involvement in the acute and subchronic effects of alcohol in the brain', *Psychopharmacology (Berl),* 231(21), pp. 4199-209. Available at: https://doi.org/10.1007/s00213-014-3562-x.

Ehlhardt, S. *et al.* (2006) 'Human endogenous retrovirus HERV-K(HML-2) Rec expression and transcriptional activities in normal and rheumatoid arthritis synovia', *J Rheumatol,* 33(1), pp. 16-23.

Eichler, E. E. (2001) 'Segmental duplications: what's missing, misassigned, and misassembled--and should we care?', *Genome Res,* 11(5), pp. 653-6. Available at: https://doi.org/10.1101/gr.188901.

Elleder, D. *et al.* (2012) 'Polymorphic integrations of an endogenous gammaretrovirus in the mule deer genome', *J Virol,* 86(5), pp. 2787-96. Available at: https://doi.org/10.1128/JVI.06859-11.

Estécio, M. R. *et al.* (2007) 'LINE-1 hypomethylation in cancer is highly variable and inversely correlated with microsatellite instability', *PLoS One,* 2(5), pp. e399. Available at: https://doi.org/10.1371/journal.pone.0000399.

Fasano, S. *et al.* (2009) 'Ras-guanine nucleotide-releasing factor 1 (Ras-GRF1) controls activation of extracellular signal-regulated kinase (ERK) signaling in the striatum and long-term behavioral responses to cocaine', *Biol Psychiatry,* 66(8), pp. 758-68. Available at: https://doi.org/10.1016/j.biopsych.2009.03.014.

Faucard, R. *et al.* (2016) 'Human Endogenous Retrovirus and Neuroinflammation in Chronic Inflammatory Demyelinating Polyradiculoneuropathy', *EBioMedicine,* 6, pp. 190-198. Available at: https://doi.org/10.1016/j.ebiom.2016.03.001.

Finnegan, D. J. (1989) 'Eukaryotic transposable elements and genome evolution', *Trends Genet,* 5(4), pp. 103-7. Available at: https://doi.org/10.1016/0168-9525(89)90039-5.

Fischer, S. *et al.* (2014) 'Human endogenous retrovirus np9 gene is over expressed in chronic lymphocytic leukemia patients', *Leuk Res Rep,* 3(2), pp. 70-2. Available at: https://doi.org/10.1016/j.lrr.2014.06.005.

Flockerzi, A. *et al.* (2008) 'Expression patterns of transcribed human endogenous retrovirus HERV-K(HML-2) loci in human tissues and the need for a HERV Transcriptome Project', *BMC Genomics,* 9, pp. 354. Available at: https://doi.org/10.1186/1471-2164-9-354.

Francis, J. M. *et al.* (2013) 'Somatic mutation of CDKN1B in small intestine neuroendocrine tumors', *Nat Genet,* 45(12), pp. 1483-6. Available at: https://doi.org/10.1038/ng.2821.

Freimanis, G. *et al.* (2010) 'A role for human endogenous retrovirus-K (HML-2) in rheumatoid arthritis: investigating mechanisms of pathogenesis', *Clin Exp Immunol,* 160(3), pp. 340-7. Available at: https://doi.org/10.1111/j.1365-2249.2010.04110.x.

Fuentes, D. R., Swigut, T. and Wysocka, J. (2018) 'Systematic perturbation of retroviral LTRs reveals widespread long-range effects on human gene regulation', *Elife,* 7. Available at: https://doi.org/10.7554/eLife.35989.

Gabriel, U. *et al.* (2010) 'Smoking increases transcription of human endogenous retroviruses in a newly established in vitro cell model and in normal urothelium', *AIDS Res Hum Retroviruses,* 26(8), pp. 883-8. Available at: https://doi.org/10.1089/aid.2010.0014.

Gandillet, A. *et al.* (2011) 'Heterogeneous sensitivity of human acute myeloid leukemia to β-catenin down-modulation', *Leukemia,* 25(5), pp. 770-780. Available at: https://doi.org/10.1038/leu.2011.17.

García-Montojo, M. *et al.* (2014) 'HERV-W polymorphism in chromosome X is associated with multiple sclerosis risk and with differential expression of MSRV', *Retrovirology,* 11, pp. 2. Available at: https://doi.org/10.1186/1742-4690-11-2.

Gattelli, A. *et al.* (2006) 'Selection of early-occurring mutations dictates hormone-independent progression in mouse mammary tumor lines', *J Virol,* 80(22), pp. 11409-15. Available at: https://doi.org/10.1128/JVI.00234-06.

Gaudin, P. *et al.* (2000) 'Infrequency of detection of particle-associated MSRV/HERV-W RNA in the synovial fluid of patients with rheumatoid arthritis', *Rheumatology (Oxford),* 39(9), pp. 950-4. Available at: https://doi.org/10.1093/rheumatology/39.9.950.

Gessain, A. *et al.* (1985) 'Antibodies to human T-lymphotropic virus type-I in patients with tropical spastic paraparesis', *Lancet,* 2(8452), pp. 407-10. Available at: https://doi.org/10.1016/s0140-6736(85)92734-5.

Gifford, R. J. *et al.* (2018) 'Nomenclature for endogenous retrovirus (ERV) loci', *Retrovirology,* 15(1), pp. 59. Available at: https://doi.org/10.1186/s12977-018-0442-1.

Goering, W., Ribarska, T. and Schulz, W. A. (2011) 'Selective changes of retroelement expression in human prostate cancer', *Carcinogenesis,* 32(10), pp. 1484-92. Available at: https://doi.org/10.1093/carcin/bgr181.

Golan, M. *et al.* (2008) 'Human endogenous retrovirus (HERV-K) reverse transcriptase as a breast cancer prognostic marker', *Neoplasia,* 10(6), pp. 521-33. Available at: https://doi.org/10.1593/neo.07986.

Golovkina, T. V. *et al.* (1992) 'Transgenic mouse mammary tumor virus superantigen expression prevents viral infection', *Cell,* 69(4), pp. 637-45. Available at: https://doi.org/10.1016/0092-8674(92)90227-4.

Grandi, N. *et al.* (2018) 'HERV-W group evolutionary history in non-human primates: characterization of ERV-W orthologs in Catarrhini and related ERV groups in Platyrrhini', *BMC Evol Biol,* 18(1), pp. 6. Available at: https://doi.org/10.1186/s12862-018-1125-1.

Grandi, N. *et al.* (2016) 'Contribution of type W human endogenous retroviruses to the human genome: characterization of HERV-W proviral insertions and processed pseudogenes', *Retrovirology,* 13(1), pp. 67. Available at: https://doi.org/10.1186/s12977-016-0301-x.

Gray, L. R. *et al.* (2019) 'HIV-1 Rev interacts with HERV-K RcREs present in the human genome and promotes export of unspliced HERV-K proviral RNA', *Retrovirology,* 16(1), pp. 40. Available at: https://doi.org/10.1186/s12977-019-0505-y.

Greenberg, M. V. C. and Bourc'his, D. (2019) 'The diverse roles of DNA methylation in mammalian development and disease', *Nat Rev Mol Cell Biol,* 20(10), pp. 590-607. Available at: https://doi.org/10.1038/s41580-019-0159-6.

Griffith, O. L. *et al.* (2016) 'A genomic case study of mixed fibrolamellar hepatocellular carcinoma', *Ann Oncol,* 27(6), pp. 1148-1154. Available at: https://doi.org/10.1093/annonc/mdw135.

Grindstad, T. *et al.* (2018) 'Progesterone Receptors in Prostate Cancer: Progesterone receptor B is the isoform associated with disease progression', *Sci Rep,* 8(1), pp. 11358. Available at: https://doi.org/10.1038/s41598-018-29520-5.

Grow, E. J. *et al.* (2015) 'Intrinsic retroviral reactivation in human preimplantation embryos and pluripotent cells', *Nature,* 522(7555), pp. 221-5. Available at: https://doi.org/10.1038/nature14308.

Guo, Y. J. *et al.* (2020) 'ERK/MAPK signalling pathway and tumorigenesis', *Exp Ther Med,* 19(3), pp. 1997-2007. Available at: https://doi.org/10.3892/etm.2020.8454.

Han, Z. J. *et al.* (2018) 'The post-translational modification, SUMOylation, and cancer (Review)', *Int J Oncol,* 52(4), pp. 1081-1094. Available at: https://doi.org/10.3892/ijo.2018.4280.

Hanger, J. J. *et al.* (2000) 'The nucleotide sequence of koala (Phascolarctos cinereus) retrovirus: a novel type C endogenous virus related to Gibbon ape leukemia virus', *J Virol,* 74(9), pp. 4264-72. Available at: https://doi.org/10.1128/jvi.74.9.4264-4272.2000.

Hayward, J. A. *et al.* (2020) 'Infectious KoRV-related retroviruses circulating in Australian bats', *Proc Natl Acad Sci U S A,* 117(17), pp. 9529-9536. Available at: https://doi.org/10.1073/pnas.1915400117.

Hegyi, H. (2013) 'GABBR1 has a HERV-W LTR in its regulatory region--a possible implication for schizophrenia', *Biol Direct,* 8, pp. 5. Available at: https://doi.org/10.1186/1745-6150-8-5.

Helena Mangs, A. and Morris, B. J. (2007) 'The Human Pseudoautosomal Region (PAR): Origin, Function and Future', *Curr Genomics,* 8(2), pp. 129-36.

Helgeson, B. E. *et al.* (2008) 'Characterization of TMPRSS2:ETV5 and SLC45A3:ETV5 gene fusions in prostate cancer', *Cancer Res,* 68(1), pp. 73-80. Available at: https://doi.org/10.1158/0008-5472.CAN-07-5352.

Hemesath, T. J. *et al.* (1998) 'MAP kinase links the transcription factor Microphthalmia to c-Kit signalling in melanocytes', *Nature,* 391(6664), pp. 298-301. Available at: https://doi.org/10.1038/34681.

Hermans, K. G. *et al.* (2008) 'Truncated ETV1, fused to novel tissue-specific genes, and full-length ETV1 in prostate cancer', *Cancer Res,* 68(18), pp. 7541-9. Available at: https://doi.org/10.1158/0008-5472.CAN-07-5930.

Herniou, E. *et al.* (1998) 'Retroviral diversity and distribution in vertebrates', *J Virol,* 72(7), pp. 5955-66. Available at: https://doi.org/10.1128/JVI.72.7.5955-5966.1998.

Hjelle, B. *et al.* (1992) 'Chronic neurodegenerative disease associated with HTLV-II infection', *Lancet,* 339(8794), pp. 645-6. Available at: https://doi.org/10.1016/0140-6736(92)90797-7.

Hohenadl, C. *et al.* (1999) 'Transcriptional activation of endogenous retroviral sequences in human epidermal keratinocytes by UVB irradiation', *J Invest Dermatol,* 113(4), pp. 587-94. Available at: https://doi.org/10.1046/j.1523-1747.1999.00728.x.

Holloway, J. R. *et al.* (2019) 'Gorillas have been infected with the HERV-K (HML-2) endogenous retrovirus much more recently than humans and chimpanzees', *Proc Natl Acad Sci U S A,* 116(4), pp. 1337-1346. Available at: https://doi.org/10.1073/pnas.1814203116.

Houzet, L. *et al.* (2006) 'Intracellular assembly and budding of the Murine Leukemia Virus in infected cells', *Retrovirology,* 3, pp. 12. Available at: https://doi.org/10.1186/1742-4690-3-12.

Hu, L. *et al.* (2006) 'Expression of human endogenous gammaretroviral sequences in endometriosis and ovarian cancer', *AIDS Res Hum Retroviruses,* 22(6), pp. 551-7. Available at: https://doi.org/10.1089/aid.2006.22.551.

Huang, W. *et al.* (2011) 'Implication of the env gene of the human endogenous retrovirus W family in the expression of BDNF and DRD3 and development of recent-onset schizophrenia', *Schizophr Bull,* 37(5), pp. 988-1000. Available at: https://doi.org/10.1093/schbul/sbp166.

Hughes, D. J. *et al.* (2012) 'Contributions of CTCF and DNA methyltransferases DNMT1 and DNMT3B to Epstein-Barr virus restricted latency', *J Virol,* 86(2), pp. 1034-45. Available at: https://doi.org/10.1128/JVI.05923-11.

Hughes, J. F. and Coffin, J. M. (2001) 'Evidence for genomic rearrangements mediated by human endogenous retroviruses during primate evolution', *Nat Genet,* 29(4), pp. 487-9. Available at: https://doi.org/10.1038/ng775.

Hughes, J. F. and Coffin, J. M. (2005) 'Human endogenous retroviral elements as indicators of ectopic recombination events in the primate genome', *Genetics,* 171(3), pp. 1183-94. Available at: https://doi.org/10.1534/genetics.105.043976.

Huh, J. W. *et al.* (2007) 'Long terminal repeats of porcine endogenous retroviruses in Sus scrofa', *Arch Virol,* 152(12), pp. 2271-6. Available at: https://doi.org/10.1007/s00705-007-1049-3.

Hurst, T. P. and Magiorkinis, G. (2017) 'Epigenetic Control of Human Endogenous Retrovirus Expression: Focus on Regulation of Long-Terminal Repeats (LTRs)', *Viruses,* 9(6). Available at: https://doi.org/10.3390/v9060130.

Ishida, Y. *et al.* (2015) 'Proliferation of endogenous retroviruses in the early stages of a host germ line invasion', *Mol Biol Evol,* 32(1), pp. 109-20. Available at: https://doi.org/10.1093/molbev/msu275.

Jabbari, K. and Bernardi, G. (2004) 'Cytosine methylation and CpG, TpG (CpA) and TpA frequencies', *Gene,* 333, pp. 143-9. Available at: https://doi.org/10.1016/j.gene.2004.02.043.

Jeong, B. H. *et al.* (2010) 'The prevalence of human endogenous retroviruses in cerebrospinal fluids from patients with sporadic Creutzfeldt-Jakob disease', *J Clin Virol,* 47(2), pp. 136-42. Available at: https://doi.org/10.1016/j.jcv.2009.11.016.

Jha, A. R. *et al.* (2009) 'Cross-sectional dating of novel haplotypes of HERV-K 113 and HERV-K 115 indicate these proviruses originated in Africa before Homo sapiens', *Mol Biol Evol,* 26(11), pp. 2617-26. Available at: https://doi.org/10.1093/molbev/msp180.

Johanning, G. L. *et al.* (2017) 'Expression of human endogenous retrovirus-K is strongly associated with the basal-like breast cancer phenotype', *Sci Rep,* 7, pp. 41960. Available at: https://doi.org/10.1038/srep41960.

Johnsen, D. O. *et al.* (1971) 'Malignant lymphoma in the gibbon', *J Am Vet Med Assoc,* 159(5), pp. 563-6.

Johnson, L. J. and Brookfield, J. F. (2006) 'A test of the master gene hypothesis for interspersed repetitive DNA sequences', *Mol Biol Evol,* 23(2), pp. 235-9. Available at: https://doi.org/10.1093/molbev/msj034.

Johnson, W. E. and Coffin, J. M. (1999) 'Constructing primate phylogenies from ancient retrovirus sequences', *Proc Natl Acad Sci U S A,* 96(18), pp. 10254-60. Available at: https://doi.org/10.1073/pnas.96.18.10254.

Johnston, J. B. *et al.* (2001) 'Monocyte activation and differentiation augment human endogenous retrovirus expression: implications for inflammatory brain diseases', *Ann Neurol,* 50(4), pp. 434-42. Available at: https://doi.org/10.1002/ana.1131.

Kahyo, T. *et al.* (2017) 'Insertionally polymorphic sites of human endogenous retrovirus-K (HML-2) with long target site duplications', *BMC Genomics,* 18(1), pp. 487. Available at: https://doi.org/10.1186/s12864-017-3872-6.

Kalluri, R. and Weinberg, R. A. (2009) 'The basics of epithelial-mesenchymal transition', *J Clin Invest,* 119(6), pp. 1420-8. Available at: https://doi.org/10.1172/JCI39104.

Kamath, P. L. *et al.* (2014) 'The population history of endogenous retroviruses in mule deer (Odocoileus hemionus)', *J Hered,* 105(2), pp. 173-87. Available at: https://doi.org/10.1093/jhered/est088.

Kamp, C. *et al.* (2000) 'Two long homologous retroviral sequence blocks in proximal Yq11 cause AZFa microdeletions as a result of intrachromosomal recombination events', *Hum Mol Genet,* 9(17), pp. 2563-72. Available at: https://doi.org/10.1093/hmg/9.17.2563.

Kanda, R. K., Tristem, M. and Coulson, T. (2013) 'Exploring the effects of immunity and life history on the dynamics of an endogenous retrovirus', *Philos Trans R Soc Lond B Biol Sci,* 368(1626), pp. 20120505. Available at: https://doi.org/10.1098/rstb.2012.0505.

Kanki, P. J., Hopper, J. R. and Essex, M. (1987) 'The origins of HIV-1 and HTLV-4/HIV-2', *Ann N Y Acad Sci,* 511, pp. 370-5. Available at: https://doi.org/10.1111/j.1749-6632.1987.tb36265.x.

Kapitonov, V. V. and Jurka, J. (2008) 'A universal classification of eukaryotic transposable elements implemented in Repbase', *Nat Rev Genet,* 9(5), pp. 411-2; author reply 414. Available at: https://doi.org/10.1038/nrg2165-c1.

Kaplan, J. E. *et al.* (1990) 'The risk of development of HTLV-I-associated myelopathy/tropical spastic paraparesis among persons infected with HTLV-I', *J Acquir Immune Defic Syndr (1988),* 3(11), pp. 1096-101.

Karamitros, T. *et al.* (2018) 'Human Endogenous Retrovirus-K HML-2 integration within', *Proc Natl Acad Sci U S A,* 115(41), pp. 10434-10439. Available at: https://doi.org/10.1073/pnas.1811940115.

Karlsson, H. *et al.* (2001) 'Retroviral RNA identified in the cerebrospinal fluids and brains of individuals with schizophrenia', *Proc Natl Acad Sci U S A,* 98(8), pp. 4634-9. Available at: https://doi.org/10.1073/pnas.061021998.

Kass, D. H., Batzer, M. A. and Deininger, P. L. (1995) 'Gene conversion as a secondary mechanism of short interspersed element (SINE) evolution', *Mol Cell Biol,* 15(1), pp. 19-25. Available at: https://doi.org/10.1128/MCB.15.1.19.

Katoh, I. *et al.* (2011) 'Activation of the long terminal repeat of human endogenous retrovirus K by melanoma-specific transcription factor MITF-M', *Neoplasia,* 13(11), pp. 1081-92.

Katzourakis, A., Rambaut, A. and Pybus, O. G. (2005) 'The evolutionary dynamics of endogenous retroviruses', *Trends Microbiol,* 13(10), pp. 463-8. Available at: https://doi.org/10.1016/j.tim.2005.08.004.

Katzourakis, A. *et al.* (2007) 'Discovery and analysis of the first endogenous lentivirus', *Proc Natl Acad Sci U S A,* 104(15), pp. 6261-5. Available at: https://doi.org/10.1073/pnas.0700471104.

Kaufmann, S. *et al.* (2010) 'Human endogenous retrovirus protein Rec interacts with the testicular zinc-finger protein and androgen receptor', *J Gen Virol,* 91(Pt 6), pp. 1494-502. Available at: https://doi.org/10.1099/vir.0.014241-0.

Kawakami, T. G. and Buckley, P. M. (1974) 'Antigenic studies on gibbon type-C viruses', *Transplant Proc,* 6(2), pp. 193-6.

Kawakami, T. G., Kollias, G. V. and Holmberg, C. (1980) 'Oncogenicity of gibbon type-C myelogenous leukemia virus', *Int J Cancer,* 25(5), pp. 641-6. Available at: https://doi.org/10.1002/ijc.2910250514.

Khodosevich, K., Lebedev, Y. and Sverdlov, E. (2002) 'Endogenous retroviruses and human evolution', *Comp Funct Genomics,* 3(6), pp. 494-8. Available at: https://doi.org/10.1002/cfg.216.

Kim, H. S., Ahn, K. and Kim, D. S. (2008) 'Quantitative expression of the HERV-W env gene in human tissues', *Arch Virol,* 153(8), pp. 1587-91. Available at: https://doi.org/10.1007/s00705-008-0159-x.

Kleiman, A. *et al.* (2004) 'HERV-K(HML-2) GAG/ENV antibodies as indicator for therapy effect in patients with germ cell tumors', *Int J Cancer,* 110(3), pp. 459-61. Available at: https://doi.org/10.1002/ijc.11649.

Kono, K. *et al.* (2021) 'Infectivity assessment of porcine endogenous retrovirus using high-throughput sequencing technologies', *Biologicals,* 71, pp. 1-8. Available at: https://doi.org/10.1016/j.biologicals.2021.05.001.

Krakower, J. M. *et al.* (1978) 'Antigenic characterization of a new gibbon ape leukemia virus isolate: seroepidemiologic assessment of an outbreak of gibbon leukemia', *Int J Cancer,* 22(6), pp. 715-20. Available at: https://doi.org/10.1002/ijc.2910220613.

Krzysztalowska-Wawrzyniak, M. *et al.* (2011) 'The distribution of human endogenous retrovirus K-113 in health and autoimmune diseases in Poland', *Rheumatology (Oxford),* 50(7), pp. 1310-4. Available at: https://doi.org/10.1093/rheumatology/ker022.

Kumar, S. *et al.* (2017) 'TimeTree: A Resource for Timelines, Timetrees, and Divergence Times', *Mol Biol Evol,* 34(7), pp. 1812-1819. Available at: https://doi.org/10.1093/molbev/msx116.

Lamprecht, B. *et al.* (2010) 'Derepression of an endogenous long terminal repeat activates the CSF1R proto-oncogene in human lymphoma', *Nat Med,* 16(5), pp. 571-9, 1p following 579. Available at: https://doi.org/10.1038/nm.2129.

Lander, E. S. *et al.* (2001) 'Initial sequencing and analysis of the human genome', *Nature,* 409(6822), pp. 860-921. Available at: https://doi.org/10.1038/35057062.

Largaespada, D. A. (2000) 'Genetic heterogeneity in acute myeloid leukemia: maximizing information flow from MuLV mutagenesis studies', *Leukemia,* 14(7), pp. 1174-84. Available at: https://doi.org/10.1038/sj.leu.2401852.

Lavialle, C. *et al.* (2013) 'Paleovirology of 'syncytins', retroviral env genes exapted for a role in placentation', *Philos Trans R Soc Lond B Biol Sci,* 368(1626), pp. 20120507. Available at: https://doi.org/10.1098/rstb.2012.0507.

Lavie, L. *et al.* (2005) 'CpG methylation directly regulates transcriptional activity of the human endogenous retrovirus family HERV-K(HML-2)', *J Virol,* 79(2), pp. 876-83. Available at: https://doi.org/10.1128/JVI.79.2.876-883.2005.

Lawson, J. S. and Glenn, W. K. (2017) 'Multiple oncogenic viruses are present in human breast tissues before development of virus associated breast cancer', *Infect Agent Cancer,* 12, pp. 55. Available at: https://doi.org/10.1186/s13027-017-0165-2.

Lee, A. *et al.* (2014) 'Novel Denisovan and Neanderthal retroviruses', *J Virol,* 88(21), pp. 12907-9. Available at: https://doi.org/10.1128/JVI.01825-14.

Lee, W. J. *et al.* (2003) 'Activation of the human endogenous retrovirus W long terminal repeat by herpes simplex virus type 1 immediate early protein 1', *Mol Cells,* 15(1), pp. 75-80.

Lemaître, C. *et al.* (2017) 'A human endogenous retrovirus-derived gene that can contribute to oncogenesis by activating the ERK pathway and inducing migration and invasion', *PLoS Pathog,* 13(6), pp. e1006451. Available at: https://doi.org/10.1371/journal.ppat.1006451.

Lernmark, A. *et al.* (1997) 'Family cell lines available for research--an endangered resource?', *Am J Hum Genet,* 61(3), pp. 778-9. Available at: https://doi.org/10.1016/S0002-9297(07)64345-6.

Lessi, F. *et al.* (2020) 'A human MMTV-like betaretrovirus linked to breast cancer has been present in humans at least since the copper age', *Aging (Albany NY),* 12(16), pp. 15978-15994. Available at: https://doi.org/10.18632/aging.103780.

Leung, D. C. *et al.* (2011) 'Lysine methyltransferase G9a is required for de novo DNA methylation and the establishment, but not the maintenance, of proviral silencing', *Proc Natl Acad Sci U S A,* 108(14), pp. 5718-23. Available at: https://doi.org/10.1073/pnas.1014660108.

Li, F. and Karlsson, H. (2016) 'Expression and regulation of human endogenous retrovirus W elements', *APMIS,* 124(1-2), pp. 52-66. Available at: https://doi.org/https://doi.org/10.1111/apm.12478.

Li, F. *et al.* (2019) 'Transcription of human endogenous retroviruses in human brain by RNA-seq analysis', *PLoS One,* 14(1), pp. e0207353. Available at: https://doi.org/10.1371/journal.pone.0207353.

Li, M. *et al.* (2017) 'Downregulation of Human Endogenous Retrovirus Type K (HERV-K) Viral env RNA in Pancreatic Cancer Cells Decreases Cell Proliferation and Tumor Growth', *Clin Cancer Res,* 23(19), pp. 5892-5911. Available at: https://doi.org/10.1158/1078-0432.CCR-17-0001.

Li, Z. *et al.* (2010) 'Expression of HERV-K correlates with status of MEK-ERK and p16INK4A-CDK4 pathways in melanoma cells', *Cancer Invest,* 28(10), pp. 1031-7. Available at: https://doi.org/10.3109/07357907.2010.512604.

Liang, W. S. *et al.* (2014) 'Long insert whole genome sequencing for copy number variant and translocation detection', *Nucleic Acids Res,* 42(2), pp. e8. Available at: https://doi.org/10.1093/nar/gkt865.

Liang, W. S. *et al.* (2012) 'Genome-wide characterization of pancreatic adenocarcinoma patients using next generation sequencing', *PLoS One,* 7(10), pp. e43192. Available at: https://doi.org/10.1371/journal.pone.0043192.

Lieber, M. M. *et al.* (1975) 'Isolation from the asian mouse Mus caroli of an endogenous type C virus related to infectious primate type C viruses', *Proc Natl Acad Sci U S A,* 72(6), pp. 2315-9. Available at: https://doi.org/10.1073/pnas.72.6.2315.

Lizarraga-Valderrama, L. R. and Sheridan, G. K. (2021) 'Extracellular vesicles and intercellular communication in the central nervous system', *FEBS Lett,* 595(10), pp. 1391-1410. Available at: https://doi.org/10.1002/1873-3468.14074.

Lowther, W. *et al.* (2005) 'A new common integration site, Int7, for the mouse mammary tumor virus in mouse mammary tumors identifies a gene whose product has furin-like and thrombospondin-like sequences', *J Virol,* 79(15), pp. 10093-6. Available at: https://doi.org/10.1128/JVI.79.15.10093-10096.2005.

Luppi, P. *et al.* (2000) 'Restricted TCR V beta gene expression and enterovirus infection in type I diabetes: a pilot study', *Diabetologia,* 43(12), pp. 1484-97. Available at: https://doi.org/10.1007/s001250051559.

Löwer, R. *et al.* (1993) 'Identification of human endogenous retroviruses with complex mRNA expression and particle formation', *Proc Natl Acad Sci U S A,* 90(10), pp. 4480-4. Available at: https://doi.org/10.1073/pnas.90.10.4480.

Ma, W. *et al.* (2016) 'Human Endogenous Retroviruses-K (HML-2) Expression Is Correlated with Prognosis and Progress of Hepatocellular Carcinoma', *Biomed Res Int,* 2016, pp. 8201642. Available at: https://doi.org/10.1155/2016/8201642.

MacArthur, C. A., Shankar, D. B. and Shackleford, G. M. (1995) 'Fgf-8, activated by proviral insertion, cooperates with the Wnt-1 transgene in murine mammary tumorigenesis', *J Virol,* 69(4), pp. 2501-7. Available at: https://doi.org/10.1128/JVI.69.4.2501-2507.1995.

Macfarlan, T. S. *et al.* (2012) 'Embryonic stem cell potency fluctuates with endogenous retrovirus activity', *Nature,* 487(7405), pp. 57-63. Available at: https://doi.org/10.1038/nature11244.

Macfarlane, C. and Simmonds, P. (2004) 'Allelic variation of HERV-K(HML-2) endogenous retroviral elements in human populations', *J Mol Evol,* 59(5), pp. 642-56. Available at: https://doi.org/10.1007/s00239-004-2656-1.

Maeurer, M. J. *et al.* (1996) 'Tumor escape from immune recognition: lethal recurrent melanoma in a patient associated with downregulation of the peptide transporter protein TAP-1 and loss of expression of the immunodominant MART-1/Melan-A antigen', *J Clin Invest,* 98(7), pp. 1633-41. Available at: https://doi.org/10.1172/JCI118958.

Mameli, G. *et al.* (2012) 'Expression and activation by Epstein Barr virus of human endogenous retroviruses-W in blood cells and astrocytes: inference for multiple sclerosis', *PLoS One,* 7(9), pp. e44991. Available at: https://doi.org/10.1371/journal.pone.0044991.

Marchetti, A. *et al.* (1995) 'Int-6, a highly conserved, widely expressed gene, is mutated by mouse mammary tumor virus in mammary preneoplasia', *J Virol,* 69(3), pp. 1932-8. Available at: https://doi.org/10.1128/JVI.69.3.1932-1938.1995.

Marchi, E. *et al.* (2013) 'Neanderthal and Denisovan retroviruses in modern humans', *Current biology : CB,* 23(22), pp. R994-R995. Available at: https://doi.org/10.1016/j.cub.2013.10.028.

Marguerat, S. *et al.* (2004) 'Association of human endogenous retrovirus K-18 polymorphisms with type 1 diabetes', *Diabetes,* 53(3), pp. 852-4. Available at: https://doi.org/10.2337/diabetes.53.3.852.

Martin, D. N., Starks, A. M. and Ambs, S. (2013) 'Biological determinants of health disparities in prostate cancer', *Curr Opin Oncol,* 25(3), pp. 235-41. Available at: https://doi.org/10.1097/CCO.0b013e32835eb5d1.

Matushansky, I., Radparvar, F. and Skoultchi, A. I. (2003) 'CDK6 blocks differentiation: coupling cell proliferation to the block to differentiation in leukemic cells', *Oncogene,* 22(27), pp. 4143-9. Available at: https://doi.org/10.1038/sj.onc.1206484.

Mayer, J. *et al.* (1999) 'An almost-intact human endogenous retrovirus K on human chromosome 7', *Nat Genet,* 21(3), pp. 257-8. Available at: https://doi.org/10.1038/6766.

Mazzanti, C. M. *et al.* (2015) 'Human saliva as route of inter-human infection for mouse mammary tumor virus', *Oncotarget,* 6(21), pp. 18355-63. Available at: https://doi.org/10.18632/oncotarget.4567.

McCarthy, E. M. and McDonald, J. F. (2004) 'Long terminal repeat retrotransposons of Mus musculus', *Genome Biol,* 5(3), pp. R14. Available at: https://doi.org/10.1186/gb-2004-5-3-r14.

McConnell, M. J. *et al.* (2003) 'Growth suppression by acute promyelocytic leukemia-associated protein PLZF is mediated by repression of c-myc expression', *Mol Cell Biol,* 23(24), pp. 9375-88. Available at: https://doi.org/10.1128/MCB.23.24.9375-9388.2003.

Medstrand, P. and Blomberg, J. (1993) 'Characterization of novel reverse transcriptase encoding human endogenous retroviral sequences similar to type A and type B retroviruses: differential transcription in normal human tissues', *J Virol,* 67(11), pp. 6778-87.

Medstrand, P. and Mager, D. L. (1998) 'Human-specific integrations of the HERV-K endogenous retrovirus family', *J Virol,* 72(12), pp. 9782-7. Available at: https://doi.org/10.1128/JVI.72.12.9782-9787.1998.

Menendez, L., Benigno, B. B. and McDonald, J. F. (2004) 'L1 and HERV-W retrotransposons are hypomethylated in human ovarian carcinomas', *Mol Cancer,* 3, pp. 12. Available at: https://doi.org/10.1186/1476-4598-3-12.

Meyer, M. *et al.* (2012) 'A high-coverage genome sequence from an archaic Denisovan individual', *Science,* 338(6104), pp. 222-6. Available at: https://doi.org/10.1126/science.1224344.

Meyer, T. J. *et al.* (2017) 'Endogenous Retroviruses: With Us and against Us', *Front Chem,* 5, pp. 23. Available at: https://doi.org/10.3389/fchem.2017.00023.

Mi, S. *et al.* (2000) 'Syncytin is a captive retroviral envelope protein involved in human placental morphogenesis', *Nature,* 403(6771), pp. 785-9. Available at: https://doi.org/10.1038/35001608.

Miller, A. D. (2003) 'Identification of Hyal2 as the cell-surface receptor for jaagsiekte sheep retrovirus and ovine nasal adenocarcinoma virus', *Curr Top Microbiol Immunol,* 275, pp. 179-99. Available at: https://doi.org/10.1007/978-3-642-55638-8_7.

Mizuuchi, K. (1984) 'Mechanism of transposition of bacteriophage Mu: polarity of the strand transfer reaction at the initiation of transposition', *Cell,* 39(2 Pt 1), pp. 395-404. Available at: https://doi.org/10.1016/0092-8674(84)90018-7.

Moloney, J. B. (1960) 'Biological studies on a lymphoid-leukemia virus extracted from sarcoma 37. I. Origin and introductory investigations', *J Natl Cancer Inst,* 24, pp. 933-51.

Murgia, C. *et al.* (2011) 'Lung adenocarcinoma originates from retrovirus infection of proliferating type 2 pneumocytes during pulmonary post-natal development or tissue repair', *PLoS Pathog,* 7(3), pp. e1002014. Available at: https://doi.org/10.1371/journal.ppat.1002014.

Murphree, A. L. and Benedict, W. F. (1984) 'Retinoblastoma: clues to human oncogenesis', *Science,* 223(4640), pp. 1028-33. Available at: https://doi.org/10.1126/science.6320372.

Muster, T. *et al.* (2003) 'An endogenous retrovirus derived from human melanoma cells', *Cancer Res,* 63(24), pp. 8735-41.

Nguyen, T. D. *et al.* (2019) 'Female Sex Hormones Activate Human Endogenous Retrovirus Type K Through the OCT4 Transcription Factor in T47D Breast Cancer Cells', *AIDS Res Hum Retroviruses,* 35(3), pp. 348-356. Available at: https://doi.org/10.1089/AID.2018.0173.

Nisole, S. and Saïb, A. (2004) 'Early steps of retrovirus replicative cycle', *Retrovirology,* 1, pp. 9. Available at: https://doi.org/10.1186/1742-4690-1-9.

Nusse, R. *et al.* (1984) 'Mode of proviral activation of a putative mammary oncogene (int-1) on mouse chromosome 15', *Nature,* 307(5947), pp. 131-6. Available at: https://doi.org/10.1038/307131a0.

O'Carroll, I. P. *et al.* (2020) 'Structural Mimicry Drives HIV-1 Rev-Mediated HERV-K Expression', *J Mol Biol,* 432(24), pp. 166711. Available at: https://doi.org/10.1016/j.jmb.2020.11.010.

Oh, S., Shin, S. and Janknecht, R. (2012) 'ETV1, 4 and 5: an oncogenic subfamily of ETS transcription factors', *Biochim Biophys Acta,* 1826(1), pp. 1-12. Available at: https://doi.org/10.1016/j.bbcan.2012.02.002.

Okano, M., Xie, S. and Li, E. (1998) 'Cloning and characterization of a family of novel mammalian DNA (cytosine-5) methyltransferases', *Nat Genet,* 19(3), pp. 219-20. Available at: https://doi.org/10.1038/890.

Oluwole, S. O. *et al.* (2007) 'Elevated levels of transcripts encoding a human retroviral envelope protein (syncytin) in muscles from patients with motor neuron disease', *Amyotroph Lateral Scler,* 8(2), pp. 67-72. Available at: https://doi.org/10.1080/17482960600864207.

Ono, M., Kawakami, M. and Takezawa, T. (1987) 'A novel human nonviral retroposon derived from an endogenous retrovirus', *Nucleic Acids Res,* 15(21), pp. 8725-37. Available at: https://doi.org/10.1093/nar/15.21.8725.

Ostertag, E. M. *et al.* (2003) 'SVA elements are nonautonomous retrotransposons that cause disease in humans', *Am J Hum Genet,* 73(6), pp. 1444-51. Available at: https://doi.org/10.1086/380207.

Palmarini, M. *et al.* (1999) 'Jaagsiekte sheep retrovirus is necessary and sufficient to induce a contagious lung cancer in sheep', *J Virol,* 73(8), pp. 6964-72. Available at: https://doi.org/10.1128/JVI.73.8.6964-6972.1999.

Pastuzyn, E. D. *et al.* (2018) 'The Neuronal Gene Arc Encodes a Repurposed Retrotransposon Gag Protein that Mediates Intercellular RNA Transfer', *Cell,* 172(1-2), pp. 275-288.e18. Available at: https://doi.org/10.1016/j.cell.2017.12.024.

Pearson, G. *et al.* (2001) 'Mitogen-activated protein (MAP) kinase pathways: regulation and physiological functions', *Endocr Rev,* 22(2), pp. 153-83. Available at: https://doi.org/10.1210/edrv.22.2.0428.

Perron, H. *et al.* (1997) 'Molecular identification of a novel retrovirus repeatedly isolated from patients with multiple sclerosis. The Collaborative Research Group on Multiple Sclerosis', *Proc Natl Acad Sci U S A,* 94(14), pp. 7583-8. Available at: https://doi.org/10.1073/pnas.94.14.7583.

Perron, H. *et al.* (1989) 'Leptomeningeal cell line from multiple sclerosis with reverse transcriptase activity and viral particles', *Res Virol,* 140(6), pp. 551-61. Available at: https://doi.org/10.1016/s0923-2516(89)80141-4.

Perron, H. *et al.* (2012) 'Human endogenous retrovirus type W envelope expression in blood and brain cells provides new insights into multiple sclerosis disease', *Mult Scler,* 18(12), pp. 1721-36. Available at: https://doi.org/10.1177/1352458512441381.

Perzova, R. *et al.* (2017) 'Is MMTV associated with human breast cancer? Maybe, but probably not', *Virol J,* 14(1), pp. 196. Available at: https://doi.org/10.1186/s12985-017-0862-x.

Perzova, R. *et al.* (2015) 'Increased seroreactivity to human T cell lymphoma/leukemia virus-related endogenous sequence-1 Gag peptides in patients with human T cell lymphoma/leukemia virus myelopathy', *AIDS Res Hum Retroviruses,* 31(2), pp. 242-9. Available at: https://doi.org/10.1089/AID.2014.0171.

Pierce, K. L., Luttrell, L. M. and Lefkowitz, R. J. (2001) 'New mechanisms in heptahelical receptor signaling to mitogen activated protein kinase cascades', *Oncogene,* 20(13), pp. 1532-9. Available at: https://doi.org/10.1038/sj.onc.1204184.

Prado-Martinez, J. *et al.* (2013) 'Great ape genetic diversity and population history', *Nature,* 499(7459), pp. 471-5. Available at: https://doi.org/10.1038/nature12228.

Quigley, B. L. *et al.* (2018) 'Molecular Dynamics and Mode of Transmission of Koala Retrovirus as It Invades and Spreads through a Wild Queensland Koala Population', *J Virol,* 92(5). Available at: https://doi.org/10.1128/JVI.01871-17.

Reiche, J., Pauli, G. and Ellerbrok, H. (2010) 'Differential expression of human endogenous retrovirus K transcripts in primary human melanocytes and melanoma cell lines after UV irradiation', *Melanoma Res,* 20(5), pp. 435-40. Available at: https://doi.org/10.1097/CMR.0b013e32833c1b5d.

Reis, B. S. *et al.* (2013) 'Prostate cancer progression correlates with increased humoral immune response to a human endogenous retrovirus GAG protein', *Clin Cancer Res,* 19(22), pp. 6112-25. Available at: https://doi.org/10.1158/1078-0432.CCR-12-3580.

Rhyu, D. W. *et al.* (2014) 'Expression of human endogenous retrovirus env genes in the blood of breast cancer patients', *Int J Mol Sci,* 15(6), pp. 9173-83. Available at: https://doi.org/10.3390/ijms15069173.

Rogers, A. R., Harris, N. S. and Achenbach, A. A. (2020) 'Neanderthal-Denisovan ancestors interbred with a distantly related hominin', *Sci Adv,* 6(8), pp. eaay5483. Available at: https://doi.org/10.1126/sciadv.aay5483.

Rolland, A. *et al.* (2005) 'Correlation between disease severity and in vitro cytokine production mediated by MSRV (multiple sclerosis associated retroviral element) envelope protein in patients with multiple sclerosis', *J Neuroimmunol,* 160(1-2), pp. 195-203. Available at: https://doi.org/10.1016/j.jneuroim.2004.10.019.

Roulois, D. *et al.* (2015) 'DNA-Demethylating Agents Target Colorectal Cancer Cells by Inducing Viral Mimicry by Endogenous Transcripts', *Cell,* 162(5), pp. 961-73. Available at: https://doi.org/10.1016/j.cell.2015.07.056.

Royuela, M. *et al.* (2001) 'Estrogen receptors alpha and beta in the normal, hyperplastic and carcinomatous human prostate', *J Endocrinol,* 168(3), pp. 447-54. Available at: https://doi.org/10.1677/joe.0.1680447.

Ruprecht, K. *et al.* (2006) 'Regulation of human endogenous retrovirus W protein expression by herpes simplex virus type 1: implications for multiple sclerosis', *J Neurovirol,* 12(1), pp. 65-71. Available at: https://doi.org/10.1080/13550280600614973.

Schanab, O. *et al.* (2011) 'Expression of human endogenous retrovirus K is stimulated by ultraviolet radiation in melanoma', *Pigment Cell Melanoma Res,* 24(4), pp. 656-65. Available at: https://doi.org/10.1111/j.1755-148X.2011.00860.x.

Schmitt, K. *et al.* (2013) 'Transcriptional profiling of human endogenous retrovirus group HERV-K(HML-2) loci in melanoma', *Genome Biol Evol,* 5(2), pp. 307-28. Available at: https://doi.org/10.1093/gbe/evt010.

Schroeder, J. A., Troyer, K. L. and Lee, D. C. (2000) 'Cooperative induction of mammary tumorigenesis by TGFalpha and Wnts', *Oncogene,* 19(28), pp. 3193-9. Available at: https://doi.org/10.1038/sj.onc.1203652.

Schumann, G. *et al.* (2011) 'Genome-wide association and genetic functional studies identify autism susceptibility candidate 2 gene (AUTS2) in the regulation of alcohol consumption', *Proc Natl Acad Sci U S A,* 108(17), pp. 7119-24. Available at: https://doi.org/10.1073/pnas.1017288108.

Schön, U. *et al.* (2001) 'Cell type-specific expression and promoter activity of human endogenous retroviral long terminal repeats', *Virology,* 279(1), pp. 280-91. Available at: https://doi.org/10.1006/viro.2000.0712.

Seifarth, W. *et al.* (2005) 'Comprehensive analysis of human endogenous retrovirus transcriptional activity in human tissues with a retrovirus-specific microarray', *J Virol,* 79(1), pp. 341-52. Available at: https://doi.org/10.1128/JVI.79.1.341-352.2005.

Serafino, A. *et al.* (2009) 'The activation of human endogenous retrovirus K (HERV-K) is implicated in melanoma cell malignant transformation', *Exp Cell Res,* 315(5), pp. 849-62. Available at: https://doi.org/10.1016/j.yexcr.2008.12.023.

She, X. *et al.* (2004) 'Shotgun sequence assembly and recent segmental duplications within the human genome', *Nature,* 431(7011), pp. 927-30. Available at: https://doi.org/10.1038/nature03062.

Shen, L. *et al.* (1994) 'Structure and genetics of the partially duplicated gene RP located immediately upstream of the complement C4A and the C4B genes in the HLA class III region. Molecular cloning, exon-intron structure, composite retroposon, and breakpoint of gene duplication', *J Biol Chem,* 269(11), pp. 8466-76.

Shin, W. *et al.* (2013) 'Human-specific HERV-K insertion causes genomic variations in the human genome', *PLoS One,* 8(4), pp. e60605. Available at: https://doi.org/10.1371/journal.pone.0060605.

Sikpi, M. O. *et al.* (1992) 'Mutations caused by gamma-radiation-induced double-strand breaks in a shuttle plasmid replicated in human lymphoblasts', *Int J Radiat Biol,* 62(5), pp. 555-62. Available at: https://doi.org/10.1080/09553009214552471.

Simpson, G. R. *et al.* (1996) 'Endogenous D-type (HERV-K) related sequences are packaged into retroviral particles in the placenta and possess open reading frames for reverse transcriptase', *Virology,* 222(2), pp. 451-6. Available at: https://doi.org/10.1006/viro.1996.0443.

Somoza, N. *et al.* (1994) 'Pancreas in recent onset insulin-dependent diabetes mellitus. Changes in HLA, adhesion molecules and autoantigens, restricted T cell receptor V beta usage, and cytokine profile', *J Immunol,* 153(3), pp. 1360-77.

Stacey, D. *et al.* (2012) 'RASGRF2 regulates alcohol-induced reinforcement by influencing mesolimbic dopamine neuron activity and dopamine release', *Proc Natl Acad Sci U S A,* 109(51), pp. 21128-33. Available at: https://doi.org/10.1073/pnas.1211844110.

Stauffer, Y. *et al.* (2004) 'Digital expression profiles of human endogenous retroviral families in normal and cancerous tissues', *Cancer Immun,* 4, pp. 2.

Steelman, L. S. *et al.* (2011) 'Roles of the Ras/Raf/MEK/ERK pathway in leukemia therapy', *Leukemia,* 25(7), pp. 1080-94. Available at: https://doi.org/10.1038/leu.2011.66.

Stewart, T. A., Pattengale, P. K. and Leder, P. (1984) 'Spontaneous mammary adenocarcinomas in transgenic mice that carry and express MTV/myc fusion genes', *Cell,* 38(3), pp. 627-37. Available at: https://doi.org/10.1016/0092-8674(84)90257-5.

Stoye, J. P. and Coffin, J. M. (1987) 'The four classes of endogenous murine leukemia virus: structural relationships and potential for recombination', *J Virol,* 61(9), pp. 2659-69. Available at: https://doi.org/10.1128/JVI.61.9.2659-2669.1987.

Subramanian, R. P. *et al.* (2011) 'Identification, characterization, and comparative genomic distribution of the HERV-K (HML-2) group of human endogenous retroviruses', *Retrovirology,* 8, pp. 90. Available at: https://doi.org/10.1186/1742-4690-8-90.

Sugaya, K. *et al.* (1994) 'Three genes in the human MHC class III region near the junction with the class II: gene for receptor of advanced glycosylation end products, PBX2 homeobox gene and a notch homolog, human counterpart of mouse mammary tumor gene int-3', *Genomics,* 23(2), pp. 408-19. Available at: https://doi.org/10.1006/geno.1994.1517.

Sun, T., Liu, Z. and Yang, Q. (2020) 'The role of ubiquitination and deubiquitination in cancer metabolism', *Mol Cancer,* 19(1), pp. 146. Available at: https://doi.org/10.1186/s12943-020-01262-x.

Sundaram, V. *et al.* (2014) 'Widespread contribution of transposable elements to the innovation of gene regulatory networks', *Genome Res,* 24(12), pp. 1963-76. Available at: https://doi.org/10.1101/gr.168872.113.

Sykes, S. M. *et al.* (2011) 'AKT/FOXO signaling enforces reversible differentiation blockade in myeloid leukemias', *Cell,* 146(5), pp. 697-708. Available at: https://doi.org/10.1016/j.cell.2011.07.032.

Szabo, S. *et al.* (2005) 'Human, rhesus macaque, and feline sequences highly similar to mouse mammary tumor virus sequences', *Microsc Res Tech,* 68(3-4), pp. 209-21. Available at: https://doi.org/10.1002/jemt.20233.

Tai, D. *et al.* (2015) 'Targeting the WNT Signaling Pathway in Cancer Therapeutics', *Oncologist,* 20(10), pp. 1189-98. Available at: https://doi.org/10.1634/theoncologist.2015-0057.

Takeichi, M. (1988) 'The cadherins: cell-cell adhesion molecules controlling animal morphogenesis', *Development,* 102(4), pp. 639-55. Available at: https://doi.org/10.1242/dev.102.4.639.

Takeuchi, Y. *et al.* (1998) 'Host range and interference studies of three classes of pig endogenous retrovirus', *J Virol,* 72(12), pp. 9986-91. Available at: https://doi.org/10.1128/JVI.72.12.9986-9991.1998.

Tarlinton, R. E., Meers, J. and Young, P. R. (2006) 'Retroviral invasion of the koala genome', *Nature,* 442(7098), pp. 79-81. Available at: https://doi.org/10.1038/nature04841.

Tatarek, J. *et al.* (2011) 'Notch1 inhibition targets the leukemia-initiating cells in a Tal1/Lmo2 mouse model of T-ALL', *Blood,* 118(6), pp. 1579-90. Available at: https://doi.org/10.1182/blood-2010-08-300343.

Tavakolian, S., Goudarzi, H. and Faghihloo, E. (2019) 'Evaluating the expression level of HERV-K env, np9, rec and gag in breast tissue', *Infect Agent Cancer,* 14, pp. 42. Available at: https://doi.org/10.1186/s13027-019-0260-7.

TCGA. (2021) Available at: https://www.cancer.gov/tcga.

Theilen, G. H. *et al.* (1971) 'C-type virus in tumor tissue of a woolly monkey (Lagothrix spp.) with fibrosarcoma', *J Natl Cancer Inst,* 47(4), pp. 881-9.

Theodorou, V. *et al.* (2004) 'Fgf10 is an oncogene activated by MMTV insertional mutagenesis in mouse mammary tumors and overexpressed in a subset of human breast carcinomas', *Oncogene,* 23(36), pp. 6047-55. Available at: https://doi.org/10.1038/sj.onc.1207816.

Thomas, J., Perron, H. and Feschotte, C. (2018) 'Variation in proviral content among human genomes mediated by LTR recombination', *Mob DNA,* 9, pp. 36. Available at: https://doi.org/10.1186/s13100-018-0142-3.

Thomas, J. H. and Schneider, S. (2011) 'Coevolution of retroelements and tandem zinc finger genes', *Genome Res,* 21(11), pp. 1800-12. Available at: https://doi.org/10.1101/gr.121749.111.

Tian, J. M. *et al.* (2018) 'Estrogen and progesterone promote breast cancer cell proliferation by inducing cyclin G1 expression', *Braz J Med Biol Res,* 51(3), pp. 1-7. Available at: https://doi.org/10.1590/1414-431X20175612.

Ting, C. H. *et al.* (2019) 'FOSB¯PCDHB13 Axis Disrupts the Microtubule Network in Non-Small Cell Lung Cancer', *Cancers (Basel),* 11(1). Available at: https://doi.org/10.3390/cancers11010107.

Todaro, G. J. *et al.* (1975) 'Infectious primate type C viruses: Three isolates belonging to a new subgroup from the brains of normal gibbons', *Virology,* 67(2), pp. 335-43. Available at: https://doi.org/10.1016/0042-6822(75)90435-3.

Tomlins, S. A. *et al.* (2007) 'Distinct classes of chromosomal rearrangements create oncogenic ETS gene fusions in prostate cancer', *Nature,* 448(7153), pp. 595-9. Available at: https://doi.org/10.1038/nature06024.

Toufaily, C. *et al.* (2011) 'Activation of LTRs from different human endogenous retrovirus (HERV) families by the HTLV-1 tax protein and T-cell activators', *Viruses,* 3(11), pp. 2146-59. Available at: https://doi.org/10.3390/v3112146.

Tristem, M. (2000) 'Identification and characterization of novel human endogenous retrovirus families by phylogenetic screening of the human genome mapping project database', *J Virol,* 74(8), pp. 3715-30. Available at: https://doi.org/10.1128/jvi.74.8.3715-3730.2000.

Turelli, P. *et al.* (2014) 'Interplay of TRIM28 and DNA methylation in controlling human endogenous retroelements', *Genome Res,* 24(8), pp. 1260-70. Available at: https://doi.org/10.1101/gr.172833.114.

Turner, G. *et al.* (2001) 'Insertional polymorphisms of full-length endogenous retroviruses in humans', *Curr Biol,* 11(19), pp. 1531-5. Available at: https://doi.org/10.1016/s0960-9822(01)00455-9.

Urnovitz, H. B. and Murphy, W. H. (1996) 'Human endogenous retroviruses: nature, occurrence, and clinical implications in human disease', *Clin Microbiol Rev,* 9(1), pp. 72-99. Available at: https://doi.org/10.1128/CMR.9.1.72.

Van Deerlin, V. M. *et al.* (2008) 'TARDBP mutations in amyotrophic lateral sclerosis with TDP-43 neuropathology: a genetic and histopathological analysis', *Lancet Neurol,* 7(5), pp. 409-16. Available at: https://doi.org/10.1016/S1474-4422(08)70071-1.

van Horssen, J. *et al.* (2016) 'Human endogenous retrovirus W in brain lesions: Rationale for targeted therapy in multiple sclerosis', *Mult Scler Relat Disord,* 8, pp. 11-8. Available at: https://doi.org/10.1016/j.msard.2016.04.006.

Venkatesh, V. *et al.* (2018) 'Targeting Notch signalling pathway of cancer stem cells', *Stem Cell Investig,* 5, pp. 5. Available at: https://doi.org/10.21037/sci.2018.02.02.

Vogt, M. A. *et al.* (2018) 'TDP-43 induces p53-mediated cell death of cortical progenitors and immature neurons', *Sci Rep,* 8(1), pp. 8097. Available at: https://doi.org/10.1038/s41598-018-26397-2.

Voisset, C. *et al.* (1999) 'Phylogeny of a novel family of human endogenous retrovirus sequences, HERV-W, in humans and other primates', *AIDS Res Hum Retroviruses,* 15(17), pp. 1529-33. Available at: https://doi.org/10.1089/088922299309810.

Volkman, H. E. and Stetson, D. B. (2014) 'The enemy within: endogenous retroelements and autoimmune disease', *Nat Immunol,* 15(5), pp. 415-22. Available at: https://doi.org/10.1038/ni.2872.

Wallace, T. A. *et al.* (2014) 'Elevated HERV-K mRNA expression in PBMC is associated with a prostate cancer diagnosis particularly in older men and smokers', *Carcinogenesis,* 35(9), pp. 2074-83. Available at: https://doi.org/10.1093/carcin/bgu114.

Wang-Johanning, F. *et al.* (2003) 'Quantitation of HERV-K env gene expression and splicing in human breast cancer', *Oncogene,* 22(10), pp. 1528-35. Available at: https://doi.org/10.1038/sj.onc.1206241.

Weiss, G. J. *et al.* (2013) 'A pilot study using next-generation sequencing in advanced cancers: feasibility and challenges', *PLoS One,* 8(10), pp. e76438. Available at: https://doi.org/10.1371/journal.pone.0076438.

Weiss, R. A. (2013) 'On the concept and elucidation of endogenous retroviruses', *Philos Trans R Soc Lond B Biol Sci,* 368(1626), pp. 20120494. Available at: https://doi.org/10.1098/rstb.2012.0494.

Wicker, T. *et al.* (2007) 'A unified classification system for eukaryotic transposable elements', *Nat Rev Genet,* 8(12), pp. 973-82. Available at: https://doi.org/10.1038/nrg2165.

Wildschutte, J. H. *et al.* (2016) 'Discovery of unfixed endogenous retrovirus insertions in diverse human populations', *Proc Natl Acad Sci U S A,* 113(16), pp. E2326-34. Available at: https://doi.org/10.1073/pnas.1602336113.

Wolff, F. *et al.* (2017) 'The double-edged sword of (re)expression of genes by hypomethylating agents: from viral mimicry to exploitation as priming agents for targeted immune checkpoint modulation', *Cell Commun Signal,* 15(1), pp. 13. Available at: https://doi.org/10.1186/s12964-017-0168-z.

Wyllie, A. H. (2002) 'E2F1 selects tumour cells for both life and death', *J Pathol,* 198(2), pp. 139-41. Available at: https://doi.org/10.1002/path.1238.

Wynyard, S. *et al.* (2014) 'Microbiological safety of the first clinical pig islet xenotransplantation trial in New Zealand', *Xenotransplantation,* 21(4), pp. 309-23. Available at: https://doi.org/10.1111/xen.12102.

Xu, W. *et al.* (2013) 'An exogenous retrovirus isolated from koalas with malignant neoplasias in a US zoo', *Proc Natl Acad Sci U S A,* 110(28), pp. 11547-52. Available at: https://doi.org/10.1073/pnas.1304704110.

Xue, B. *et al.* (2020) 'Identification of the distribution of human endogenous retroviruses K (HML-2) by PCR-based target enrichment sequencing', *Retrovirology,* 17(1), pp. 10. Available at: https://doi.org/10.1186/s12977-020-00519-z.

Yap, M. W. *et al.* (2004) 'Trim5alpha protein restricts both HIV-1 and murine leukemia virus', *Proc Natl Acad Sci U S A,* 101(29), pp. 10786-91. Available at: https://doi.org/10.1073/pnas.0402876101.

Yi, J. M., Kim, H. M. and Kim, H. S. (2004) 'Expression of the human endogenous retrovirus HERV-W family in various human tissues and cancer cells', *J Gen Virol,* 85(Pt 5), pp. 1203-1210. Available at: https://doi.org/10.1099/vir.0.79791-0.

York, D. F. *et al.* (1991) 'Isolation, identification, and partial cDNA cloning of genomic RNA of jaagsiekte retrovirus, the etiological agent of sheep pulmonary adenomatosis', *J Virol,* 65(9), pp. 5061-7. Available at: https://doi.org/10.1128/JVI.65.9.5061-5067.1991.

Young, G. R., Stoye, J. P. and Kassiotis, G. (2013) 'Are human endogenous retroviruses pathogenic? An approach to testing the hypothesis', *Bioessays,* 35(9), pp. 794-803. Available at: https://doi.org/10.1002/bies.201300049.

Young, G. R. *et al.* (2018) 'HIV-1 Infection of Primary CD4', *J Virol,* 92(1). Available at: https://doi.org/10.1128/JVI.01507-17.

Zahn, J. *et al.* 2015. Expansion of a novel endogenous retrovirus

throughout the pericentromeres of modern humans. *Genome biology*.

Zare, M. *et al.* (2017) 'Human endogenous retrovirus env genes: Potential blood biomarkers in lung cancer', *Microb Pathog,* 115, pp. 189-193. Available at: https://doi.org/10.1016/j.micpath.2017.12.040.

Zhan, T., Rindtorff, N. and Boutros, M. (2017) 'Wnt signaling in cancer', *Oncogene,* 36(11), pp. 1461-1473. Available at: https://doi.org/10.1038/onc.2016.304.

Zhang, W. *et al.* (2015) 'Structural basis of arc binding to synaptic proteins: implications for cognitive disease', *Neuron,* 86(2), pp. 490-500. Available at: https://doi.org/10.1016/j.neuron.2015.03.030.

Zhao, J. *et al.* (2011) 'Expression of Human Endogenous Retrovirus Type K Envelope Protein is a Novel Candidate Prognostic Marker for Human Breast Cancer', *Genes Cancer,* 2(9), pp. 914-22. Available at: https://doi.org/10.1177/1947601911431841.

Zhao, Q. *et al.* (2020) 'The Function of SUMOylation and Its Role in the Development of Cancer Cells under Stress Conditions: A Systematic Review', *Stem Cells Int,* 2020, pp. 8835714. Available at: https://doi.org/10.1155/2020/8835714.

Zheng, H. *et al.* (2020) 'Koala retrovirus diversity, transmissibility, and disease associations', *Retrovirology,* 17(1), pp. 34. Available at: https://doi.org/10.1186/s12977-020-00541-1.

Zhou, F. *et al.* (2016) 'Activation of HERV-K Env protein is essential for tumorigenesis and metastasis of breast cancer cells', *Oncotarget,* 7(51), pp. 84093-84117. Available at: https://doi.org/10.18632/oncotarget.11455.

Ávila-Arcos, M. C. *et al.* (2013) 'One hundred twenty years of koala retrovirus evolution determined from museum skins', *Mol Biol Evol,* 30(2), pp. 299-304. Available at: https://doi.org/10.1093/molbev/mss223.

6. Appendix 1

Scripts used in the analysis.

1.i. extractfromref_optimal.py

```
# This script extracts ranges from reference genome files and prints the sequences on screen. Instructions for usage below.

# Dictionary for sequences in reverse order.
reverseatcg={"A":"T",
"T":"A",
"C":"G",
"G":"C",
"N":"N",
'g':'c',
'c':'g',
'a':'t',
't':'a',
'n':'n'
}
# Dictionary for chromosome names in the official files.
# There are differences in the names of the chromosomes between build 37 and 38 of the human genome. Tbe following lines of code standardise the chromosome name to just the numerical identifier of the chromosome (i.e. gi|224589800|ref|NC_000001.10| (build 37) and chr1 (build 38) are now both recognised by "1")
chrname={
"gi|224589800|ref|NC_000001.10|":"1",
"gi|224589811|ref|NC_000002.11|":"2",
"gi|224589815|ref|NC_000003.11|":"3",
"gi|224589816|ref|NC_000004.11|":"4",
"gi|224589817|ref|NC_000005.9|":"5",
"gi|224589818|ref|NC_000006.11|":"6",
"gi|224589819|ref|NC_000007.13|":"7",
"gi|224589820|ref|NC_000008.10|":"8",
"gi|224589821|ref|NC_000009.11|":"9",
"gi|224589801|ref|NC_000010.10|":"10",
"gi|224589802|ref|NC_000011.9|":"11",
"gi|224589803|ref|NC_000012.11|":"12",
"gi|224589804|ref|NC_000013.10|":"13",
"gi|224589805|ref|NC_000014.8|":"14",
"gi|224589806|ref|NC_000015.9|":"15",
"gi|224589807|ref|NC_000016.9|":"16",
"gi|224589808|ref|NC_000017.10|":"17",
"gi|224589809|ref|NC_000018.9|":"18",
"gi|224589810|ref|NC_000019.9|":"19",
"gi|224589812|ref|NC_000020.10|":"20",
"gi|224589813|ref|NC_000021.8|":"21",
"gi|224589814|ref|NC_000022.10|":"22",
"gi|224589822|ref|NC_000023.10|":"X",
"gi|224589823|ref|NC_000024.9|":"Y",
"chr1":"1",
"chr2":"2",
"chr3":"3",
"chr4":"4",
"chr5":"5",
"chr6":"6",
"chr7":"7",
"chr8":"8",
"chr9":"9",
"chr10":"10",
"chr11":"11",
"chr12":"12",
"chr13":"13",
```

```
"chr14":"14",
"chr15":"15",
"chr16":"16",
"chr17":"17",
"chr18":"18",
"chr19":"19",
"chr20":"20",
"chr21":"21",
"chr22":"22",
"chrX":"X",
"chrY":"Y",
"1":"1",
"2":"2",
"3":"3",
"4":"4",
"5":"5",
"6":"6",
"7":"7",
"8":"8",
"9":"9",
"10":"10",
"11":"11",
"12":"12",
"13":"13",
"14":"14",
"15":"15",
"16":"16",
"17":"17",
"18":"18",
"19":"19",
"20":"20",
"21":"21",
"22":"22",
"X":"X",
"Y":"Y",
"chr1_KI270765v1_alt":"chr1_KI270765v1_alt",
"chr2_KI270768v1_alt":"chr2_KI270768v1_alt",
"chr3_KI270779v1_alt":"chr3_KI270779v1_alt",
"chr3_KI270780v1_alt":"chr3_KI270780v1_alt",
"chr3_KI270895v1_alt":"chr3_KI270895v1_alt",
"chr3_KI270924v1_alt":"chr3_KI270924v1_alt",
"chr3_KI270934v1_alt":"chr3_KI270934v1_alt",
"chr3_KI270935v1_alt":"chr3_KI270935v1_alt",
"chr3_KI270936v1_alt":"chr3_KI270936v1_alt",
"chr3_KI270937v1_alt":"chr3_KI270937v1_alt",
"chrUn_KI270749v1":"chrUn_KI270749v1",
"chrUn_gl000219":"chrUn_gl000219",
"chrUn_gl000222":"chrUn_gl000222",
"chrUn_gl000223":"chrUn_gl000223",
"chrUn_gl000231":"chrUn_gl000231",
"chrUn_gl000212":"chrUn_gl000212",
"chrUn_gl000219":"chrUn_gl000219",
"chrUn_gl000232":"chrUn_gl000232",
"chr17_ctg5_hap1":"chr17_ctg5_hap1",
"chrUn_GL000219v1":"chrUn_GL000219v1"
}



# List for extracted locations.
# The script requires a list of the locations of the insertions, as a text file in
    the format:
# [cytogenetic identifier] [chromosome(numeric)] [orientation] [5' position in the
    genome] [3' position in the genome] <- these fields are tab seperated
```

```
# The output format combines the above fields to create a new name for each extracted sequence, followed ny the DNA sequence extracted ***
locs=[]

# The following section opens TSV file with viral location to extract and loads it into memory.
with open('toextrac.left.txt','r') as f:
    for line in f:
        locs.append(tuple(line.rstrip("\n").split("\t")))
f.close()
locs=tuple(locs)
# Create variables for sequences and headers.
genome=[]
header=''
sequence=''


# This section opens fasta reference genome sequence and loads individual chromosomes into memory, also scanning the list of known HERV-K insertions and extracting sequences of insertions from each analysed chromosome.
with open('hg38.fa','r') as g:
    for line in g:
# If line starts with header sign, get the chromosome name.
        if line.startswith(">"):
            if header=='':
                analysed=[]
                header=chrname[line.rstrip("\n").lstrip(">")]
# Pick insertions that belong to the analysed chromosome
                for i in locs:

                    if header in set(i) or ("chr"+str(header)) in set(i):
                        analysed.append(i)

            else:
                analysed=tuple(analysed)
# Extract insertions from chromosome loaded to the memory before.
                for i in analysed:
                    start=int(i[3])
                    end=int(i[4])
                    print(">"+i[0]+"_chr"+chrname[i[1]]+":"+str(i[3])+"-"+str(i[4]))
# Print sequences on screen, change orientation if necessary.
                    if i[2]=="+":
                        print(sequence[start-1:end])
                    else:
                        seq=sequence[start-1:end][::-1]
                        seq1=''
                        for char in seq:
                            seq1+=reverseatcg[char]
                        print(seq1)
                analysed=[]
                sequence=''
# If line starts with header sign, get the chromosome name. Pass on chromosomes that you don't know according to dictionary at the beginning.
                try:
                    header=chrname[line.rstrip("\n").lstrip(">")]
                except KeyError:
                    continue
# Pick insertions that belong to the analysed chromosome.
                for i in locs:
                    if header in set(i) or ("chr"+str(header)) in set(i):
                        analysed.append(i)
# If line doesn't have a header sign, assemble it into sequence.
        else:
            sequence+=line.rstrip("\n")
```

```
176
177
178 g.close()
179 analysed=tuple(analysed)
180 # Extract insertions from chromosome loaded to the memory before.
181 for i in analysed:
182     print(i)
183     start=int(i[3])
184     end=int(i[4])
185     print(">"+i[0]+"_chr"+chrname[i[1]]+":"+str(i[3])+"-"+str(i[4]))
186 # Print sequences on screen, change orientation if necessary
187     if i[2]=="+":
188         print(sequence[start-1:end])
189     else:
190         seq=sequence[start-1:end][::-1]
191         seq1=''
192         for char in seq:
193             seq1+=reverseatcg[char]
194         print(seq1)
```

1.ii. tsd_wooble.py

```
1  # This script scans all the insertions and looks for single basepair differences
       between flanking TSDs. If it finds one, it introduces it into the table for
       future use. The single basepair difference between flanking TSDs is allowed due
       to random mutation rate and the fact that the polymerase that introduces TSDs has
        no repair mechanism which might make the mutations more prone to occur in TSD
       zone.
2
3  # Open file with TSD result.
4  with open('solotsdv31.csv','r') as f:
5      for line in f:
6          line1=line.rstrip("\n")
7          line=line.rstrip("\n").rstrip(",").split(",")
8  # If there is no matching TSD or insertion is truncated print the line.
9          if line[-1]=="-" or "truncated" in line[-1]:
10             print(line1)
11         else:
12 # If there is just a matching TSD print the line.
13             if (line[-1] in line[-3][-len(line[-1]):]) and (line[-1] in line[-2][:
       len(line[-1])]):
14                 print(line1)
15             else:
16 # If the TSD matches with a mutation, print both variants.
17                 if line[-1] in line[-3][-len(line[-1]):]:
18                     print(",".join(line)+"/"+line[-2][:len(line[-1])])
19                 else:
20                     print(",".join(line)+"/"+line[-3][-len(line[-1]):])
21
22 f.close()
```

1.iii. tsd.py

```
1  # This script analyses TSD data of all the insertions and outputs the numbers of
       insertions involved in duplication and recombination events. It compares TSDs and
        flanking sequences of insertions to create lists of inserts that have the same
       TSD and tries to categorize if the TSDs come from recombinations or segmental
       duplications. It requires output of the 'aligner.py' script to get the
       duplication data.
2
3
4  # Lists for insertions that are not truncated but have different flanks, 5'
       truncated ones, 3' truncated ones and normal inserts with proper TSDs.
5  recombinators=[]
6  five=[]
7  three=[]
```

```
normal=[]
fiventhree=[]
fiven=0
threen=0
fntn=0
notsdn=0
tsdn=0

# Added after the report - in results table there were inserts with non-matching TSDs (8) and 3' (8) or 5' (6) truncated that i reported as no activity. How can this be? The function below takes the 8bp flank in each of these cases, and systematically introduces 1 mutation at each position and compares this with the established list of TSDs (provided in table MikeDBv43.grch.csv, big table from report - (back in the report had mixed up locations - MikeDBv41 - in google drive )).
def singlemut(seq):
    if type(seq)==str:
        nucl=tuple(["A","T","C","G"])
        seq=seq.upper()
        result=[]
        count=0
        for i in seq:
            for j in nucl:
                seq2=list(seq)
                seq2[count]=j
                result.append("".join(seq2))
            count+=1
        result=tuple(set(result))
    elif type(seq)==list or type(seq)==tuple:
            nucl=tuple(["A","T","C","G"])
            result=[]
            for z in seq:
                count=0
                z=z.upper()
                for i in z:
                    for j in nucl:
                        seq2=list(z)
                        seq2[count]=j
                        result.append("".join(seq2))
                    count+=1
            result=tuple(set(result))
    else:
        result="Unknown input type, please provide a flank as a string or a list of potential tsds"
    return result


# Open data source with insertion information, along with TSDs, in format : name, presence in HG19 genome, presence in HG38 geneome, chromosome, orientation, hg19 location, length, famili, age, age ,range, comment, hg38 location, 5' end, 3' end, 5' flank, 3' flank, TSD.
with open('tsd_wooble_final.csv','r') as f:
    for line in f:
        lines=line.rstrip("\n").lstrip("\ufeff").split(",")

# Get insertion name
        ident=tuple([lines[0],line.rstrip('\n')])
# Get left and right flanks
        ltsd0=tuple([lines[16][-4:],lines[16][-5:],lines[16][-6:],lines[16][-7:],lines[16][-8:]])
        rtsd0=tuple([lines[17][:4],lines[17][:5],lines[17][:6],lines[17][:7],lines[17][:8]])
        ltsd=singlemut(tuple([lines[16][-4:],lines[16][-5:],lines[16][-6:],lines[16][-7:],lines[16][-8:]]))
```

```
        rtsd=singlemut(tuple([lines[17][:4],lines[17][:5],lines[17][:6],lines
    [17][:7],lines[17][:8]]))
# If no proper TSD, add to recombinators.
        if lines[18]=="-":
            recombinators.append(tuple([ident,ltsd0,rtsd0,ltsd,rtsd]))
            notsdn+=1
# If 5' in TSD section, add to 5' truncated
        elif "5'" in lines[18] and "3'" not in lines[18]:
            five.append(tuple([ident,"-",rtsd0,"-",rtsd]))
            fiven+=1
# If 3' in TSD section, add to 3' truncated
        elif "3'" in lines[18] and "5'" not in lines[18]:
            threen+=1
            three.append(tuple([ident,ltsd0,"-",ltsd,"-"]))
# If both 5' and 3' in TSD section, add to normal ones
        elif "5'" not in lines[18] and "3'" not in lines[18] and lines[18]!="-":
            tsdn+=1
            normal.append(tuple([ident,lines[18]]))
        elif "5'" in lines[18] and "3'" in lines[18]:
            fntn+=1
            fiventhree.append(tuple([ident,"-"]))
normaltsds=[]


# Function that outputs intersection of two lists.
def intersection(lst1, lst2):
    return list(set(lst1) & set(lst2))
# Load real duplications from a separate file.
realduplication=[]
dupins=[]
with open('duplications.all.good.txt','r') as f:
    for line in f:
        realduplication.append(tuple(line.rstrip("\n").split(",")))
f.close()
realduplication=tuple(realduplication)
for i in realduplication:
    for j in i:
        dupins.append(j)
dupins=set(dupins)
dupfive=0
dupthree=0
dupfnt=0
duptsd=0
dupnotsd=0
# Count insertions in different categories.
#print(dupins)
for i in five:
    if i[0][0] in dupins:
        dupfive+=1
#        five.remove(i)
for i in tuple(dupins):
    for j in five:
        if j[0][0]==i:
            five.remove(j)
            break

for i in three:
    if i[0][0] in dupins:
        dupthree+=1
#        three.remove(i)
for i in tuple(dupins):
    for j in three:
        if j[0][0]==i:
            three.remove(j)
```

```
            break

for i in normal:
    if i[0][0] in dupins:
        duptsd+=1
#         normal.remove(i)
for i in tuple(dupins):
    for j in normal:
        if j[0][0]==i:
            normal.remove(j)
            break


for i in recombinators:
    if i[0][0] in dupins:
        dupnotsd+=1
#         recombinators.remove(i)
for i in tuple(dupins):
    for j in recombinators:
        if j[0][0]==i:
            recombinators.remove(j)
            break
for i in fiventhree:
    if i[0][0] in dupins:
        dupfnt+=1

for i in tuple(dupins):
    for j in fiventhree:
        if j[0][0]==i:
            fiventhree.remove(j)
            break
''''
for i in fiventhree:
    if i[0][0] in dupins:
        dupfnt+=1

#         fiventhree.remove(i)
'''

# Analyse all insertions with good TSDs recognisable on both ends. The script
    compares the TSDs against each other and groups them according to TSD recognised.
     It also creates a list of all TSDs from the good insertions to further compare
    the truncated ones with them and try to predict if any of the truncated ones
    might have been created by recombination/duplication of a good one. If a mutation
     occurs, both of the TSD variants are taken into account.

# Add all TSDs from normal inserts to a list, create a set.
#for i in normal:
#    print(i)
for i in normal:
    normaltsds.append(i[1])
normaltsds=list(set(normaltsds))
# Comparing normal to normal inserts (Proper TSDs)

# Put every TSD into a list.
for i in normaltsds:
    normaltsds[normaltsds.index(i)]=[normaltsds[normaltsds.index(i)]]
#print()
# Scan every normal insertion and put it into groups that have their TSDs.
for i in range(0,len(normaltsds)):
    for j in normal:
        if j[-1]==normaltsds[i][0]:
            normaltsds[i].append(j)
count=0
```

```
181
182 normaltonormal=[]
183
184
185 # Count the number of insertion involved in recombination.
186
187 for i in normaltsds:
188     if len(i)>2:
189         normaltonormal.append([])
190         j=0
191
192         for j in i[1:]:
193
194             normaltonormal[-1].append(j[0][1].split(",")[0]+","+j[0][1].split(",")
    [3]+","+j[0][1].split(",")[5]+","+j[0][1].split(",")[6]+","+j[0][1].split(",")
    [7]+","+j[0][1].split(",")[4]+","+j[0][1].split(",")[3]+","+j[0][1].split(",")
    [18])
195
196
197
198 # Split results if mutation happened to account both TSD variants.
199 normaltsds=[]
200 for i in normal:
201     if "/" in i[-1]:
202         normaltsds.append(i[-1].split("/")[0])
203         normaltsds.append(i[-1].split("/")[1])
204     else:
205         normaltsds.append(i[-1])
206 print("good to good")
207 for i in normaltonormal:
208     print(i[0]+","+i[1])
209     if len(i)>2:
210         for j in i[2:]:
211             print(i[0]+","+j)
212
213
214
215 print("--------------------")
216
217 # This section analyses all recombined insertions - ones that are not truncated but
    the TSDs don't match. The recognised flanking sequences are extended from 4-8bp
    and compared against each other, inserts with recognisable TSDs and 3' as well as
     5' truncated insertions.
218
219 #print("Recombined")
220 retsds=[]
221 # Append all TSDs of recombined insertions into a list.
222 for i in recombinators:
223     retsds.append((i[-4],i[-3],i[-2],i[-1]))
224 # Create variables for 5' to 3' and 3' to 5' recombination, 5' to 5' or 3' to 3' or
    (5' to 5' AND 3' to 3')
225 proper=0
226 half=0
227 segrecomdup=0
228 alreadyfound=[]
229 recomtorecomres=[]
230 halfhalf=[]
231 # Compare recombined insertion to other recombined insertions.
232
233 # Checking truncated insertions against each other for TSDs. Using flanks extending
    from 4 to 8bp and cross checking 5' to 3' and 3' to 5' as well as 5' to 5' and 3'
     to 3' ends.
234
235 # Go backwards on TSDs
```

```
for i in range(len(retsds) - 1, -1, -1):
    # Get a list of inserts minus the checked ones
    temp=retsds[:i]
    # Remove the ones that already recombined
    if len(alreadyfound)>0:
        for e in alreadyfound:
            if e in temp:
                temp.remove(e)
    # Go through rest inserts.
    for j in temp:
        # Check both extending flanks from 4-8bp for overlaps. Put the ones you find in a list.
        if any(elem in j[0]  for elem in retsds[i][3]) and any(elem in j[1]  for elem in retsds[i][2]):
            proper+=1
            alreadyfound.append(j)
            break
        # If just one extending flank appears in other insertion, put it in half recombined list.
        if any(elem in j[0]  for elem in retsds[i][3]) or any(elem in j[1]  for elem in retsds[i][2]):
            half+=1
            alreadyfound.append(j)
            halfhalf.append(j)
            break

# Check if 4-8bp from both flanks appear in other insertion in the same fashion.
for i in range(len(retsds) - 1, -1, -1):
    temp=retsds[:i]
    for j in temp:

        if any(elem in j[0]  for elem in retsds[i][2]) and any(elem in j[1]  for elem in retsds[i][3]):
            segrecomdup+=1
            break
recombpairs=[]

# Go through all the recombinating insertions again again. Create paris of insertions recombining 5' to 3' end and 3' to 5' end to find properly recombined insertions.
for i in range(len(recombinators)-1,-1,-1):
# Get a list of inserts minus the checked ones
    temp=recombinators[:i]
# Remove ones that have been found in previous iterations.
    if len(recombpairs)>0:
        for e in recombpairs:
            if e in temp:
                temp.remove(e)
# Scan inserts for 5' to 5' and 3' to 3' matches.
    for j in temp:

        if any(elem in j[4]  for elem in recombinators[i][2]):
            recombpairs.append([j[0][1],recombinators[i][0][1],max(intersection(j[4],recombinators[i][2]), key=len)])
        if any(elem in j[3]  for elem in recombinators[i][1]):
            recombpairs.append([j[0][1],recombinators[i][0][1],max(intersection(j[3],recombinators[i][1]), key=len)])
#         if any(elem in j[3]  for elem in recombinators[i][2]):
#             recombpairs.append([j[0][1],recombinators[i][0][1],max(intersection(j[3],recombinators[i][2]), key=len)])
#         if any(elem in j[4]  for elem in recombinators[i][1]):
#             recombpairs.append([j[0][1],recombinators[i][0][1],max(intersection(j[4],recombinators[i][1]), key=len)])
recombpairsres=[]
```

```
# Add results to list.

for i in recombpairs:
    recombpairsres.append(i[0].split(",")[0])
    recombpairsres.append(i[1].split(",")[0])

print("recom to recom")
for i in recombpairs:
    print(i[0].split(",")[0]+","+i[0].split(",")[3]+","+i[0].split(",")[5]+","+i[0].split(",")[6]+","+i[0].split(",")[7]+","+i[0].split(",")[4]+","+i[-1]+","+i[1].split(",")[0]+","+i[1].split(",")[3]+","+i[1].split(",")[5]+","+i[0].split(",")[6]+","+i[1].split(",")[7]+","+i[1].split(",")[4]+","+i[2]+",")
# The flanks of recombined insertions are compared to the list of TSDs recognised in the proper insertions to find if any of them have inserted due to duplication of another, existing insertion prior to recombination, or may be a result of a different event involving a known, proper insertion. The process involves scanning for a single mutation if it was observed to occur and outputting the longest match from the selected ones, since the 4-8bp window is compared from the recombined side.
#import sys
#sys.exit("breakpoint")

recomfull=[]
recomtofl=[]
for i in range(len(retsds) - 1, -1, -1):
    longest=[]
    for j in normal:
# If there is mutation, compare each of the variants to the recombined TSDs.
        if "/" in j[-1]:
            if any(elem==j[-1].split("/")[0]  for elem in retsds[i][2]):# or any(elem==j[-1].split("/")[0]  for elem in retsds[i][3]):
                longest.append(j[-1].split("/")[0])
            if any(elem==j[-1].split("/")[1]  for elem in retsds[i][3]):# or any(elem==j[-1].split("/")[1]  for elem in retsds[i][2]) :
                longest.append(j[-1].split("/")[1])
# If there's no wooble, just do the comparison to the TSD.
        else:
            if any(elem==j[-1]  for elem in retsds[i][2]) or any(elem==j[-1]  for elem in retsds[i][3]):
                longest.append(j[-1])
# Get the longest match i.e. found ATCT in HERV1 vs HERV2, but found ATCTGT in HERV1 vs HERV3, so the pair HERV1-HERV3 is most likely correct.

    if len(longest)!=0:
        recomfull.append(max(longest, key=len))
        for nor in normal:
# Compare both TSDs if you see wooble. Assemble them in pairs. Put longest TSDs you find for a possible pair.
            if "/" in nor[-1]:
                if nor[-1].split("/")[0]==max(longest, key=len) or nor[-1].split("/")[1]==max(longest, key=len):
                    pair=nor[0][1].split(",")[0]+","+nor[0][1].split(",")[3]+","+nor[0][1].split(",")[5]+","+nor[0][1].split(",")[6]+","+nor[0][1].split(",")[4]+","+nor[0][1].split(",")[-1]
            else:
# Do the same for single, good TSDs. Assemble found inserts in paris.
                if nor[-1]==max(longest, key=len):

                    pair=nor[0][1].split(",")[0]+","+nor[0][1].split(",")[3]+","+nor[0][1].split(",")[5]+","+nor[0][1].split(",")[6]+","+nor[0][1].split(",")[4]+","+nor[0][1].split(",")[-1]
# Get the names of the insertions for each found TSDs to create complete pairs with all data on recombining inserts
        for z in recombinators:
```

```
            if z[-2]==retsds[i][-2] and z[-1]==retsds[i][-1]:
                recomtofl.append([z[0][1].split(",")[0]+","+z[0][1].split(",")[3]+","+z[0][1].split(",")[5]+","+z[0][1].split(",")[6]+","+z[0][1].split(",")[4]+","+max(longest, key=len),pair])

# 5'

# The 5' truncated insertions are compared to all of the other insertions in a similar fashion as the recombined inserts before. The 4-8bp possible matches following the 3' end of the insertion are taken into account, as well as the possibility of mutation. 5' inserts are checked against each ofther for duplications, 3' truncated inserts for recombinations as well as agains insertions with proper TSDs and ones with recognisable ends but non-matching TSDs.

# Create some lists for results.
fivefullres=[]
fivefull=[]

# Go through 5' truncated inserts, backwards.
for i in range(len(five) - 1, -1, -1):
    longest=[]
# Compare them with all known good TSDs.
    for j in normal:
# Use variants if mutation happens.
        if "/" in j[-1]:
            if any(elem==j[-1].split("/")[1]  for elem in five[i][-1]):
                longest.append(j[-1].split("/")[1])
        else:
            if any(elem==j[-1]  for elem in five[i][-1]):
                longest.append(j[-1])
# Choose the longest overlapping TSD combo.
    if len(longest)!=0:
        fivefull.append(max(longest, key=len))
# Get name of the insertion along with other data and put it into a pair. Account wooble.
        for zg in normal:
            if "/" in zg[-1] and (zg[-1].split("/")[1]==max(longest, key=len)):
                fivefullres.append([five[i][0][1].split(",")[0]+","+five[i][0][1].split(",")[3]+","+five[i][0][1].split(",")[5]+","+five[i][0][1].split(",")[6]+","+five[i][0][1].split(",")[4]+","+max(longest, key=len),zg[0][1].split(",")[0]+","+zg[0][1].split(",")[3]+","+zg[0][1].split(",")[5]+","+zg[0][1].split(",")[6]+","+zg[0][1].split(",")[4]+","+max(longest, key=len)])
# Get name of the insertion along with other data and put it into a pair.
            else:
                if zg[-1]==max(longest, key=len):
                    fivefullres.append([five[i][0][1].split(",")[0]+","+five[i][0][1].split(",")[3]+","+five[i][0][1].split(",")[5]+","+five[i][0][1].split(",")[6]+","+five[i][0][1].split(",")[4]+","+max(longest, key=len),zg[0][1].split(",")[0]+","+zg[0][1].split(",")[3]+","+zg[0][1].split(",")[5]+","+zg[0][1].split(",")[6]+","+zg[0][1].split(",")[4]+","+max(longest, key=len)])



# Compare 5' truncated insertions to 3' truncated insertions. 4-8bp possible TSDs are taken from each of the insertion and longest overlapping TSDs are selected.
fivethreeres=[]
fivethree=[]

# 5' insertions are compared to the recombined insertions (insertions with good ends but non-matching TSDs) below. Both of the ends of recombined insertion are compared against 3' end of the insert to find 4-8bp matches.

fiverecom=[]
```

```
377
378 # Go through all 5' trucated insertions, backwards.
379 for i in range(len(five) - 1, -1, -1):
380     longest=[]
381     counter=0
382 # Go through all recombined insertion TSDs.
383     for j in retsds:
384 # Find longest matches on both ends
385         if any(elem in j[-1]  for elem in five[i][-3]):
386             longest+=intersection(j[-1], five[i][-3])
387
388         counter+=1
389
390 # Choose the lontest match.
391     if len(longest)!=0:
392         for re in recombinators:
393             if max(longest, key=len) in re[-1]:
394                 break
395 # Add the results to list with all the details for each pair of insertion.
396         fiverecom.append([five[i][0][1].split(",")[0]+","+five[i][0][1].split(",")
    [3]+","+five[i][0][1].split(",")[5]+","+five[i][0][1].split(",")[6]+","+five[i
    ][0][1].split(",")[4]+","+max(longest, key=len),re[0][1].split(",")[0]+","+re
    [0][1].split(",")[3]+","+re[0][1].split(",")[12]+","+re[0][1].split(",")[13]+","+
    re[0][1].split(",")[4]+","+max(longest, key=len)])
397
398
399 # The 3' truncated insertions are compared to all of the other insertions in a
    similar fashion as the recombined inserts before. The 4-8bp possible matches
    following the 5' end of the insertion are taken into account, as well as the
    possibility of mutation. 3' inserts are checked against each ofther for
    duplications, 5' truncated inserts for recombinations as well as agains
    insertions with proper TSDs and ones with recognisable ends but non-matching TSDs
    .
400
401 threefullres=[]
402 threefull=[]
403 # Go through 3' truncated insertions, backwards.
404 for i in range(len(three) - 1, -1, -1):
405     longest=[]
406 # Go through normal ones.
407     for j in normal:
408 # Get both TSD variants if wooble happens and compare with 3' truncated flank 4-8bp.
409         if "/" in j[-1]:
410             if any(elem==j[-1].split("/")[0]  for elem in three[i][-4]):
411
412                 longest.append(j[-1].split("/")[0])
413 # In case of no wooble, use the TSD and compare with 3' truncated flank 4-8bp.
414         else:
415             if any(elem==j[-1]  for elem in three[i][-4]):
416
417                 longest.append(j[-1])
418 # Get longest variant of recombination found.
419     if len(longest)!=0:
420
421         threefull.append(max(longest, key=len))
422 # Find details about each insertions, put pairs to a list.
423         for zg in normal:
424             if zg[-1]==max(longest, key=len):
425                 threefullres.append([three[i][0][1].split(",")[0]+","+three[i
    ][0][1].split(",")[3]+","+three[i][0][1].split(",")[5]+","+three[i][0][1].split("
    ,")[6]+","+three[i][0][1].split(",")[4]+","+max(longest, key=len),zg[0][1].split(
    ",")[0]+","+zg[0][1].split(",")[3]+","+zg[0][1].split(",")[5]+","+zg[0][1].split(
    ",")[6]+","+zg[0][1].split(",")[4]+","+max(longest, key=len)])
426
```

```
# Compare 3' truncated insertions to recombined insertions.
threerecom=[]
for i in range(len(three) - 1, -1, -1):
    longest=[]
    counter=0
# Get intersections of possible TSD combinations and choose the longest pair.
    for j in retsds:

        if any(elem in j[-2]  for elem in three[i][-4]):
            longest+=intersection(j[-2], three[i][-4])

        counter+=1
# Get the longest pair for recombinations. Append pairs to a list.
    if len(longest)!=0:
        for re in recombinators:
            if max(longest, key=len) in re[-1] or max(longest, key=len) in re[-2]:
                break
        threerecom.append([three[i][0][1].split(",")[0]+","+three[i][0][1].split(","
    )[3]+","+three[i][0][1].split(",")[5]+","+three[i][0][1].split(",")[6]+","+three[
    i][0][1].split(",")[4]+","+max(longest, key=len),re[0][1].split(",")[0]+","+re
    [0][1].split(",")[3]+","+re[0][1].split(",")[12]+","+re[0][1].split(",")[13]+","+
    re[0][1].split(",")[4]+","+max(longest, key=len)])

# Compare 3' truncated inserts against each other to find possible duplications:

threetothree=[]
for i in range(len(three) - 1, -1, -1):
    temp=three[:i]
    for j in temp:
# Accept results only if the TSD matches completely - possible duplication
        if any(elem in three[i][-2]  for elem in j[-4]):
            tsdl=[]
            for t in three[i][-2]:
                if t in j[-4]:
                    tsdl.append(t)
# Append results to a list.
            threetothree.append([three[i][0][1].split(",")[0]+","+three[i][0][1].
    split(",")[3]+","+three[i][0][1].split(",")[11]+","+three[i][0][1].split(",")
    [12]+","+three[i][0][1].split(",")[4]+","+max(tsdl,key=len),j[0][1].split(",")
    [0]+","+j[0][1].split(",")[3]+","+j[0][1].split(",")[11]+","+j[0][1].split(",")
    [12]+","+j[0][1].split(",")[4]+","+max(tsdl,key=len)])


# Compare 5' truncated inserts against each other to find possible duplications:

fivetofive=[]
for i in range(len(five) - 1, -1, -1):
    temp=five[:i]
    for j in temp:
# Accept results only if the TSD matches completely - possible duplication
        if any(elem in five[i][-1]  for elem in j[-3]):
            tsdl=[]
            for t in five[i][-1]:
                if t in j[-3]:
                    tsdl.append(t)
# Append results to a list.

            fivetofive.append([five[i][0][1].split(",")[0]+","+five[i][0][1].split("
    ,")[3]+","+five[i][0][1].split(",")[11]+","+five[i][0][1].split(",")[12]+","+five
    [i][0][1].split(",")[4]+","+max(tsdl,key=len),j[0][1].split(",")[0]+","+j[0][1].
    split(",")[3]+","+j[0][1].split(",")[11]+","+j[0][1].split(",")[12]+","+j[0][1].
    split(",")[4]+","+max(tsdl,key=len)])


```

```
479
480 fivenames=[]
481 threenames=[]
482 allnames=[]
483 all5names=[]
484 all3names=[]
485 #for i in normaltonormal:
486 #    allnames.append(i[0].split(",")[0])
487 #    allnames.append(i[1].split(",")[0])
488 '''
489 # Go through all the produced lists. Get the names of each insertion that belongs to
        each category in the results table.
490 print("5' tr to recombined")
491 for i in fiverecom:
492     for j in i:
493         print(j,end=',')
494     print("")
495 print("3' tr to recombined")
496 for i in threerecom:
497     for j in i:
498         print(j,end=',')
499     print("")
500 print("Recombined to recombined")
501 for i in fivethreeres:
502     for j in i:
503         print(j,end=',')
504     print("")
505 print("Recombined to normal")
506 for i in recomtofl:
507     for j in i:
508         print(j,end=',')
509     print("")
510 print("Normal to normal")
511 for i in normaltonormal:
512     for j in i:
513         print(j,end=',')
514     print("")
515 print("Normal to 5'tr")
516 for i in fivefullres:
517     for j in i:
518         print(j,end=',')
519     print("")
520 print("Normal to 3'tr")
521 for i in threefullres:
522     for j in i:
523         print(j,end=',')
524     print("")
525 print("5' Truncated to 3' truncated")
526 for i in fivethreeres:
527     for j in i:
528         print(j,end=',')
529     print("")
530 print("5' Truncated to 5' truncated")
531 for i in fivetofive:
532     for j in i:
533         print(j,end=',')
534     print("")
535 print("3' Truncated to 3' truncated")
536 for i in threetothree:
537     for j in i:
538         print(j,end=',')
539     print("")
540 '''
541
```

```
print("Recombined to FL")
for i in recomtofl:
    recombpairsres.append(i[0].split(",")[0])
    allnames.append(i[1].split(",")[0])
    for j in i[0:-1]:
        print(j,end=',')
    print(i[-1])
print("Recombined to 5'tr")
for i in fiverecom:
    fivenames.append(i[0].split(",")[0])
    recombpairsres.append(i[1].split(",")[0])
    for j in i[0:-1]:
        print(j,end=',')
    print(i[-1])
print("Recombined to 3'tr")
for i in threerecom:
    threenames.append(i[0].split(",")[0])
    recombpairsres.append(i[1].split(",")[0])
    for j in i[0:-1]:
        print(j,end=',')
    print(i[-1])
print("5'tr to 5'tr")
for i in fivetofive:
    fivenames.append(i[0].split(",")[0])
    fivenames.append(i[1].split(",")[0])
    for j in i[0:-1]:
       print(j,end=',')
    print(i[-1])
print("3'tr to 3'tr")
for i in threetothree:
    threenames.append(i[0].split(",")[0])
    threenames.append(i[1].split(",")[0])
    for j in i[0:-1]:
        print(j,end=',')
    print(i[-1])
print("5'tr to 3'tr")
for i in fivethreeres:
    fivenames.append(i[0].split(",")[0])
    threenames.append(i[1].split(",")[0])
    for j in i[0:-1]:
        print(j,end=',')
    print(i[-1])
print("5'tr to good")
for i in fivefullres:
    allnames.append(i[1].split(",")[0])
    fivenames.append(i[0].split(",")[0])
    for j in i[0:-1]:
        print(j,end=',')
    print(i[-1])
print("3'tr to good")
for i in threefullres:
    allnames.append(i[1].split(",")[0])
    threenames.append(i[0].split(",")[0])
    for j in i[0:-1]:
        print(j,end=',')
    print(i[-1])
# Create sets of results to remove insertions that have been recombining multiple
    times etc.
duplicationtsds=[]
with open('duplication_sum.csv','r') as f:
    for line in f:
        line=line.lstrip("\ufeff").rstrip("\n").split(",")
        if line[-2]=="-":
            line[-2]=""
```

```
        if line[-3]=="-":
            line[-3]=""
        if line[-1]=="t" or line[-1]=="m":
            line[-3]="/".join(singlemut(tuple([line[-3][-4:],line[-3][-5:],line[-3][-6:],line[-3][-7:],line[-3][-8:]])))
            line[-2]="/".join(singlemut(tuple([line[-2][:4],line[-2][:5],line[-2][:6],line[-2][:7],line[-2][:8]])))
        duplicationtsds.append(tuple(line))
f.close()
#print(duplicationtsds)
#print(duplicationtsds)

# Good to segdup

duplicationsfound=[]
for j in duplicationtsds:
    duplicationsfound.append([j])
    for i in normal:
        for z in j[1:]:
            z=z.split("/")
            if i[-1] in z:
                duplicationsfound[-1].append(i)
                break
print("dup to good")
for i in duplicationsfound:
    if len(i)>1:
        a=0
        print(i[0][0]+",,,,"+i[0][1]+"/"+i[0][2],end=',')
        for j in i[1:]:
#            if a>0:
#                print(",,,,",end=',')
            allnames.append(j[0][0])
            print(j[0][0]+","+j[0][1].split(",")[3]+","+j[0][1].split(",")[5]+","+j[0][1].split(",")[6]+","+j[0][1].split(",")[16]+","+j[0][1].split(",")[17]+","+j[0][1].split(",")[18])
            a+=1
#            for i in j:
#                print(i[0]+" "+i[1][-1],end=' ')


print("")
#print(duplicationsfound)
# Five to segdup

duplicationsfound=[]
for j in duplicationtsds:
    duplicationsfound.append([j])
    for i in five:
        for z in j[1:]:
            z=z.split("/")
            if any(g in z for g in i[2]):
                duplicationsfound[-1].append(i)
                break

#print(duplicationsfound)
print("dup to 5tr")
for i in duplicationsfound:
    if len(i)>1:
        a=0
        print(i[0][0]+",,,,"+i[0][1]+"/"+i[0][2],end=',')

        for j in i[1:]:
            longest=[]

```

```
            if any(elem in i[0][1].split("/")  for elem in j[2]):
                longest+=intersection(i[0][1].split("/"), j[2])
            if any(elem in i[0][2].split("/") for elem in j[2]):
                longest+=intersection(i[0][2].split("/"), j[2])
#             if a>0:
#                 print(",,,,",end=',')


            fivenames.append(j[0][0])
            print(j[0][0]+","+j[0][1].split(",")[3]+","+j[0][1].split(",")[5]+","+j
    [0][1].split(",")[6]+","+j[0][1].split(",")[16]+","+j[0][1].split(",")[17]+","+j
    [0][1].split(",")[18]+","+max(longest))
            a+=1

#for i in five:
#    print(i)

# three to segdup
duplicationsfound=[]
for j in duplicationtsds:
    duplicationsfound.append([j])
    for i in three:
        for z in j[1:]:
            z=z.split("/")
            if any(g in z for g in i[1]):
                duplicationsfound[-1].append(i)
                break
#print(duplicationsfound)
print("dup to 3tr")
for i in duplicationsfound:
    if len(i)>1:
        a=0
        print(i[0][0]+",,,,"+i[0][1]+"/"+i[0][2],end=',')
        for j in i[1:]:
            longest=[]
            if any(elem in i[0][1].split("/")  for elem in j[1]):
                longest+=intersection(i[0][1].split("/"), j[1])
            if any(elem in i[0][2].split("/") for elem in j[1]):
                longest+=intersection(i[0][2].split("/"), j[1])
#             if a>0:
#                 print(",,,,",end=',')


            threenames.append(j[0][0])
            print(j[0][0]+","+j[0][1].split(",")[3]+","+j[0][1].split(",")[5]+","+j
    [0][1].split(",")[6]+","+j[0][1].split(",")[16]+","+j[0][1].split(",")[17]+","+j
    [0][1].split(",")[18]+","+max(longest))
            a+=1
# recombinators to segdup
#for i in recombinators:
#    print(i)

duplicationsfound=[]
for j in duplicationtsds:
    duplicationsfound.append([j])
    for i in recombinators:
        for z in j[1:]:
            z=z.split("/")
            if any(g in z for g in i[1]) or any(g in z for g in i[2]):
                duplicationsfound[-1].append(i)
                break
#print(duplicationsfound)
print("dup to recom")
for i in duplicationsfound:
```

```
    if len(i)>1:
        a=0
        print(i[0][0]+",,,,"+i[0][1]+"/"+i[0][2],end=',')
        for j in i[1:]:
            longest=[]
            if any(elem in i[0][1].split("/")  for elem in j[1]):
                longest+=intersection(i[0][1].split("/"), j[1])
            if any(elem in i[0][2].split("/") for elem in j[1]):
                longest+=intersection(i[0][2].split("/"), j[1])
            if any(elem in i[0][1].split("/")  for elem in j[2]):
                longest+=intersection(i[0][1].split("/"), j[2])
            if any(elem in i[0][2].split("/") for elem in j[2]):
                longest+=intersection(i[0][2].split("/"), j[2])
#             if a>0:
#                 print(",,,,",end=',')

            recombpairsres.append(j[0][0])
            print(j[0][0]+","+j[0][1].split(",")[3]+","+j[0][1].split(",")[5]+","+j
    [0][1].split(",")[6]+","+j[0][1].split(",")[16]+","+j[0][1].split(",")[17]+","+j
    [0][1].split(",")[18]+","+max(longest))
            a+=1

for i in normaltonormal:
#         print(i)
    for j in i:
        allnames.append(j.split(",")[0])
recombpairsres=list(set(recombpairsres))
#print(recombpairsres)
allnames=list(set(allnames))
fivenames=list(set(fivenames))
threenames=list(set(threenames))




# Print results in form of a table.

print("--------------")



print("1) Matching TSDs")
print("2) Non-matching TSDs")
print("3) 5' truncaded insertions")
print("4) 3' truncaded insertions")
print("5) 5' and 3' truncaded insertions")
print("-------------------------")
print("")
print("")
a1=tsdn-(len(allnames))-duptsd
a2=notsdn-len(recombpairsres)-dupnotsd
a3=fiven-len(fivenames)-dupfive
a4=threen-len(threenames)-dupthree
a5=fntn-dupfnt
b1=duptsd
b2=dupnotsd
b3=dupfive
b4=dupthree
b5=dupfnt
c1=len(allnames)
c2=len(recombpairsres)
c3=len(fivenames)
c4=len(threenames)
```

```
c5=0



#print(fivefullres)
print("")
print("                              |  1  |  2  |  3  |  4  |  5  |")
print("------------------------------|-----|-----|-----|-----|-----|")
print("No evidence of recombination |                               |")
print("External flanks don't match  | "+str(a1)+" |"+str(a2)+" |"+str(a3)+" |"+str(a4)+" |"+str(a5)+" |")
print("                              |                               |")
print("------------------------------|-------------------------------|")
print("External flanks match        |                               |")
print("Segmental duplications       | "+str(b1)+" |"+str(b2)+" |"+str(b3)+" |"+str(b4)+" |"+str(b5)+" |")
print("                              |                               |")
print("------------------------------|-------------------------------|")
print("Insertions sharing TSDs      |                               |")
print("Ext flanks don't match       | "+str(c1)+" |"+str(c2)+" |"+str(c3)+" |"+str(c4)+" |"+str(c5)+" |")
print("Recombinations               |                               |")
print("------------------------------|-------------------------------|")
j=0
#for i in allnames:
#    print(i)

#for i in recombpairsres:
#    print(i)

#for i in fivenames:
#    print(i)

#for i in threenames:
#    print(i)
#for i in recombinators:
#    if i[0]
'''
recombined=[]
with open('recom1.txt','r') as f:
    for line in f:
        recombined.append(line.rstrip("\n"))
f.close()
recombined=set(recombined)
print("both ends might be recombining")


print("B1")
for i in recombinators:
    if i[0][0] not in recombined:
        print(i[0][0])
'''
```

1.iv. splitends.py

```
# This script opens extracted sequences and creates separate files for 500bp of 5' flanking sequence and 3' flanking sequences.

with open('allhervk.fasta','r') as f:
    for line in f:
        if line.startswith(">"):
            name=line
        else:
            line=line.rstrip("\n")
            with open('allreference.5.fasta','a') as res1:
```

```
                res1.write(name)
                res1.write(line[:500]+"\n")
            res1.close()
            with open('allreference.3.fasta','a') as res2:
                res2.write(name)
                res2.write(line[-500:]+"\n")
            res2.close()
f.close()
```

1.v. cleanup.py

```
lines=[]
# This script removes duplicated results from BLAST, including same subject - query pairs (A vs A) and inverted results (A-B vs B-A)

# Open blast results
with open("allreference.5.txt",'r') as f:
    for line in f:
        line=tuple(line.rstrip("\n").split("\t"))
        # Disregard A vs A results.
        if line[0]==line[1]:
            pass
        else:
            if len(lines)==0:
                lines.append(line)
            else:
                found=0
                # Disregard B vs A if A vs B is already there.
                for i in lines:
                    if (i[0]==line[1] and i[1]==line[0]):
                        found=1
                        break

                if found==0:
                    lines.append(line)
f.close()
# Print filtered results.
for i in lines:
    print("\t".join(i))
```

1.vi. group.py

```
# This script groups all of the fragmented results between subject-query pairs and prints the grouped results.

allleft=[]

with open("allreference.5.clean.txt","r") as f:
    for line in f:
        line=line.rstrip("\n").split("\t")
        if len(allleft)==0:
            allleft.append(line[0:2]+line[6:10])
        else:
            found=0
            counter=0
            for i in allleft:
                if line[0]==i[0] and line[1]==i[1]:
                    allleft[counter].append(line[6])
                    allleft[counter].append(line[7])
                    allleft[counter].append(line[8])
                    allleft[counter].append(line[9])
                    found=1
                counter+=1
            if found==0:
                allleft.append(line[0:2]+line[6:10])

```

```
f.close()
for i in allleft:
    print(i[0]+","+i[1],end=',')
    numbers=i[2:]
    a=0
    for j in numbers:
        print(j,end=",")

    print("")
```

1.vii. selector.py

```
# This script selects groups from BLAST matches, that match within the 10bp
    immidiate flank, to find possible segmental duplications with matching TSDs.

# 5' end selection

"""
with open("allreference.5.grouped.txt","r") as f:
    for line in f:
        good="n"
        line=line.rstrip("\n").split(",")
        if line[-1]=="":
            line=line[:-1]
        counter=0

        for j in line[2:]:
            if counter==0 or counter==2:
                counter+=1
            elif counter==1:
                firstnum=int(j)
                counter+=1
            else:
                secondnum=int(j)
                if firstnum>=490 and secondnum>=490:
                    print(",".join(line))
                counter=0
f.close()
"""
# 3' end selection

with open("allreference.3.grouped.txt","r") as f:
    for line in f:
        good="n"
        line=line.rstrip("\n").split(",")
        if line[-1]=="":
            line=line[:-1]
        counter=0
# extracts the matches and looks for the range
        for j in line[2:]:
            if counter==0:
                firstnum=int(j)
                counter+=1
            elif counter==2:
                secondnum=int(j)
# if the range encompasses first 10bp, select it
                if firstnum<=10 and secondnum<=10:
                    print(",".join(line))
                counter+=1
            elif counter==1:
                counter+=1
            else:
                counter=0


```

```
f.close()
```

1.viii. comparetsd_segdupv2.py

```

lines=[]
tsds=[]
# This script loads TSD infromation from the table and grouped results to produce possible pairs of duplications that share the same TSD and produce positive BLAST results.

# Load TSD status list
with open("tsd_wooble_final.csv",'r') as f:
    for line in f:
        line=tuple(line.split(","))
        if "TRUNCATED" in line[18].rstrip("\n") and "3" in line[18].rstrip("\n"):
            tsds.append(tuple([line[0],line[16].rstrip("\n")]))
        elif "TRUNCATED" in line[18].rstrip("\n") and "5" in line[18].rstrip("\n"):
            tsds.append(tuple([line[0],line[17].rstrip("\n")]))
        elif "-" in line[18].rstrip("\n"):
            tsds.append(tuple([line[0],(line[16].rstrip("\n")+"/"+line[17].rstrip("\n"))]))
        else:
            tsds.append(tuple([line[0],line[18].rstrip("\n")]))
f.close()
# Open 3' ends selected for duplications
with open("allreference.3.selected.txt",'r') as f:
    for line in f:
        line=line.lstrip("\ufeff")
        tsd1=""
        tsd2=""
        liner=line.rstrip("\n").split(",")
        liner[0]=liner[0].split("_")[0]
        liner[1]=liner[1].split("_")[0]
        liner=tuple(liner)
# Search for TSDs in the status list
        for i in tsds:
            if i[0]==liner[0]:
                tsd1=i[1].rstrip("\n")
            if i[0]==liner[1]:
                tsd2=i[1].rstrip("\n")


        if ("/" in tsd1 and "/" not in tsd2 and (tsd2==tsd1.split("/")[0] or tsd2==tsd1.split("/")[1])) or ("/" in tsd2 and "/" not in tsd1 and (tsd1==tsd2.split("/")[0] or tsd1==tsd2.split("/")[1])):
            print(tsd1+","+tsd2+","+line,end='')
        else:
            if tsd1==tsd2:
                print(tsd1+","+tsd2+","+line,end='')
            else:
#### FOR 3' ENDS
## compare BLAST match, select if a pair on 3' end matches at worst from 2nd basepair of TSD/flank
                lines=line.rstrip("\n").split(",")
                if lines[-1]=="":
                    lines=lines[:-1]
                counter=0

                for j in lines[2:]:
                    if counter==0:
                        firstnum=int(j)
                        counter+=1
                    elif counter==2:
                        secondnum=int(j)
```

```
                        if firstnum<=2 and secondnum<=2:
                            print(tsd1+","+tsd2+","+line,end='')
                        counter+=1
                    elif counter==1:
                        counter+=1
                    else:
                        counter=0


### FOR 5' ENDS
## compare BLAST match, select if a pair on 3' end matches at worst from 2nd basepair of TSD/flank
"""
                lines=line.rstrip("\n").split(",")
                if lines[-1]=="":
                    line=line[:-1]
                counter=0

                for j in lines[2:]:
                    if counter==0 or counter==2:
                        counter+=1
                    elif counter==1:
                        firstnum=int(j)
                        counter+=1
                    else:
                        secondnum=int(j)
                        if firstnum>=499 and secondnum>=499:
                            print(tsd1+","+tsd2+","+line,end='')
                        counter=0
"""
f.close()
```

1.ix. connectends.py

```
# This script loads lists of detected duplication on 5' and 3' flanks and creates a list of proper duplications, that occur on both ends of a pair of insertions.
five=[]
three=[]
# Load 5' duplication list
with open("allreference.5.duplications.txt","r") as f:
    for line in f:
        five.append(line.rstrip("\n").split(","))
f.close()
# Load 3' duplication list
with open("allreference.3.duplications.txt","r") as f:
    for line in f:
        three.append(line.rstrip("\n").split(","))
f.close()
# Compile pairs of insertions that form duplications on 5' and 3' flanks
for i in tuple(five):
    for j in tuple(three):
        if (i[2].split("_")[0]==j[2].split("_")[0] and i[3].split("_")[0]==j[3].split("_")[0]) or (i[2].split("_")[0]==j[3].split("_")[0] and i[3].split("_")[0]==j[2].split("_")[0]):
            a=",".join(i).rstrip(",")
            b=",".join(j).rstrip(",")
            print(a)
            print(b)
            print("")
```

1.x. getduplist.py

```
pairs=[]
pairnames=[]
flag=0

```

```
# This script loads pairs of duplicated sequences and creates a list of insertions that are involved in segmental duplication, grouping them into appropriate groups.

# Open file with listed duplications.
with open('duplications.txt','r') as f:
    for line in f:
        if line=="\n":
            pairs.append(entry)
        else:
            line=",".join(line.split(",")[2:])
            if "\t" in line:
                line=line.replace("\t",",")
            if flag==0:
                entry=[line.rstrip("\n").lstrip("\ufeff")]
                flag=1
            else:
                flag=0
                entry.append(line.rstrip("\n").lstrip("\ufeff"))
f.close()
# Load duplications into memory.

for i in tuple(pairs):
    pairnames.append(tuple([i[0].split(",")[0].split("_")[0],i[0].split(",")[1].split("_")[0]]))

listofdup=[list(pairnames[0])]

# Group insertions and print them.

for i in tuple(pairnames[1:]):
    found=0
    counter=0
    for j in listofdup:
        if i[0] in j and i[1] in j:
            found=1
            break
        if i[0] in j and i[1] not in j:
            listofdup[counter].append(i[1])
            found=1
            break
        if i[1] in j and i[0] not in j:
            listofdup[counter].append(i[0])
            found=1
            break
        counter+=1
    if found==0:
        listofdup.append(list(i))
counter=0
for i in listofdup:
    for j in i[0:-1]:
        print(j,end=',')
        counter+=1
    print(i[-1])
    counter+=1
print(counter)
```

1.xi. getoneside.py

```
# This script extracts duplications observed only on 5' or 3' flanks of pairs of insertions, for manual inspection.
dups=[]
refins=[]
ref113=''
# Load reference herv-k113 sequences
```

```
with open("LTR113.fasta","r") as a:
    for line in a:
        ref113+=line
a.close()
# Load source file with sequences of extracted viral loci
with open("allhervk.fasta","r") as refi:
    for line in refi:
        if line.startswith(">"):
            insert=[line]
        else:
            insert.append(line)
            refins.append(tuple(insert))
# Load duplication list
with open("duplications.all.txt","r") as f:
    for line in f:
        for i in tuple(line.rstrip("\n").split(",")):
            dups.append(i)
f.close()
dups=set(dups)
pairs=[]
# Load list of duplications on 5' flank
with open("allreference.5.duplications.txt","r") as f:
    start=1
    for line in f:
        if tuple(line.split(","))[2].split("_")[0] in dups and tuple(line.split(",")
    )[3].split("_")[0] in dups:
            pass
        else:
            if start==1:
                start=0
                #print("Single 5' ends:")
            #print(line.rstrip("\n"))
            pairs.append(tuple([line.split(",")[2],line.split(",")[3]]))
f.close()
counter=0
# Save 5' flank only duplications
for i in tuple(pairs):
    found=0
    five=[]
    for j in refins:
        if i[0]==j[0].lstrip(">").rstrip("\n") or i[1]==j[0].lstrip(">").rstrip("\n"
    ):
            five.append(j[0]+j[1])
            found+=1
        if found==2:
            break
    if len(five)>1:
        counter+=1
        with open("5end.duplications"+str(counter)+".fasta",'a') as fiveend:
            fiveend.write(ref113)
            fiveend.write(five[0])
            fiveend.write(five[1])
fiveend.close()
pairs=[]
# Load list of duplications on 3' flank
with open("allreference.3.duplications.txt","r") as f:
    start=1
    for line in f:
#        print(line)
        #print(tuple(line.split(",")))

        if tuple(line.split(","))[2].split("_")[0] in dups and tuple(line.split(",")
    )[3].split("_")[0] in dups:
            pass
```

```
        else:
            if start==1:
                start=0
                #print("Single 3' ends:")
            #print(line.rstrip("\n"))
            pairs.append(tuple([line.split(",")[2],line.split(",")[3]]))
counter=0
# Save 3' flank only duplications
for i in tuple(pairs):
    found=0
    three=[]
    for j in refins:
        if i[0]==j[0].lstrip(">").rstrip("\n") or i[1]==j[0].lstrip(">").rstrip("\n"
    ):
            three.append(j[0]+j[1])
            found+=1
        if found==2:
            break
    if len(three)>1:
        counter+=1
        with open("3end.duplications"+str(counter)+".fasta",'a') as threeend:
            threeend.write(ref113)
            threeend.write(three[0])
            try:
                threeend.write(three[1])
            except IndexError:
                print(three)
                break
        threeend.close()
f.close()
```

1.xii. isolate.py

```
# This script loads the known insertions list and divides the found loci in .fasta
    format into previously known insertions and novel loci.
from os import listdir
from os import mkdir

inserts=[]
# Load list of known insertions.
with open('mikedbpoly.csv','r') as db:
    for line in db:
    # Load each line, with name and location of the locus.
        line=line.rstrip("\n").split(",")
        line[2]=int(line[2])
        line[3]=int(line[3])
        inserts.append(tuple(line))
db.close()
inserts=tuple(inserts)

arr = listdir()
temp=[]
# Load all sam files in the folder.
for i in arr:
    if ".sam" in i:
        temp.append(i)
arr=tuple(temp)
#print(inserts)

# Open each sam file in the folder
for fi in arr:
    with open(fi,'r') as f:
        for line in f:
            if line.startswith("@"):
                continue
```

```
            linel=line.rstrip("\n")
            line=line.rstrip("\n").split("\t")
            # Find chromosome for each read
            chro=line[2].lstrip("chr")
            # Read location for each read.
            try:
                loc1=int(line[3])
            except ValueError:
                print(linel)
            try:
                loc2=int(line[7])
            except ValueError:
                print(linel)
            # See if the read comes from a known location.
            for k in inserts:
              # If the read comes from +-1200bp of known location, put it in known folder
                if chro==k[1] and (k[2]-1200<=loc1<=k[3]+1200 or k[2]-1200<=loc2<=k[3]+1200):
                    try:
                        mkdir(fi.rstrip(".sam")+"known")
                    except FileExistsError:
                        pass
                    with open(fi.rstrip(".sam")+"known/"+k[0]+".sam","a") as res:
                        res.write(linel+"\n")
                    res.close()
                    break
    f.close()
```

1.xiii. coverage3.py

```
# This script calculates coverage of sam files to see if the insertion is a real thing or just a misalinment.

from numpy import std

from os import listdir


folders=listdir()
temp=[]
for i in folders:
    if "." not in i:
        temp.append(i)
folders=temp

knownins=[]
# Load a list of known insertions.
with open('allncbi36.csv','r') as kn:
    for line in kn:
        knownins.append(tuple(line.rstrip("\n").split(",")))
    kn.close()
knownins=tuple(knownins)
# Procedure that calculates coverage for a given location
def coverage(sam,l): # OFC zero-based location numbering.
    locations=[]

    for i in tuple(sam):
        loc=int(i[1])-1
        cigar=[]
        cons=""
# Load cigar string
        for j in i[2]:
            cons+=j
            if j.isupper()==True:
```

```
                cigar.append(cons)
                cons=""
        s=1
 # Calculate coverage from cigar string by adding up cigar operations
        for c in tuple(cigar):
            if c[-1]=="S" or c[-1]=="H" or c[-1]=="P" or c[-1]=="N":
                pass
            elif c[-1]=="M" or  c[-1]=="X" or  c[-1]=="=":
                if s==1:
                    for i in range(1,int(c[:-1])+1):
                        locations.append(loc+i)
                        s=0
                else:
                    for i in range(1,int(c[:-1])+1):
                        locations.append(locations[-1]+i)
    locations=tuple(locations)
    cat=list(set(locations))
    counter=0
    for c in tuple(cat):
        if type(l)==int:
            if c==l:
                cat[counter]=tuple([c,locations.count(c)])
                counter+=1
        else:
            if c in set(l):
                cat[counter]=tuple([c,locations.count(c)])
                counter+=1
    cat1=[]
    for j in cat:
        if type(j)==int:
            pass
        else:
            cat1.append(j)
    cat=tuple(cat1)
    if type(l)==int:
        for j in cat:
            try:
                if int(l)==j[0]:
                    return j[1]
            except TypeError:
                print(line)
                print(j)
                print(l)
                break
    elif (type(l)==list or type(l)==tuple) and len(l)==2:
        res=[]
        for i in range(l[0],l[1]):
            for j in cat:
                if int(i)==j[0]:
                    res.append(int(j[1]))
        return tuple(res)
    else:
        raise ValueError('Please specify location or a list/tuple [start,end]')




samfile=[]
# Load all sam files in a folder
for fol in folders:
    files=listdir(fol)
    # Load sam file
    for fi in files:
```

```
        samfile=[]
        with open(fol+"/"+fi,'r') as source:
            # Go through all the reads in sam file
            for line in source:
                line=line.split("\t")
                samfile.append(tuple([line[0],line[3],line[5]]))
        source.close()
        samfile=tuple(samfile)
        for k in knownins:
        # See if the read comes from a known insert
            if k[0]==fi.rstrip(".sam"):
                loc=int(k[2])
               # calculate coverage for the location cover=[coverage(samfile,loc-30),coverage(samfile,loc-20),coverage(samfile,loc-10),coverage(samfile,loc),coverage(samfile,loc+10),coverage(samfile,loc+20),coverage(samfile,loc+30)]
                for i in cover:
                    if type(i)!=int:
                        cover[cover.index(i)]=0

                #cov=coverage(samfile,[loc-31,loc+29])
                #print(k[0]+" "+str(cov[0])+","+str(cov[9])+","+str(cov[19])+","+str(cov[29])+","+str(cov[39])+","+str(cov[49])+","+str(cov[59]),end=", 3end:")
                # Print coverage for the insert -20 -10 0 +10 +20bp from its location.
                print(fol+" "+k[0]+" "+str(cover[0])+","+str(cover[1])+","+str(cover[2])+","+str(cover[3])+","+str(cover[4])+","+str(cover[5])+","+str(cover[6]),end=",")
                print("{:.4f}".format(std(cover[0:4]))+","+"{:.4f}".format(std(cover[4:]))+","+"{:.4f}".format(std(cover)),end=', 3end:')
                loc=int(k[3])
                #cov=coverage(samfile,[loc-31,loc+29])
                #print(str(cov[0])+","+str(cov[9])+","+str(cov[19])+","+str(cov[29])+","+str(cov[39])+","+str(cov[49])+","+str(cov[59]))
                cover=[coverage(samfile,loc-30),coverage(samfile,loc-20),coverage(samfile,loc-10),coverage(samfile,loc),coverage(samfile,loc+10),coverage(samfile,loc+20),coverage(samfile,loc+30)]
                for i in cover:
                    if type(i)!=int:
                        cover[cover.index(i)]=0
                print(fol+" "+str(cover[0])+","+str(cover[1])+","+str(cover[2])+","+str(cover[3])+","+str(cover[4])+","+str(cover[5])+","+str(cover[6]),end=',')
                print("{:.4f}".format(std(cover[0:4]))+","+"{:.4f}".format(std(cover[4:]))+","+"{:.4f}".format(std(cover)))
                break
```

1.xiv. sam2fasta.py

```
# This script converts .SAM source files to .fasta files, including necessary properties.

#!/usr/bin/env python3
from intro import sam2fasta_intro
import pathlib, psutil, sys
import subprocess
from hervmodules import sam2fasta




fastalist=sam2fasta('sam')

if len(sys.argv)>2:
    sam2fasta_intro("toomany")
else:
```

```
    if len(sys.argv)==2 and sys.argv[1]=="HELP":
        sam2fasta_intro("HELP")
    elif len(sys.argv)==2 and (sys.argv[1].isdigit()==False or sys.argv[1]=="0"):
        sam2fasta_intro("notint")
    else:
        if len(sys.argv)==2:
            threads=int(sys.argv[1])
        else:
            threads=psutil.cpu_count(logical=False)
        for i in range(1,int(threads)+1):
            pathlib.Path(str(i)).mkdir(parents=False, exist_ok=True)
        k=1
        for j in fastalist:
            a=subprocess.Popen(['/bin/bash', '-c', 'mv '+str(j)+' '+str(k)+'/'],
   stdin=subprocess.PIPE, stderr=subprocess.STDOUT)
            a.wait()
            k+=1
            if k>int(threads):
                k=1
```

1.xv. extractunmapped.py

```
# This script opens BLAST results txt file and extracts reads that have been marked
    with blast to have at least 30bp long matches from source fasta file.

from os import listdir
from os.path import isfile, join
onlyfiles = [f for f in listdir(".") if isfile(join(".", f))]
blastres=[]
# open blast result in folder
for i in onlyfiles:
  if i.endswith(".blast.txt"):
    blastres.append(i)


for i in blastres:
  source=i.split(".")[0]+".sra."+i.split(".")[2]+".sam.fasta"
  if "SRR" not in i:
    continue
  blastfile=[]
  toextract=[]
# open appropriate files
  with open(i,'r') as f:
    for line in f:
      line=line.split("\t")
# select reads with matches >=30bp
      if int(line[3])>30:
        toextract.append(line[0])
        blastfile.append(line)
  f.close()
  toextract=set(toextract)
# Extract the reads from source files.
  with open(source,'r') as fastafile:
    with open(source.rstrip('fasta')+"filtered.30.fasta","a") as result:
      flag=0
      for line in fastafile:
        if flag==1:
          result.write(line)
          flag=0
        if line.startswith(">"):
          if line.rstrip("\n").lstrip(">") in toextract:
            flag=1
            result.write(line.rstrip("\n"))
            for i in tuple(blastfile):
              if i[0]==line.rstrip("\n").lstrip(">"):
```

```
                if "_5" in i[1]:
                  result.write("_five\n")
                elif "_3" in i[1]:
                  result.write("_three\n")
                break

    result.close()
  fastafile.close()
```

1.xvi. sort.py

```
# This script sorts results of read extraction in *.fasta format by location, ascending.
from glob import glob
# Open all fasta files in the folder
for file in glob('*30.fasta'):
  with open(file,'r') as source:
    data=[]
    for line in source:
      # Load the description lines.
      if line.startswith(">"):
        dual=[int(line.rstrip("\n").split("_")[-2]),line]
      else:
        # load the sequence lines
        dual.append(line)
        data.append(tuple(dual))
        dual=[]
  source.close()
  # sort reads ascending, by location
  data=sorted(data,key=lambda read: read[0])
  # save sorted results into sorted subfolder
  with open("sorted/"+file,'a') as res:
    for i in tuple(data):
      res.write(i[1])
      res.write(i[2])
  res.close()
```

1.xvii. contigs.py

```
# This script loads existing reads in fasta files and assembles them into 300bp contigs.

from glob import glob
# Open each fasta file.
for file in glob('*.fasta'):
  with open(file,'r') as source:
    contig=[]
    read=[]
    # Load the location data.
    for line in source:
      if line.startswith(">"):
        loc=int(line.rstrip("\n").split("_")[-2])
        read=[loc,line]
      else:
        read.append(line)
        if len(contig)==0:
          contig.append(tuple(read))
        else:
          # Combine the reads into contigs coming from 300bp range. Since reads are 150-250bp long this should combine reads from single DNA fragments.
          if abs(contig[-1][0]-read[0])<300:
            contig.append(tuple(read))
          else:
            # create new file names with locations
            newfile=file[:-6]+"."+str(contig[0][0])+".fasta"
            contig=tuple(contig)
```

```
            # save contigs into contigs subfolder
            with open('contigs/'+newfile,'a') as res:
              for re in contig:
                res.write(re[1])
                res.write(re[2])
            res.close()
            contig=[tuple(read)]
    newfile=file[:-6]+"."+str(contig[0][0])+".fasta"
    contig=tuple(contig)
    with open('contigs/'+newfile,'a') as res:
      for re in contig:
        res.write(re[1])
        res.write(re[2])
    res.close()
  source.close()
```

1.xviii. removeknown.py

```
# This script loads a list of known insertions and divides results saved in *.fasta
    format into a group of loci coming from known, previously reported insertions and
     previously unreported loci.
from shutil import copyfile

import glob
known=[]
# load a list of known reference insertions  from a csv file, formatted as : name,
    chromosome, loc start, loc end, icludes insertionally polymorphic that are
    present in the reference.
with open('allncbi37.csv','r') as f:
  for line in f:
    line=line.rstrip("\n").split(",")
    line[2]=int(line[2])
    line[3]=int(line[3])
    known.append(tuple(line))
f.close()
known=tuple(known)
poly=[]
# load a list of polymorphic insertions (this includes reference and non-reference
    insertions) from a csv file, formatted as : name, chromosome, loc start, loc end
with open('polymorphic37.csv','r') as f:
  for line in f:
    line=line.rstrip("\n").split(",")
    line[2]=int(line[2])
    line[3]=int(line[3])
    poly.append(tuple(line))
f.close()
poly=tuple(poly)
analysed=[]

# do for all fasta files in foleder
for files in glob.glob("*.fasta"):
  analysed=[]
  analysedp=[]
  # focus on inserts on a particular chromosome.
  for i in known:
    if i[1]==files.split(".")[2]:
      analysed.append(i)
  analysed=tuple(analysed)
  for i in poly:
    if i[1]==files.split(".")[2]:
      analysedp.append(i)
  analysedp=tuple(analysedp)
  # open a fasta file
  with open(files,'r') as source:
    name=''
```

```
    for line in source:
      # scan for header rows
      if line.startswith(">"):
        # look if the read comes from within +/- 1000bp of any known inserts
        for a in analysed:
          # if it does, mark it
          if a[2]-1000<int(line.rstrip("\n").split("_")[-2])<a[3]+1000:
              name=a[0]
              break
        # if found and marked, stop looking
        if name!='':
          break
        for a in analysedp:
          # if it does, mark it
          if a[2]-1000<int(line.rstrip("\n").split("_")[-2])<a[3]+1000:
              name=a[0]+"poly"
              break
        # if found and marked, stop looking
        if name!='':
          break
  source.close()
  # if found and marked, copy and mark locus file with appropriate insert name
  if name!='' and name[-4:]!="poly":
    copyfile(files,'div/known/'+files.rstrip("fasta")+name+'.fasta')
  # else just copy file as a potential new insert
  elif name[-4:]=="poly":
    copyfile(files,'div/poly/'+files.rstrip("fasta")+name.rstrip("poly")+'.fasta')
  else:
    copyfile(files,'div/'+files)
```

1.xix. 53pw.py

```
# This script selects results by their alignment to both 5' and 3' end of the HERV-K113 and selects results contating
# both ends within 1000bp alingment location (read length is approx. 250bp, so every DNA fragment is approx. 500bp max,
# whereas two consecutive fragments can point to the same location, so the location of novel insert can be +-1000bp).

# import modules for file operations
from glob import glob
from shutil import copyfile
# Load all fasta files in the folder
files=[]
for i in glob("*.fasta"):
  with open(i,'r') as f:
    five=0
    three=0
    min=0
    max=0
    # For each fasta file, load reads aligned to either ends of 113 ltr.
    for line in f:
      if line.startswith(">"):
        if line.endswith("_three\n"):
          three=1
        elif line.endswith("_five\n"):
          five=1
        line=line.split("_")
        # Load chromosome number
        chr=line[4]
        try:
          # Load location and chromosome number, in case of slightly different formatting load it according to except formula.
          loc=int(line[5])
        except ValueError:
```

```
          chr=line[5]
          loc=int(line[6])
          # get the minimal and maximal alignment locations.
        if loc<min or min==0:
          min=loc
        if loc>max or max==0:
          max=loc
  # If there are no results, just add this to results.
  if len(files)==0:
    files.append([five,three,chr,min,max])
  # Else check if there already are any reads from within +-1000bp of the reads
    location and add locations if there are outside of existing ones, i.e. lower than
     start or higher than end.
  else:
    found=0
    j=0
    for i in files:
      # check the location +-1000bp and chromosome of the read
      if chr==i[2] and (i[3]-1000<=max<=i[4]+1000 or i[3]-1000<=min<=i[4]+1000):
        newfile=[i[0],i[1],i[2],i[3],i[4]]
        # update 5' end status
        if five==1:
          newfile[0]=1
        # update 3' end status
        if three==1:
          newfile[1]=1
        # update 5' end outer location
        if min<i[3]:
          newfile[3]=min
        # update 3'end outer location
        if max>i[4]:
          newfile[4]=max
        files[j]=newfile
        found=1
        break
      j+=1
    # If location found for the first time, add to list
    if found==0:
      files.append([five,three,chr,min,max])

  f.close()
# Loop through the results
for j in tuple(files):
  # If the result contains both 5' and 3' ends, continue
  if j[0]==1 and j[1]==1:
    # find the fasta file with reads that are selected.
    for i in glob("*.fasta"):
      i1=tuple(i.split("."))
      # copy the fasta file with appropriate result to 53pw folder.
      if i1[2]==j[2].lstrip("chr") and j[3]<=int(i1[-2])<=j[4]:
        with open('53pw/'+j[2]+"."+str(j[3])+".fasta",'a') as res:
          with open(i,'r') as source:
            for line in source:
              res.write(line)
              if line[-1]!="\n":
                res.write("\n")
          source.close()
        res.close()
```

1.xx. identifycancer.py

```
# This script identifies cancer names within OTP reference results. It extracts the
    disease names from known cells in the spreadsheet and prints them on screen.
counter=0
cancernames=[]
```

```
import csv
# open results csv file
with open('results.csv', newline='') as csvfile:
  res = csv.reader(csvfile, delimiter=',', quotechar='"')
  # Go through lines
  for liner in res:
    counter+=1
    # If the line is empty, skip
    if len(liner)<2:
      continue
    # If there's data in the line, analyse
    if liner[0]=="" and liner[1]!="":
      fieldcount=0
      found=0
      # go through data cells, looking for disease name
      for i in liner:
        if i=="disease.name":
          found=1
          break
        fieldcount+=1
      if found==0:
        fieldcount=0
        # go through data cells, looking for disease info
        for i in liner:
          if i=="disease.efo_info.therapeutic_area.labels":
            found=1
            break
          fieldcount+=1
        # If disease not found, print error
        if found==0:
          print("Line "+str(counter)+" disease.id not found.")
    # If theres data on disease, append the name to a list.
    elif liner[0].isdigit()==True:
      if found==1:
        try:
          cancernames.append(liner[fieldcount])
        # Print lines if there are any formatting errors.
        except IndexError:
          print(liner)
          print(fieldcount)
          raise IndexError

csvfile.close()
# Print disease names on screen.
for c in list(set(cancernames)):
  print(c)
```

1.xxi. filtercancer.py

```
# This script filters all results from OpenTargetsPlatform by cancer keywords, to
    extract all the links associated with cancer.

# All of the found keywords regarding different cancers in literature
keywords=["cancer","tumor","tumour","carcinoid","neoplas","lymphom","sarcom","
    carcinom","leukaem","astocytom","malignan","melanom","myelom","glioma","blastom",
    "adenom","papill","glaucom","meningiom"]
cancernames=[]
others=[]
# Open all the disease names extracted from OTP results and select those, that
    contain keywords.
with open("cancerlist.txt",'r') as source:
  for line in source:
    for word in line.rstrip("\n").split(" "):
      # Select results containting keywords.
      if word in set(keywords):
```

```
        cancernames.append(line.lstrip("."))
        break
      else:
        # Select results containing parts of keywords.
        for k in tuple(keywords):
          if k in word.lower() and len(word)>3:
            cancernames.append(line.lstrip("."))
            break
    if line.lstrip(".") not in set(cancernames) or line not in set(cancernames):
      others.append(line)

source.close()
# Add cancer names to autocancer file
with open('autocancer.txt','a') as canres:
  for i in tuple(set(cancernames)):
    canres.write(i)
canres.close()
# Add other disease names to autoother file
with open('autoother.txt','a') as ores:
  for i in tuple(set(others)):
    ores.write(i)
ores.close()
```

1.xxii. getdata.py

```
# This script uses OpenTargetsClient library to download all literature cancer data, using found gene names.

# Import modules
from time import sleep
from opentargets import OpenTargetsClient

ot = OpenTargetsClient()
genes=[]
# Load list of genes form opentargetsplatform files.
with open('genelist.otp.txt','r') as src1:
  for line in src1:
    genes.append(line.rstrip("\n"))
src1.close()

# Load list of other genes extracted from other databases
with open('genelist.other.txt','r') as src2:
  for line in src2:
    genes.append(line.rstrip("\n"))
src2.close()
# Loop through the collection of gene names
for g in tuple(set(genes)):
  # Get all results from the online search and save it to csv file, skip if genes produce errors and save problematic gene names.
  try:
    e = ot.get_evidence_for_target(g)
    print(e.to_csv())
  except StopIteration:
    with open('errorlist.log','a') as er:
      er.write("stopiteration: "+g+"\n")
er.close()
```

1.xxiii. gentable.py

```
# This script counts presence of reads aligning to 5' end 3' end of HERV-K113 in sam alignments produced in Geneious.
from glob import glob
title=['Filename','Size[GB]','Case','Is Tumor?']
lines=[]
# Read results from done SRA files.
with open('donefiles.csv','r') as f:
```

```
  for line in f:
    lines.append(line.rstrip("\n"))
f.close()
# Open sam files with alignments of novel inserts
for name in glob("*.sam"):
  title.append(name.rstrip(".sam"))
  inserts=[]
  with open(name,'r') as source:
    # Load reads into memory
    for line in source:
      if line.startswith("@"):
        continue
      else:
        info=tuple([line.split("\t")[0].split("_")[0],line.split("\t")[0].split("_")[-1]])
        inserts.append(info)
  source.close()
  count=0
  # go through results from csv file
  for i in lines:
    lines[count]=lines[count]+","
    # Find data on specific result in sam file.
    for j in tuple(inserts):
      # Count if there are 5' ends
      if i.split(",")[0]==j[0] and j[1]=='five':
        if "5" in lines[count].split(",")[-1]:
          pass
        else:
          lines[count]=lines[count]+'5'
      # Count if there are 3' ends
      if i.split(",")[0]==j[0] and j[1]=='three':
        if "3" in lines[count].split(",")[-1]:
          pass
        else:
          lines[count]=lines[count]+'3'

    count+=1
# Print results
print(",".join(title))

for i in lines:
  print(i)
```

1.xxiv. calculatefreq.py

```
# This script calculates insertion frequencies from 2016 Wildschutte paper data
from glob import glob
projects=[]
# open csv for each dataset and load to a list.
for i in glob("*.csv"):
  if "freq" in i:
    continue
  proj=[i.rstrip(".csv")]
  with open(i,'r') as f:
    for line in f:
      proj.append(line.rstrip("\n").split(";"))
  f.close()
  projects.append(tuple(proj))
files=[]
res=[]
# go through all the projects loaded.
for j in projects:
  cat=[]

  # extract data for presence / absence of instert in individuals.
```

```
  for z in j[12:]:
    indiv=0
    ref=0
    alt=0
    hetero=0
    missing=0
    for fre in z[10:]:
      # Count presence if the individual is indicated to have insert and preinsert site(1/1) or only insert reads (1/0)
      if fre.split(":")[0]=="1/1":
        alt+=1
        indiv+=1
      elif fre.split(":")[0]=="0/0":
        ref+=1
        indiv+=1
      # do not count if theres no data
      elif fre.split(":")[0]=="0/1":
        hetero+=1
        indiv+=1
      # do not count if theres no data
      elif fre.split(":")[0]==".":
        missing+=1
#       indiv+=1
      # just count the individual if there's data but no inserts.

# put the data into a table of all inserts
    if len(cat)==0:
      cat.append(z[2]+","+str(float(alt)/float(indiv))+";"+str(float(ref)/float(indiv))+";"+str(float(hetero)/float(indiv))+";"+str(missing))
    else:
      found=0
      counter=0
      for i in cat:
        if i.split(",")[0]==z[2]:
          # calculate frequencies
          cat[counter]=i+","+str(float(alt)/float(indiv))+";"+str(float(ref)/float(indiv))+";"+str(float(hetero)/float(indiv))+";"+str(missing)
          found=1
          break
        counter+=1
      if found==0:
        cat.append(z[2]+","+str(float(alt)/float(indiv))+";"+str(float(ref)/float(indiv))+";"+str(float(hetero)/float(indiv))+";"+str(missing))
  files.append(j[0]+" "+str(indiv+missing))
  res.append(cat)

print("",end=',')
# Print filenames
for i in files[0:-1]:
  print(i,end=',')
print(files[-1])
inserts=[]
# get all insert names
for i in res[0]:
  inserts.append(i.split(",")[0])
count=0


for i in inserts:
  # print insert names
  print(i,end=',')
  # print appropriate frequencies for each files in a line
  for j in res[0:-1]:
    print(j[count].split(",")[1],end=",")
```

```
  print(res[-1][count].split(",")[1])
  count+=1
# print(j[12:])
```

1.xxv. canceranalysis.Rmd

```
---
title: "Popn Analysis of novel insertions found from cancer"
output: html_document
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```

Intro
Novel HERV-K inserts that you found in the cancer genomes that have been previously described by Wildschutte et al as being in the general population. Ultimately, we want to compare these allele freqs to the allele freqs that you observed for these particular insertions. But first need to check whether there is a significant difference between the alelle freqs of these insertions between populations....

```{r Read in Data}
PopnData <- read.csv("NovelCancerFreqs.csv", header = T)
colnames(PopnData)
unique(PopnData["Population"])
which(PopnData$Population=="HGDP")
unique(PopnData$Name)
which(PopnData$Name =="HERV-K-c1")
PopnData[which(PopnData$Name=="HERV-K-c1"),]
PopnData[which(PopnData$Name=="HERV-K-c1"),] -> dfc1
PopnData[which(PopnData$Name=="HERV-K-c2"),] -> dfc2
PopnData[which(PopnData$Name=="HERV-K-c3"),] -> dfc3
PopnData[which(PopnData$Name=="HERV-K-c6"),] -> dfc6
PopnData[which(PopnData$Name=="HERV-K-c7"),] -> dfc7
PopnData[which(PopnData$Name=="HERV-K-c8"),] -> dfc8
PopnData[which(PopnData$Name=="HERV-K-c10"),] -> dfc10
PopnData[which(PopnData$Name=="HERV-K-c12"),] -> dfc12
PopnData[which(PopnData$Name=="HERV-K-c13"),] -> dfc13
PopnData[which(PopnData$Name=="HERV-K-c14"),] -> dfc14
PopnData[which(PopnData$Name=="HERV-K-c15"),] -> dfc15
PopnData[which(PopnData$Name=="HERV-K-c17"),] -> dfc17
PopnData[which(PopnData$Name=="HERV-K-c18"),] -> dfc18
PopnData[which(PopnData$Name=="HERV-K-c19"),] -> dfc19
PopnData[which(PopnData$Name=="HERV-K-c20"),] -> dfc20
PopnData[which(PopnData$Name=="HERV-K-c22"),] -> dfc22
PopnData[which(PopnData$Name=="HERV-K-c23"),] -> dfc23
PopnData[which(PopnData$Name=="HERV-K-c24"),] -> dfc24
PopnData[which(PopnData$Name=="HERV-K-c25"),] -> dfc25
PopnData[which(PopnData$Name=="HERV-K-c27"),] -> dfc27
```

Plotting the data for each locus shows varying patterns...

```{r echo=FALSE}
plot(1:27,dfc1$No_of_insertions/(2*dfc1$No_of_samples))
```

To see if these distributions are statistically different at each locus, we use Fisher's exact test (can't use Chi squared because some of your data points are zero). Have to use the "simulate.p.value" option as memory requirements are otherwise too large <- READ UP ON THIS!!!
```{R code}
fisher.test(rbind(dfc1$No_of_insertions, dfc1$No_of_PIS), simulate.p.value = TRUE, B
```
```

```
    =1000000)
fisher.test(rbind(dfc2$No_of_insertions, dfc2$No_of_PIS), simulate.p.value = TRUE, B
    =1000000)
fisher.test(rbind(dfc3$No_of_insertions, dfc3$No_of_PIS), simulate.p.value = TRUE, B
    =1000000)
fisher.test(rbind(dfc6$No_of_insertions, dfc6$No_of_PIS), simulate.p.value = TRUE, B
    =1000000)
fisher.test(rbind(dfc7$No_of_insertions, dfc7$No_of_PIS), simulate.p.value = TRUE, B
    =1000000)
fisher.test(rbind(dfc8$No_of_insertions, dfc8$No_of_PIS), simulate.p.value = TRUE, B
    =1000000)
fisher.test(rbind(dfc10$No_of_insertions, dfc10$No_of_PIS), simulate.p.value = TRUE,
    B=1000000)
fisher.test(rbind(dfc12$No_of_insertions, dfc12$No_of_PIS), simulate.p.value = TRUE,
    B=1000000)
fisher.test(rbind(dfc13$No_of_insertions, dfc13$No_of_PIS), simulate.p.value = TRUE,
    B=1000000)
fisher.test(rbind(dfc14$No_of_insertions, dfc14$No_of_PIS), simulate.p.value = TRUE,
    B=1000000)
fisher.test(rbind(dfc15$No_of_insertions, dfc15$No_of_PIS), simulate.p.value = TRUE,
    B=1000000)
fisher.test(rbind(dfc17$No_of_insertions, dfc17$No_of_PIS), simulate.p.value = TRUE,
    B=1000000)
fisher.test(rbind(dfc18$No_of_insertions, dfc18$No_of_PIS), simulate.p.value = TRUE,
    B=1000000)
fisher.test(rbind(dfc19$No_of_insertions, dfc19$No_of_PIS), simulate.p.value = TRUE,
    B=1000000)
fisher.test(rbind(dfc20$No_of_insertions, dfc20$No_of_PIS), simulate.p.value = TRUE,
    B=1000000)
fisher.test(rbind(dfc22$No_of_insertions, dfc22$No_of_PIS), simulate.p.value = TRUE,
    B=1000000)
fisher.test(rbind(dfc23$No_of_insertions, dfc23$No_of_PIS), simulate.p.value = TRUE,
    B=1000000)
fisher.test(rbind(dfc24$No_of_insertions, dfc24$No_of_PIS), simulate.p.value = TRUE,
    B=1000000)
fisher.test(rbind(dfc25$No_of_insertions, dfc25$No_of_PIS), simulate.p.value = TRUE,
    B=1000000)
fisher.test(rbind(dfc27$No_of_insertions, dfc27$No_of_PIS), simulate.p.value = TRUE,
    B=1000000)
```

1.xxvi. fishertestall.R

```
## This script loads counts of novel insertions detected in cancer genomes and
    performs a Fishers Exact test on them against 1KGP and HGDP populations tested in
     Wildschutte et al. 2016

setwd("~/Desktop/wildschutte data/rav stats")
PopnData <- read.csv("mixeddatahetero.csv", header = T)

PopnData[which(PopnData$Name=="HERV-K-c1" & (PopnData$Population=="HGDP" | PopnData$
    Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c1" & (PopnData$Population=="1KGP" | PopnData$
    Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c1" & (PopnData$Population=="HGDP" | PopnData$
    Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c1" & (PopnData$Population=="1KGP" | PopnData$
    Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))

PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="HGDP" | PopnData$
    Population=="BrT")),] -> dfc
```

```
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="1KGP" | PopnData$
    Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="HGDP" | PopnData$
    Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="1KGP" | PopnData$
    Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="HGDP" | PopnData$
    Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="1KGP" | PopnData$
    Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="HGDP" | PopnData$
    Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="1KGP" | PopnData$
    Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="HGDP" | PopnData$
    Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="1KGP" | PopnData$
    Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="HGDP" | PopnData$
    Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="1KGP" | PopnData$
    Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="HGDP" | PopnData$
    Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="1KGP" | PopnData$
    Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="HGDP" | PopnData$
    Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="1KGP" | PopnData$
    Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="HGDP" | PopnData$
    Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="1KGP" | PopnData$
    Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="HGDP" | PopnData$
    Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="1KGP" | PopnData$
    Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="HGDP" | PopnData$
    Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="1KGP" | PopnData$
    Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
```

```
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="HGDP" | PopnData$
    Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="1KGP" | PopnData$
    Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="HGDP" | PopnData$
    Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="1KGP" | PopnData$
    Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="HGDP" | PopnData$
    Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="1KGP" | PopnData$
    Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="HGDP" | PopnData$
    Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="1KGP" | PopnData$
    Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="HGDP" | PopnData$
    Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="1KGP" | PopnData$
    Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="HGDP" | PopnData$
    Population=="BpT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="1KGP" | PopnData$
    Population=="BpT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="HGDP" | PopnData$
    Population=="BpC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c2" & (PopnData$Population=="1KGP" | PopnData$
    Population=="BpC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))

PopnData[which(PopnData$Name=="HERV-K-c3" & (PopnData$Population=="HGDP" | PopnData$
    Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c3" & (PopnData$Population=="1KGP" | PopnData$
    Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c3" & (PopnData$Population=="HGDP" | PopnData$
    Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c3" & (PopnData$Population=="1KGP" | PopnData$
    Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c3" & (PopnData$Population=="HGDP" | PopnData$
    Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c3" & (PopnData$Population=="1KGP" | PopnData$
    Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c3" & (PopnData$Population=="HGDP" | PopnData$
    Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
```

```
PopnData[which(PopnData$Name=="HERV-K-c3" & (PopnData$Population=="1KGP" | PopnData$
    Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))

PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
```

```
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="PaT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="PaT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="PaC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="PaC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="BpT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="BpT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="HGDP" | PopnData$
    Population=="BpC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c6" & (PopnData$Population=="1KGP" | PopnData$
    Population=="BpC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))

PopnData[which(PopnData$Name=="HERV-K-c7" & (PopnData$Population=="HGDP" | PopnData$
    Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
```

```
PopnData[which(PopnData$Name=="HERV-K-c7" & (PopnData$Population=="1KGP" | PopnData$
    Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c7" & (PopnData$Population=="HGDP" | PopnData$
    Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c7" & (PopnData$Population=="1KGP" | PopnData$
    Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c7" & (PopnData$Population=="HGDP" | PopnData$
    Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c7" & (PopnData$Population=="1KGP" | PopnData$
    Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c7" & (PopnData$Population=="HGDP" | PopnData$
    Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c7" & (PopnData$Population=="1KGP" | PopnData$
    Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c7" & (PopnData$Population=="HGDP" | PopnData$
    Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c7" & (PopnData$Population=="1KGP" | PopnData$
    Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c7" & (PopnData$Population=="HGDP" | PopnData$
    Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c7" & (PopnData$Population=="1KGP" | PopnData$
    Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c7" & (PopnData$Population=="HGDP" | PopnData$
    Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c7" & (PopnData$Population=="1KGP" | PopnData$
    Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c7" & (PopnData$Population=="HGDP" | PopnData$
    Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c7" & (PopnData$Population=="1KGP" | PopnData$
    Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))

PopnData[which(PopnData$Name=="HERV-K-c8" & (PopnData$Population=="HGDP" | PopnData$
    Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c8" & (PopnData$Population=="1KGP" | PopnData$
    Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c8" & (PopnData$Population=="HGDP" | PopnData$
    Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c8" & (PopnData$Population=="1KGP" | PopnData$
    Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c8" & (PopnData$Population=="HGDP" | PopnData$
    Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c8" & (PopnData$Population=="1KGP" | PopnData$
    Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
```

```
PopnData[which(PopnData$Name=="HERV-K-c8" & (PopnData$Population=="HGDP" | PopnData$
    Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c8" & (PopnData$Population=="1KGP" | PopnData$
    Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c8" & (PopnData$Population=="HGDP" | PopnData$
    Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c8" & (PopnData$Population=="1KGP" | PopnData$
    Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c8" & (PopnData$Population=="HGDP" | PopnData$
    Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c8" & (PopnData$Population=="1KGP" | PopnData$
    Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c8" & (PopnData$Population=="HGDP" | PopnData$
    Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c8" & (PopnData$Population=="1KGP" | PopnData$
    Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c8" & (PopnData$Population=="HGDP" | PopnData$
    Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c8" & (PopnData$Population=="1KGP" | PopnData$
    Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c8" & (PopnData$Population=="HGDP" | PopnData$
    Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c8" & (PopnData$Population=="1KGP" | PopnData$
    Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))

PopnData[which(PopnData$Name=="HERV-K-c10" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c10" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c10" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c10" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c10" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c10" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c10" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c10" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c10" & (PopnData$Population=="HGDP" | PopnData
    $Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
```

```
PopnData[which(PopnData$Name=="HERV-K-c10" & (PopnData$Population=="1KGP" | PopnData$Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c10" & (PopnData$Population=="HGDP" | PopnData$Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c10" & (PopnData$Population=="1KGP" | PopnData$Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))


PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="HGDP" | PopnData$Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="1KGP" | PopnData$Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="HGDP" | PopnData$Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="1KGP" | PopnData$Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="HGDP" | PopnData$Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="1KGP" | PopnData$Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="HGDP" | PopnData$Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="1KGP" | PopnData$Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="HGDP" | PopnData$Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="1KGP" | PopnData$Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="HGDP" | PopnData$Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="1KGP" | PopnData$Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="HGDP" | PopnData$Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="1KGP" | PopnData$Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="HGDP" | PopnData$Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="1KGP" | PopnData$Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="HGDP" | PopnData$Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="1KGP" | PopnData$Population=="MyT")),] -> dfc
```

```
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="HGDP" | PopnData
    $Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="1KGP" | PopnData
    $Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="HGDP" | PopnData
    $Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="1KGP" | PopnData
    $Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="HGDP" | PopnData
    $Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="1KGP" | PopnData
    $Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="HGDP" | PopnData
    $Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c12" & (PopnData$Population=="1KGP" | PopnData
    $Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))

PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="HGDP" | PopnData
    $Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="1KGP" | PopnData
    $Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="HGDP" | PopnData
    $Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="1KGP" | PopnData
    $Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="HGDP" | PopnData
    $Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="1KGP" | PopnData
    $Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="HGDP" | PopnData
    $Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="1KGP" | PopnData
    $Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="HGDP" | PopnData
    $Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="1KGP" | PopnData
    $Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="HGDP" | PopnData
    $Population=="HeC")),] -> dfc
```

```
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="1KGP" | PopnData
    $Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="HGDP" | PopnData
    $Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="1KGP" | PopnData
    $Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="HGDP" | PopnData
    $Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="1KGP" | PopnData
    $Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="HGDP" | PopnData
    $Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="1KGP" | PopnData
    $Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="HGDP" | PopnData
    $Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="1KGP" | PopnData
    $Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="HGDP" | PopnData
    $Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="1KGP" | PopnData
    $Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="HGDP" | PopnData
    $Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="1KGP" | PopnData
    $Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
```

```
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="HGDP" | PopnData$Population=="BpT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="1KGP" | PopnData$Population=="BpT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="HGDP" | PopnData$Population=="BpC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c13" & (PopnData$Population=="1KGP" | PopnData$Population=="BpC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))

PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="HGDP" | PopnData$Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="1KGP" | PopnData$Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="HGDP" | PopnData$Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="1KGP" | PopnData$Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="HGDP" | PopnData$Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="1KGP" | PopnData$Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="HGDP" | PopnData$Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="1KGP" | PopnData$Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="HGDP" | PopnData$Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="1KGP" | PopnData$Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="HGDP" | PopnData$Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="1KGP" | PopnData$Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="HGDP" | PopnData$Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="1KGP" | PopnData$Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="HGDP" | PopnData$Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="1KGP" | PopnData$Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="HGDP" | PopnData$Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
```

```
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="1KGP" | PopnData$Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="HGDP" | PopnData$Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="1KGP" | PopnData$Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="HGDP" | PopnData$Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="1KGP" | PopnData$Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="HGDP" | PopnData$Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="1KGP" | PopnData$Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="HGDP" | PopnData$Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="1KGP" | PopnData$Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="HGDP" | PopnData$Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="1KGP" | PopnData$Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="HGDP" | PopnData$Population=="BpT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="1KGP" | PopnData$Population=="BpT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="HGDP" | PopnData$Population=="BpC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c14" & (PopnData$Population=="1KGP" | PopnData$Population=="BpC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))

PopnData[which(PopnData$Name=="HERV-K-c15" & (PopnData$Population=="HGDP" | PopnData$Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c15" & (PopnData$Population=="1KGP" | PopnData$Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c15" & (PopnData$Population=="HGDP" | PopnData$Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c15" & (PopnData$Population=="1KGP" | PopnData$Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c15" & (PopnData$Population=="HGDP" | PopnData$Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c15" & (PopnData$Population=="1KGP" | PopnData$Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
```

```
PopnData[which(PopnData$Name=="HERV-K-c15" & (PopnData$Population=="HGDP" | PopnData$Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c15" & (PopnData$Population=="1KGP" | PopnData$Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c15" & (PopnData$Population=="HGDP" | PopnData$Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c15" & (PopnData$Population=="1KGP" | PopnData$Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c15" & (PopnData$Population=="HGDP" | PopnData$Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c15" & (PopnData$Population=="1KGP" | PopnData$Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))

PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="HGDP" | PopnData$Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="1KGP" | PopnData$Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="HGDP" | PopnData$Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="1KGP" | PopnData$Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="HGDP" | PopnData$Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="1KGP" | PopnData$Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="HGDP" | PopnData$Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="1KGP" | PopnData$Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="HGDP" | PopnData$Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="1KGP" | PopnData$Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="HGDP" | PopnData$Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="1KGP" | PopnData$Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="HGDP" | PopnData$Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="1KGP" | PopnData$Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="HGDP" | PopnData$Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
```

```
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="1KGP" | PopnData$Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="HGDP" | PopnData$Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="1KGP" | PopnData$Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="HGDP" | PopnData$Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="1KGP" | PopnData$Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="HGDP" | PopnData$Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="1KGP" | PopnData$Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="HGDP" | PopnData$Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="1KGP" | PopnData$Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="HGDP" | PopnData$Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="1KGP" | PopnData$Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="HGDP" | PopnData$Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="1KGP" | PopnData$Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="HGDP" | PopnData$Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="1KGP" | PopnData$Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="HGDP" | PopnData$Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c17" & (PopnData$Population=="1KGP" | PopnData$Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))

PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="HGDP" | PopnData$Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="1KGP" | PopnData$Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="HGDP" | PopnData$Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="1KGP" | PopnData$Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
```

```
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="HGDP" | PopnData$Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="1KGP" | PopnData$Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="HGDP" | PopnData$Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="1KGP" | PopnData$Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="HGDP" | PopnData$Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="1KGP" | PopnData$Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="HGDP" | PopnData$Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="1KGP" | PopnData$Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="HGDP" | PopnData$Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="1KGP" | PopnData$Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="HGDP" | PopnData$Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="1KGP" | PopnData$Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="HGDP" | PopnData$Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="1KGP" | PopnData$Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="HGDP" | PopnData$Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="1KGP" | PopnData$Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="HGDP" | PopnData$Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="1KGP" | PopnData$Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="HGDP" | PopnData$Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="1KGP" | PopnData$Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="HGDP" | PopnData$Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="1KGP" | PopnData
```

```
    $Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="HGDP" | PopnData
    $Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="1KGP" | PopnData
    $Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="HGDP" | PopnData
    $Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="1KGP" | PopnData
    $Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="HGDP" | PopnData
    $Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c18" & (PopnData$Population=="1KGP" | PopnData
    $Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))

PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="HGDP" | PopnData
    $Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="1KGP" | PopnData
    $Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="HGDP" | PopnData
    $Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="1KGP" | PopnData
    $Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="HGDP" | PopnData
    $Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="1KGP" | PopnData
    $Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="HGDP" | PopnData
    $Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="1KGP" | PopnData
    $Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="HGDP" | PopnData
    $Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="1KGP" | PopnData
    $Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="HGDP" | PopnData
```

```
    $Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="1KGP" | PopnData
    $Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="HGDP" | PopnData
    $Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="1KGP" | PopnData
    $Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="HGDP" | PopnData
    $Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="1KGP" | PopnData
    $Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="HGDP" | PopnData
    $Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="1KGP" | PopnData
    $Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="HGDP" | PopnData
    $Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c19" & (PopnData$Population=="1KGP" | PopnData
    $Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))

PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
```

```
    $Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="PaT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="PaT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="PaC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="PaC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="ADT")),] -> dfc
```

```
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="BpT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="BpT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="HGDP" | PopnData
    $Population=="BpC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c20" & (PopnData$Population=="1KGP" | PopnData
    $Population=="BpC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))

PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="HGDP" | PopnData
    $Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="1KGP" | PopnData
    $Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="HGDP" | PopnData
    $Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="1KGP" | PopnData
    $Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="HGDP" | PopnData
    $Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="1KGP" | PopnData
    $Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="HGDP" | PopnData
    $Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="1KGP" | PopnData
    $Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="HGDP" | PopnData
    $Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="1KGP" | PopnData
    $Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="HGDP" | PopnData
    $Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="1KGP" | PopnData
    $Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MyT")),] -> dfc
```

```
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="HGDP" | PopnData$Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="1KGP" | PopnData$Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="HGDP" | PopnData$Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="1KGP" | PopnData$Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="HGDP" | PopnData$Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="1KGP" | PopnData$Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="HGDP" | PopnData$Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="1KGP" | PopnData$Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="HGDP" | PopnData$Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="1KGP" | PopnData$Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="HGDP" | PopnData$Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="1KGP" | PopnData$Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="HGDP" | PopnData$Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="1KGP" | PopnData$Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="HGDP" | PopnData$Population=="BpT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="1KGP" | PopnData$Population=="BpT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="HGDP" | PopnData$Population=="BpC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c22" & (PopnData$Population=="1KGP" | PopnData$Population=="BpC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))

PopnData[which(PopnData$Name=="HERV-K-c23" & (PopnData$Population=="HGDP" | PopnData$Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c23" & (PopnData$Population=="1KGP" | PopnData$Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c23" & (PopnData$Population=="HGDP" | PopnData$Population=="PrC")),] -> dfc
```

```
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c23" & (PopnData$Population=="1KGP" | PopnData
    $Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))

PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="HGDP" | PopnData
    $Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="1KGP" | PopnData
    $Population=="BrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="HGDP" | PopnData
    $Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="1KGP" | PopnData
    $Population=="BrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="HGDP" | PopnData
    $Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="1KGP" | PopnData
    $Population=="CeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="HGDP" | PopnData
    $Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="1KGP" | PopnData
    $Population=="CeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="HGDP" | PopnData
    $Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="1KGP" | PopnData
    $Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="HGDP" | PopnData
    $Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="1KGP" | PopnData
    $Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MyT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="HGDP" | PopnData
    $Population=="MyC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="1KGP" | PopnData
    $Population=="MyC")),] -> dfc
```

```
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="HGDP" | PopnData
    $Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="1KGP" | PopnData
    $Population=="NeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="HGDP" | PopnData
    $Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="1KGP" | PopnData
    $Population=="NeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="HGDP" | PopnData
    $Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="1KGP" | PopnData
    $Population=="PrT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="HGDP" | PopnData
    $Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="1KGP" | PopnData
    $Population=="PrC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="HGDP" | PopnData
    $Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="1KGP" | PopnData
    $Population=="ADT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="HGDP" | PopnData
    $Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="1KGP" | PopnData
    $Population=="ADC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="HGDP" | PopnData
    $Population=="BpT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="1KGP" | PopnData
    $Population=="BpT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="HGDP" | PopnData
    $Population=="BpC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c24" & (PopnData$Population=="1KGP" | PopnData
    $Population=="BpC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))

PopnData[which(PopnData$Name=="HERV-K-c25" & (PopnData$Population=="HGDP" | PopnData
    $Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c25" & (PopnData$Population=="1KGP" | PopnData
    $Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c25" & (PopnData$Population=="HGDP" | PopnData
    $Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c25" & (PopnData$Population=="1KGP" | PopnData
    $Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c27" & (PopnData$Population=="HGDP" | PopnData
    $Population=="HeT")),] -> dfc
```

```
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c27" & (PopnData$Population=="1KGP" | PopnData
    $Population=="HeT")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c27" & (PopnData$Population=="HGDP" | PopnData
    $Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
PopnData[which(PopnData$Name=="HERV-K-c27" & (PopnData$Population=="1KGP" | PopnData
    $Population=="HeC")),] -> dfc
fisher.test(rbind(dfc$Noins, dfc$NoPIS))
```

7. Appendix 2

List of solo LTR elements found in the Human reference genome, originally reported by Subramanian et al. (2011).

Chromosome	Name	GRCh38 / Hg38 position		Type	Reference	TSD
1	HML2.LTR1	1409807	1410773	LTR5_Hs	(Subramanian et al.; 2011)	CTCAC
1	HML2.LTR2	1580343	1581251	LTR5_Hs	(Subramanian et al.; 2011)	TTGCCC
1	HML2.LTR3	3750401	3751390	LTR5B	(Subramanian et al.; 2011)	AATTTC
1	HML2.LTR4	5573601	5574610	LTR5B	(Subramanian et al.; 2011)	ACTCCA
1	HML2.LTR5	9001820	9002808	LTR5B	(Subramanian et al.; 2011)	CCCTGT
1	HML2.LTR6	10425395	10427626	LTR5_Hs	(Subramanian et al.; 2011)	AGTTT
1	HML2.LTR7	11889866	11891158	LTR5B	(Subramanian et al.; 2011)	ACAAAT
1	HML2.LTR8	13113521	13114390	LTR5B	(Subramanian et al.; 2011)	ACCATG
1	HML2.LTR9	15135338	15136296	LTR5_Hs	(Subramanian et al.; 2011)	GCTTTG
1	HML2.LTR10	15379018	15379976	LTR5_Hs	(Subramanian et al.; 2011)	CTTCAG
1	HML2.LTR11	20030028	20031015	LTR5B	(Subramanian et al.; 2011)	ACAAC
1	HML2.LTR12	25587850	25588817	LTR5_Hs	(Subramanian et al.; 2011)	-
1	HML2.LTR13	29338082	29339047	LTR5_Hs	(Subramanian et al.; 2011)	GCCCAC
1	HML2.LTR14	29359333	29360290	LTR5_Hs	(Subramanian et al.; 2011)	CTGTAA
1	HML2.LTR15	37559675	37560664	LTR5B	(Subramanian et al.; 2011)	GCTGG
1	HML2.LTR16	39504134	39505581	LTR5B	(Subramanian et al.; 2011)	AAGTG
1	HML2.LTR17	40471747	40472757	LTR5A	(Subramanian et al.; 2011)	ATGCC
1	HML2.LTR18	40572026	40572994	LTR5_Hs	(Subramanian et al.; 2011)	GCCGG/GCTGG
1	HML2.LTR19	45513511	45514465	LTR5_Hs	(Subramanian et al.; 2011)	TACCAC
1	HML2.LTR20	45528099	45529068	LTR5_Hs	(Subramanian et al.; 2011)	ACTCC/CCATC
1	HML2.LTR21	46320296	46321263	LTR5_Hs	(Subramanian et al.; 2011)	GCTATG
1	HML2.LTR22	46329763	46330515	LTR5_Hs	(Subramanian et al.; 2011)	5' truncated
1	HML2.LTR23	46395864	46396842	LTR5B	(Subramanian et al.; 2011)	ACATG
1	HML2.LTR24	46788569	46789541	LTR5_Hs	(Subramanian et al.; 2011)	GATAA
1	HML2.LTR25	47133175	47134203	LTR5A	(Subramanian et al.; 2011)	TAAAAG
1	HML2.LTR26	52006245	52007211	LTR5_Hs	(Subramanian et al.; 2011)	AGAGCT
1	HML2.LTR27	53168237	53169249	LTR5A	(Subramanian et al.; 2011)	CTTAGG/GTTAGG
1	HML2.LTR28	54636121	54637097	LTR5_Hs	(Subramanian et al.; 2011)	ATTAAC/TTTAAC
1	HML2.LTR29	59914345	59915366	LTR5B	(Subramanian et al.; 2011)	CACCT/TACCT
1	HML2.LTR30	59938159	59939165	LTR5B	(Subramanian et al.; 2011)	TTCAAT
1	HML2.LTR31	62170977	62171967	LTR5B	(Subramanian et al.; 2011)	GGTCT/AGTCT
1	HML2.LTR32	65139796	65140762	LTR5_Hs	(Subramanian et al.; 2011)	CCCGTC/CCCATC
1	HML2.LTR33	66424815	66425774	LTR5_Hs	(Subramanian et al.; 2011)	GAAGG
1	HML2.LTR34	66591568	66592554	LTR5B	(Subramanian et al.; 2011)	CACTC
1	HML2.LTR35	70460425	70461399	LTR5_Hs	(Subramanian et al.; 2011)	GCCCTC
1	HML2.LTR36	73129298	73130265	LTR5_Hs	(Subramanian et al.; 2011)	CATGT
1	HML2.LTR37	77982896	77983837	LTR5_Hs	(Subramanian et al.; 2011)	TTAACC
1	HML2.LTR38	79702341	79703294	LTR5_Hs	(Subramanian et al.; 2011)	TACAAT/TACAGT

1	HML2.LTR39	89066863	89067823	LTR5_Hs	(Subramanian et al.; 2011)	CTTCC
1	HML2.LTR40	93276782	93277749	LTR5_Hs	(Subramanian et al.; 2011)	CAATTA
1	HML2.LTR41	99793555	99794555	LTR5B	(Subramanian et al.; 2011)	AATAAG
1	HML2.LTR42	99835834	99836768	LTR5_Hs	(Subramanian et al.; 2011)	CATGAA
1	HML2.LTR43	104134739	104135712	LTR5_Hs	(Subramanian et al.; 2011)	CGCATG/CGCATA
1	HML2.LTR44	108517445	108518412	LTR5_Hs	(Subramanian et al.; 2011)	CTGGGT
1	HML2.LTR45	109766616	109767614	LTR5B	(Subramanian et al.; 2011)	GGTCCC
1	HML2.LTR46	111359721	111360685	LTR5_Hs	(Subramanian et al.; 2011)	CAAAC
1	HML2.LTR47	111361416	111362383	LTR5B	(Subramanian et al.; 2011)	5' truncated
1	HML2.LTR48	112816778	112817803	LTR5A	(Subramanian et al.; 2011)	AGTAT
1	HML2.LTR49	114058148	114059120	LTR5_Hs	(Subramanian et al.; 2011)	CTGGCT
1	HML2.LTR50	120424751	120425723	LTR5_Hs	(Subramanian et al.; 2011)	CTGAAC
1	HML2.LTR63	144450894	144451866	LTR5_Hs	(Subramanian et al.; 2011)	CTGAAC
1	HML2.LTR52	145414873	145415845	LTR5_Hs	(Subramanian et al.; 2011)	CTGAAC
1	HML2.LTR51	145932414	145933382	LTR5_Hs	(Subramanian et al.; 2011)	CTTCT/CTTTT
1	HML2.LTR54	146907212	146908212	LTR5_Hs	(Subramanian et al.; 2011)	-
1	HML2.LTR55	146948313	146949300	LTR5_Hs	(Subramanian et al.; 2011)	CTGAAC
1	HML2.LTR56	147106117	147107077	LTR5_Hs	(Subramanian et al.; 2011)	AGTC
1	HML2.LTR57	147240100	147241067	LTR5_Hs	(Subramanian et al.; 2011)	CTACCT
1	HML2.LTR58	147709408	147710433	LTR5A	(Subramanian et al.; 2011)	-
1	HML2.LTR59	147733842	147734800	LTR5_Hs	(Subramanian et al.; 2011)	GAAACC
1	HML2.LTR60	148137833	148138805	LTR5_Hs	(Subramanian et al.; 2011)	CTGAAC
1	HML2.LTR61	148188817	148189817	LTR5_Hs	(Subramanian et al.; 2011)	-
1	HML2.LTR62	149092877	149093849	LTR5_Hs	(Subramanian et al.; 2011)	CTGAAC
1	HML2.LTR64	150709190	150710213	LTR5A	(Subramanian et al.; 2011)	GTTAT
1	HML2.LTR65	151872509	151873495	LTR5B	(Subramanian et al.; 2011)	AGTCTC/ACTCTC
1	HML2.LTR66	152455109	152456081	LTR5_Hs	(Subramanian et al.; 2011)	CCTAGC
1	HML2.LTR67	153088297	153089275	LTR5A	(Subramanian et al.; 2011)	3' truncated
1	HML2.LTR68	155599537	155600505	LTR5_Hs	(Subramanian et al.; 2011)	TATGC
1	HML2.LTR69	156179223	156180190	LTR5_Hs	(Subramanian et al.; 2011)	TGGCAC
1	HML2.LTR70	156181461	156182421	LTR5_Hs	(Subramanian et al.; 2011)	AGGTAG
1	HML2.LTR71	159767796	159768763	LTR5_Hs	(Subramanian et al.; 2011)	CATAAG
1	HML2.LTR72	160652139	160653095	LTR5_Hs	(Subramanian et al.; 2011)	3' truncated
1	HML2.LTR73	160913142	160914122	LTR5_Hs	(Subramanian et al.; 2011)	TGAGCC
1	HML2.LTR74	160946904	160947898	LTR5B	(Subramanian et al.; 2011)	CCACG/CCATG
1	HML2.LTR75	160948825	160949791	LTR5_Hs	(Subramanian et al.; 2011)	ATAAT
1	HML2.LTR76	161016835	161017804	LTR5_Hs	(Subramanian et al.; 2011)	CACCAG
1	HML2.LTR77	161315708	161316604	LTR5_Hs	(Subramanian et al.; 2011)	GTGGAC

1	HML2.LTR78	161412410	161413620	LTR5B	(Subramanian et al.; 2011)	5' truncated
1	HML2.LTR79	161433900	161434926	LTR5A	(Subramanian et al.; 2011)	ACTTT
1	HML2.LTR80	165092583	165093558	LTR5B	(Subramanian et al.; 2011)	TCCTGC
1	HML2.LTR81	173474425	173475408	LTR5B	(Subramanian et al.; 2011)	-
1	HML2.LTR82	188751236	188752206	LTR5_Hs	(Subramanian et al.; 2011)	AATATA
1	HML2.LTR83	196108732	196109705	LTR5_Hs	(Subramanian et al.; 2011)	AAGAGG
1	HML2.LTR84	198129684	198130677	LTR5B	(Subramanian et al.; 2011)	CTTTAC
1	HML2.LTR85	209141191	209142173	LTR5_Hs	(Subramanian et al.; 2011)	CTCAAG
1	HML2.LTR86	211222132	211223100	LTR5B	(Subramanian et al.; 2011)	GAGAG
1	HML2.LTR87	219618579	219619569	LTR5B	(Subramanian et al.; 2011)	GAGATG/GAGACG
1	HML2.LTR88	224339831	224340798	LTR5_Hs	(Subramanian et al.; 2011)	AATTAC
1	HML2.LTR89	225226603	225227591	LTR5B	(Subramanian et al.; 2011)	AAATG
1	HML2.LTR90	225965843	225966858	LTR5A	(Subramanian et al.; 2011)	CATAG
1	HML2.LTR91	227606016	227607021	LTR5B	(Subramanian et al.; 2011)	CTTTC
1	HML2.LTR92	227868340	227869308	LTR5_Hs	(Subramanian et al.; 2011)	GTGAGG/ATGAGG
1	HML2.LTR93	234011810	234012819	LTR5B	(Subramanian et al.; 2011)	GTGCT
1	HML2.LTR94	236504498	236505526	LTR5A	(Subramanian et al.; 2011)	CATCTT
1	HML2.LTR95	246082685	246083636	LTR5_Hs	(Subramanian et al.; 2011)	GTTTTG
1	HML2.LTR96	247015783	247016742	LTR5_Hs	(Subramanian et al.; 2011)	ATTTCA/ATTACA
1	HML2.LTR97	247582010	247582998	LTR5B	(Subramanian et al.; 2011)	TTAAC
1	HML2.LTR98	248906915	248907908	LTR5B	(Subramanian et al.; 2011)	CCTAAC/CCCAAC
1	HML2.LTR53	-	-	LTR5_Hs	(Subramanian et al.; 2011)	-
2	HML2.LTR99	94046	95055	LTR5A	(Subramanian et al.; 2011)	GAGAGA
2	HML2.LTR100	702549	703588	LTR5A	(Subramanian et al.; 2011)	ATTTT
2	HML2.LTR101	1560272	1561258	LTR5B	(Subramanian et al.; 2011)	CTGCTG
2	HML2.LTR102	3104478	3105423	LTR5_Hs	(Subramanian et al.; 2011)	GTCTAC/GCCTAC
2	HML2.LTR103	20501550	20502198	LTR5A	(Subramanian et al.; 2011)	3' truncated
2	HML2.LTR104	26749865	26750871	LTR5_Hs	(Subramanian et al.; 2011)	ATTTT
2	HML2.LTR105	27459979	27460946	LTR5_Hs	(Subramanian et al.; 2011)	AGAGG
2	HML2.LTR106	30613472	30614439	LTR5_Hs	(Subramanian et al.; 2011)	CTTCCA
2	HML2.LTR107	32275582	32276549	LTR5_Hs	(Subramanian et al.; 2011)	GGTGAT/GGTGAC
2	HML2.LTR108	37225318	37226285	LTR5_Hs	(Subramanian et al.; 2011)	TCACAG
2	HML2.LTR109	37750592	37751580	LTR5B	(Subramanian et al.; 2011)	CTGCAT
2	HML2.LTR110	38277784	38278760	LTR5_Hs	(Subramanian et al.; 2011)	TCACTT
2	HML2.LTR111	39321158	39322127	LTR5_Hs	(Subramanian et al.; 2011)	ATTTC
2	HML2.LTR112	55279271	55280234	LTR5_Hs	(Subramanian et al.; 2011)	ATTGC
2	HML2.LTR113	73119185	73120136	LTR5B	(Subramanian et al.; 2011)	5' truncated
2	HML2.LTR114	86261299	86262267	LTR5_Hs	(Subramanian et al.; 2011)	TTTCCT
2	HML2.LTR115	88866498	88867492	LTR5B	(Subramanian et al.; 2011)	AGCCAG
2	HML2.LTR116	94744663	94745630	LTR5_Hs	(Subramanian et al.; 2011)	TGTGG

2	HML2.LTR117	95191340	95192369	LTR5A	(Subramanian et al.; 2011)	TGTTG
2	HML2.LTR118	95567179	95568184	LTR5A	(Subramanian et al.; 2011)	CATAAA/CATAAT
2	HML2.LTR119	98129887	98130831	LTR5_Hs	(Subramanian et al.; 2011)	TGGGCC
2	HML2.LTR120	100269773	100270806	LTR5B	(Subramanian et al.; 2011)	CGTTTT
2	HML2.LTR121	100684866	100685837	LTR5_Hs	(Subramanian et al.; 2011)	ATATAA/ATTTAA
2	HML2.LTR122	105425026	105426056	LTR5A	(Subramanian et al.; 2011)	TATAT
2	HML2.LTR123	109636838	109637851	LTR5A	(Subramanian et al.; 2011)	GCCTCT
2	HML2.LTR124	110056070	110057040	LTR5_Hs	(Subramanian et al.; 2011)	TCGTGG
2	HML2.LTR125	110314426	110315396	LTR5_Hs	(Subramanian et al.; 2011)	TCGTGG
2	HML2.LTR126	111962843	111963809	LTR5_Hs	(Subramanian et al.; 2011)	CTGTC
2	HML2.LTR127	112934254	112935221	LTR5_Hs	(Subramanian et al.; 2011)	TTGCT
2	HML2.LTR128	127542575	127543549	LTR5_Hs	(Subramanian et al.; 2011)	TCCTT
2	HML2.LTR129	129701350	129702385	LTR5A	(Subramanian et al.; 2011)	AATGT
2	HML2.LTR130	131721200	131722222	LTR5A	(Subramanian et al.; 2011)	CTGTT/GTGTT
2	HML2.LTR131	132175021	132176042	LTR5B	(Subramanian et al.; 2011)	TATTT
2	HML2.LTR132	134966893	134967857	LTR5A	(Subramanian et al.; 2011)	3' truncated
2	HML2.LTR133	142695091	142696086	LTR5B	(Subramanian et al.; 2011)	CTCTGG
2	HML2.LTR134	152769078	152770072	LTR5B	(Subramanian et al.; 2011)	CAGGC/CAGCC
2	HML2.LTR135	170263304	170264250	LTR5_Hs	(Subramanian et al.; 2011)	AACGAA/AATGAA
2	HML2.LTR136	192541942	192542927	LTR5_Hs	(Subramanian et al.; 2011)	CACATG
2	HML2.LTR137	193142416	193143387	LTR5_Hs	(Subramanian et al.; 2011)	ATCTT
2	HML2.LTR138	194504342	194505296	LTR5_Hs	(Subramanian et al.; 2011)	TTTTTG/TTTTCG
2	HML2.LTR139	201139002	201139967	LTR5_Hs	(Subramanian et al.; 2011)	-
2	HML2.LTR140	207036833	207037793	LTR5_Hs	(Subramanian et al.; 2011)	CTTGTA
2	HML2.LTR141	208256757	208257744	LTR5B	(Subramanian et al.; 2011)	-
2	HML2.LTR142	214802892	214803852	LTR5_Hs	(Subramanian et al.; 2011)	GATCAG
2	HML2.LTR143	218095540	218096528	LTR5B	(Subramanian et al.; 2011)	GGTGTA
2	HML2.LTR144	218311250	218312230	LTR5B	(Subramanian et al.; 2011)	TTTACA/ATTACA
2	HML2.LTR145	223190521	223191443	LTR5_Hs	(Subramanian et al.; 2011)	GATCAG
2	HML2.LTR146	227722205	227723197	LTR5B	(Subramanian et al.; 2011)	-
2	HML2.LTR147	228363642	228364609	LTR5_Hs	(Subramanian et al.; 2011)	ATAGTC
2	HML2.LTR148	230843639	230844596	LTR5_Hs	(Subramanian et al.; 2011)	CAATAG
2	HML2.LTR149	231576784	231577743	LTR5_Hs	(Subramanian et al.; 2011)	GGCTGC
2	HML2.LTR150	232422993	232423965	LTR5_Hs	(Subramanian et al.; 2011)	TGTTCT
2	HML2.LTR151	234168065	234169033	LTR5_Hs	(Subramanian et al.; 2011)	TATTTC
2	HML2.LTR152	240671345	240671994	LTR5A	(Subramanian et al.; 2011)	3' truncated
3	HML2.LTR153	5081627	5082560	LTR5_Hs	(Subramanian et al.; 2011)	GAAGCC
3	HML2.LTR154	7941676	7942697	LTR5A	(Subramanian et al.; 2011)	TCAACC
3	HML2.LTR155	12674115	12675082	LTR5_Hs	(Subramanian et al.; 2011)	-

3	HML2.LTR156	14091185	14092152	LTR5_Hs	(Subramanian et al.; 2011)	TCTTCT
3	HML2.LTR157	14272845	14273811	LTR5_Hs	(Subramanian et al.; 2011)	CTCCC
3	HML2.LTR158	15145529	15146369	LTR5A	(Subramanian et al.; 2011)	5' truncated
3	HML2.LTR159	23544663	23545625	LTR5_Hs	(Subramanian et al.; 2011)	GATCT
3	HML2.LTR160	27531245	27532269	LTR5B	(Subramanian et al.; 2011)	ATGTT
3	HML2.LTR161	36141079	36142094	LTR5A	(Subramanian et al.; 2011)	CATCT/CGTCT
3	HML2.LTR162	38134480	38135067	LTR5B	(Subramanian et al.; 2011)	3' truncated
3	HML2.LTR163	38135237	38136292	LTR5B	(Subramanian et al.; 2011)	GGCAG/GGCAA
3	HML2.LTR164	39418299	39419272	LTR5_Hs	(Subramanian et al.; 2011)	GACCC
3	HML2.LTR165	42811191	42812187	LTR5B	(Subramanian et al.; 2011)	ATTTTC/GTTTTC
3	HML2.LTR166	43258305	43259266	LTR5_Hs	(Subramanian et al.; 2011)	GAGGT
3	HML2.LTR167	45756789	45757724	LTR5B	(Subramanian et al.; 2011)	3' truncated
3	HML2.LTR168	47259534	47260501	LTR5_Hs	(Subramanian et al.; 2011)	GGTAAG
3	HML2.LTR169	50519892	50520848	LTR5_Hs	(Subramanian et al.; 2011)	TCCTG
3	HML2.LTR170	53977514	53978481	LTR5_Hs	(Subramanian et al.; 2011)	AAGATT
3	HML2.LTR171	57343184	57344213	LTR5A	(Subramanian et al.; 2011)	GGAAAT
3	HML2.LTR172	75374163	75375179	LTR5A	(Subramanian et al.; 2011)	AGTGAG
3	HML2.LTR173	75393193	75394152	LTR5_Hs	(Subramanian et al.; 2011)	ATCAG
3	HML2.LTR174	75592672	75593700	LTR5A	(Subramanian et al.; 2011)	AGTGAG
3	HML2.LTR175	75645328	75646291	LTR5_Hs	(Subramanian et al.; 2011)	-
3	HML2.LTR176	98316135	98317116	LTR5B	(Subramanian et al.; 2011)	ACATTG
3	HML2.LTR177	98362555	98363561	LTR5A	(Subramanian et al.; 2011)	-
3	HML2.LTR178	100272494	100273445	LTR5_Hs	(Subramanian et al.; 2011)	GCTTT
3	HML2.LTR179	100336326	100337291	LTR5_Hs	(Subramanian et al.; 2011)	ATTAG
3	HML2.LTR180	100666141	100667097	LTR5_Hs	(Subramanian et al.; 2011)	AGGCGG
3	HML2.LTR181	101703833	101704801	LTR5_Hs	(Subramanian et al.; 2011)	GGACAG
3	HML2.LTR182	110367053	110368048	LTR5B	(Subramanian et al.; 2011)	CTGATA/TTGATA
3	HML2.LTR183	112152009	112153000	LTR5B	(Subramanian et al.; 2011)	AAAGG
3	HML2.LTR184	112564080	112565009	LTR5_Hs	(Subramanian et al.; 2011)	CTCAAT
3	HML2.LTR185	118539850	118540839	LTR5B	(Subramanian et al.; 2011)	GTTTAA
3	HML2.LTR186	121611104	121612102	LTR5A	(Subramanian et al.; 2011)	ACATAT
3	HML2.LTR187	125730453	125731468	LTR5A	(Subramanian et al.; 2011)	AGTGAG
3	HML2.LTR188	125753034	125754061	LTR5A	(Subramanian et al.; 2011)	AGCGAG/AGTGAG
3	HML2.LTR189	125890459	125891262	LTR5_Hs	(Subramanian et al.; 2011)	-
3	HML2.LTR190	127411653	127412622	LTR5_Hs	(Subramanian et al.; 2011)	AGTATC
3	HML2.LTR191	129641252	129642062	LTR5_Hs	(Subramanian et al.; 2011)	TTGCTC
3	HML2.LTR192	130015575	130016587	LTR5A	(Subramanian et al.; 2011)	AGTGAG
3	HML2.LTR193	130038287	130039299	LTR5A	(Subramanian et al.; 2011)	AGTGAG
3	HML2.LTR194	130057289	130058265	LTR5_Hs	(Subramanian et al.; 2011)	ATCAA/ATCAG
3	HML2.LTR195	130123340	130124347	LTR5A	(Subramanian et al.; 2011)	ATTTG

3	HML2.LTR196	132648797	132649792	LTR5B	(Subramanian et al.; 2011)	GGTTTG
3	HML2.LTR197	134515939	134516890	LTR5_Hs	(Subramanian et al.; 2011)	AACAAAG
3	HML2.LTR198	142752045	142753074	LTR5A	(Subramanian et al.; 2011)	CTTAAT
3	HML2.LTR199	145709249	145710214	LTR5_Hs	(Subramanian et al.; 2011)	GTATGT
3	HML2.LTR200	146522264	146523231	LTR5_Hs	(Subramanian et al.; 2011)	TTATGC/TTATGT
3	HML2.LTR201	175905550	175906517	LTR5_Hs	(Subramanian et al.; 2011)	ATTAAT
3	HML2.LTR202	183169981	183171007	LTR5A	(Subramanian et al.; 2011)	ATTGA
3	HML2.LTR203	183510924	183511927	LTR5B	(Subramanian et al.; 2011)	GCATTG
3	HML2.LTR204	183520353	183521348	LTR5B	(Subramanian et al.; 2011)	-
3	HML2.LTR205	186865611	186866486	LTR5_Hs	(Subramanian et al.; 2011)	3' truncated
3	HML2.LTR206	186886874	186887834	LTR5_Hs	(Subramanian et al.; 2011)	ACTACC
3	HML2.LTR207	186893220	186894186	LTR5_Hs	(Subramanian et al.; 2011)	AGAAGA
3	HML2.LTR208	189251371	189252338	LTR5_Hs	(Subramanian et al.; 2011)	AAGAG
3	HML2.LTR209	195728099	195729092	LTR5A	(Subramanian et al.; 2011)	CTTTCC
3	HML2.LTR210	195927525	195928492	LTR5_Hs	(Subramanian et al.; 2011)	TCAGG
3	HML2.LTR211	196944080	196945028	LTR5_Hs	(Subramanian et al.; 2011)	CTATCC/CTACCC
3	HML2.LTR212	196962041	196963050	LTR5B	(Subramanian et al.; 2011)	GTCTG/GTCTC
4	HML2.LTR213	19023	20027	LTR5B	(Subramanian et al.; 2011)	AACGT/AACAT
4	HML2.LTR214	40541	41493	LTR5B	(Subramanian et al.; 2011)	5' truncated
4	HML2.LTR215	151742	152709	LTR5_Hs	(Subramanian et al.; 2011)	TTGGTT
4	HML2.LTR216	207800	208766	LTR5_Hs	(Subramanian et al.; 2011)	GTATAA
4	HML2.LTR217	2021350	2022342	LTR5B	(Subramanian et al.; 2011)	ATCTC
4	HML2.LTR218	3883351	3884360	LTR5A	(Subramanian et al.; 2011)	AGTGAG
4	HML2.LTR219	3905321	3906335	LTR5A	(Subramanian et al.; 2011)	AGTGAG
4	HML2.LTR220	3925643	3926609	LTR5_Hs	(Subramanian et al.; 2011)	ATCAG/GTCAG
4	HML2.LTR221	4120309	4121338	LTR5A	(Subramanian et al.; 2011)	AGTGAG
4	HML2.LTR222	4150510	4151525	LTR5A	(Subramanian et al.; 2011)	AGTGAG
4	HML2.LTR223	8955580	8956594	LTR5A	(Subramanian et al.; 2011)	AGTGAG
4	HML2.LTR224	8987402	8988426	LTR5A	(Subramanian et al.; 2011)	AGTGAG
4	HML2.LTR225	9176783	9177814	LTR5A	(Subramanian et al.; 2011)	CTTGGA
4	HML2.LTR226	9465209	9466220	LTR5A	(Subramanian et al.; 2011)	AGTGAG
4	HML2.LTR227	9489688	9490702	LTR5A	(Subramanian et al.; 2011)	AGTGAG
4	HML2.LTR228	9518857	9519886	LTR5A	(Subramanian et al.; 2011)	AGTGAG
4	HML2.LTR229	9731774	9732691	LTR5_Hs	(Subramanian et al.; 2011)	5' truncated
4	HML2.LTR230	9751053	9752065	LTR5A	(Subramanian et al.; 2011)	AGTGAG
4	HML2.LTR231	10127377	10128304	LTR5_Hs	(Subramanian et al.; 2011)	AATTT/AATCT
4	HML2.LTR232	15878547	15879554	LTR5B	(Subramanian et al.; 2011)	GGAAAC/GAAAAC
4	HML2.LTR233	16076658	16077632	LTR5_Hs	(Subramanian et al.; 2011)	-
4	HML2.LTR234	22597763	22598736	LTR5_Hs	(Subramanian et al.; 2011)	CCTGTG
4	HML2.LTR235	30232903	30233913	LTR5A	(Subramanian et al.; 2011)	GTTGTC
4	HML2.LTR236	43001634	43002661	LTR5A	(Subramanian et al.; 2011)	-

4	HML2.LTR237	48285356	48286331	LTR5_Hs	(Subramanian et al.; 2011)	GAGATG
4	HML2.LTR238	49029102	49030097	LTR5B	(Subramanian et al.; 2011)	TGGGT
4	HML2.LTR239	62940878	62941845	LTR5_Hs	(Subramanian et al.; 2011)	TCTGTC
4	HML2.LTR240	64355878	64356868	LTR5B	(Subramanian et al.; 2011)	ACAAT
4	HML2.LTR241	66358757	66359723	LTR5_Hs	(Subramanian et al.; 2011)	AATATG
4	HML2.LTR242	68281467	68282426	LTR5_Hs	(Subramanian et al.; 2011)	-
4	HML2.LTR243	68528239	68529271	LTR5A	(Subramanian et al.; 2011)	CTTGCATA/CTTGCTAA
4	HML2.LTR244	68693467	68694431	LTR5_Hs	(Subramanian et al.; 2011)	GTGTCT/GTATCT
4	HML2.LTR245	72129224	72130191	LTR5_Hs	(Subramanian et al.; 2011)	CATAAT
4	HML2.LTR246	75286596	75287589	LTR5B	(Subramanian et al.; 2011)	GAAATG/GAAAGG
4	HML2.LTR247	87746312	87747305	LTR5B	(Subramanian et al.; 2011)	TCCCCC
4	HML2.LTR248	96524106	96524865	LTR5B	(Subramanian et al.; 2011)	5' truncated
4	HML2.LTR249	104170967	104171994	LTR5A	(Subramanian et al.; 2011)	-
4	HML2.LTR250	107728699	107729703	LTR5B	(Subramanian et al.; 2011)	GATGG/GATCG
4	HML2.LTR251	117183120	117184080	LTR5_Hs	(Subramanian et al.; 2011)	TGTAGC
4	HML2.LTR252	119342534	119343499	LTR5_Hs	(Subramanian et al.; 2011)	GGTAG
4	HML2.LTR253	119396366	119397332	LTR5_Hs	(Subramanian et al.; 2011)	TAAGA
4	HML2.LTR254	121396554	121397554	LTR5A	(Subramanian et al.; 2011)	TTGTG/ATGTG
4	HML2.LTR255	121446172	121447044	LTR5A	(Subramanian et al.; 2011)	3' truncated
4	HML2.LTR256	127439477	127440477	LTR5A	(Subramanian et al.; 2011)	CAGTGA
4	HML2.LTR257	134045987	134046985	LTR5B	(Subramanian et al.; 2011)	GTTTTT
4	HML2.LTR258	135538048	135539039	LTR5B	(Subramanian et al.; 2011)	ATACTG/GTACTG
4	HML2.LTR259	156304762	156305718	LTR5_Hs	(Subramanian et al.; 2011)	TCTTTT
4	HML2.LTR260	156386354	156387371	LTR5A	(Subramanian et al.; 2011)	TTACC/TTACT
4	HML2.LTR261	159976488	159977451	LTR5_Hs	(Subramanian et al.; 2011)	ATATCT
4	HML2.LTR262	162418478	162419472	LTR5B	(Subramanian et al.; 2011)	CCCTT
4	HML2.LTR263	164905514	164906483	LTR5_Hs	(Subramanian et al.; 2011)	AGTTG
4	HML2.LTR264	164975370	164976382	LTR5A	(Subramanian et al.; 2011)	CAAAG/TAAAG
4	HML2.LTR265	164990383	164991350	LTR5_Hs	(Subramanian et al.; 2011)	GAATGG
4	HML2.LTR266	165073132	165074123	LTR5B	(Subramanian et al.; 2011)	ATCAAC
4	HML2.LTR267	166093063	166094086	LTR5B	(Subramanian et al.; 2011)	CATGAC
4	HML2.LTR268	174380062	174380987	LTR5_Hs	(Subramanian et al.; 2011)	GTTTCC
4	HML2.LTR269	174383029	174384002	LTR5_Hs	(Subramanian et al.; 2011)	GAAGAA/GAAGAG
4	HML2.LTR270	174400760	174401732	LTR5_Hs	(Subramanian et al.; 2011)	TACCAG
4	HML2.LTR271	179553172	179554140	LTR5_Hs	(Subramanian et al.; 2011)	ACCTGG
4	HML2.LTR272	183726268	183727281	LTR5A	(Subramanian et al.; 2011)	GTGTGT
4	HML2.LTR273	186050057	186051007	LTR5_Hs	(Subramanian et al.; 2011)	ATCCTG
5	HML2.LTR274	196784	197781	LTR5A	(Subramanian et al.; 2011)	CAGCGG
5	HML2.LTR275	1567392	1568346	LTR5B	(Subramanian et al.; 2011)	-

5	HML2.LTR276	1595977	1596944	LTR5_Hs	(Subramanian et al.; 2011)	TGTGG
5	HML2.LTR277	4924928	4925896	LTR5_Hs	(Subramanian et al.; 2011)	GAGTTG
5	HML2.LTR278	8937742	8938708	LTR5_Hs	(Subramanian et al.; 2011)	CTAAAT
5	HML2.LTR279	18578454	18579422	LTR5_Hs	(Subramanian et al.; 2011)	CCAAGT
5	HML2.LTR280	24465739	24466778	LTR5A	(Subramanian et al.; 2011)	TTTTC/TTTCC
5	HML2.LTR281	26197418	26198424	LTR5A	(Subramanian et al.; 2011)	-
5	HML2.LTR282	35176385	35177353	LTR5_Hs	(Subramanian et al.; 2011)	CAGTTC
5	HML2.LTR283	37671897	37672890	LTR5B	(Subramanian et al.; 2011)	-
5	HML2.LTR284	39828054	39829084	LTR5A	(Subramanian et al.; 2011)	TTCAAA
5	HML2.LTR285	43058135	43059111	LTR5B	(Subramanian et al.; 2011)	AAGAC
5	HML2.LTR286	43077022	43078006	LTR5B	(Subramanian et al.; 2011)	AGCAT/AGTAT
5	HML2.LTR287	43580448	43581389	LTR5_Hs	(Subramanian et al.; 2011)	GTTAT
5	HML2.LTR288	44730487	44731454	LTR5_Hs	(Subramanian et al.; 2011)	CAGACT
5	HML2.LTR289	55571080	55572039	LTR5_Hs	(Subramanian et al.; 2011)	GGTACT
5	HML2.LTR290	56156997	56157967	LTR5_Hs	(Subramanian et al.; 2011)	CTCAC
5	HML2.LTR291	57546637	57547664	LTR5A	(Subramanian et al.; 2011)	TGTCTC
5	HML2.LTR292	59463786	59464754	LTR5_Hs	(Subramanian et al.; 2011)	CTCGGG
5	HML2.LTR293	75605833	75606792	LTR5_Hs	(Subramanian et al.; 2011)	CTTTGC
5	HML2.LTR294	76944959	76945950	LTR5B	(Subramanian et al.; 2011)	GGAGT/GGAAT
5	HML2.LTR295	78909366	78910363	LTR5B	(Subramanian et al.; 2011)	TTAAGT/TAAAGT
5	HML2.LTR296	80111147	80112653	LTR5B	(Subramanian et al.; 2011)	TATGA/TGTGA
5	HML2.LTR297	95811015	95811999	LTR5B	(Subramanian et al.; 2011)	CATGT
5	HML2.LTR298	99328393	99329417	LTR5A	(Subramanian et al.; 2011)	CAGTC
5	HML2.LTR299	100606039	100607048	LTR5A	(Subramanian et al.; 2011)	-
5	HML2.LTR300	100983049	100984020	LTR5_Hs	(Subramanian et al.; 2011)	ACAGA/ATAGA
5	HML2.LTR301	106259169	106260131	LTR5_Hs	(Subramanian et al.; 2011)	AAAAAGA/AAAGAGA
5	HML2.LTR302	115346574	115347566	LTR5B	(Subramanian et al.; 2011)	ATAAG/ATAGG
5	HML2.LTR303	115448387	115449373	LTR5B	(Subramanian et al.; 2011)	CATTG
5	HML2.LTR304	116821139	116822098	LTR5_Hs	(Subramanian et al.; 2011)	TTTTC
5	HML2.LTR305	120194839	120195651	LTR5_Hs	(Subramanian et al.; 2011)	ATTGTG
5	HML2.LTR306	149942473	149943464	LTR5B	(Subramanian et al.; 2011)	GTGCAC/GTTCAC
5	HML2.LTR307	150453761	150454717	LTR5_Hs	(Subramanian et al.; 2011)	CCACCT
5	HML2.LTR308	150990480	150991473	LTR5B	(Subramanian et al.; 2011)	CTGTAT/CTGTGT
5	HML2.LTR309	151426155	151427141	LTR5B	(Subramanian et al.; 2011)	GATAC/GACAC
5	HML2.LTR310	152797290	152798315	LTR5A	(Subramanian et al.; 2011)	ACTGC
5	HML2.LTR311	157034002	157034989	LTR5B	(Subramanian et al.; 2011)	AGCTC
5	HML2.LTR312	157685958	157686925	LTR5B	(Subramanian et al.; 2011)	CCTGGG
5	HML2.LTR313	162370106	162371124	LTR5A	(Subramanian et al.; 2011)	CTCAGG
5	HML2.LTR314	169996875	169997834	LTR5_Hs	(Subramanian et al.; 2011)	GCAAG
5	HML2.LTR315	178798076	178799040	LTR5_Hs	(Subramanian et al.; 2011)	CAGAAG/CTGAAG
5	HML2.LTR316	179513963	179514930	LTR5_Hs	(Subramanian et al.; 2011)	GATAAA/GATACA

5	HML2.LTR317	180689552	180690566	LTR5A	(Subramanian et al.; 2011)	-
5	HML2.LTR318	180827269	180828232	LTR5_Hs	(Subramanian et al.; 2011)	ACGTGC
5	HML2.LTR319	181266862	181267826	LTR5_Hs	(Subramanian et al.; 2011)	CCAGTC
6	HML2.LTR320	2908663	2909630	LTR5_Hs	(Subramanian et al.; 2011)	ACCTGG
6	HML2.LTR321	2925855	2926704	LTR5B	(Subramanian et al.; 2011)	AATTGT
6	HML2.LTR322	10583934	10584906	LTR5B	(Subramanian et al.; 2011)	-
6	HML2.LTR323	18226766	18227696	LTR5_Hs	(Subramanian et al.; 2011)	ACAGAG
6	HML2.LTR324	24400047	24401019	LTR5_Hs	(Subramanian et al.; 2011)	GATAA
6	HML2.LTR325	24661818	24662791	LTR5B	(Subramanian et al.; 2011)	TCTTAC
6	HML2.LTR326	25748483	25749478	LTR5B	(Subramanian et al.; 2011)	CTCCT
6	HML2.LTR327	26641492	26642501	LTR5B	(Subramanian et al.; 2011)	CAAATA/GAAATA
6	HML2.LTR328	26744675	26745634	LTR5_Hs	(Subramanian et al.; 2011)	CAATC/CGATC
6	HML2.LTR329	27596721	27597701	LTR5B	(Subramanian et al.; 2011)	ATAAG
6	HML2.LTR330	27774460	27775428	LTR5_Hs	(Subramanian et al.; 2011)	TTCTC
6	HML2.LTR331	32657126	32658083	LTR5_Hs	(Subramanian et al.; 2011)	GTAGC/TGTGGC
6	HML2.LTR332	32778068	32779035	LTR5_Hs	(Subramanian et al.; 2011)	ACCAC
6	HML2.LTR333	33121938	33122934	LTR5B	(Subramanian et al.; 2011)	-
6	HML2.LTR334	33809960	33810919	LTR5_Hs	(Subramanian et al.; 2011)	CCCAGA
6	HML2.LTR335	34719460	34720428	LTR5_Hs	(Subramanian et al.; 2011)	GTGGT
6	HML2.LTR336	43239491	43240525	LTR5A	(Subramanian et al.; 2011)	TGTTT
6	HML2.LTR337	44328048	44329009	LTR5_Hs	(Subramanian et al.; 2011)	CTCTGA
6	HML2.LTR338	52748279	52749305	LTR5A	(Subramanian et al.; 2011)	CCTCT
6	HML2.LTR339	52761830	52762798	LTR5_Hs	(Subramanian et al.; 2011)	GGGATG
6	HML2.LTR340	52883249	52884264	LTR5A	(Subramanian et al.; 2011)	CCTGGG
6	HML2.LTR341	52922680	52923640	LTR5_Hs	(Subramanian et al.; 2011)	TCATTG
6	HML2.LTR342	52947698	52948713	LTR5A	(Subramanian et al.; 2011)	-
6	HML2.LTR343	75704079	75705095	LTR5B	(Subramanian et al.; 2011)	GGAATG/GGACTG
6	HML2.LTR344	78858788	78859755	LTR5_Hs	(Subramanian et al.; 2011)	AGTAG
6	HML2.LTR345	80634674	80635670	LTR5A	(Subramanian et al.; 2011)	AGCTG/AGTGT
6	HML2.LTR346	88381588	88382555	LTR5_Hs	(Subramanian et al.; 2011)	GAAGA
6	HML2.LTR347	89890821	89891765	LTR5_Hs	(Subramanian et al.; 2011)	TTCTTT/CTCTTT
6	HML2.LTR348	93173365	93174330	LTR5_Hs	(Subramanian et al.; 2011)	CCCCAG
6	HML2.LTR349	95560627	95561589	LTR5_Hs	(Subramanian et al.; 2011)	-
6	HML2.LTR350	99429371	99430327	LTR5_Hs	(Subramanian et al.; 2011)	AAACT/AATCT
6	HML2.LTR351	111255097	111256055	LTR5_Hs	(Subramanian et al.; 2011)	TTTCTC
6	HML2.LTR352	120931438	120932413	LTR5_Hs	(Subramanian et al.; 2011)	CCAGAT
6	HML2.LTR353	125775847	125776815	LTR5_Hs	(Subramanian et al.; 2011)	CCAAGG
6	HML2.LTR354	131478658	131479644	LTR5B	(Subramanian et al.; 2011)	TCTGC/TCTAC
6	HML2.LTR355	132720806	132721774	LTR5B	(Subramanian et al.; 2011)	TCCCTG
6	HML2.LTR356	134659110	134660077	LTR5_Hs	(Subramanian et al.; 2011)	CTGCTT
6	HML2.LTR357	137844469	137845341	LTR5_Hs	(Subramanian et al.; 2011)	GAGGAG
6	HML2.LTR358	141096209	141097236	LTR5A	(Subramanian et al.; 2011)	CATTG

6	HML2.LTR359	151455210	151456185	LTR5_Hs	(Subramanian et al.; 2011)	ATAACA
6	HML2.LTR360	157752214	157753181	LTR5_Hs	(Subramanian et al.; 2011)	ATCCTG
6	HML2.LTR361	159794737	159795695	LTR5_Hs	(Subramanian et al.; 2011)	TAACT
6	HML2.LTR362	160082911	160083941	LTR5A	(Subramanian et al.; 2011)	GGGGAG
6	HML2.LTR363	165500278	165501246	LTR5_Hs	(Subramanian et al.; 2011)	TGGCAT
6	HML2.LTR364	169346285	169347246	LTR5_Hs	(Subramanian et al.; 2011)	CACTGG
7	HML2.LTR365	2389407	2390372	LTR5_Hs	(Subramanian et al.; 2011)	CTTTC
7	HML2.LTR366	6860235	6861249	LTR5A	(Subramanian et al.; 2011)	AGTGAG
7	HML2.LTR367	6883375	6884389	LTR5A	(Subramanian et al.; 2011)	AGTGAG
7	HML2.LTR368	6901997	6902963	LTR5_Hs	(Subramanian et al.; 2011)	ATCAA/ATCAG
7	HML2.LTR369	7001777	7002700	LTR5_Hs	(Subramanian et al.; 2011)	CTTGGA
7	HML2.LTR370	16197722	16198689	LTR5_Hs	(Subramanian et al.; 2011)	-
7	HML2.LTR371	23039856	23040823	LTR5_Hs	(Subramanian et al.; 2011)	GTCTCA
7	HML2.LTR372	23757207	23758179	LTR5_Hs	(Subramanian et al.; 2011)	GGTAG/GGAAG
7	HML2.LTR373	27741660	27742621	LTR5_Hs	(Subramanian et al.; 2011)	GTATG/GTGTG
7	HML2.LTR374	32513608	32514421	LTR5B	(Subramanian et al.; 2011)	5' truncated
7	HML2.LTR375	38230844	38231857	LTR5A	(Subramanian et al.; 2011)	GAAGT
7	HML2.LTR376	47989679	47990639	LTR5_Hs	(Subramanian et al.; 2011)	TATGTA
7	HML2.LTR377	54675201	54676133	LTR5_Hs	(Subramanian et al.; 2011)	CAATCC
7	HML2.LTR378	55942892	55943903	LTR5A	(Subramanian et al.; 2011)	AGTTCT
7	HML2.LTR379	62619212	62620211	LTR5A	(Subramanian et al.; 2011)	GTTTGT
7	HML2.LTR380	65243034	65243993	LTR5_Hs	(Subramanian et al.; 2011)	GTGCAG
7	HML2.LTR381	65260267	65261141	LTR5_Hs	(Subramanian et al.; 2011)	3' truncated
7	HML2.LTR382	66618955	66619926	LTR5_Hs	(Subramanian et al.; 2011)	-
7	HML2.LTR383	68103753	68104765	LTR5A	(Subramanian et al.; 2011)	TTGTC
7	HML2.LTR384	72951214	72952196	LTR5B	(Subramanian et al.; 2011)	CAATCC/TAATCC
7	HML2.LTR385	73749692	73750685	LTR5B	(Subramanian et al.; 2011)	GCATTG/GCACTG
7	HML2.LTR386	75412888	75413870	LTR5B	(Subramanian et al.; 2011)	CAATCC/TAATCC
7	HML2.LTR387	79750825	79751819	LTR5B	(Subramanian et al.; 2011)	ATATG
7	HML2.LTR388	97941017	97942046	LTR5A	(Subramanian et al.; 2011)	AGTGAG
7	HML2.LTR389	99431755	99432769	LTR5B	(Subramanian et al.; 2011)	-
7	HML2.LTR390	101149854	101150821	LTR5_Hs	(Subramanian et al.; 2011)	CCTGGT
7	HML2.LTR391	102838511	102839506	LTR5B	(Subramanian et al.; 2011)	CAGTGG
7	HML2.LTR392	105148640	105149665	LTR5A	(Subramanian et al.; 2011)	ATCCAA
7	HML2.LTR393	113341518	113342477	LTR5_Hs	(Subramanian et al.; 2011)	GACATC
7	HML2.LTR394	119100247	119101210	LTR5_Hs	(Subramanian et al.; 2011)	ATGTG
7	HML2.LTR395	123780637	123781604	LTR5_Hs	(Subramanian et al.; 2011)	CACATA/CACGTA
7	HML2.LTR396	125221170	125222137	LTR5_Hs	(Subramanian et al.; 2011)	TAAAG
7	HML2.LTR397	125939706	125940692	LTR5B	(Subramanian et al.; 2011)	CATTA/CTTTA
7	HML2.LTR398	126168160	126169127	LTR5_Hs	(Subramanian et al.; 2011)	ATTTT
7	HML2.LTR399	139467405	139468373	LTR5_Hs	(Subramanian et al.; 2011)	ATTCCT

7	HML2.LTR400	140552984	140553942	LTR5_Hs	(Subramanian et al.; 2011)	CCCCT
7	HML2.LTR401	143996126	143997139	LTR5B	(Subramanian et al.; 2011)	CCGCCC/CCCCCC
7	HML2.LTR402	151027014	151027969	LTR5_Hs	(Subramanian et al.; 2011)	CGTGAC
7	HML2.LTR403	158236791	158237758	LTR5_Hs	(Subramanian et al.; 2011)	CTCTGC
8	HML2.LTR404	252774	253782	LTR5B	(Subramanian et al.; 2011)	AACAT
8	HML2.LTR405	7079944	7080947	LTR5A	(Subramanian et al.; 2011)	AGTGAG
8	HML2.LTR406	7240654	7241669	LTR5A	(Subramanian et al.; 2011)	AGTGAG
8	HML2.LTR407	7260529	7261560	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR408	7268151	7269182	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR409	7275773	7276804	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR410	7283395	7284426	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR411	7291017	7292048	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR412	7298637	7299668	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR413	7547398	7548440	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR414	7555053	7556085	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR415	7562703	7563735	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR416	7570349	7571380	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR417	7577998	7579029	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR418	7597900	7598915	LTR5A	(Subramanian et al.; 2011)	AGTGAG
8	HML2.LTR419	7699198	7700213	LTR5A	(Subramanian et al.; 2011)	AGTGAG
8	HML2.LTR420	7719068	7720099	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR421	7726716	7727747	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR422	7734364	7735395	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR423	7742011	7743042	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR424	7749659	7750690	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR425	7757307	7758338	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR426	7764955	7765987	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR427	7772603	7773644	LTR5A	(Subramanian et al.; 2011)	CGTGGA/CTTGGA
8	HML2.LTR428	8010698	8011729	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR429	8018342	8019373	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR430	8025988	8027021	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR431	8045897	8046912	LTR5A	(Subramanian et al.; 2011)	AGTGAG
8	HML2.LTR432	12027614	12028643	LTR5A	(Subramanian et al.; 2011)	AGTGAA/AGTGAG
8	HML2.LTR433	12171126	12172146	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR434	12413445	12414465	LTR5A	(Subramanian et al.; 2011)	CTTGGA
8	HML2.LTR435	12678408	12679433	LTR5A	(Subramanian et al.; 2011)	AGTGAA
8	HML2.LTR436	18793949	18794916	LTR5_Hs	(Subramanian et al.; 2011)	TGAAGC
8	HML2.LTR437	37193368	37194335	LTR5_Hs	(Subramanian et al.; 2011)	AAAATC
8	HML2.LTR438	39646983	39647993	LTR5B	(Subramanian et al.; 2011)	CATTT
8	HML2.LTR439	39650794	39651758	LTR5_Hs	(Subramanian et al.; 2011)	AAATTT/AAATTG
8	HML2.LTR440	42795694	42796663	LTR5_Hs	(Subramanian et al.; 2011)	CCCACC
8	HML2.LTR441	43739539	43740506	LTR5_Hs	(Subramanian et al.; 2011)	TTTGTA
8	HML2.LTR442	46430976	46432004	LTR5A	(Subramanian et al.; 2011)	CTTCTG

8	HML2.LTR443	46439919	46440931	LTR5A	(Subramanian et al.; 2011)	CCAC
8	HML2.LTR444	46455508	46456524	LTR5A	(Subramanian et al.; 2011)	TGGATG
8	HML2.LTR445	47176033	47176992	LTR5_Hs	(Subramanian et al.; 2011)	TTCTC
8	HML2.LTR446	54031681	54032648	LTR5_Hs	(Subramanian et al.; 2011)	CACAC
8	HML2.LTR447	57199603	57200564	LTR5_Hs	(Subramanian et al.; 2011)	CAAATG
8	HML2.LTR448	58671739	58672704	LTR5_Hs	(Subramanian et al.; 2011)	CTAGG
8	HML2.LTR449	66313056	66314079	LTR5A	(Subramanian et al.; 2011)	GTGGAC
8	HML2.LTR450	78438767	78439733	LTR5_Hs	(Subramanian et al.; 2011)	ATATA
8	HML2.LTR451	90684032	90684999	LTR5_Hs	(Subramanian et al.; 2011)	GTGACC
8	HML2.LTR452	90685001	90685846	LTR5_Hs	(Subramanian et al.; 2011)	5' truncated
8	HML2.LTR453	112772174	112773147	LTR5_Hs	(Subramanian et al.; 2011)	AAAAC
8	HML2.LTR454	119917332	119918260	LTR5_Hs	(Subramanian et al.; 2011)	CTTCCT
8	HML2.LTR455	137414346	137415312	LTR5_Hs	(Subramanian et al.; 2011)	CATAAA/TATAAA
8	HML2.LTR456	139048898	139049879	LTR5B	(Subramanian et al.; 2011)	3' truncated
8	HML2.LTR457	139050125	139051123	LTR5B	(Subramanian et al.; 2011)	-
8	HML2.LTR458	142828843	142829804	LTR5_Hs	(Subramanian et al.; 2011)	AAATG
8	HML2.LTR459	142968680	142969680	LTR5B	(Subramanian et al.; 2011)	AAAATT
8	HML2.LTR460	143000773	143001759	LTR5B	(Subramanian et al.; 2011)	CCCACT/CCCATT
8	HML2.LTR461	143067466	143068456	LTR5B	(Subramanian et al.; 2011)	3' truncated
8	HML2.LTR462	143287290	143288251	LTR5_Hs	(Subramanian et al.; 2011)	AGACC/AGACT
8	HML2.LTR463	143831989	143832957	LTR5_Hs	(Subramanian et al.; 2011)	AAGAC
8	HML2.LTR464	143849709	143850662	LTR5_Hs	(Subramanian et al.; 2011)	GCAGTG
8	HML2.LTR465	144145715	144146714	LTR5A	(Subramanian et al.; 2011)	TTAGG
8	HML2.LTR467	144410231	144411259	LTR5B	(Subramanian et al.; 2011)	AAAAC/AAAAT
8	HML2.LTR468	144795894	144796861	LTR5_Hs	(Subramanian et al.; 2011)	CCTGTG/CATGTG
8	HML2.LTR469	144949447	144950446	LTR5B	(Subramanian et al.; 2011)	CCCCA/CCCCG
8	HML2.LTR466	-	-	LTR5A	(Subramanian et al.; 2011)	TTAGG
9	HML2.LTR470	5082036	5082999	LTR5B	(Subramanian et al.; 2011)	TTCAGT
9	HML2.LTR471	11894520	11895489	LTR5_Hs	(Subramanian et al.; 2011)	TAAAC/TAAAT
9	HML2.LTR472	17445361	17446328	LTR5_Hs	(Subramanian et al.; 2011)	GACCC
9	HML2.LTR473	26715150	26716178	LTR5_Hs	(Subramanian et al.; 2011)	CCTGG
9	HML2.LTR474	28880205	28881214	LTR5A	(Subramanian et al.; 2011)	CCAGG/CCGGG
9	HML2.LTR475	31632879	31633846	LTR5_Hs	(Subramanian et al.; 2011)	AAAGTT
9	HML2.LTR476	32950333	32951317	LTR5B	(Subramanian et al.; 2011)	GTTATG
9	HML2.LTR477	33514870	33515836	LTR5_Hs	(Subramanian et al.; 2011)	TAGAA/CAGAA
9	HML2.LTR478	40338734	40339708	LTR5_Hs	(Subramanian et al.; 2011)	TGTGG
9	HML2.LTR482	40781491	40782501	LTR5B	(Subramanian et al.; 2011)	TATTT
9	HML2.LTR484	41105161	41106192	LTR5A	(Subramanian et al.; 2011)	TCCACC
9	HML2.LTR480	42965007	42965981	LTR5_Hs	(Subramanian et al.; 2011)	TGTGG
9	HML2.LTR483	63681609	63682568	LTR5_Hs	(Subramanian et al.; 2011)	CCACTG
9	HML2.LTR481	64478771	64479744	LTR5_Hs	(Subramanian et al.; 2011)	TGTGG

9	HML2.LTR486	64902286	64903296	LTR5B	(Subramanian et al.; 2011)	TATTT
9	HML2.LTR479	66036608	66037582	LTR5_Hs	(Subramanian et al.; 2011)	TGTGG
9	HML2.LTR487	68325230	68326261	LTR5A	(Subramanian et al.; 2011)	TCCACC
9	HML2.LTR488	69763137	69764104	LTR5_Hs	(Subramanian et al.; 2011)	TTTTCA
9	HML2.LTR489	73087207	73088175	LTR5_Hs	(Subramanian et al.; 2011)	ATGCA
9	HML2.LTR490	77303423	77304452	LTR5A	(Subramanian et al.; 2011)	CTTCAT
9	HML2.LTR491	84367444	84368439	LTR5B	(Subramanian et al.; 2011)	ATCAG/ATCAA
9	HML2.LTR492	87784943	87785932	LTR5B	(Subramanian et al.; 2011)	AACCC
9	HML2.LTR493	90166243	90167269	LTR5B	(Subramanian et al.; 2011)	GTCCCA
9	HML2.LTR494	97232136	97233105	LTR5_Hs	(Subramanian et al.; 2011)	TAGAA
9	HML2.LTR495	108594613	108595572	LTR5_Hs	(Subramanian et al.; 2011)	CTCAG
9	HML2.LTR496	110826251	110827218	LTR5_Hs	(Subramanian et al.; 2011)	TACTGC
9	HML2.LTR497	111875303	111876269	LTR5_Hs	(Subramanian et al.; 2011)	CTTTTC/CTTTCC
9	HML2.LTR498	113379060	113380053	LTR5B	(Subramanian et al.; 2011)	CGTGTA/CATGTA
9	HML2.LTR499	121429106	121430073	LTR5_Hs	(Subramanian et al.; 2011)	TAAAAA
9	HML2.LTR500	122447929	122448924	LTR5B	(Subramanian et al.; 2011)	TGGG/TCGG
9	HML2.LTR501	126716512	126717527	LTR5A	(Subramanian et al.; 2011)	GTTTAG
9	HML2.LTR502	131356719	131357676	LTR5_Hs	(Subramanian et al.; 2011)	GGTT
9	HML2.LTR503	133283398	133284394	LTR5B	(Subramanian et al.; 2011)	GCGGGT/GCAGGT
9	HML2.LTR504	133957780	133958741	LTR5_Hs	(Subramanian et al.; 2011)	GTGGAG
9	HML2.LTR505	134092436	134093404	LTR5_Hs	(Subramanian et al.; 2011)	ACTGC/ACTAC
9	HML2.LTR506	134714933	134715895	LTR5_Hs	(Subramanian et al.; 2011)	AGAAT
9	HML2.LTR507	136796533	136797817	LTR5B	(Subramanian et al.; 2011)	CTTTT
9	HML2.LTR508	137116360	137117229	LTR5B	(Subramanian et al.; 2011)	-
9	HML2.LTR509	138132949	138133926	LTR5B	(Subramanian et al.; 2011)	-
9	HML2.LTR485	-	-	LTR5_Hs	(Subramanian et al.; 2011)	TGTGG
10	HML2.LTR510	3061469	3062440	LTR5_Hs	(Subramanian et al.; 2011)	ATTCAC
10	HML2.LTR511	4902957	4903956	LTR5B	(Subramanian et al.; 2011)	GTCCTA
10	HML2.LTR512	6164485	6165457	LTR5B	(Subramanian et al.; 2011)	GAAGT
10	HML2.LTR513	16249991	16250977	LTR5B	(Subramanian et al.; 2011)	AAAAAAAA
10	HML2.LTR514	19509807	19510764	LTR5_Hs	(Subramanian et al.; 2011)	-
10	HML2.LTR515	25881103	25882129	LTR5A	(Subramanian et al.; 2011)	TGGATT/TGTATT
10	HML2.LTR516	26376885	26377844	LTR5_Hs	(Subramanian et al.; 2011)	ATGAA
10	HML2.LTR517, K103, K(C10)	26893470	26894437	LTR5_Hs	(Barbulescu et al.; 1999)	ATGGGG
10	HML2.LTR518	29426711	29427740	LTR5A	(Subramanian et al.; 2011)	CTGCC
10	HML2.LTR519	32894456	32895483	LTR5A	(Subramanian et al.; 2011)	CATCTC
10	HML2.LTR520	37493325	37494305	LTR5_Hs	(Subramanian et al.; 2011)	GTCAGG/GTAAGG
10	HML2.LTR521	41713188	41714153	LTR5_Hs	(Subramanian et al.; 2011)	GAAAT
10	HML2.LTR522	43337203	43338170	LTR5_Hs	(Subramanian et al.; 2011)	TGGCA/TGGCG
10	HML2.LTR523	47396850	47397838	LTR5B	(Subramanian et al.; 2011)	GTTTT/CTTTT
10	HML2.LTR524	65409230	65410196	LTR5_Hs	(Subramanian et al.; 2011)	ATCCCT

10	HML2.LTR525	68525672	68526636	LTR5_Hs	(Subramanian et al.; 2011)	CAACC
10	HML2.LTR526	70257191	70258221	LTR5A	(Subramanian et al.; 2011)	CAGGAG
10	HML2.LTR527	79744593	79745590	LTR5B	(Subramanian et al.; 2011)	-
10	HML2.LTR528	80805882	80806911	LTR5A	(Subramanian et al.; 2011)	CTTATT
10	HML2.LTR529	87266813	87267810	LTR5B	(Subramanian et al.; 2011)	AATGAT/GATGAT
10	HML2.LTR530	87470166	87471163	LTR5B	(Subramanian et al.; 2011)	AATGAT/GATGAT
10	HML2.LTR531	90301653	90302607	LTR5_Hs	(Subramanian et al.; 2011)	TGTAAA
10	HML2.LTR532	95197194	95198222	LTR5A	(Subramanian et al.; 2011)	TGATA
10	HML2.LTR533	96403267	96404268	LTR5A	(Subramanian et al.; 2011)	-
10	HML2.LTR534	97699820	97700846	LTR5A	(Subramanian et al.; 2011)	CTAAAC/CTAAAA
10	HML2.LTR535	101033533	101034560	LTR5A	(Subramanian et al.; 2011)	ACTAGA/ACTAGG
10	HML2.LTR536	102391445	102392396	LTR5_Hs	(Subramanian et al.; 2011)	ATTTT
10	HML2.LTR537	102454375	102455335	LTR5_Hs	(Subramanian et al.; 2011)	AACTCC/CACTCC
10	HML2.LTR538	121929935	121930906	LTR5B	(Subramanian et al.; 2011)	ACAAG
10	HML2.LTR539	125940081	125941075	LTR5B	(Subramanian et al.; 2011)	CTGAC
10	HML2.LTR540	130627957	130628919	LTR5_Hs	(Subramanian et al.; 2011)	TAAGAA
10	HML2.LTR541	133541050	133542017	LTR5_Hs	(Subramanian et al.; 2011)	AATAG
10	HML2.LTR542	133609964	133611180	LTR5A	(Subramanian et al.; 2011)	ATCACC
11	HML2.LTR543	201708	202704	LTR5B	(Subramanian et al.; 2011)	GGTGAG
11	HML2.LTR544	3394552	3395567	LTR5A	(Subramanian et al.; 2011)	AGTGAG
11	HML2.LTR545	3593659	3594685	LTR5A	(Subramanian et al.; 2011)	AGTGAG
11	HML2.LTR546	9318776	9319704	LTR5_Hs	(Subramanian et al.; 2011)	TGTGC
11	HML2.LTR547	10391309	10392276	LTR5_Hs	(Subramanian et al.; 2011)	CTATT
11	HML2.LTR548	18900117	18901076	LTR5B	(Subramanian et al.; 2011)	AAACT
11	HML2.LTR549	23092503	23093370	LTR5A	(Subramanian et al.; 2011)	5' truncated
11	HML2.LTR550	24446094	24447052	LTR5_Hs	(Subramanian et al.; 2011)	GAACAA
11	HML2.LTR551	25540715	25541727	LTR5A	(Subramanian et al.; 2011)	ATATT
11	HML2.LTR552	33049342	33050285	LTR5_Hs	(Subramanian et al.; 2011)	AAAAAAT/AAATAAT
11	HML2.LTR553	37858938	37859901	LTR5_Hs	(Subramanian et al.; 2011)	TTCCTT/TTCCCT
11	HML2.LTR554	37930572	37931589	LTR5A	(Subramanian et al.; 2011)	TAATT
11	HML2.LTR555	50131140	50132165	LTR5A	(Subramanian et al.; 2011)	GTTCA
11	HML2.LTR556	50346425	50347450	LTR5A	(Subramanian et al.; 2011)	GTTCA
11	HML2.LTR557	55430995	55431959	LTR5_Hs	(Subramanian et al.; 2011)	ATTGAC
11	HML2.LTR558	56057658	56058622	LTR5_Hs	(Subramanian et al.; 2011)	AACAAC
11	HML2.LTR559	56502580	56503706	LTR5B	(Subramanian et al.; 2011)	AAGGTA
11	HML2.LTR560	56618318	56619311	LTR5B	(Subramanian et al.; 2011)	ACTGA/ATTGA
11	HML2.LTR561	56718658	56719681	LTR5B	(Subramanian et al.; 2011)	CATAAG/CATAAC
11	HML2.LTR562	59136291	59137277	LTR5B	(Subramanian et al.; 2011)	-
11	HML2.LTR563	59681165	59682136	LTR5B	(Subramanian et al.; 2011)	GAAAA
11	HML2.LTR564	60759108	60760106	LTR5B	(Subramanian et al.; 2011)	ATCAT
11	HML2.LTR565	61655013	61655976	LTR5_Hs	(Subramanian et al.; 2011)	GAAGGG
11	HML2.LTR566	62194660	62195628	LTR5_Hs	(Subramanian et al.; 2011)	AAAAAC

11	HML2.LTR567	62215948	62216939	LTR5B	(Subramanian et al.; 2011)	CCCCGG/GCCCGG
11	HML2.LTR568	62326600	62327566	LTR5_Hs	(Subramanian et al.; 2011)	GTTCTT
11	HML2.LTR569	62382197	62383091	LTR5_Hs	(Subramanian et al.; 2011)	3' truncated
11	HML2.LTR570	62858518	62859476	LTR5_Hs	(Subramanian et al.; 2011)	CTCTAA
11	HML2.LTR571	63527971	63528938	LTR5_Hs	(Subramanian et al.; 2011)	CTTACC
11	HML2.LTR572	63530314	63531281	LTR5_Hs	(Subramanian et al.; 2011)	ATGTGG
11	HML2.LTR573	65023204	65024528	LTR5B	(Subramanian et al.; 2011)	AACAG/CACAG
11	HML2.LTR574	65522789	65523765	LTR5B	(Subramanian et al.; 2011)	-
11	HML2.LTR575	67603354	67604313	LTR5_Hs	(Subramanian et al.; 2011)	AATTAG
11	HML2.LTR576	67716490	67717502	LTR5A	(Subramanian et al.; 2011)	AGTGAG
11	HML2.LTR577	67739337	67740349	LTR5A	(Subramanian et al.; 2011)	AGTGAG
11	HML2.LTR578	67757889	67758849	LTR5_Hs	(Subramanian et al.; 2011)	ATCAA/ATCAG
11	HML2.LTR579	67829985	67831010	LTR5A	(Subramanian et al.; 2011)	ATTTG
11	HML2.LTR580	67867964	67868923	LTR5_Hs	(Subramanian et al.; 2011)	GGTACA
11	HML2.LTR581	67968248	67969277	LTR5A	(Subramanian et al.; 2011)	AGTGAG
11	HML2.LTR582	71534703	71535694	LTR5B	(Subramanian et al.; 2011)	CACATA
11	HML2.LTR583	71599375	71600389	LTR5A	(Subramanian et al.; 2011)	AATGAG/AGTGAG
11	HML2.LTR584	71625997	71627026	LTR5A	(Subramanian et al.; 2011)	AGTGAA/AGTGAG
11	HML2.LTR585	71899369	71900382	LTR5A	(Subramanian et al.; 2011)	AGTGAG
11	HML2.LTR586	72164374	72165341	LTR5_Hs	(Subramanian et al.; 2011)	TATGC
11	HML2.LTR587	76711086	76712112	LTR5A	(Subramanian et al.; 2011)	AAGGC
11	HML2.LTR588	93744697	93745652	LTR5B	(Subramanian et al.; 2011)	GTTTTG
11	HML2.LTR589	96586133	96587100	LTR5_Hs	(Subramanian et al.; 2011)	ATTTT
11	HML2.LTR590	105264851	105265754	LTR5_Hs	(Subramanian et al.; 2011)	3' truncated
11	HML2.LTR591	113777040	113778056	LTR5A	(Subramanian et al.; 2011)	TCTTC
11	HML2.LTR592	119008102	119009131	LTR5A	(Subramanian et al.; 2011)	ATAAGT
11	HML2.LTR593	119039272	119040160	LTR5_Hs	(Subramanian et al.; 2011)	GCTACC
11	HML2.LTR594	123279703	123280710	LTR5A	(Subramanian et al.; 2011)	CCCAG/CCCAA
12	HML2.LTR595	4721231	4722200	LTR5_Hs	(Subramanian et al.; 2011)	ATAGG
12	HML2.LTR596	5537017	5537956	LTR5_Hs	(Subramanian et al.; 2011)	3' truncated
12	HML2.LTR597	6191437	6192226	LTR5_Hs	(Subramanian et al.; 2011)	3' truncated
12	HML2.LTR598	6797403	6798680	LTR5B	(Subramanian et al.; 2011)	GAGAGG
12	HML2.LTR599	6885860	6886821	LTR5_Hs	(Subramanian et al.; 2011)	GATATA
12	HML2.LTR600	8220179	8221207	LTR5A	(Subramanian et al.; 2011)	CTTGGA
12	HML2.LTR601	8264684	8265713	LTR5A	(Subramanian et al.; 2011)	ATTTG
12	HML2.LTR602	8409218	8410247	LTR5A	(Subramanian et al.; 2011)	AGTGAG
12	HML2.LTR603	8431407	8432439	LTR5A	(Subramanian et al.; 2011)	AGTGAG
12	HML2.LTR604	8461932	8462899	LTR5_Hs	(Subramanian et al.; 2011)	TAATTT
12	HML2.LTR605	8635661	8636609	LTR5_Hs	(Subramanian et al.; 2011)	ACTGGG
12	HML2.LTR606	9600465	9601432	LTR5_Hs	(Subramanian et al.; 2011)	GAGAT
12	HML2.LTR607	10571268	10572236	LTR5_Hs	(Subramanian et al.; 2011)	TTGCC
12	HML2.LTR608	21644942	21645931	LTR5_Hs	(Subramanian et al.; 2011)	TTGTAC

12	HML2.LTR609	29932054	29933021	LTR5_Hs	(Subramanian et al.; 2011)	CTTATG
12	HML2.LTR610	32099511	32100478	LTR5_Hs	(Subramanian et al.; 2011)	GGGTAC
12	HML2.LTR611	34180273	34181302	LTR5A	(Subramanian et al.; 2011)	ATGAT
12	HML2.LTR612	37738088	37739055	LTR5_Hs	(Subramanian et al.; 2011)	TTTCC/TTTCT
12	HML2.LTR613	37995532	37996554	LTR5A	(Subramanian et al.; 2011)	AATGT/AATAT
12	HML2.LTR614	42329778	42331094	LTR5B	(Subramanian et al.; 2011)	TCCAGA/TCCAGG
12	HML2.LTR615	42387719	42388711	LTR5B	(Subramanian et al.; 2011)	TGAGAT
12	HML2.LTR616	49115641	49116673	LTR5A	(Subramanian et al.; 2011)	ACCGTT/ACTGTT
12	HML2.LTR617	51454287	51455254	LTR5_Hs	(Subramanian et al.; 2011)	CTCAC
12	K134, HML2.LTR618	55333431	55334399	LTR5_Hs	(Belshaw; 2005)	GCTAT
12	HML2.LTR619	56400361	56401325	LTR5_Hs	(Subramanian et al.; 2011)	GGCACA/GGCACC
12	HML2.LTR620	56822108	56823135	LTR5A	(Subramanian et al.; 2011)	CTTTT
12	HML2.LTR621	58543162	58544122	LTR5_Hs	(Subramanian et al.; 2011)	TCAATC
12	HML2.LTR622	59108214	59109184	LTR5_Hs	(Subramanian et al.; 2011)	TGCAAT
12	HML2.LTR623	73008723	73009695	LTR5_Hs	(Subramanian et al.; 2011)	GAATTG/GAATTA
12	HML2.LTR624	73347934	73348922	LTR5B	(Subramanian et al.; 2011)	GACAG
12	HML2.LTR625	75444832	75445817	LTR5B	(Subramanian et al.; 2011)	CAAAGC
12	HML2.LTR626	85468940	85469903	LTR5_Hs	(Subramanian et al.; 2011)	CTAAC
12	HML2.LTR627	93074403	93075369	LTR5_Hs	(Subramanian et al.; 2011)	AGTTAC
12	HML2.LTR628	105478559	105479527	LTR5_Hs	(Subramanian et al.; 2011)	CTCTT
12	HML2.LTR629	107997036	107998061	LTR5A	(Subramanian et al.; 2011)	TACTTA/TAATTA
12	HML2.LTR630	118105984	118106943	LTR5_Hs	(Subramanian et al.; 2011)	GTTAG
12	HML2.LTR631	122750860	122751831	LTR5_Hs	(Subramanian et al.; 2011)	GGAATA
12	HML2.LTR632	123492642	123493573	LTR5_Hs	(Subramanian et al.; 2011)	ATGAGA
12	HML2.LTR633	126958677	126959664	LTR5B	(Subramanian et al.; 2011)	ATAGG
13	HML2.LTR634	18719995	18720917	LTR5_Hs	(Subramanian et al.; 2011)	GATTTC
13	HML2.LTR635	19600218	19601185	LTR5_Hs	(Subramanian et al.; 2011)	ACTGGT
13	HML2.LTR636	22813650	22814620	LTR5_Hs	(Subramanian et al.; 2011)	-
13	HML2.LTR637	24591593	24592530	LTR5_Hs	(Subramanian et al.; 2011)	AGGCT
13	HML2.LTR638	24956546	24957486	LTR5_Hs	(Subramanian et al.; 2011)	AGGCT
13	HML2.LTR639	42710788	42711803	LTR5A	(Subramanian et al.; 2011)	AATGT
13	HML2.LTR640	43040544	43041509	LTR5_Hs	(Subramanian et al.; 2011)	TATAG
13	HML2.LTR641	49599022	49599990	LTR5_Hs	(Subramanian et al.; 2011)	GTAAT
13	HML2.LTR642	54266807	54267766	LTR5_Hs	(Subramanian et al.; 2011)	TATGTT
13	HML2.LTR643	69614878	69615842	LTR5_Hs	(Subramanian et al.; 2011)	GTTTTT
13	HML2.LTR644	78646802	78647787	LTR5B	(Subramanian et al.; 2011)	ATGAC/ATGAT
13	HML2.LTR645	95268021	95269003	LTR5_Hs	(Subramanian et al.; 2011)	CTACTA/CTAATA
13	HML2.LTR646	98764338	98765351	LTR5A	(Subramanian et al.; 2011)	AGTAT/AGTGT
13	HML2.LTR647	110837234	110838214	LTR5B	(Subramanian et al.; 2011)	GCAAAC/GTAAAC
13	HML2.LTR648	114181911	114182938	LTR5A	(Subramanian et al.; 2011)	GGCAT
14	HML2.LTR649	18319453	18320475	LTR5B	(Subramanian et al.; 2011)	TATTT
14	HML2.LTR651	19129412	19130441	LTR5A	(Subramanian et al.; 2011)	CTGTT/GTGTT

14	HML2.LTR650	19269575	19270584	LTR5B	(Subramanian et al.; 2011)	CTGTT/GTGTT
14	HML2.LTR652	20084587	20085556	LTR5_Hs	(Subramanian et al.; 2011)	CCCTG
14	HML2.LTR653	20241556	20242578	LTR5A	(Subramanian et al.; 2011)	GACAC
14	HML2.LTR654	20268923	20269890	LTR5_Hs	(Subramanian et al.; 2011)	TGCCAA
14	HML2.LTR655	22724685	22725652	LTR5_Hs	(Subramanian et al.; 2011)	AGCAA
14	HML2.LTR656	23981494	23982489	LTR5B	(Subramanian et al.; 2011)	GCTATA
14	HML2.LTR657	24024299	24025286	LTR5B	(Subramanian et al.; 2011)	GCTATA
14	HML2.LTR658	38118099	38119067	LTR5_Hs	(Subramanian et al.; 2011)	TTCTG
14	HML2.LTR659	45213722	45214690	LTR5_Hs	(Subramanian et al.; 2011)	AACCAT/AACTAT
14	HML2.LTR660	45393582	45394565	LTR5B	(Subramanian et al.; 2011)	3' truncated
14	HML2.LTR661	54752605	54753622	LTR5A	(Subramanian et al.; 2011)	TATATG/TATACG
14	HML2.LTR662	55024550	55025503	LTR5_Hs	(Subramanian et al.; 2011)	CAAACT
14	HML2.LTR663	64978588	64979555	LTR5_Hs	(Subramanian et al.; 2011)	CTCCAT
14	HML2.LTR664	77635277	77636245	LTR5_Hs	(Subramanian et al.; 2011)	AAGAG
14	HML2.LTR665	77661398	77662365	LTR5_Hs	(Subramanian et al.; 2011)	-
14	HML2.LTR666	77794674	77795634	LTR5_Hs	(Subramanian et al.; 2011)	GCCTGA
14	HML2.LTR667	87020142	87021007	LTR5_Hs	(Subramanian et al.; 2011)	ATGCT
14	HML2.LTR668	88024340	88025306	LTR5B	(Subramanian et al.; 2011)	GTTCTA
14	HML2.LTR669	92819980	92820978	LTR5B	(Subramanian et al.; 2011)	GATGG/GATGA
14	HML2.LTR670	94482206	94483210	LTR5B	(Subramanian et al.; 2011)	TTTGC
14	HML2.LTR671	94626138	94627143	LTR5A	(Subramanian et al.; 2011)	CAGCG/CAGTG
14	HML2.LTR672	95729488	95730101	LTR5B	(Subramanian et al.; 2011)	3' truncated
14	HML2.LTR673	100311791	100312758	LTR5_Hs	(Subramanian et al.; 2011)	AGTTGG
14	HML2.LTR674	103524916	103525904	LTR5B	(Subramanian et al.; 2011)	GGAAAG
14	HML2.LTR675	105273468	105274474	LTR5B	(Subramanian et al.; 2011)	CCATGG
14	HML2.LTR676	105554875	105555804	LTR5B	(Subramanian et al.; 2011)	GAATT
14	HML2.LTR677	105732432	105733432	LTR5B	(Subramanian et al.; 2011)	TCAC
14	HML2.LTR678	106031229	106032258	LTR5A	(Subramanian et al.; 2011)	-
14	HML2.LTR679	106447279	106448239	LTR5B	(Subramanian et al.; 2011)	-
15	HML2.LTR680	19866100	19867111	LTR5B	(Subramanian et al.; 2011)	TATTT
15	HML2.LTR681	23081717	23082703	LTR5B	(Subramanian et al.; 2011)	GGAGAG
15	HML2.LTR682	40643952	40644969	LTR5A	(Subramanian et al.; 2011)	-
15	HML2.LTR683	44228595	44229613	LTR5A	(Subramanian et al.; 2011)	AAAAGA/AAAAAA
15	HML2.LTR684	58833342	58834296	LTR5_Hs	(Subramanian et al.; 2011)	AGATAT
15	HML2.LTR685	65226542	65227509	LTR5_Hs	(Subramanian et al.; 2011)	TGACCC
15	HML2.LTR686	65733653	65734621	LTR5_Hs	(Subramanian et al.; 2011)	ATTAA
15	HML2.LTR687	75323292	75324260	LTR5B	(Subramanian et al.; 2011)	ATGGAA/GTGGAA
15	HML2.LTR688	75870854	75871822	LTR5_Hs	(Subramanian et al.; 2011)	AATAAT
15	HML2.LTR689	78221052	78222039	LTR5B	(Subramanian et al.; 2011)	GGGTTG
15	HML2.LTR690	79935248	79936241	LTR5B	(Subramanian et al.; 2011)	GAGCCC
15	HML2.LTR691	88540552	88541519	LTR5_Hs	(Subramanian et al.; 2011)	TCTTGG
15	HML2.LTR692	90346561	90347586	LTR5A	(Subramanian et al.; 2011)	CCAAG

15	HML2.LTR693	90973106	90974068	LTR5B	(Subramanian et al.; 2011)	AAGA
15	HML2.LTR694	95784723	95785682	LTR5_Hs	(Subramanian et al.; 2011)	-
15	HML2.LTR695	97629102	97630131	LTR5A	(Subramanian et al.; 2011)	GTGGT/GTCGT
15	HML2.LTR696	101859977	101860944	LTR5_Hs	(Subramanian et al.; 2011)	TAGATA/TCGATA
16	HML2.LTR697	387824	388797	LTR5B	(Subramanian et al.; 2011)	AAAAG
16	HML2.LTR698	1930829	1931857	LTR5A	(Subramanian et al.; 2011)	GCTGG
16	HML2.LTR699	5754388	5755355	LTR5_Hs	(Subramanian et al.; 2011)	-
16	HML2.LTR700	8178761	8179728	LTR5_Hs	(Subramanian et al.; 2011)	GAGTGTG
16	HML2.LTR701	10829449	10830485	LTR5A	(Subramanian et al.; 2011)	CAGCCC/CGGCCC
16	HML2.LTR702	14636147	14637106	LTR5_Hs	(Subramanian et al.; 2011)	TGGGGG
16	HML2.LTR703	21221787	21222748	LTR5_Hs	(Subramanian et al.; 2011)	TCCCAT
16	HML2.LTR704	23599484	23600443	LTR5_Hs	(Subramanian et al.; 2011)	GTTACA
16	HML2.LTR705	29740241	29741225	LTR5B	(Subramanian et al.; 2011)	CTCAC/CTCAT
16	HML2.LTR706	31504698	31505539	LTR5B	(Subramanian et al.; 2011)	GTTCT
16	HML2.LTR707	34144086	34145098	LTR5A	(Subramanian et al.; 2011)	CTCAC
16	HML2.LTR708	35153437	35154456	LTR5A	(Subramanian et al.; 2011)	GATCA
16	HML2.LTR709	35204578	35205582	LTR5B	(Subramanian et al.; 2011)	-
16	HML2.LTR710	35628809	35629805	LTR5B	(Subramanian et al.; 2011)	AACAAC
16	HML2.LTR711	35955555	35956522	LTR5_Hs	(Subramanian et al.; 2011)	CAACG/CAAAG
16	HML2.LTR712	47864459	47865426	LTR5_Hs	(Subramanian et al.; 2011)	TTTAGA
16	HML2.LTR713	66627729	66628736	LTR5B	(Subramanian et al.; 2011)	GAAGA
16	HML2.LTR714	66955375	66956403	LTR5B	(Subramanian et al.; 2011)	CTGAT
16	HML2.LTR715	74799404	74800360	LTR5_Hs	(Subramanian et al.; 2011)	3' truncated
16	HML2.LTR716	83879186	83880182	LTR5B	(Subramanian et al.; 2011)	CACCAT/CAGCAT
17	HML2.LTR717	5074680	5075646	LTR5_Hs	(Subramanian et al.; 2011)	CTTGAC
17	HML2.LTR718	6646383	6647360	LTR5B	(Subramanian et al.; 2011)	GGAAG
17	HML2.LTR719	7088333	7089633	LTR5B	(Subramanian et al.; 2011)	ATCAAC
17	HML2.LTR720	7159072	7160057	LTR5B	(Subramanian et al.; 2011)	ATCTAT/ACCTAT
17	HML2.LTR721	16399849	16401128	LTR5A	(Subramanian et al.; 2011)	CTTTGG
17	HML2.LTR722	16834880	16835857	LTR5_Hs	(Subramanian et al.; 2011)	GGTGGG
17	HML2.LTR723	17610928	17611884	LTR5_Hs	(Subramanian et al.; 2011)	GATTCA
17	HML2.LTR724	18302964	18303940	LTR5B	(Subramanian et al.; 2011)	GTAAC/ATAAC
17	HML2.LTR725	18436884	18437858	LTR5_Hs	(Subramanian et al.; 2011)	GGTGGG
17	HML2.LTR726	19504958	19505922	LTR5_Hs	(Subramanian et al.; 2011)	CAACAC
17	HML2.LTR727	19519711	19520716	LTR5B	(Subramanian et al.; 2011)	GATGT/AATGT
17	HML2.LTR728	20506550	20507524	LTR5_Hs	(Subramanian et al.; 2011)	GGTGGG
17	HML2.LTR729	22229630	22230620	LTR5A	(Subramanian et al.; 2011)	CTTTGG/CTTTTG
17	HML2.LTR730	30563389	30564162	LTR5_Hs	(Subramanian et al.; 2011)	3' truncated
17	HML2.LTR731	30594444	30595336	LTR5B	(Subramanian et al.; 2011)	5' truncated
17	HML2.LTR732	30700041	30701008	LTR5_Hs	(Subramanian et al.; 2011)	GCGTG
17	HML2.LTR733	36140053	36141024	LTR5_Hs	(Subramanian et al.; 2011)	CATAC
17	HML2.LTR734	43354954	43355913	LTR5_Hs	(Subramanian et al.; 2011)	-
17	HML2.LTR735	46283614	46284581	LTR5_Hs	(Subramanian et al.; 2011)	CCACAC

17	HML2.LTR736	54201574	54202541	LTR5_Hs	(Subramanian et al.; 2011)	-
17	HML2.LTR737	59290107	59290917	LTR5_Hs	(Subramanian et al.; 2011)	3' truncated
17	HML2.LTR738	64930022	64930983	LTR5_Hs	(Subramanian et al.; 2011)	CCACAC
17	HML2.LTR739	67326846	67327771	LTR5_Hs	(Subramanian et al.; 2011)	GTAAG
17	HML2.LTR740	68603992	68604951	LTR5_Hs	(Subramanian et al.; 2011)	AGGTTG
17	HML2.LTR741	80551208	80552176	LTR5_Hs	(Subramanian et al.; 2011)	CACTTT
17	HML2.LTR742	82744532	82745497	LTR5_Hs	(Subramanian et al.; 2011)	TGAGAG
18	HML2.LTR743	90214	91194	LTR5B	(Subramanian et al.; 2011)	-
18	HML2.LTR744	2000814	2001781	LTR5_Hs	(Subramanian et al.; 2011)	CTGTG
18	HML2.LTR745	4917279	4918247	LTR5_Hs	(Subramanian et al.; 2011)	TGAGAT
18	HML2.LTR746	15343747	15344757	LTR5B	(Subramanian et al.; 2011)	TATTT
18	HML2.LTR747	26201835	26202802	LTR5_Hs	(Subramanian et al.; 2011)	TCTTTC
18	HML2.LTR748	26927277	26928239	LTR5_Hs	(Subramanian et al.; 2011)	CCTGTG
18	HML2.LTR749	31810501	31811482	LTR5_Hs	(Subramanian et al.; 2011)	CAACTG
18	HML2.LTR750	41183868	41184837	LTR5_Hs	(Subramanian et al.; 2011)	GTTGGC
18	HML2.LTR751	46088310	46089299	LTR5B	(Subramanian et al.; 2011)	-
18	HML2.LTR752	50404195	50405171	LTR5B	(Subramanian et al.; 2011)	GACCTT
18	HML2.LTR753	67076783	67077752	LTR5_Hs	(Subramanian et al.; 2011)	ATACT
18	HML2.LTR754	68105242	68106208	LTR5_Hs	(Subramanian et al.; 2011)	GTATC
18	HML2.LTR755	68942269	68943233	LTR5_Hs	(Subramanian et al.; 2011)	TCTTT
18	HML2.LTR756	69042852	69043820	LTR5_Hs	(Subramanian et al.; 2011)	AGGAAC
18	HML2.LTR757	70479125	70480147	LTR5A	(Subramanian et al.; 2011)	AATCA
18	HML2.LTR758	79960166	79961063	LTR5_Hs	(Subramanian et al.; 2011)	5’ truncated
19	HML2.LTR759	8314892	8315814	LTR5_Hs	(Subramanian et al.; 2011)	GCCTGG
19	HML2.LTR760	9676796	9677759	LTR5B	(Subramanian et al.; 2011)	CCTAT
19	HML2.LTR761	9697021	9699079	LTR5B	(Subramanian et al.; 2011)	CCACTG/CCACTA
19	HML2.LTR762	9784066	9785035	LTR5B	(Subramanian et al.; 2011)	-
19	HML2.LTR763	9789358	9790296	LTR5B	(Subramanian et al.; 2011)	CTCAGT/CTCGGT
19	HML2.LTR764	11889618	11890605	LTR5B	(Subramanian et al.; 2011)	GGTGCA/GGTGCG
19	HML2.LTR765	12231835	12232790	LTR5_Hs	(Subramanian et al.; 2011)	GATGG/GATAG
19	HML2.LTR766	15782597	15783593	LTR5B	(Subramanian et al.; 2011)	GCCTGA
19	HML2.LTR767	18402468	18403463	LTR5B	(Subramanian et al.; 2011)	-
19	HML2.LTR768	19955503	19956488	LTR5B	(Subramanian et al.; 2011)	CCAGG/CCCGGG
19	HML2.LTR769	19961589	19962582	LTR5B	(Subramanian et al.; 2011)	CCACCA
19	HML2.LTR770	20082342	20083303	LTR5_Hs	(Subramanian et al.; 2011)	GCATT
19	HML2.LTR771	20156338	20157300	LTR5_Hs	(Subramanian et al.; 2011)	GGGGA
19	HML2.LTR772	20466331	20467299	LTR5_Hs	(Subramanian et al.; 2011)	AATTTA
19	HML2.LTR773	20961559	20962547	LTR5B	(Subramanian et al.; 2011)	GTCCGT/GTCCAT
19	HML2.LTR774	21567942	21568911	LTR5_Hs	(Subramanian et al.; 2011)	GCCCAG
19	HML2.LTR775	21620224	21621240	LTR5A	(Subramanian et al.; 2011)	GGGGC
19	HML2.LTR776	22283562	22284503	LTR5_Hs	(Subramanian et al.; 2011)	GCTCA
19	HML2.LTR777	23077744	23078765	LTR5_Hs	(Subramanian et al.; 2011)	GTCTG
19	HML2.LTR778	23692688	23693586	LTR5_Hs	(Subramanian et al.; 2011)	TTGTAG

19	HML2.LTR779	23723658	23724624	LTR5_Hs	(Subramanian et al.; 2011)	GAAGG/GAGGG
19	HML2.LTR780	27814254	27815552	LTR5B	(Subramanian et al.; 2011)	CAGCC
19	HML2.LTR781	28130630	28131598	LTR5_Hs	(Subramanian et al.; 2011)	ATGGCA
19	HML2.LTR782	28689312	28690280	LTR5_Hs	(Subramanian et al.; 2011)	TCTTT
19	HML2.LTR783	34920178	34921118	LTR5_Hs	(Subramanian et al.; 2011)	3' truncated
19	HML2.LTR784	35129353	35130347	LTR5B	(Subramanian et al.; 2011)	-
19	HML2.LTR785	36233354	36234332	LTR5_Hs	(Subramanian et al.; 2011)	GGCTG
19	HML2.LTR786	36247121	36247924	LTR5_Hs	(Subramanian et al.; 2011)	CTGGG
19	HML2.LTR787	36332169	36333136	LTR5_Hs	(Subramanian et al.; 2011)	-
19	HML2.LTR788	36799599	36800807	LTR5B	(Subramanian et al.; 2011)	-
19	HML2.LTR789	37006159	37007120	LTR5_Hs	(Subramanian et al.; 2011)	ATGGGT
19	HML2.LTR790	37320788	37321682	LTR5_Hs	(Subramanian et al.; 2011)	CCTGG
19	HML2.LTR791	37331391	37332372	LTR5_Hs	(Subramanian et al.; 2011)	CGCCCA
19	HML2.LTR792	37501745	37502718	LTR5B	(Subramanian et al.; 2011)	GTTTGT
19	HML2.LTR793	37529809	37530767	LTR5_Hs	(Subramanian et al.; 2011)	TAAAT/TAAAC
19	HML2.LTR794	37629951	37630912	LTR5_Hs	(Subramanian et al.; 2011)	AAAGAG
19	HML2.LTR795	37866177	37867144	LTR5_Hs	(Subramanian et al.; 2011)	CAAGCT
19	HML2.LTR796	38631139	38632089	LTR5_Hs	(Subramanian et al.; 2011)	GTAATG
19	HML2.LTR797	39479462	39480441	LTR5_Hs	(Subramanian et al.; 2011)	GACAA
19	HML2.LTR798	39688349	39689354	LTR5B	(Subramanian et al.; 2011)	GGTTTG
19	HML2.LTR799	39959231	39960162	LTR5_Hs	(Subramanian et al.; 2011)	TGGAGG
19	HML2.LTR800	40822624	40823609	LTR5B	(Subramanian et al.; 2011)	GAGAC
19	HML2.LTR801	40922204	40923191	LTR5B	(Subramanian et al.; 2011)	ACGCT/ATGCT
19	HML2.LTR802	41036045	41037073	LTR5A	(Subramanian et al.; 2011)	CTGCTG
19	HML2.LTR803	41544578	41545607	LTR5A	(Subramanian et al.; 2011)	AATATA
19	HML2.LTR804	44053110	44054098	LTR5B	(Subramanian et al.; 2011)	ATATC
19	HML2.LTR805	44280661	44281558	LTR5_Hs	(Subramanian et al.; 2011)	3' truncated
19	HML2.LTR806	44594096	44595063	LTR5_Hs	(Subramanian et al.; 2011)	ATTTC
19	HML2.LTR807	44639054	44640041	LTR5B	(Subramanian et al.; 2011)	AAAAG/AAATG
19	HML2.LTR808	46768372	46769368	LTR5B	(Subramanian et al.; 2011)	TGCCTG
19	HML2.LTR809	48889636	48890598	LTR5_Hs	(Subramanian et al.; 2011)	CCTAA
19	HML2.LTR810	48891605	48892893	LTR5B	(Subramanian et al.; 2011)	GGCTG
19	HML2.LTR811	51868708	51869699	LTR5B	(Subramanian et al.; 2011)	ATGTG
19	HML2.LTR812	51905692	51906661	LTR5_Hs	(Subramanian et al.; 2011)	GTGCAC/CTGCAC
19	HML2.LTR813	51908192	51909157	LTR5_Hs	(Subramanian et al.; 2011)	TCAAC
19	HML2.LTR814	51909158	51909980	LTR5B	(Subramanian et al.; 2011)	5' truncated
19	HML2.LTR815	51921149	51922154	LTR5B	(Subramanian et al.; 2011)	3' truncated
19	HML2.LTR816	51924895	51925884	LTR5B	(Subramanian et al.; 2011)	-
19	HML2.LTR817	51957782	51958776	LTR5B	(Subramanian et al.; 2011)	-
19	HML2.LTR818	51960243	51961231	LTR5B	(Subramanian et al.; 2011)	AGAGAG
19	HML2.LTR819	51982962	51983946	LTR5B	(Subramanian et al.; 2011)	GGAAGG/GGCAGG
19	HML2.LTR820	52026209	52027168	LTR5B	(Subramanian et al.; 2011)	AGGTG
19	HML2.LTR821	52043132	52044100	LTR5_Hs	(Subramanian et al.; 2011)	CAAATT

19	HML2.LTR822	52051722	52052691	LTR5B	(Subramanian et al.; 2011)	GGCAGC
19	HML2.LTR823	52338780	52339732	LTR5B	(Subramanian et al.; 2011)	GTGGTG
19	HML2.LTR824	52420956	52421918	LTR5_Hs	(Subramanian et al.; 2011)	ACTTT/CCTTT
19	HML2.LTR825	52485538	52486425	LTR5_Hs	(Subramanian et al.; 2011)	3' truncated
19	HML2.LTR826	52519595	52520931	LTR5B	(Subramanian et al.; 2011)	CTGGAT
19	HML2.LTR827	52558697	52559661	LTR5_Hs	(Subramanian et al.; 2011)	GGAAT
19	HML2.LTR828	52697740	52698706	LTR5_Hs	(Subramanian et al.; 2011)	CTTTCA
19	HML2.LTR829	52723561	52724548	LTR5B	(Subramanian et al.; 2011)	ATAAGG
19	HML2.LTR830	52740758	52742027	LTR5B	(Subramanian et al.; 2011)	CTGGAT
19	HML2.LTR831	52847055	52848043	LTR5B	(Subramanian et al.; 2011)	GAAAG
19	HML2.LTR832	52872325	52873620	LTR5B	(Subramanian et al.; 2011)	-
19	HML2.LTR833	52936054	52937035	LTR5B	(Subramanian et al.; 2011)	-
19	HML2.LTR834	52944582	52945561	LTR5B	(Subramanian et al.; 2011)	CTTCTG
19	HML2.LTR835	53027908	53028880	LTR5B	(Subramanian et al.; 2011)	CCATCT
19	HML2.LTR836	53420531	53421487	LTR5B	(Subramanian et al.; 2011)	GAAAAT
19	HML2.LTR837	53443775	53444761	LTR5B	(Subramanian et al.; 2011)	GGGTT
19	HML2.LTR838	53459937	53460913	LTR5B	(Subramanian et al.; 2011)	CTTTC/CTGTC
19	HML2.LTR839	53468335	53469326	LTR5B	(Subramanian et al.; 2011)	ACAGGA
19	HML2.LTR840	53497641	53498598	LTR5B	(Subramanian et al.; 2011)	TCACT
19	HML2.LTR841	53838626	53839592	LTR5_Hs	(Subramanian et al.; 2011)	GGCTC/TGCTC
19	HML2.LTR842	54943736	54944664	LTR5_Hs	(Subramanian et al.; 2011)	GTGACG/GTGAGG
19	HML2.LTR843	54950284	54951249	LTR5_Hs	(Subramanian et al.; 2011)	TATTTT
19	HML2.LTR844	55059690	55060689	LTR5A	(Subramanian et al.; 2011)	-
19	HML2.LTR845	55717908	55718885	LTR5B	(Subramanian et al.; 2011)	CACTCG/CACTCA
19	HML2.LTR846	56559332	56560285	LTR5_Hs	(Subramanian et al.; 2011)	TTCAG
19	HML2.LTR847	56686015	56687004	LTR5B	(Subramanian et al.; 2011)	ATTTTC
19	HML2.LTR848	57556708	57557700	LTR5B	(Subramanian et al.; 2011)	CTAAG
19	HML2.LTR849	57810328	57811568	LTR5_Hs	(Subramanian et al.; 2011)	TTAGCC
19	HML2.LTR850	57823625	57824595	LTR5B	(Subramanian et al.; 2011)	-
19	HML2.LTR851	57828963	57829955	LTR5B	(Subramanian et al.; 2011)	GACAGGA/AACAGGA
20	HML2.LTR852	811836	812820	LTR5B	(Subramanian et al.; 2011)	AGAAAG/ATAAAG
20	HML2.LTR853	3022940	3023895	LTR5B	(Subramanian et al.; 2011)	TGTGAT
20	HML2.LTR854	5885006	5885997	LTR5B	(Subramanian et al.; 2011)	TCCTGG
20	HML2.LTR855	7956200	7957167	LTR5_Hs	(Subramanian et al.; 2011)	TAGTCC
20	HML2.LTR856	15946666	15947661	LTR5B	(Subramanian et al.; 2011)	TTGAG
20	HML2.LTR857	23829322	23830335	LTR5A	(Subramanian et al.; 2011)	ACCTG
20	HML2.LTR858	23860595	23861558	LTR5_Hs	(Subramanian et al.; 2011)	AATCAC
20	HML2.LTR859	23982641	23983609	LTR5_Hs	(Subramanian et al.; 2011)	GGTCCC
20	HML2.LTR860	25233838	25234802	LTR5_Hs	(Subramanian et al.; 2011)	CTATTA
20	HML2.LTR861	25241246	25242215	LTR5_Hs	(Subramanian et al.; 2011)	ATAGAT
20	HML2.LTR862	25550343	25551333	LTR5B	(Subramanian et al.; 2011)	-
20	HML2.LTR863	34126944	34128531	LTR5B	(Subramanian et al.; 2011)	-
20	HML2.LTR864	34158608	34159575	LTR5B	(Subramanian et al.; 2011)	GCTTGG

20	HML2.LTR865	34252112	34253109	LTR5B	(Subramanian et al.; 2011)	5' truncated
20	HML2.LTR866	34424999	34425992	LTR5B	(Subramanian et al.; 2011)	CCTAAA/TCTAAA
20	HML2.LTR867	35262124	35263091	LTR5_Hs	(Subramanian et al.; 2011)	AGAGAT
20	HML2.LTR868	36967836	36968808	LTR5B	(Subramanian et al.; 2011)	CTTTGA/CTTTGG
20	HML2.LTR869	41970896	41971863	LTR5_Hs	(Subramanian et al.; 2011)	AATACC
20	HML2.LTR870	56211781	56212781	LTR5B	(Subramanian et al.; 2011)	-
20	HML2.LTR871	60041179	60042140	LTR5_Hs	(Subramanian et al.; 2011)	ACAGA/TCAGA
20	HML2.LTR872	63659690	63660674	LTR5B	(Subramanian et al.; 2011)	AAGGAC
21	HML2.LTR873	13023725	13024739	LTR5B	(Subramanian et al.; 2011)	TATTT
21	HML2.LTR874	13819880	13820848	LTR5_Hs	(Subramanian et al.; 2011)	GATTTC
21	HML2.LTR875	14280940	14281912	LTR5_Hs	(Subramanian et al.; 2011)	ATAAG
21	HML2.LTR876	17694216	17695183	LTR5_Hs	(Subramanian et al.; 2011)	GCAGAA
21	HML2.LTR877	17932719	17933686	LTR5_Hs	(Subramanian et al.; 2011)	GTATT
21	HML2.LTR878	19103346	19104326	LTR5B	(Subramanian et al.; 2011)	-
21	HML2.LTR879	32996300	32997287	LTR5B	(Subramanian et al.; 2011)	TTATT/TTGTT
21	HML2.LTR880	38572857	38573882	LTR5A	(Subramanian et al.; 2011)	GATAAT
21	HML2.LTR881	41420869	41421836	LTR5_Hs	(Subramanian et al.; 2011)	GGTGCC
21	HML2.LTR882	43147130	43148089	LTR5_Hs	(Subramanian et al.; 2011)	AATCC
21	HML2.LTR883	44282860	44283777	LTR5_Hs	(Subramanian et al.; 2011)	GAGGC
22	HML2.LTR884	15851785	15852795	LTR5B	(Subramanian et al.; 2011)	CTGTT/GTGTT
22	HML2.LTR885	17059133	17060168	LTR5A	(Subramanian et al.; 2011)	CAGTA
22	HML2.LTR888	18347506	18348467	LTR5_Hs	(Subramanian et al.; 2011)	GGTCCC
22	HML2.LTR886	18513933	18514894	LTR5_Hs	(Subramanian et al.; 2011)	GGTCCC
22	HML2.LTR887	18771254	18772215	LTR5_Hs	(Subramanian et al.; 2011)	GGTCCC
22	HML2.LTR889	21093340	21094329	LTR5B	(Subramanian et al.; 2011)	CAGGCC
22	HML2.LTR890	21205502	21206463	LTR5_Hs	(Subramanian et al.; 2011)	GGTCCC
22	HML2.LTR891	22649348	22650319	LTR5_Hs	(Subramanian et al.; 2011)	GGTCCC
22	HML2.LTR892	23907084	23908049	LTR5_Hs	(Subramanian et al.; 2011)	GGCTAGGCTG
22	HML2.LTR893	24210892	24211849	LTR5_Hs	(Subramanian et al.; 2011)	GTTTTG
22	HML2.LTR894	24630446	24631411	LTR5_Hs	(Subramanian et al.; 2011)	GGTCCC
22	HML2.LTR895	39011516	39012506	LTR5B	(Subramanian et al.; 2011)	GTTCT/GTTCC
chrUn_gl000219	K105; K111	175210	176178	LTR5_Hs	(Subramanian et al.; 2011)	GAATC
X	HML2.LTR896	251171	252143	LTR5A	(Subramanian et al.; 2011)	-
X	HML2.LTR897	1292288	1293545	LTR5B	(Subramanian et al.; 2011)	CTCTG/CACTG
X	HML2.LTR898	1899479	1900433	LTR5B	(Subramanian et al.; 2011)	GCCAGC/GCCAAC
X	HML2.LTR899	17115011	17115977	LTR5_Hs	(Subramanian et al.; 2011)	GAAAG
X	HML2.LTR900	18846490	18847415	LTR5_Hs	(Subramanian et al.; 2011)	5' truncated
X	HML2.LTR901	18979890	18980854	LTR5B	(Subramanian et al.; 2011)	AATGAT/AATGGT
X	HML2.LTR902	23659883	23660832	LTR5_Hs	(Subramanian et al.; 2011)	GGATCC
X	HML2.LTR903	26525467	26526440	LTR5B	(Subramanian et al.; 2011)	CTATG/CTGTG
X	HML2.LTR904	48555523	48556520	LTR5B	(Subramanian et al.; 2011)	CCTTA/CCTTG
X	HML2.LTR905	55835213	55836208	LTR5B	(Subramanian et al.; 2011)	-
X	HML2.LTR906	55997770	55998730	LTR5_Hs	(Subramanian et al.; 2011)	TCCAG

X	HML2.LTR907	57335164	57336541	LTR5_Hs	(Subramanian et al.; 2011)	CTGTAA
X	HML2.LTR908	58065321	58066304	LTR5B	(Subramanian et al.; 2011)	CTGTGG
X	HML2.LTR909	72733185	72734156	LTR5_Hs	(Subramanian et al.; 2011)	-
X	HML2.LTR910	87972052	87973043	LTR5B	(Subramanian et al.; 2011)	-
X	HML2.LTR911	90213331	90214282	LTR5_Hs	(Subramanian et al.; 2011)	CCTAC
X	HML2.LTR912	90950615	90951583	LTR5_Hs	(Subramanian et al.; 2011)	GGTGGG
X	HML2.LTR913	91170266	91171296	LTR5A	(Subramanian et al.; 2011)	TCTATG
X	HML2.LTR914	123679648	123680632	LTR5_Hs	(Subramanian et al.; 2011)	GATCCA
X	HML2.LTR915	125779511	125780478	LTR5_Hs	(Subramanian et al.; 2011)	GCTCCT
X	HML2.LTR916	127353937	127354944	LTR5A	(Subramanian et al.; 2011)	TTGGCT
X	HML2.LTR917	135030345	135031305	LTR5_Hs	(Subramanian et al.; 2011)	TGACT
X	HML2.LTR918	135302739	135303701	LTR5_Hs	(Subramanian et al.; 2011)	-
X	HML2.LTR919	135406098	135407066	LTR5_Hs	(Subramanian et al.; 2011)	AAAAA
X	HML2.LTR920	138295781	138296749	LTR5_Hs	(Subramanian et al.; 2011)	TTCCT
X	HML2.LTR921	138333487	138334454	LTR5_Hs	(Subramanian et al.; 2011)	TTTGAG
X	HML2.LTR922	139341915	139342888	LTR5_Hs	(Subramanian et al.; 2011)	CCAAGA/CCAAGG
X	HML2.LTR923	145720841	145721800	LTR5_Hs	(Subramanian et al.; 2011)	CTGGAA
X	HML2.LTR924	149626519	149627509	LTR5B	(Subramanian et al.; 2011)	GTACCT/GTATCT
X	HML2.LTR925	152129619	152130583	LTR5_Hs	(Subramanian et al.; 2011)	GCTTTT
X	HML2.LTR926	154521158	154522177	LTR5A	(Subramanian et al.; 2011)	CCAAG
X	HML2.LTR927	154909979	154911318	LTR5B	(Subramanian et al.; 2011)	TTTTAC
Y	HML2.LTR928	251171	252143	LTR5A	(Subramanian et al.; 2011)	-
Y	HML2.LTR929	1292288	1293545	LTR5B	(Subramanian et al.; 2011)	CTCTG/CACTG
Y	HML2.LTR930	1899479	1900433	LTR5B	(Subramanian et al.; 2011)	GCCAGC/GCCAAC
Y	HML2.LTR931	3782433	3783386	LTR5_Hs	(Subramanian et al.; 2011)	CCTAC
Y	HML2.LTR932	4634023	4635053	LTR5A	(Subramanian et al.; 2011)	TCTATG
Y	HML2.LTR933	6748889	6749855	LTR5_Hs	(Subramanian et al.; 2011)	GTGGTC
Y	HML2.LTR934	7179865	7180823	LTR5_Hs	(Subramanian et al.; 2011)	-
Y	HML2.LTR935	7711338	7712304	LTR5B	(Subramanian et al.; 2011)	GTCCAT
Y	HML2.LTR936	10124402	10125302	LTR5_Hs	(Subramanian et al.; 2011)	ATATAG
Y	HML2.LTR937	10176052	10176921	LTR5A	(Subramanian et al.; 2011)	5' truncated
Y	HML2.LTR938	12462876	12463843	LTR5_Hs	(Subramanian et al.; 2011)	-
Y	HML2.LTR939	13103823	13104789	LTR5_Hs	(Subramanian et al.; 2011)	GCAGT
Y	HML2.LTR940	14133176	14134199	LTR5A	(Subramanian et al.; 2011)	AAGGA/AATGA
Y	HML2.LTR941	15773167	15774133	LTR5_Hs	(Subramanian et al.; 2011)	AAAAT
Y	HML2.LTR942	19699481	19700447	LTR5_Hs	(Subramanian et al.; 2011)	CAAGAG
Y	HML2.LTR943	20976083	20977040	LTR5_Hs	(Subramanian et al.; 2011)	CTGTG/CTATG
Y	HML2.LTR944	22893039	22894006	LTR5_Hs	(Subramanian et al.; 2011)	CCTTTT
Y	HML2.LTR945	24526788	24527755	LTR5_Hs	(Subramanian et al.; 2011)	CCTTTT
Y	HML2.LTR946	25142348	25143315	LTR5_Hs	(Subramanian et al.; 2011)	CCTTTT

8. Appendix 3

List of segmentally duplicated insertions in the human reference genome.

Name	Duplication no.	Chromosome	GRCh38 / Hg38 position		State / TSD
15q25.2	1	15	84160268	84164397	5' and 3' truncated
Yq11.23a	1	Y	24250775	24254888	5 and 3' truncated
Yq11.23b	1	Y	25415255	25419369	5' and 3' truncated
16p11.2	2	16	34997024	34999771	5' truncated
LTRW15	2	16	34412057	34414806	5' TRUNCATED
20p11.21a	3	20	23693935	23695327	3' TRUNCATED
20p11.21b	3	20	23755863	23757247	3' TRUNCATED
HML2.LTR205	4	3	186865611	186866486	3' truncated
HML2.LTR381	4	7	65260266	65261141	3' truncated
LTRW19	4	20	4034107	4034982	3' TRUNCATED
1p36.21b	5	1	13206972	13216513	AGCGTA/AGTGTA
1p36.21c	5	1	13352736	13362257	AGTGTA
HML2.LTR50	6	1	120424751	120425723	CTGAAC
HML2.LTR52	6	1	145414873	145415845	CTGAAC
HML2.LTR55	6	1	146948313	146949300	CTGAAC
HML2.LTR60	6	1	148137833	148138805	CTGAAC
HML2.LTR62	6	1	149092877	149093849	CTGAAC
HML2.LTR63	6	1	144450894	144451866	CTGAAC
HML2.LTR54	7	1	146907212	146908212	FL, non-matching TSD
HML2.LTR61	7	1	148188817	148189817	FL, non-matching TSD
LTRM9	8	1	144555550	144556489	FL, non-matching TSD
LTRW2	8	1	146381151	146382090	FL, non-matching TSD
HML2.LTR527	9	10	79744593	79745590	FL, non-matching TSD
HML2.LTR529	9	10	87266813	87267810	AATGAT/GATGAT
HML2.LTR530	9	10	87470166	87471163	AATGAT/GATGAT
11p15.4	10	11	3447426	3456979	ATTTG/ATTTTG
4p16.1b	10	4	9657956	9667550	ATTTG
8p23.1b	10	8	8197178	8206699	ATTTG
8p23.1c	10	8	12216461	12225988	ATTTG
8p23.1d	10	8	12458983	12468498	ATTTG
HML2.LTR195	10	3	130123340	130124347	ATTTG
HML2.LTR579	10	11	67829985	67831010	ATTTG
HML2.LTR601	10	12	8264684	8265713	ATTTG
4p16.1a	10	4	9121786	9131367	ATTTG
4p16.3b	10	4	3977324	3986912	ATTTG
HML2.LTR172	11	3	75374163	75375179	AGTGAG
HML2.LTR174	11	3	75592672	75593700	AGTGAG
HML2.LTR187	11	3	125730453	125731468	AGTGAG
HML2.LTR188	11	3	125753034	125754061	AGCGAG/AGTGAG
HML2.LTR192	11	3	130015575	130016587	AGTGAG
HML2.LTR193	11	3	130038287	130039299	AGTGAG
HML2.LTR218	11	4	3883351	3884360	AGTGAG
HML2.LTR219	11	4	3905321	3906335	AGTGAG
HML2.LTR221	11	4	4120309	4121338	AGTGAG

HML2.LTR222	11	4	4150510	4151525	AGTGAG
HML2.LTR223	11	4	8955580	8956594	AGTGAG
HML2.LTR224	11	4	8987402	8988426	AGTGAG
HML2.LTR226	11	4	9465209	9466220	AGTGAG
HML2.LTR227	11	4	9489688	9490702	AGTGAG
HML2.LTR228	11	4	9518857	9519886	AGTGAG
HML2.LTR230	11	4	9751053	9752065	AGTGAG
HML2.LTR366	11	7	6860235	6861249	AGTGAG
HML2.LTR367	11	7	6883375	6884389	AGTGAG
HML2.LTR388	11	7	97941017	97942046	AGTGAG
HML2.LTR405	11	8	7079944	7080947	AGTGAG
HML2.LTR406	11	8	7240654	7241669	AGTGAG
HML2.LTR418	11	8	7597900	7598915	AGTGAG
HML2.LTR419	11	8	7699198	7700213	AGTGAG
HML2.LTR431	11	8	8045897	8046912	AGTGAG
HML2.LTR432	11	8	12027614	12028643	AGTGAA/AGTGAG
HML2.LTR435	11	8	12678408	12679433	AGTGAA
HML2.LTR544	11	11	3394552	3395567	AGTGAG
HML2.LTR545	11	11	3593659	3594685	AGTGAG
HML2.LTR576	11	11	67716490	67717502	AGTGAG
HML2.LTR577	11	11	67739337	67740349	AGTGAG
HML2.LTR581	11	11	67968248	67969277	AGTGAG
HML2.LTR583	11	11	71599375	71600389	AATGAG/AGTGAG
HML2.LTR584	11	11	71625997	71627026	AGTGAA/AGTGAG
HML2.LTR585	11	11	71899369	71900382	AGTGAG
HML2.LTR602	11	12	8409218	8410247	AGTGAG
HML2.LTR603	11	12	8431407	8432439	AGTGAG
HML2.LTR555	12	11	50131140	50132165	GTTCA
HML2.LTR556	12	11	50346425	50347450	GTTCA
HML2.LTR173	13	3	75393193	75394152	ATCAG
HML2.LTR194	13	3	130057289	130058265	ATCAA/ATCAG
HML2.LTR220	13	4	3925643	3926609	ATCAG/GTCAG
HML2.LTR368	13	7	6901997	6902963	ATCAA/ATCAG
HML2.LTR578	13	11	67757889	67758849	ATCAA/ATCAG
11p15.4a	14	11	3545352	3546375	5' TRUNCATED
3p12.3a	14	3	75537719	75538747	5' TRUNCATED
3q21.2a	14	3	125799251	125800275	5' TRUNCATED
4p16.1c	14	4	9034417	9035451	5' TRUNCATED
4p16.1d	14	4	9567314	9568339	5' TRUNCATED
4p16.3	14	4	4074290	4075313	5' TRUNCATED
8p23.1e	14	8	12624092	12625104	5' TRUNCATED
8p23.1f	14	8	7126987	7128007	5' TRUNCATED
8p23.1g	14	8	8100098	8101124	5' TRUNCATED
LTRM15	14	8	12565980	12567003	5' TRUNCATED
LTRM7	14	8	7186378	7187404	5' TRUNCATED
HML2.LTR225	15	4	9176783	9177814	CTTGGA
HML2.LTR407	15	8	7260529	7261560	CTTGGA
HML2.LTR408	15	8	7268151	7269182	CTTGGA
HML2.LTR409	15	8	7275773	7276804	CTTGGA
HML2.LTR410	15	8	7283395	7284426	CTTGGA

HML2.LTR411	15	8	7291017	7292048	CTTGGA
HML2.LTR412	15	8	7298637	7299668	CTTGGA
HML2.LTR413	15	8	7547398	7548440	CTTGGA
HML2.LTR414	15	8	7555053	7556085	CTTGGA
HML2.LTR415	15	8	7562703	7563735	CTTGGA
HML2.LTR416	15	8	7570349	7571380	CTTGGA
HML2.LTR417	15	8	7577998	7579029	CTTGGA
HML2.LTR420	15	8	7719068	7720099	CTTGGA
HML2.LTR421	15	8	7726716	7727747	CTTGGA
HML2.LTR422	15	8	7734364	7735395	CTTGGA
HML2.LTR423	15	8	7742011	7743042	CTTGGA
HML2.LTR424	15	8	7749659	7750690	CTTGGA
HML2.LTR425	15	8	7757307	7758338	CTTGGA
HML2.LTR426	15	8	7764955	7765987	CTTGGA
HML2.LTR427	15	8	7772603	7773644	CGTGGA/CTTGGA
HML2.LTR428	15	8	8010698	8011729	CTTGGA
HML2.LTR429	15	8	8018342	8019373	CTTGGA
HML2.LTR430	15	8	8025988	8027021	CTTGGA
HML2.LTR433	15	8	12171126	12172146	CTTGGA
HML2.LTR434	15	8	12413445	12414465	CTTGGA
HML2.LTR600	15	12	8220179	8221207	CTTGGA
HML2.LTR634	16	13	18719995	18720917	GATTTC
HML2.LTR874	16	21	13819880	13820848	GATTTC
HML2.LTR637	17	13	24591593	24592530	AGGCT
HML2.LTR638	17	13	24956546	24957486	AGGCT
HML2.LTR859	18	20	23982641	23983609	GGTCCC
HML2.LTR886	18	22	18513933	18514894	GGTCCC
HML2.LTR887	18	22	18771254	18772215	GGTCCC
HML2.LTR888	18	22	18347506	18348467	GGTCCC
HML2.LTR890	18	22	21205502	21206463	GGTCCC
HML2.LTR891	18	22	22649348	22650319	GGTCCC
HML2.LTR894	18	22	24630446	24631411	GGTCCC
LTRW13	18	13	18250955	18251919	GGTCCC
HML2.LTR131	19	2	132175021	132176042	TATTT
HML2.LTR482	19	9	40781491	40782501	TATTT
HML2.LTR486	19	9	64902286	64903296	TATTT
HML2.LTR649	19	14	18319453	18320475	TATTT
HML2.LTR680	19	15	19866100	19867111	TATTT
HML2.LTR746	19	18	15343747	15344757	TATTT
HML2.LTR873	19	21	13023725	13024739	TATTT
HML2.LTR130	20	2	131721200	131722222	CTGTT/GTGTT
HML2.LTR650	20	14	19269575	19270584	CTGTT/GTGTT
HML2.LTR651	20	14	19129412	19130441	CTGTT/GTGTT
HML2.LTR884	20	22	15851785	15852795	CTGTT/GTGTT
HML2.LTR656	21	14	23981494	23982489	GCTATA
HML2.LTR657	21	14	24024299	24025286	GCTATA
HML2.LTR707	22	16	34144086	34145098	CTCAC
LTRM1	22	20	29316282	29317182	CTCAC
HML2.LTR722	23	17	16834880	16835857	GGTGGG
HML2.LTR725	23	17	18436884	18437858	GGTGGG

HML2.LTR728	23	17	20506550	20507524	GGTGGG
HML2.LTR735	24	17	46283614	46284581	CCACAC
HML2.LTR738	24	17	64930022	64930983	CCACAC
HML2.LTR509	25	9	138132949	138133926	FL, non-matching TSD
HML2.LTR743	25	18	90214	91194	FL, non-matching TSD
HML2.LTR234	26	4	22597763	22598736	CCTGTG
HML2.LTR748	26	18	26927277	26928239	CCTGTG
HML2.LTR826	27	19	52519595	52520931	CTGGAT
HML2.LTR830	27	19	52740758	52742027	CTGGAT
HML2.LTR116	28	2	94744663	94745630	TGTGG
HML2.LTR478	28	9	40338734	40339708	TGTGG
HML2.LTR479	28	9	66036608	66037582	TGTGG
HML2.LTR480	28	9	42965007	42965981	TGTGG
HML2.LTR481	28	9	64478771	64479744	TGTGG
HML2.LTR124	29	2	110056070	110057040	TCGTGG
HML2.LTR125	29	2	110314426	110315396	TCGTGG
HML2.LTR175	30	3	75645328	75646291	FL, non-matching TSD
LTRM2	30	20	29440303	29441256	FL, non-matching TSD
HML2.LTR936	31	Y	10124402	10125302	ATATAG
LTRM5	31	22	11323908	11324867	ATATAG
LTRM6	31	22	11555908	11556858	ATATAG/ATATAC
LTRW22	31	20	30849917	30850867	ATATAG/ATATAC
HML2.LTR882	32	21	43147130	43148089	AATCC
LTRW23	32	21	6538815	6539774	AATCC
5q33.3	33	5	156657706	156666885	ACTGC
HML2.LTR310	33	5	152797290	152798315	ACTGC
HML2.LTR354	33	6	131478658	131479644	TCTGC/TCTAC
HML2.LTR384	34	7	72951214	72952196	CAATCC/TAATCC
HML2.LTR386	34	7	75412885	75413870	CAATCC/TAATCC
HML2.LTR477	35	9	33514870	33515836	TAGAA/CAGAA
HML2.LTR494	35	9	97232136	97233105	TAGAA
HML2.LTR484	36	9	41105161	41106192	TCCACC
HML2.LTR487	36	9	68325230	68326261	TCCACC
HML2.LTR896	37	X	251171	252143	FL, non-matching TSD
HML2.LTR928	37	Y	251171	252143	FL, non-matching TSD
HML2.LTR897	38	X	1292288	1293545	CTCTG/CACTG
HML2.LTR929	38	Y	1292288	1293545	CTCTG/CACTG
HML2.LTR898	39	X	1899479	1900433	GCCAGC/GCCAAC
HML2.LTR930	39	Y	1899479	1900433	GCCAGC/GCCAAC
HML2.LTR911	40	X	90213331	90214282	CCTAC
HML2.LTR931	40	Y	3782433	3783386	CCTAC
HML2.LTR913	41	X	91170266	91171296	TCTATG
HML2.LTR932	41	Y	4634023	4635053	TCTATG
HML2.LTR944	42	Y	22893039	22894006	CCTTTT
HML2.LTR945	42	Y	24526788	24527755	CCTTTT
HML2.LTR946	42	Y	25142348	25143315	CCTTTT
Yq11.23	43	Y	23880366	23881090	GAAAT
Yq11.223	43	Y	25789017	25789741	GAAAT

HML2.LTR785	44	19	36233354	36234332	GGCTG
HML2.LTR810	44	19	48891605	48892893	GGCTG
HML2.LTR721	45	17	16399849	16401128	CTTTGG
HML2.LTR729	45	17	22229630	22230620	CTTTGG/CTTTTG
HML2.LTR868	45	20	36967836	36968808	CTTTGA/CTTTGG
HML2.LTR213	46	4	19027	20027	AACGT/AACAT
4q35.2	46	4	190106259	190113550	AACGT/AACAT