Računalna analiza retrotranspozona MT kod sojeva miševa C57BL/6J i PWK/PHJ

Štancl, Paula

University of Zagreb

Faculty of Science

Department of Biology

Paula Štancl

Computational analysis of MT retrotransposons in mouse strains C57BL/6J and PWK/PhJ

Graduation thesis

Zagreb, 2020

This thesis is created in the bioinformatics group at the Division of Molecular Biology, under the supervision of Professor Kristian Vlahoviček and co-supervision of Assistant Maja Kuzman. The thesis is submitted for grading to the Department of Biology at the Faculty of Science, University of Zagreb, with the aim of obtaining the Master's degree in molecular biology.

I express my deepest and sincere gratitude to my mentor, Professor Kristian Vlahoviček, for many given opportunities, professional guidance, patience, and provided knowledge during these past years.

I am thankful to all the members from the Bioinformatics group; Maja, Dunja, Filip, Antonio and Rosa, for all the advice and amazing time spent in the laboratory. I am especially grateful to Maja whom I cannot thank enough for every piece of advice and everything she has taught me. I also thank Pepa for a fantastic time spent at the lab in Prague and the idea for this thesis.

Thank you, Mom, for everything you had done for me and the rest of my family for supporting me. Also, I am thankful to my friends who have always lifted my spirit when needed.

BASIC DOCUMENTATION CARD

University of Zagreb
Faculty of Science
Division of Biology

Graduation thesis

Computational analysis of MT retrotransposons in mouse strains C57BL/6J and PWK/PhJ

Paula Štancl
Department of Molecular Biology, Horvatovac 102A, 10000 Zagreb, Croatia

Rodent-specific MT retrotransposons belong to a class of LTR retrotransposons. Full-length MT elements contain two long terminal repeats (LTRs) and an internal sequence. Due to ectopic recombination, the internal sequence of full-length MT elements is excised producing a solo LTR. The standard annotation of retrotransposons is done with RepeatMasker program which one of the major drawbacks is the precise annotation of full-length elements. The goal of this thesis was to improve the RepeatMasker annotation of MT elements and to identify homologous between a classical mouse strain C57BL/6J and a wild-derived mouse strain PWK/PhJ. The developed computational method focuses on the conserved length of their LTRs and has resulted in a larger number of annotated full-length MT elements in in the mouse strain C57BL/6J. Phylogenetically younger and still active annotated full-length MT elements, MTA and MTB, had more new C57BL/6J-strains specific elements that have emerged after the separations of the strains. On the other hand, older MT elements were more conserved among strains. Conserved and strain-specific MT element integrations into genes mostly occurred in introns of protein-coding genes. Identified new C57BL/6J-strains specific elements integrated into exons of olfactory receptor genes, may contribute to phenotypic differences between mouse strains C57BL/6J and PWK/PhJ.

(49 pages, 25 figures, 4 tables, 58 references, original in English)

Thesis deposited in the Central Biological Library

Key words: LTR retrotransposons, mouse strains, computational genomics

Supervisor: Professor Kristian Vlahoviček, PhD

Assistant Supervisor: Maja Kuzman, MSc

Reviewers: Professor Kristian Vlahoviček, PhD

Assoc. Prof. Damjan Franjević, PhD

Asst. Prof. Duje Lisičić, PhD

Substitution: Asst. Prof. Rosa Karlić, PhD

Thesis accepted: 14.07.2020.

TEMELJNA DOKUMENTACIJSKA KARTICA

Sveučilište u Zagrebu
Prirodoslovno-matematički fakultet
Biološki odsjek Diplomski rad

Računalna analiza retrotranspozona MT kod sojeva miševa C57BL/6J i PWK/PhJ

Paula Štancl
Zavod za molekularnu biologiju, Horvatovac 102A, 10000 Zagreb, Hrvatska

Retrotranspozoni MT pripadaju klasi LTR retrotranspozona, specifičnih za glodavce. Cjeloviti MT elementi sadrže dva duga terminalna ponavljanja (engl. long terminal repeat, LTR) i internu sekvencu. Zbog ektopične rekombinacije, izrezuje se interna sekvenca cjelovitog MT elementa stvarajući samostalni LTR. Standardna anotacija retrotranspozona radi se programom RepeatMasker, čiji je glavni nedostatak precizna anotacija cjelovitih elemenata. Cilj ovog diplomskog rada bio je poboljšati već postojeću anotaciju cjelovitih MT elemenata dobivenih upotrebom programa RepeatMasker i identificirati homologne MT elemente između klasičnog C57BL/6J i divljeg PWK/PhJ mišjeg soja. Razvijena računalna metoda anotacije cjelovitih MT elemenata usredotočena je na filtriranje njihovih očuvanih duljina LTR-ova i rezultirala je većim brojem anotiranih cjelovitih MT elemenata u C57BL/6J genomu za razliku od PWK/PhJ genoma. Filogenetski mlađi i jedini aktivni cjeloviti MT elementi, MTA i MTB, imali su više novih specifičnih elemenata kod C57BL/6J soja, koji su se pojavili nakon razdvajanja sojeva, dok su stariji MT elementi očuvani među mišjim sojevima. Integracije očuvanih i specifičnih MT elemenata kod sojeva većinom su u intronima protein-kodirajućih gena. Identificirani novi specifični elementi C57BL/6J soja integrirani su u egzone gena olfaktornih receptora te mogu pridonijeti fenotipskim razlikama između mišjih sojeva C57BL/6J i PWK/PhJ.

(49 stranica, 25 slika, 4 tablice, 58 literaturna navoda , jezik izvornika: engleski)

Rad je pohranjen u Središnjoj biološkoj knjižnici.

Ključne riječi: LTR retrotranspozoni, sojevi miševa, računalna genomika

Voditelj: prof. dr. sc. Kristian Vlahoviček

Neposredni voditelj: Maja Kuzman, mag. biol. mol.

Ocjenitelji: prof. dr. sc. Kristian Vlahoviček
izv. prof. dr. sc. Damjan Franjević
doc. dr. sc. Duje Lisičić

Zamjena: doc. dr. sc. Rosa Karlić

Rad prihvaćen: 14.07.2020.

Abbreviations

TE	transposable element
LINE	long interspersed nuclear element
SINE	short interspersed nuclear element
LTR	long terminal repeat
ERV	endogenous retrovirus
EST	expressed sequence tag
ncRNA	non-coding RNA
miRNA	micro RNA
lncRNA	long-non-coding RNA
bp	base pair

Table of content

1 Introduction

1.1 Transposons: repetitive mobile sequences

Most eukaryotic genomes consist of highly repeated DNA sequences. Britten and Kohne (1964) were the first ones to describe the presence of these sequences in hundreds of thousands of copies per genome using reassociation studies. Term “repeats” is often applied to these sequences and they can be roughly divided into following categories (Smit et al. 2013-2015):

1. Simple repeats: a very short sequence typically 1-5 bp long (e.g. A, CGG) repeated multiple times.
2. Tandem repeats: longer sequences of 100-200 bp repeated multiple times. They are typically found at centromeres and telomeres of chromosomes.
3. Segmental duplications: large blocks of 10-300 kilobases that have been copied to another region in the genome. Interchromosomal duplication occurs onto non-homologous chromosomes while intrachromosomal onto the same chromosome.
4. Interspersed repeat: copies of sequences of varying length dispersed through the genome such as processed pseudogene, retrotranscripts, SINES, DNA transposons, retrovirus retrotransposons and non-retrovirus retrotransposons.

Transposable elements (TEs), discovered by Barbara McClintock in the 1950s (McClintock 1950), compromise the majority of interspersed repeats and have the ability to change their position in the genome. They are sometimes referred to as “selfish” DNA because their success (that is, an increase in copy number) is either neutral or harmful to the host organism (Werren et al. 1988). Transposable elements are classified into two groups based on their transposition intermediate (Finnegan 1989): RNA (class I or retrotransposons) or DNA (class II or DNA transposons). The transposition mechanism of retrotransposons is commonly called “copy and paste” and that of DNA transposons, “cut and paste”. Retrotransposons and DNA transposons contain elements that can be further divided and classified as either autonomous or non-autonomous. Autonomous elements encode all the proteins of machinery required for their transposition, whereas non-autonomous elements do not. Non-autonomous elements are unable to transpose on their own because their internal coding sequence may contain mutations such as substitutions, large internal deletions, internal rearrangements or even contain insertions of unrelated transposable elements. However, many non-autonomous elements still contain the sequences necessary for recognition by the transposase and other factors involved in transposition, and these elements can be mobilized by other autonomous elements (Hartl et al. 1992).

1.2 Mammalian retrotransposons

All retrotransposons use the mechanism of transposition via RNA mediators, known as the "copy and paste" mechanism. Although each type of retrotransposon has a slight variation in the mechanisms of transposition, the main steps are in general similar. In the first "copy" step, the RNA intermediate is transcribed by RNA polymerase II and then, in the "paste" step reverse-transcribed into DNA by a TE-encoded reverse transcriptase (RT), followed by reintegration into the genome (Figure 1). This process is not perfect and many truncated or mutated new retrotransposon copies can be produced (Martin et al. 2005; Szak et al. 2002). Therefore, many retrotransposons in the genome are only partial and are no longer capable of retrotransposition (de Parseval et al. 2003). There are two main classes of retrotransposon: long terminal repeat (LTR) and non-LTR retrotransposons (Padeken et al. 2015). Each of these retrotransposon classes has different characteristics and properties; and can be in different forms present in the genomes (Figure 2A).

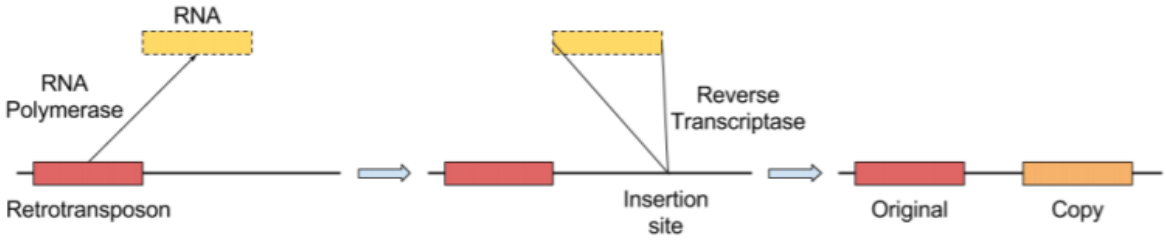

Figure 1. The basic steps in the mechanism of transposition for retrotransposons. Retrotransposon is transcribed into RNA which is reverse transcribed into DNA by reverse transcriptase and integrated into the genome at a new position. Adapted from (Gardner 2019).

1.2.1 Non-LTR retrotransposons

The non-LTR retrotransposons include long interspersed elements (LINEs) and short interspersed elements (SINEs). Both LINEs and SINEs can be identified by the presence of a repetitive tail, usually poly-A, and a lack of LTRs (Platt et al. 2018).

The most widespread LINEs among vertebrates are LINE1 (L1) and LINE2 (L2). Only L1 is still active in the human genome (Mills et al. 2007) and contributes to more than 20% of genome in human and mouse (Waterston et al. 2002). Complete full-length LINEs are 4-7 kilobases long and contain a variable internal promoter and two open reading frames (ORFs). Open reading frame or ORF1 encodes a trimeric protein with RNA-binding properties and nucleic-acid chaperone activity. ORF2 encodes endonuclease and reverse-transcriptase. Both ORFs are

necessary for L1 retrotransposition and have a strong cis-preference for the mRNA that encodes them to ensure that the element is retrotransposed. Most copies of L1 are not active in mammalian genomes due to imperfect retrotransposition. They are usually present in the genome in the 5-truncated form as a result of the incomplete reverse-transcription (Figure 2A.) (Crichton et al. 2014). It is most likely that LINEs have originated from group II introns that are mobile genetic elements in bacterial and mitochondrial genes. They have similar reverse transcriptases and use the similar mechanism for retrotransposition (Lambowitz and Belfort 2014).

Unlike LINEs that are autonomous retrotransposons, SINEs depend on the LINE protein machinery for retrotransposition (Dewannieux et al. 2003). Also, they are much shorter than the LINEs, up to 1000 bp long. There are several SINE classes, like B1 and B2, in the mouse genome that together comprise around 8% of the genome (Waterston et al. 2002). SINEs evolved from RNA genes; B1 (homolog to human Alu) is derived from 7SL RNA, and B2 is derived from tRNA (Gagnier et al. 2019).

1.2.2 LTR retrotransposons

The LTR retrotransposons are characterized by the presence of long terminal repeats (LTRs) that flank the internal sequence. The reason why they are also known as endogenous retroviruses (ERVs) is because both exogenous retroviruses and LTR retrotransposons contain a *gag* and *pol* gene. The *gag* gene encodes a viral particle coat and the *pol* gene encodes a reverse transcriptase, ribonuclease H, and an integrase, which provide the necessary enzymatic machinery for reverse transcription and integration into the host genome (Anisimova 2019). Exogenous retroviruses contain an *env* gene that encodes an envelope which enables retroviruses to infiltrate into other cells whereas only some LTR retrotransposon may contain the *env* gene (Kazazian 2004).

LTR retrotransposons have lost their ability to horizontally transfer from cell to cell, instead their transposition is limited within a genome of a single cell. Like already mentioned LINEs, most of the LTR retrotransposons present in the genomes are not complete full-length elements (Figure 2A). Homologous recombination between identical 5’ and 3’ LTR of a full-length element leads to excision of the internal sequence and one LTR, producing a solo or solitary LTR (Figure 2B). It is estimated that there are about 90 000 copies in both primate and rodent genomes, primarily in the form of solitary LTRs (Smit 1996). The formation of solo LTRs may be significant for the host organisms, not only due to the removal of the entire coding sequence of a provirus, but it can also alter the cis-regulatory or transcriptional activity of LTR (Thomas et al. 2018).

LTR retrotransposon compromises about 8% of the human genome and around 10% of the mouse genome (Lander et al. 2001; Waterston et al. 2002). Around 150 different types of LTR retrotransposons are classified into ERV1, ERVK, ERVL and ERVL-MaLR families depending on their phylogenetic relationship (Crichton et al. 2014). These families are retrotranspositionally active in the mouse genome, but not in the human genome, and their *de novo* insertions can cause spontaneous mutations in mice (Maksakova et al. 2006).

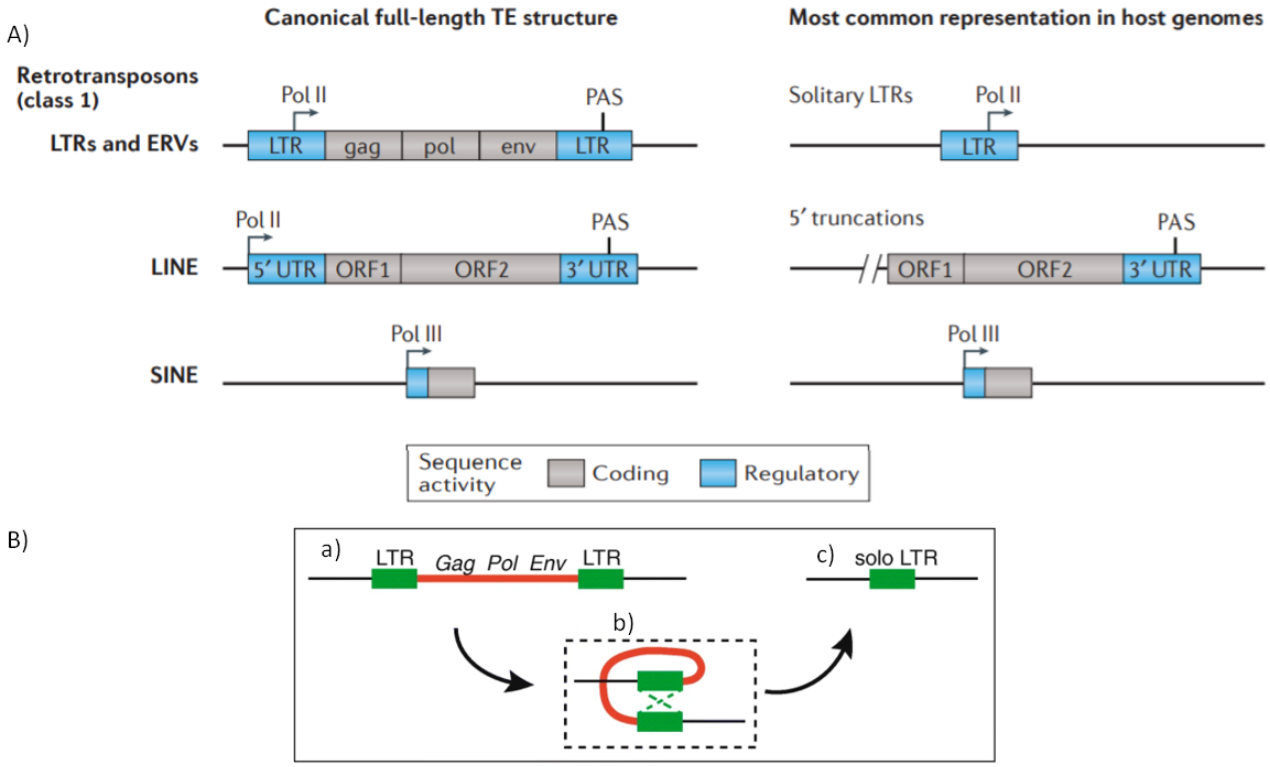

Figure 2. A) Schematic representation of retrotransposons (Class I) and their most common representation in the host genomes. Complete full-length version of each retrotransposon group is shown on the left while their most occurring structures in the genome are on the right. Adapted from (Chuong et al. 2017). B) Structure of a full-length LTR retrotransposon (a) with its internal coding sequence (red line) flanked by two long terminal repeats (LTR). Ectopic recombination occurs between the 5' and 3' LTRs (b) leading to the excision of the internal sequence alongside one LTR, resulting in the formation of a solo LTR (c). Adapted from (Thomas et al. 2018).

1.2.2.1 MT retrotransposons

MT retrotransposons belong to a class of LTR retrotransposons that is present only in rodent lineage. Because MT elements lack open reading frames that are present in other LTR retrotransposons, they are classified as Mammalian apparent LTR retrotransposons or Mammalian LTR-Retrosequence (MaLR) (Smit 1996). Another characteristic that differentiates MT elements from other LTR retrotransposons, is that their average size is much shorter due to the loss of open reading frames. MT elements are around 1 to 1.5 kilobases while full-length LTR retrotransposons can range from 5 to 11 kilobases (Franke 2016). Even though MT elements lack protein machinery necessary for retrotransposition and therefore are non-autonomous elements, they have multiple regulatory signals that enable them to influence the transcriptome of the host (Franke et al. 2017; Peaston et al. 2007). MT 5' LTR has three types of regulatory signals: TATA box, which is part of the promoter sequence of most eukaryotic genes; a GT splicing donor and a polyadenylation signal (Figure 3.) (Smit 1996).

MT elements are split into 5 categories: MTA, MTB, MTC, MTD and MTE. MTA elements represent the youngest element while MTE represents the oldest element of MT retrotransposons (Smit 1996). Like LTR retrotransposons, all MT elements are not present in their full-length rather they can be found mainly as solo LTRs, as a result of ectopic recombination of the full-length element, in the mouse genome.

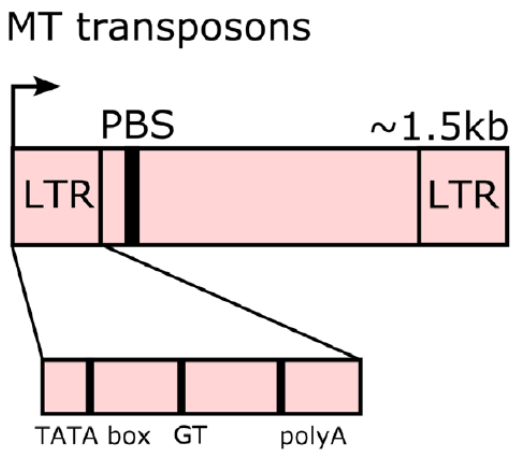

Figure 3. Structure of the full-length MT retrotransposons. The element contains two LTRs, internal primer binding site (PBS) which serves as the initiator of reverse transcription and lacks any open reading frames. 5' LTR contains a TATA box, a GT splice donor and a putative polyA binding site. Adapted from (Franke 2016).

1.3 Identification of transposable elements

One of the main challenges in studying retrotransposons is their accurate identification and classification in host genomes. Various innovative computational methods have been developed to tackle this problem. There are two distinct methods regarding the identification and annotation of TEs based on the genome; homology-based and *de novo* methods. Homology-based or repository-based methods rely on finding similarities between genome sequences and known TE consensus sequences. An alternative method of identification of TEs that does not rely on the genome is *de novo* annotation using raw reads (Goerner-Potvin and Bourque 2018). Short summary of these methods is shown in Figure 4.

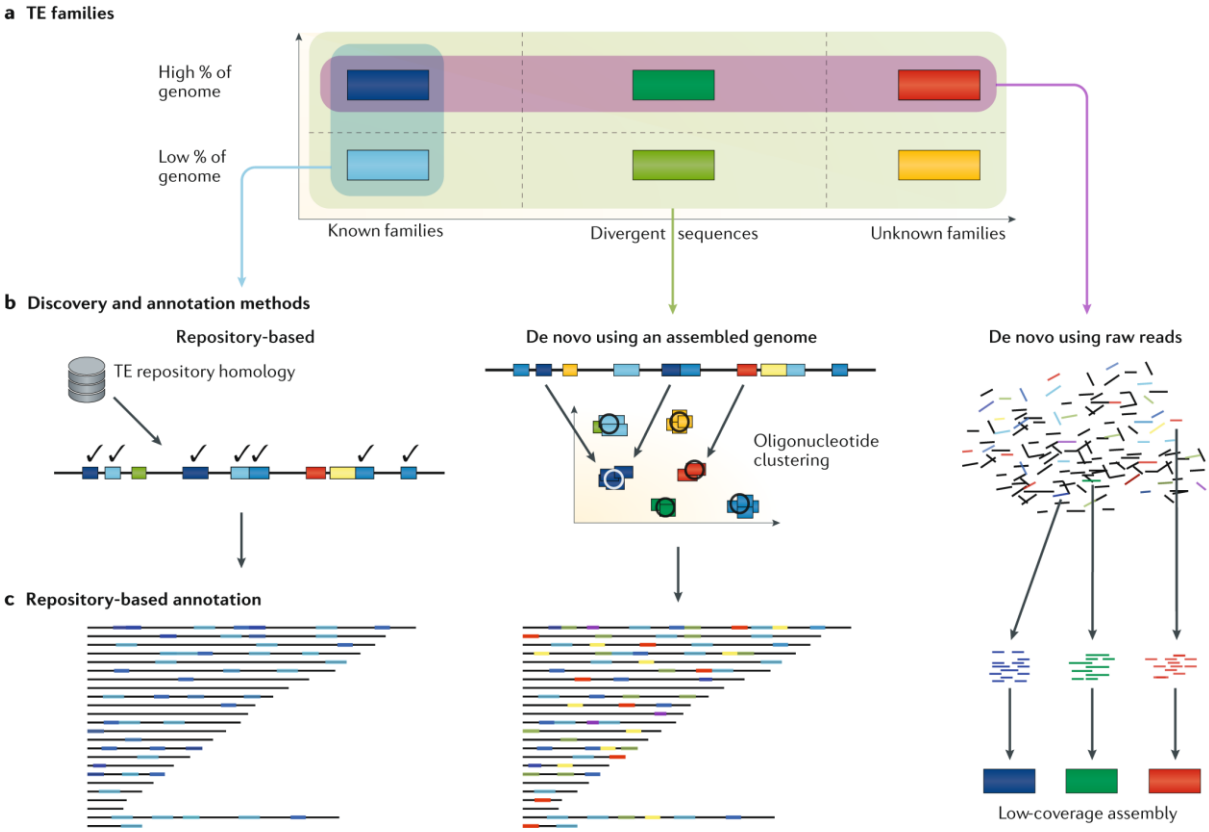

Figure 4. Main approaches of TEs annotation. A) Groups of TEs sequences based on sequence: sequences from known families of TEs consensus sequences (left), more divergent TEs sequences from a known consensus sequence (middle), sequences from unknown families of TEs that are not present in the repository (right). High (top) and low (bottom) occurring sequences of these sequences in the genome. B) The assembled genome is searched for sequences that are similar to TE consensus sequences from repository in the repository-based approach. De novo methods takes the oligonucleotide sequences from assembled genome and clusters them by sequence similarity. In de novo methods using raw reads, the sequencing reads are directly assembled into TE sequences. C) Repository based annotations returns annotated TEs

sequences that are homologous to known TE consensus sequences and excludes sequences that are too divergent from known consensuses. De novo method using an assembled genome annotates both known and unknown TE sequences and it classifies the annotated sequences into TE families based on the existing TE repositories. De novo method using raw reads returns annotated TEs based on their genomic frequencies regardless of genome annotation. Adapated from (Goerner-Potvin and Bourque 2018).

The most commonly used software for TE annotation is homology-based RepeatMasker (Smit et al. 2013-2015). RepeatMasker is a program that screens DNA sequences for interspersed repeats and low complexity DNA sequences. The output of the program is a detailed annotation of the repeats that are present in the query sequence as well as a modified version of the query sequence in which all the annotated repeats have been masked (replaced by Ns) (Smit et al. 2013-2015). RepeatMasker identifies repetitive sequences in genome sequences based on homology with a pre-existing library of annotated repeating elements. The reference sequences for TEs are based on Repbase (Jurka et al. 2005): a library of manually curated submissions from researches, maintained by the Genetic Information Research Institute (GIRI). There are several reasons why RepeatMasker is considered a gold standard for the annotation of repetitive elements. To begin with, it is persistently maintained and updated. Also, its library Repbase is manually curated and represents the most comprehensive library of repeats available, especially for human and mouse genomes. As it is one of the most popular tools in the field, many researchers can get reproducible and comparable results. Despite all that, it is not without flaws. Since Repbase is manually curated it is susceptible to biases because it is based on people's knowledge and expertise in that field of research. As well as Repbase data and methods are not publicly available which makes it difficult to compare with other methods. Besides the issues with its library Repbase, RepeatMasker is prone to error in the annotation of full-length and solo repetitive elements. Gaps in the sequences or other inserted repetitive elements within a full-length element can mislead RepeatMaker to annotate the element as solo. So, to obtain accurate annotation of TEs and throw light upon the evolution of specific TEs between different species and within different strains, the assembled genomes of an organism must be of the highest quality. All of the publicly available genomes are annotated using RepeatMasker and most of them are of low quality. Best assembled genomes are those of humans and mice with a lower number of gaps in sequences compared to other assembled genomes.

1.4 The laboratory mouse

The house mouse (*Mus musculus* Linnaeus, 1758) has become the preferred mammalian model since the early days of genetics. Scientists have long been fascinated with various spontaneously arising phenotypes such as coat colour, waltzing, albino and yellow mice. Historical

records show that Mendel bred mice to study inheritance using the coat color traits of mice until he was requested by his bishop to stop experimenting with mice and turned to continue his work with garden peas (Paigen 2003). Nowadays, there are numerous different mouse strains obtained by genetic manipulation of their genome that have enriched our knowledge of mammalian biology, ranging from embryonic development to metabolic disease, histocompatibility, immunology, and cancer. The major event that enabled numerous research on mice was the production of a high-quality draft sequence of the mouse genome from female mice of the C57BL/6J strain (Waterston et al. 2002). Not only did this allow scientists to manipulate the mouse genome in order to produce different mouse strains, but also to infer about homology between human and mouse. For instance, while the mouse genome is smaller compared to the human genome, the estimated number of genes in the mouse genome is higher, around 25 000, while humans have about 20 000. Although the first drafts of mouse sequences were of high-quality, they still contained several gaps due to the limitations in the sequencing protocol (Waterston et al. 2002). Since then, there were many improvements of the assembly of the mouse genome and the latest assembly was released by the Genome Reference Consortium (Genome Reference Consortium Mouse Build 38 patch 6 or GRCm38.p6).

Despite the many improvements on the reference mouse genome of C57BL/6J, there was a lack of complete high-quality genetic information for other common laboratory strains that were extensively being used in research. This lack of information has been a confounding factor in many experimental designs and interpretations. As an example, different structures of transcripts between C57BL/6J and the non-obese diabetic (NOD) mouse are due to large structural differences in specific loci (Steward et al. 2013). So, The Mouse Genomes Project (MPG) was established in 2009 with the goal to sequence the genomes of the most common laboratory mouse strains, and a selected set of wild-derived inbred strains (Adams et al. 2015). They have produced high-quality *de novo* assemblies and strain-specific gene annotation for 16 mouse strains (129S1/SvImJ, A/J, ARK/J, BALB/cJ, C3H/HeJ, C57BL/6NJ, CAST/EiJ, CBA/J, DBA/2J, FVB/NJ, LP/J, NOD/ShiLtJ, NZO/HlLtJ, PWK/PhJ, SPRET/EiJ, and WSB/EiJ).

1.5.1 Classical and wild-derived mouse strains

Most commonly used mouse models are inbred strains. An inbred strain is defined as a strain that has been through more than 20 generations of brother-sister mating, making mice from the same inbred strain genetically identical. Inbred mice are referred to by a combination of capitalized letters and numbers, like NOD. Substrains are colonies of the same inbred strain that have been separated for at least 18 generations. Substrains are identified by a slash followed by a number of and/or letters, followed by the institution or laboratory responsible for the substrain.

For example, DBA/1J and DBA/2J are two different substrains of DBA and both are maintained at the Jackson Laboratory (J) (Suckow et al. 2001).

Inbred laboratory strains of mice are organized into two groups, classical and wild-derived strains. Classical inbred laboratory strains descended from a relatively small number of genetic founders such as M. m. musculus, M. m. castaneus, M. m. domesticus and the hybrid M. m. molossinus (Frazer et al. 2007). Wild-derived strains are captured in the wild and are genetically distinct from classical laboratory mice. They have a number of complex phenotypic characteristics and are valuable tools for genetic mapping, evolution and systemic research. Sul and colleague (2018) observed that these two strains are close to each other in the phylogeny and that many genetic differences clearly separate the two using genetic variant information at 140 000 SNPs for each strain (Figure 5). A single nucleotide polymorphism (SNP) is a variation of a single base at a certain position in the genome that is present in over 1% of the population. Besides the genetic differences, there is also contrast in the body weights between classical and wild-derived strain. Classical inbred strains have larger body weight compared to wild-derived strains (Figure 5).

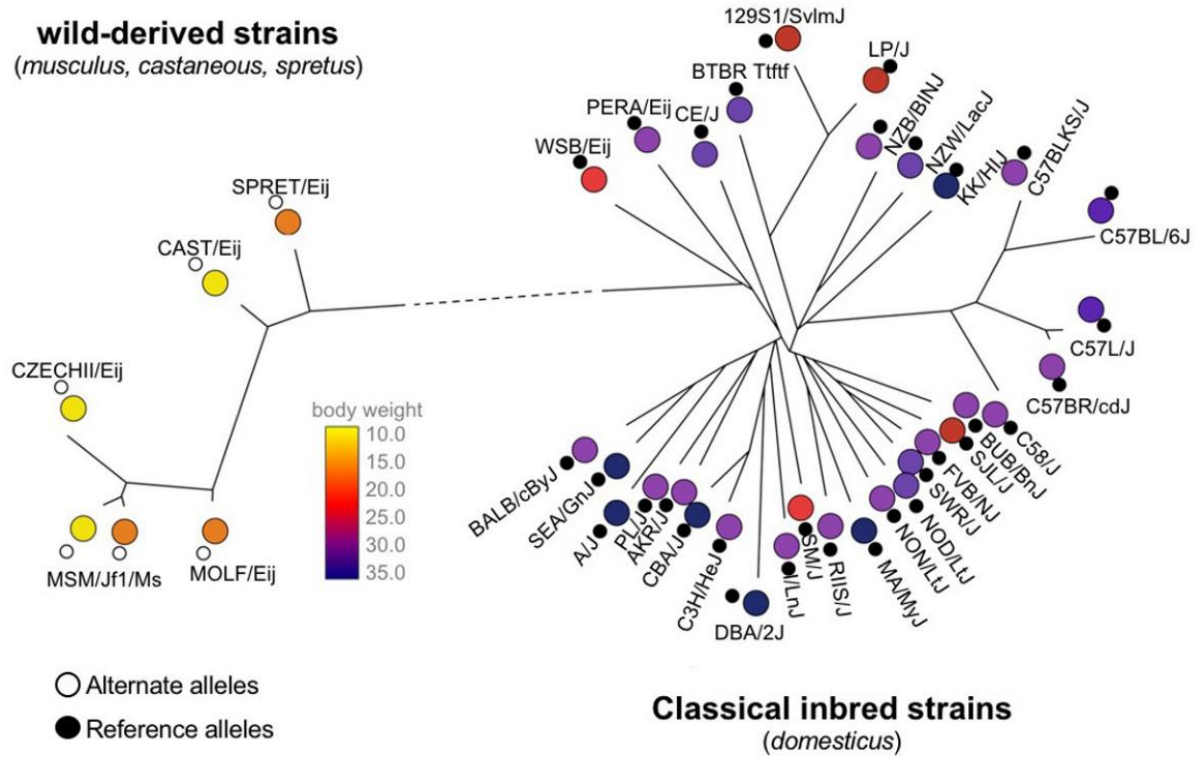

Figure 5. A phylogenetic tree between 38 inbred mouse strains using 1400 000 mouse HapMap SNPs. Strains are clustered into two groups: classical inbred strains and wild-derived strains. Colours represent the body weight phenotypes, obtained from the Mouse Phenome Database, of the strains. Adapted and adjusted from (Sul et al. 2018).

The inbred mouse strain C57BL/6J is the most commonly cited and well-characterized laboratory strain used in biomedical research. It is a preferred model in researches that include developmental biology, genetics, neurobiology and many more. The C57BL/6J mouse strain has a number of available genetic, phenotypic and genomic data such as already mention a high-quality reference genome (GRCm38.p6). Most of our knowledge about retrotransposons and their impact on the mouse genome comes from research done on the C57BL/6J mouse strain.

1.5 Impact of retrotransposons on mouse genome evolution and function

The majority of the repetitive sequences originated in transposable elements (TEs) (Anisimova 2019). So, it is not surprising that because of their large contribution in the genome which can range up to 40 % mouse genome (Waterston et al. 2002) they have a significant influence in shaping the host genome and its evolution. Most of the inserted TEs are still active in the genome and can influence the transcriptome even though most of them are truncated or highly mutated and therefore incapable of transposition. TEs can generate a variety of mutations depending on the exact insertion location of TEs in the genome (Gogvadze and Buzdin 2009).

For instance, if a new TE copy inserts in or near an existing gene it can influence the gene's expression. This way it can either serve as a new regulatory element leading to gene overexpression or it can disrupt existing regulatory regions and inactivate the gene. The integration of TEs in exons often disrupts the gene by introducing a stop codon or a shift in the open reading frame (Casacuberta and González 2013). LTR retrotransposons are the primary group of TEs responsible for generating new gene promoters (Gardner 2019). The best known and well-studied example of an LTR serving as a promoter is the insertion of an intracisternal A particle (IAP), a mouse-specific LTR retrotransposon, upstream of Agouti viable yellow gene in mouse (Figure 6A) (Dolinoy 2008). Depending on the degree of methylation of the 5′ LTR of IAP, the coat color of mice can range from yellow (unmethylated) to pseudoaguti (methylated) (Figure 6B). Therefore, the IAP element is an example when an LTR retrotransposon contributes to the production of metastable epialleles in mice. Metastable epialleles are alleles that are differentially expressed in genetically identical individuals due to epigenetic modifications (Dolinoy et al. 2007). IAP elements account for most of the reported mutations in mice (Gagnier et al. 2019) and alongside MaLR class are responsible for the majority of the variation between laboratory mouse strains (Nellåker et al. 2012).

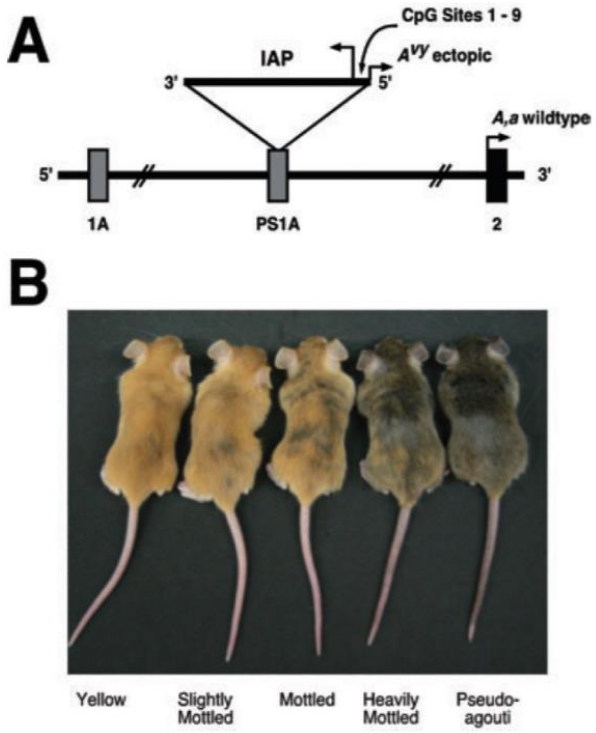

Figure 6. The IAP elements influence the gene expression of the Agouti gene in mice. A) The insertion of intracisternal A particle (IAP), a mouse-specific LTR retrotransposon, upstream of Agouti viable yellow gene. B) Genetically identical individuals of mice with different levels of methylation of IAP. Adapted from (Dolinoy 2008).

Since active TE can be highly mutagenic leading to deletions, duplications, and sequence rearrangements, the host genome has evolved an array of mechanisms to suppress their activity. TEs are primarily repressed by the epigenetic silencing pathways including histone modification, DNA methylation and small RNA machinery (Molaro and Malik 2016). Although genome-wide epigenetic silencing of TEs is a crucial step in early mouse (and other animals) development, recent studies revealed that LTR retrotransposons have a significant impact on the gene expression in the germline and later on in the development of mice.

Peaston and colleagues (Peaston et al. 2004) showed that TEs, mainly retrotransposons from the MaLR family, can significantly influence the transcriptome in full-grown oocytes in mice. By sequencing short fragments of cDNA sequences (expressed sequence tags, ESTs), they were able to detect that MT retrotransposons, members of the MaLR family, account for over 12% of the total sequences in the library. Moreover, they found that the MT retrotransposons act as alternative promoters and first exon in a subset of mouse genes. MT elements do not only contribute to the tissue-specific reprogramming of the transcriptome in oocytes but are also required for the normal phenotype of a mouse. For instance, excision of the MTC element

integrated into the sixth intron of the mouse DICER1 gene creates an oocyte specific DICER1 isoform which leads to female mouse sterility (Flemr et al. 2013). Another research, conducted by Franke and colleagues (Franke et al. 2017), has also shown the importance of MT elements during the oocyte-to-embryo transition. They found that not only do the MT elements contribute to the formation of new long-non-coding RNA (lncRNA) genes but that they also support the evolution of new protein-coding genes. The youngest members of the MaLR family, MTA elements, had a greater contribution to serving as promoters or first exons of genes in contrast to the oldest MTE elements.

2 Goals

Analysis and annotation of repetitive elements can present quite a challenge. Fortunately, there currently exist various software tools utilizing different approaches that tackle these challenges. The most popular and widely used program RepeatMasker offers reproducible identification of transposons in genomes, but relies on the consensuses and can be flawed when in identifying full-length retrotransposons. Today, there are many high-quality assembled genomes coupled with RepeatMakser annotations of retrotransposons for many different laboratory mouse strains. However, most of our knowledge about rodent specific MT retrotransposons and their contribution to the mouse genome is limited to studies done on the reference genome of the C57BL/6J mouse strain. Although some studies have explored the differences in the abundance of the MaLR family to which MT elements belong, there is still a lack of research about new (strain-specific) and conserved homologous MT elements between mouse strains.

The goals of this research are:

1. compare the composition of repetitive elements between a reference mouse strain C57BL/6J and a wild-derived mouse strain PWK/PhJ;
2. develop a computational method for improving the existing RepeatMasker annotation of full-length MT elements
3. identify shared conserved and strain-specific full-length MT elements between these strains and analyse their distributions in the genomes.

3 Materials and methods

3.1 Reference genomes and annotations

Biological databases are organized collections of biological data, generally stored and accessed electronically from a computer system. They contain information from many different research areas such as genomics, proteomics, metabolomics and many others. Databases that provide well organized and annotated biological features for a large number of genomes are UCSC Genome Browser and Ensembl.

The University of California Santa Cruz (UCSC) Genome Browser (genome.ucsc.edu) is a popular Web-based tool for quickly displaying a certain location of a genome at any scale. It also provides users to visualize different types of features in annotation “tracks”. Annotation “tracks” are generated either by UCSC Genome Bioinformatics Group or external collaborators. The annotations include gene and gene predictions; mapping and sequencing results; simple nucleotide polymorphisms, expression and regulatory data; repeats and pairwise and multiple-species comparative genomics data. All information relevant to a certain location is presented in one window, allowing for biological analysis and interpretation. The database tables underlying the Genome Browser tracks can be viewed, downloaded, and manipulated using another Web-based application, the UCSC Table Browser (Karolchik et al. 2009).

Ensembl (http://www.ensembl.org/) is a bioinformatics project and database that provides organized biological information and different genomic features associated with the sequences of large genomes. It is a comprehensive source of stable automatic annotation of many individual genomes. Moreover, it is a framework for the integration of any biological data that can be mapped onto features derived from the genomic sequence. Ensembl is available as an interactive Web site, a set of flat files, and as a complete, portable open source software system for handling genomes (Birney 2004). It enables users to easily access and download genomes and gene annotations for many organisms, as well as to explore certain features through its interactive Website.

I analysed data from the C57BL/6J mouse strain, which is the mouse reference genome strain, and PWK/PhJ, which is an inbred wild-strain. I have downloaded the C57BL/6J genome (GCA_000001635.2) and the PWK/PhJ genome (GCA_001624775.1) from the UCSC database alongside with their RepeatMasker annotations of repetitive elements. Gene annotations for laboratory mouse strains C57BL/6J and PWK/PhJ were downloaded from the Ensembl release 100. Summary of global assembly statistics for both mouse genome assemblies is shown in Table 1.

Table 1. Global statistics of C57BL/6J mouse genome assembly (GCA_000001635.2) and PWK/PhJ mouse genome assembly (GCA_001624775.1)

	C57BL/6J mouse genome assembly	PWK/PhJ mouse genome assembly
Total sequence length	2 730 855 475	2 559 987 392
Total ungapped length	2 652 767 259	2 323 507 398
Number of scaffolds	162	3 140
N50	54 517 951	128 387 505

3.2 Computational tools and programming softwares

3.2.1 R

R is a free software environment for statistical computing and graphics (R core team 2019). It can be run on all sorts of platforms, such as UNIX, Windows, and MacOS platforms. R provides a wide variety of statistical and graphical techniques. Packages are one of the unique features of R. Packages are free publicly available collections of functions and data sets developed by any member of the community. They increase the power of R by improving existing base R functionalities, or by adding new ones.

3.2.2 Bioconductor

One of the most comprehensive collections of R packages for analysis of genomic data is Bioconductor (Huber et al. 2015). This collection of packages provides more than 1900 wide range of powerful statistical and graphical tools for the analysis of genomic data.

Most of the analysis, including development of a method for annotating full-length MT elements in mice, I have entirely written in R. I have incorporated various methods using relevant packages to efficiently analyse vast amounts of genomic data. A list and description of used packages is present in Supplementary 1. Unless it is stated otherwise, all the analysis and methods described below were done in R.

3.2.3 D-GENIES

D-GENIES (Dot plot large Genomes in an Interactive, Efficient and Simple way) is an interactive online tool designed to compare two genome sequences. Dot plots are widely used to quickly compare sequence sets and to identify highly similar and repetitive regions. They provide a synthetic similarity overview, highlighting repetitions, breaks and inversions (Cabanettes and Klopp 2018). In dotplot each axis represents one sequence that is being compared to the other one. A short segment of one sequence, called widow size, is compared with all other segments of

the same length in the second sequence. Then for each pair of comparison (alignment), the similarity between the sequences residues is scored based on the probability with which the various pairs of aligned residues replace one another. If the overall score between any two segments exceeds a certain threshold value, a dot is plotted in the dotplot. By changing the threshold value and window site, weaker regions of similarity between sequences can be detected.

I have made a dotplot of mouse genomes C57BL/6J and PWK/PhJ using D-GENIES with default parameters. The use of dotplot has enabled me to do a quick detection of similarity (homologous regions) between these two mouse genomes.

3.2.4 BLAT (BLAST-like alignment tool)

An important goal of genomics is to search for similarities between biological sequences. Similarity search between sequences enables researchers to infer phylogenetic relationships between species, identifications of homologous genes, search for conserved proteins domains and many more. One of the tools used for this purpose is the BLAT (BLAST-like alignment tool) (Altschul et al. 1990). BLAT, like BLAST, takes a subject protein or nucleotide sequence (called a query) and compares it with a database of sequences (Figure 7). The program first builds an index of the genome by splitting the sequences into smaller non-overlapping words, kmers, of length 11. This step is different from the BLAST algorithm and it allows BLAT to more quickly find matches between large database and query sequence. The query sequence is also split into kmers that are then compared against all generated non-overlapping kmers in the database. When searching for homologous regions in nucleotide sequences, the algorithm looks for perfect hits. Perfect hits are then extended in both directions until there are no mismatches and overlapping hits are merged. Any existing gaps in the alignment of query or database sequences are filled by the algorithm. BLAT connects extended hits which follow each other in both query and database coordinates (the homologous region) into one larger alignment (Kent 2002).

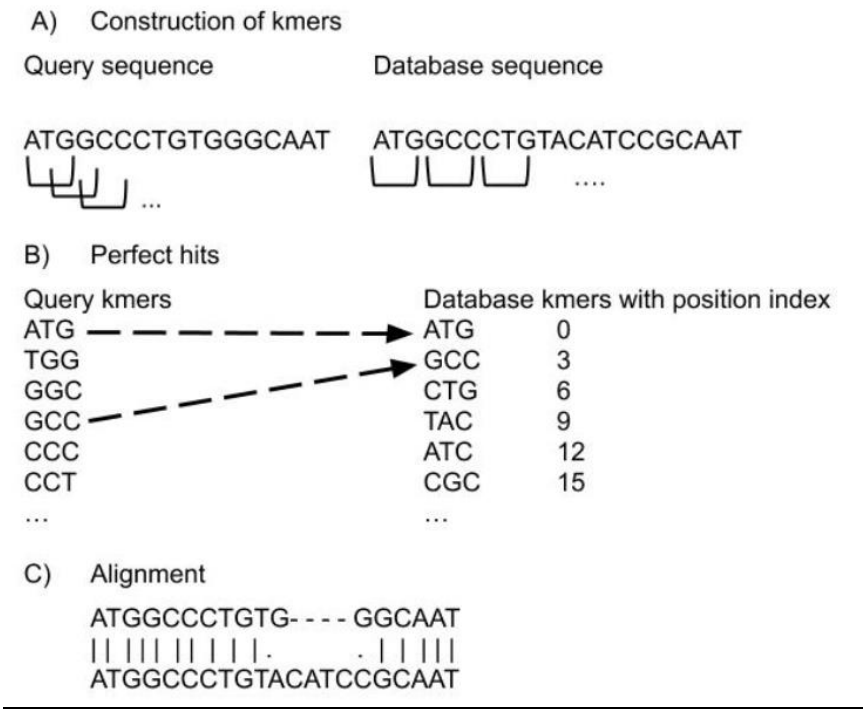

Figure 7. A) Generation of overlapping kmers from query sequence and non-overlapping kmers from database sequence of length 3. B) Finding perfect hits between query and database kmers C) BLAT alignment of query and database sequence. Vertical lines represent exact matches, dots represent mismatches and horizontal lines in query sequence indicate gaps.

I have used BLAT with default parameters to identify homologous full-length MT elements between mouse strains C57BL/6J and PWK/PhJ.

3.3 Annotation of full-length MT elements

Retrieved annotation files of repetitive elements from the UCSC, like most of the publicly available annotations of repeats, were created with the program RepeatMasker. As a part of the comparative analysis of repetitive elements in C57BL/6J and PWK/PhJ genomes, I have calculated the percentage of base pairs belonging to each repetitive class based on RepeatMasker annotation for both strains using custom R script (Supplementary 2). To test if there is a statistical significance in the contribution of certain repetitive classes between C57BL/6J and PWK/PhJ mouse genome, I have used Chi-square test.

To obtain high-quality annotation of full-length MT elements, I have developed a method that centers around their conserved LTRs. RepeatMasker can differentiate between LTRs and internal sequences. It assigns a unique number or ID to each part of the same element. For

instance, a full-length element would contain 2 LTRs and one internal sequence under the same ID whereas in solo or fragment elements some parts would not be present. I decided to disregard the unique IDs assigned by RepeatMasker so I could get a bit more independent result and better annotation of full-length MT elements. The steps for improving the annotation of full-length MT elements in both mouse strains are the following (Figure 8):

1. I have discarded all of the internal sequences and keep only LTRs of MT elements annotated by RepeatMasker.
2. To obtain full-length elements, I have filtered the LTRs by length for each MT element group and mouse strain separately. I have performed two sets of filtering on MTA and MTB elements in mouse PWK/PhJ. First filtering is less strict and allows for a wider range of LTR length (PWK/PhJ relaxed), while second filtering is more rigid in order to detect only conserved elements that are most similar to MT elements in C57BL/6J (PWK/PhJ conserved). Distributions of LTR length for each MT group and mouse strain were visualized in R and filtering values were decided based on the highest numbers of LTR lengths, referred to as peak, in each MT group. Filtering lengths of LTRs are shown in Table 2. In the steps that follow, I have analysed only those full-length MT elements that were more stringently filtered in both mouse strains.

Table 2. Filtered lengths of LTRs for each group of MT elements in mouse strains C57BL/6J and PWK/PhJ. PWK/PhJ conserved represents more stringent filtering for MTA and MTB that was also done in C57BL/6J. PWK/PhJ relaxed represents less stringent filtering that differs for MTA and MTB in C57BL/6J.

	MTA/ bp	**MTB/ bp**	**MTC/ bp**	**MTE/ bp**	**MTD/ bp**
C57BL/6J	390-400	380-400	300-410	300-410	300-410
PWK/PhJ conserved	390-400	380-400	300-410	300-410	300-410
PWK/PhJ relaxed	250-500	320-450	300-410	300-410	300-410

3. Identify the closest pairs of LTRs to each other. I have kept only those pairs whose distance from each other is smaller than 2200 bp and greater than 1800 bp. Filtering lengths were decided based on the distribution of distances between annotated pairs of MT LTRs.

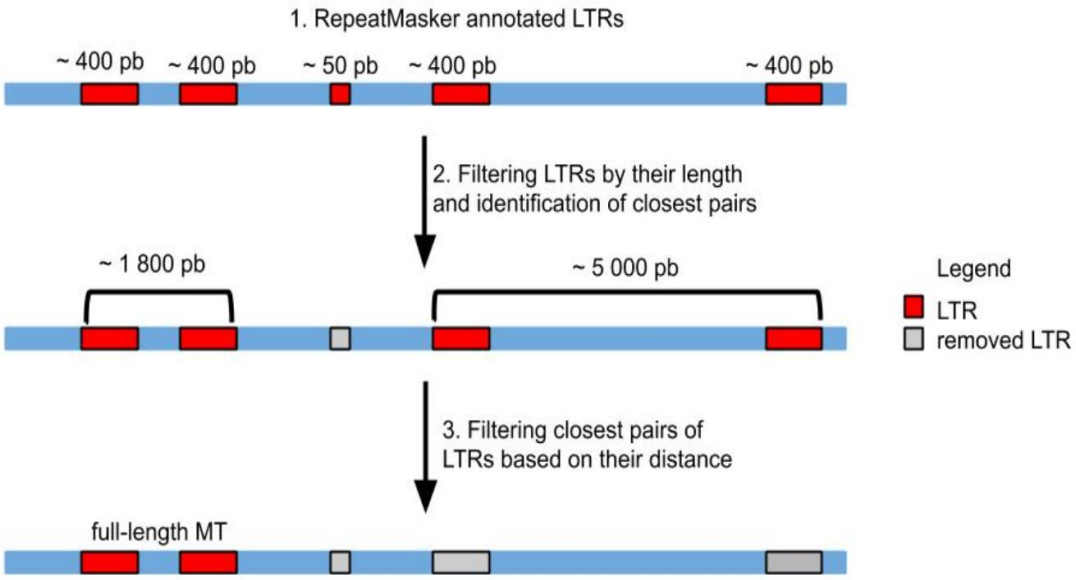

Figure 8. Schematic representation of filtering steps in annotation of full-length MT elements in the mouse genome based on the RepeatMasker annotation of LTRs.

I analyzed the distribution of annotated full-length MT elements on chromosomes of both strains. For a full-length MT element to be fully conserved, its LTRs have to have a high degree of similarity one to each other. I have done Needleman-Wunsch pairwise alignment of defined LTRs for each full-length MT element in both strains with more conserved filtering and also in PWK/PhJ with less stringent filtering. From pairwise alignment, I have extracted the percentage of identical nucleotides, also known as the percentage of identity, between those two LTRs of the same full-length element. I have visualized their percent identities in relation to their alignment length. To examine if my approach of annotating full-length MT elements by ignoring their assigned unique ID by RepeatMasker was a reasonable decision, I did the same length and distance filtering of LTRs, as described earlier, based on their unique IDs assigned by RepeatMasker. R code for the mentioned annotation process and analysis of data is in Supplementary 2.

3.4 Analysis of homologous MT elements between mouse strains C57BL/6J and PWK/PhJ

Sequences of newly annotated full-length MT elements obtained by a strict filtering approach were extracted from both mouse strain genomes. In order to find homologous full-length MT elements between strains, I have aligned the more abundant annotated full-length MT elements from C57BL/6J genome on the PWK/PhJ genome using program BLAT with default parameters (Figure 9). For each query sequence from C57BL/6J, I have selected only the first best scoring hit in PWK/PhJ (the one with highest percent of identity). I have visualized the BLAT results by plotting starting positions of full-length MT elements of mouse strain C57BL/6J versus starting position of the alignment on the strain PWK/PhJ. To validate the homologous MT elements obtained by BLAT results, I have examined the following criteria (Figure 9):

1. Annotation of the mapped MT element had to be the same as the annotation of the element onto which it is mapped. I analyzed the overlap of RepeatMasker annotation on PWK/PhJ mouse with mapped positions of MT elements from C57BL/6J onto PWK/PhJ genome.
2. Genes surrounding an MT element in PWK/PhJ genome should be the same as the genes surrounding the homologous element in C57BL/6J. I have selected genes from the gene annotation files that are homologous protein-coding genes between strains. In the gene annotation file of PWK/PhJ it is stated which genes are homologous to which genes in reference mouse genome C57BL/6J, so I did not have to perform any additional analysis to identify homologous genes. However, I have filtered those homologs that are on different chromosomes. MT elements that passed both criteria were considered to be conserved elements between strains. I have performed a Wilcoxon test to test whether there is a significant difference in length or percent of identity of mapping results between conserved MT elements and those that are not.

3. To get a more reliable identification of shared conserved homologous full-length MT elements between C57BL/6J and PWK/PhJ, I have aligned the annotated full-length MT elements from PWK/PhJ on the C57BL/6J genome using program BLAT with default parameters. Those elements that occurred as pairs in both BLAT analyses, later referred to as reciprocal BLAT approach, I have labeled them as conserved full-length MT elements.

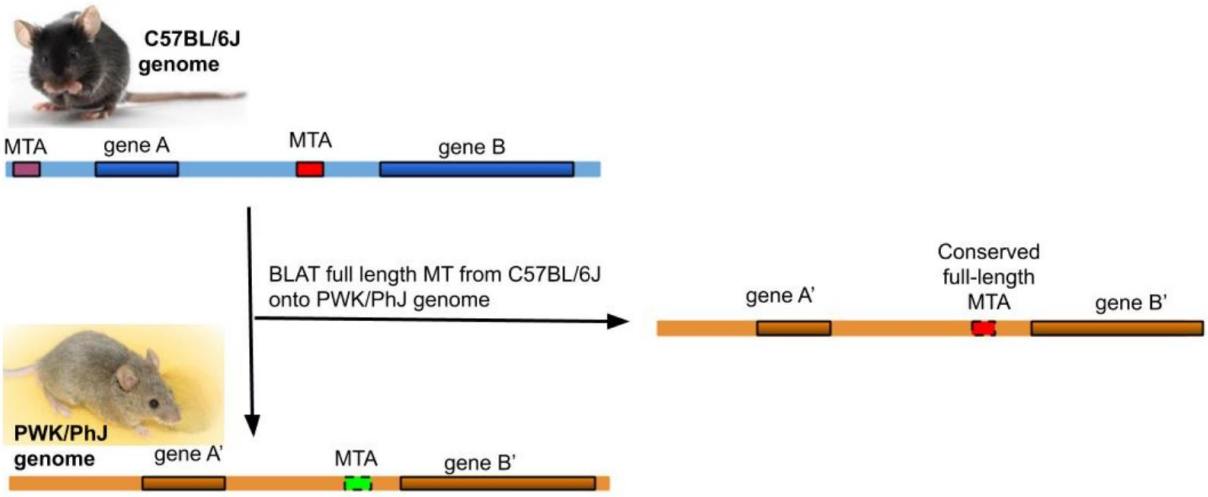

Figure 9. Schematic representation of identifying homologous MT elements between mouse strains C57BL/6J and PWK/PhJ. Two full-length MTA elements (purple and red) are mapped onto the PWK/PhJ genome using BLAT. One MTA element (red) is mapped onto the same position as the MTA elements (green) from PWK/PhJ and has the same homologous surrounding genes in both mice. Therefore, this MTA is labeled as conserved full-length MTA while the other MTA (purple) does not pass this filtering criteria hence it is C57BL/6J strain specific.

As already mentioned, only full-length MT elements that occurred as pairs in reciprocal BLAT approach are conserved full-length retrotransposons between strains. The rest of the elements were then divided into groups depending on the 1. and 2. filtering criteria. MT elements that haven't passed both of the criteria were labeled strain-specific elements while those that passed were labeled uncertain. I have manually examined some MT elements belonging to these assigned categories in the UCSC genome browser.

3.4.1 Integration of shared conserved and strain-specific full-length MT elements in genomic regions

For the analysis of MT elements integration in genes of both strains, I have overlapped exons, introns, and intergenic regions with conserved and strain-specific full-length MT elements. I took only exons from gene annotation of both strains, from which I made a list of non-overlapping exons. I have defined introns as the regions between exons and intergenic regions as regions between genes for both mouse strains. After defining the regions, I have made an overlap with conserved and strain-specific full-length MT elements separately for each strain. To determine if there is a significant difference in orientation biases of conserved and new MT elements integrated into genes I have used the Chi-square test. The code for the described method is shown in Supplementary 2. Furthermore, to analyse into which types of genes are MT elements integrated in both strains, I have divided the gene types into these categories based on the Ensembl

annotation of genes: protein-coding genes, non-coding RNA genes (miRNA (micro RNA) and different types of lncRNA) and pseudogenes (processed/unprocessed and /or transcribed)

As an additional validation of my results, I have manually examined some cases of conserved and strain-specific MT elements integrated into protein-coding genes in the UCSC Genome Browser.

3.5 Statistical methods

Chi-square test is used to test the relationship between two categorical variables. The null hypothesis of the test is that no relationship exists on the categorical variables meaning they are independent while the alternative hypothesis is that they are related. Chi-square test uses a contingency table in which frequency distribution of the variables are stored. The Chi-square statistic is calculated by formula: $\chi = \sum \frac{(observed - expected)^2}{expected}$. Observed values are present in the table and expected value for a given cell is calculated as $e = \frac{row.sum * col.sum}{total.sum}$. If the result of the Chi-square test is a p-value below a certain threshold, which is usually 0.05, the null hypothesis is rejected.

Wilcoxon rank sum test is a non-parametric test used to compare two independent samples when the data is not normally distributed. It is an alternative test to unpaired two-samples t-test. Wilcoxon rank sum test assumes that observations within each sample and the two samples must be independent of one another. The test assigns ranks to each observation in data and then calculates the means of the ranks. Under the null hypothesis, random distributions of two samples are equal, it compares the different rank means. Like Chi-square test if the calculated p-value is below 0.05 the null hypothesis is rejected.

4 Results

4.1 Whole-Genome Alignment

Whole-genome alignment of C57BL/6J and PWK/PhJ mouse genomes is shown in Figure 10. The main diagonal represents the regions that have the highest similarity between genomic sequences. Both mouse strains contain all the autosomal chromosomes, but PWK/PhJ assembly does not contain chromosome Y. Alignment of chromosome X from these strains contains the highest number of gaps and overall lower identity.

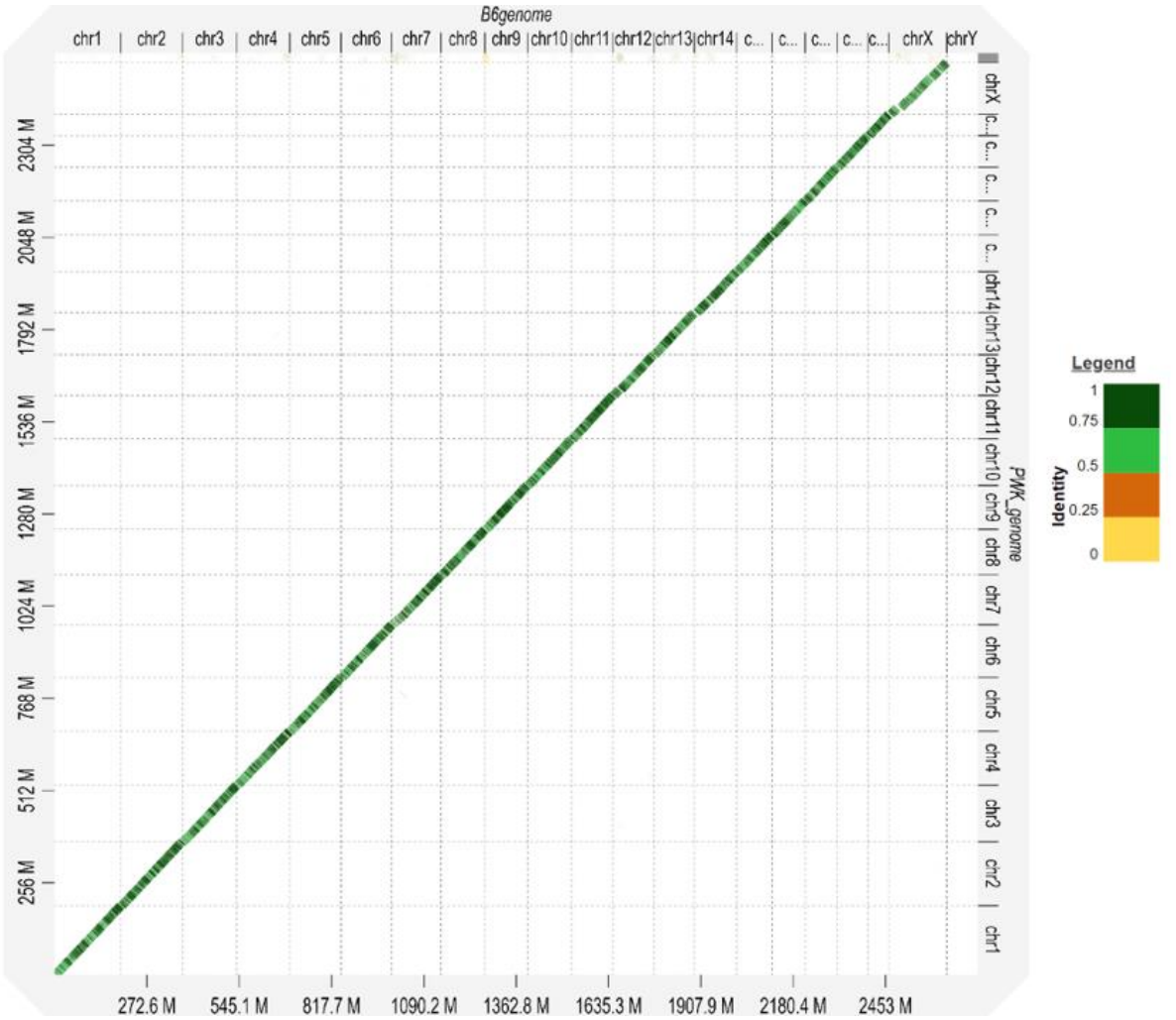

Figure 10. A dotplot of genomes of the C57BL/6J (GCA_000001635.2) and PWK/PhJ (GCA_001624775.1) mouse genome assembly generated by D-GENIES with default parameters and visualization option “strong precision”. The C57BL/6J genome is assigned to the horizontal axis and the PWK/PhJ genome is assigned to the vertical axis.

4.2 Repeat content comparison between C57BL/6J and PWK/PhJ mouse genomes

To compare the repeat content between C57BL/6J and PWK/PhJ mouse genomes, I calculated the percentage of base pairs belonging to a certain repetitive class annotated by program RepeatMasker. In both mouse genomes over 85% of repetitive elements belong to the retrotransposon class; LINE, LTR and SINE as shown in Figure 11A. SINE elements and simple repeats contribute more to total genomic repeat content in PWK/PhJ genome than in C57BL/6J whereas LINE elements contribute more in the C57BL/6J genome. LTR retrotransposons account for around 26% of total genomic repeat content in both strains. Classes that mostly contribute to LTR retrotransposons composition are ERVL-MaLR (containing MT elements) and ERVK, followed by ERVL and ERV1 in both strains (Figure 11B). ERVL-MaLR has a slightly higher contribution to overall content of LTR retrotransposons in PWK/PhJ genome than in C57BL/6J. Contributions of these repetitive elements between C57BL/6J and PWK/PhJ mouse genomes are not statistically significant (Chi-square, p>0.05).

Figure 11. A) Percentage of base pairs belonging to different classes of repetitive elements in the mouse genomes. B) Percentage of base pairs belonging to differentLTR retrotransposon classes. The percentages were calculated byRepeatMasker annotation of repetitive elements in C57BL/6J and PWK/PhJ mouse genomes.

4.2.1 MT element comparison C57BL/6J and PWK/PhJ mouse genomes

The MT elements that mostly contribute to class ERVL-MaLR (the most abundant class of LTR retrotransposon) are MTE and MTD, followed by MTC (Figure 12). The lowest percentage of base pairs belonging to ERVL-MaLR in both strains, besides the category “other”, are MTA and MTB elements. MTE and MTB elements contribute more to repeat content of ERVL-MaLR in PWK/PhJ genome than in C57BL/6J genome. While MTA elements contribute more in the C57BL/6 genome. Contributions of these repetitive elements between C57BL/6J and PWK/PhJ mouse genomes are not statistically significant (Chi-square, p>0.05).

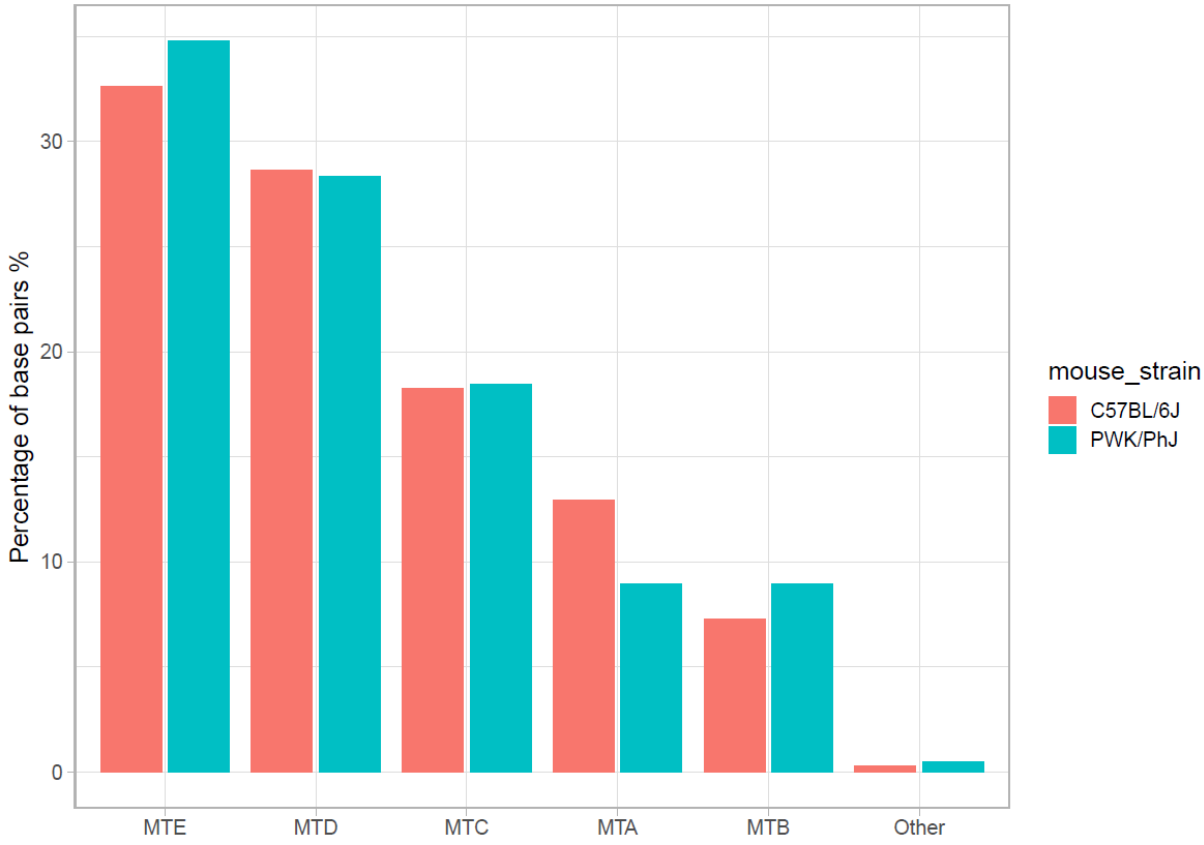

Figure 12. Percentage of base pairs belonging to each of the repetitive classes of ERVL-MaLR in the C57BL/6J and PWK/PhJ mouse genomes. The percentages were calculated from the RepeatMasker annotation of repetitive elements in C57BL/6j and PWK/PhJ mouse genome.

4.3 Identification and annotation of full-length MT elements

I have developed the method for identification and detection of full-length MT elements based on RepeatMasker annotation of long terminal repeat, LTRs. Distribution of LTR length of each type of MT element for C57BL/6J and PWK/PhJ genomes is shown in Figure 13. All LTRs of MT elements in both strains have the greatest number of LTRs around length from 380 to 400

bp. MTA elements from C57BL/6J have the highest number of LTRs, close to 10 000 LTRs length around 400 bp, unlike the LTRs of MTA elements in PWK/PhJ where there are less than 1 250 of the same length. The rest of the MT elements in both strains have similar numbers of LTRs in their peaks.

After pairing the closest LTRs from the same class, I have visualized their distances from each other from 1500 to 3200 bp (Figure 14). The highest number of MT elements is for the MTA elements in the C57BL/6J genome around length of 1800 bp. The rest of the MT elements do not have a clear shape of distribution or even a peak in their distribution besides MTA and MTB in the C57BL/6J genome.

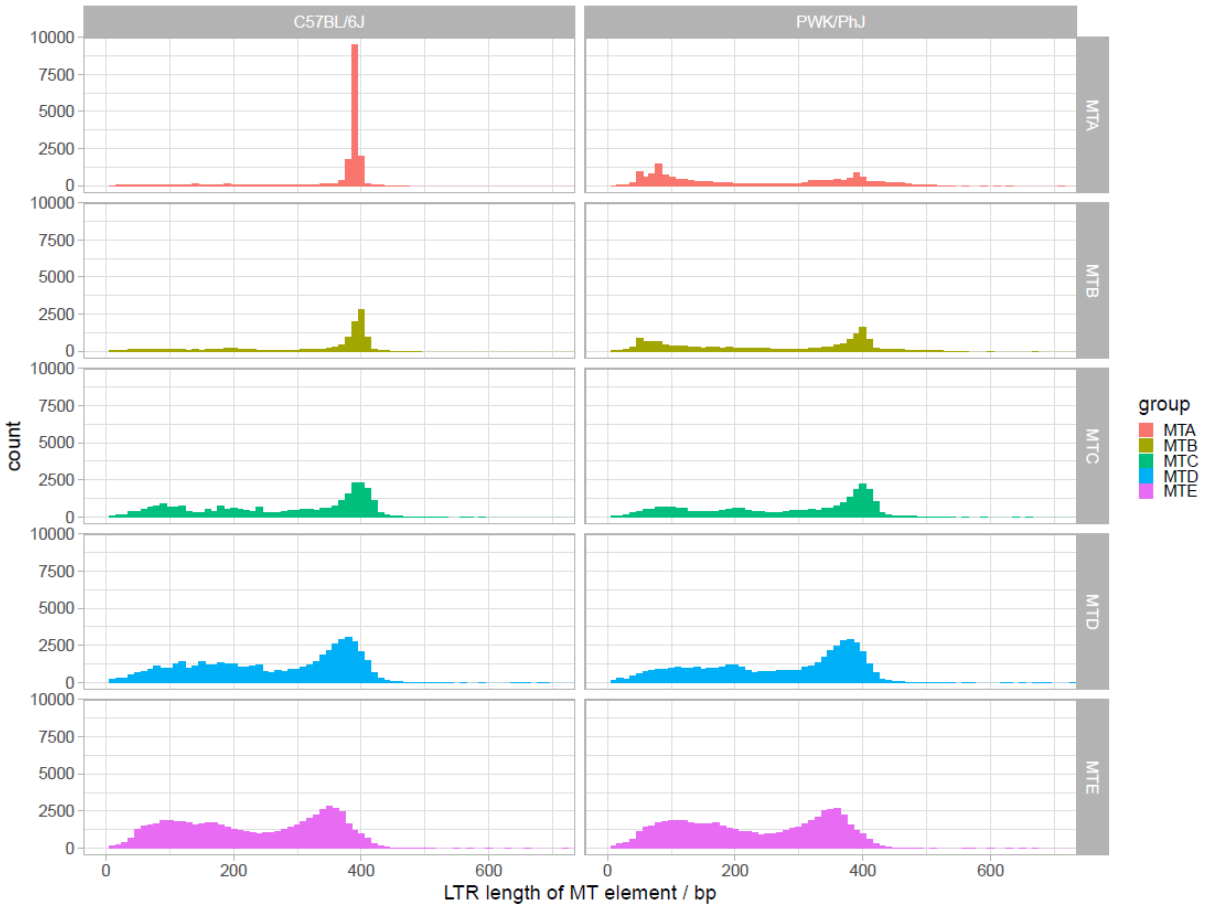

Figure 13. Distribution of LTR lengths of MT elements from RepeatMasker annotation of the C57BL/6J mouse genome (left) and PWK/PhJ mouse genome (right).

Figure 14. Distribution of distance between annotated pairs of full-length MT LTRs from 1 500 bp to 3 500 bp on the C57BL/6J (left) and PWK/PhJ mouse genome (right).

4.3.1 Comparison of annotated full-length MT elements in C57BL/6J and PWK/PhJ mouse genomes

MT elements that have passed the filtering criteria for their LTRs by more strict approach, I have annotated as full-length elements and their numbers for the C57BL/6J and PWK/PhJ are presented in Table 3. MTA elements in the C57BL/6J genome are the most abundant group of annotated full-length MT elements. While the MTA elements in PWK/PhJ are the sparsest ones alongside MTB elements. Numbers of annotated full-length MTC, MTD and MTE elements are very similar in both mouse strains. Annotated full-length MT elements are not uniformly distributed along chromosomes in both mouse genomes (Figure 15). In certain areas on chromosomes in the C57BL/6J genome there is a high concentration of MTA elements like on chromosome 1 around 35 Mbp.

Table 3. Number of annotated full-length MT elements using my approach in C57BL/6J and PWK/PhJ mouse genome

Mouse genome	MTA	MTB	MTC	MTD	MTE
C57BL/6j	1217	94	95	130	114
PWK/PhJ	13	29	97	113	99

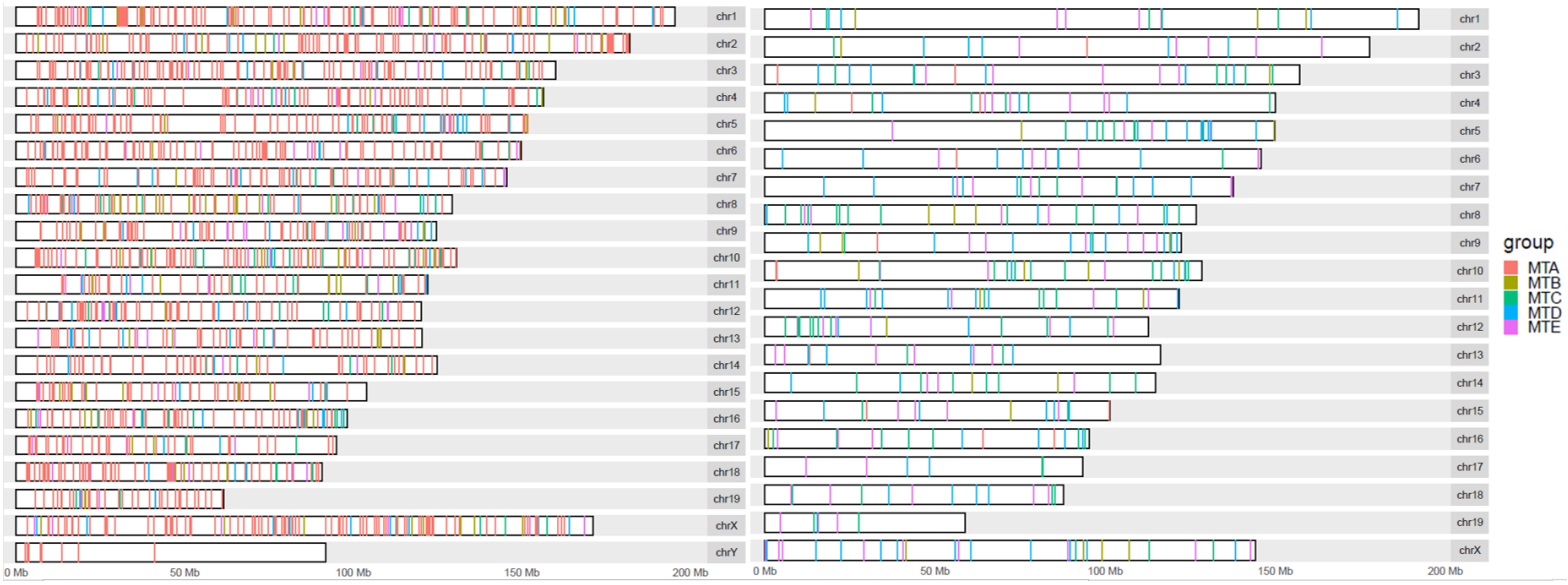

Figure 15. Distribution of annotated full-length MT LTRs on the C57BL/6J mouse genome (left) and PWK/PhJ mouse genome (right).

The numbers of MTC, MTD and MTE elements in PWK/PhJ are the same for strict and less strict filtering, numbers shown in Table 3. However, the number of MTA elements obtained with less rigid filtering is 120 and for MTB elements is 103. Figure 16 represents the results of pairwise alignment of LTRs from annotated full-length MT elements in C57BL/6J and PWK/PhJ mouse genome. Gray dots represent full-length elements that were obtained with less rigid filtering of LTR lengths in the PWK/PhJ genome. The number of MTC, MTD and MTE elements is the same for both filtering, numbers shown in Table 3.

Relations of percent identity and alignment length are more similar for MT elements obtained with strict filtering of LTR lengths between C57BL/6J and PWK/PhJ than those obtained with less strict approach.

Figure 16. Percent identities in relation to their alignment length percent from pairwise alignment of annotated full-length MT LTRs from the C57BL/6J mouse genome (left) and PWK/PhJ mouse genome (right). Grey dots represent full-length MT elements obtained by less strict filtering of LTR length and colored dots by MT group are obtained by more rigid filtering of LTR length.

4.3.2 Comparison of my annotated full-length MT element and RepeatMasker annotation of elements

One way of assessing if my approach of identification and annotation is better than the original RepeatMasker annotation, was to perform the same filtering steps of LTRs on the unique IDs of full-length MT elements assigned by RepeatMasker. I have identified more full-length elements in all groups of MT elements in C57BL/6J and PWK/PhJ genomes using my approach than completely relying on the RepeatMasker annotation (Figure 17). The biggest increase in the number of annotated full-length elements is for MTD and MTE elements.

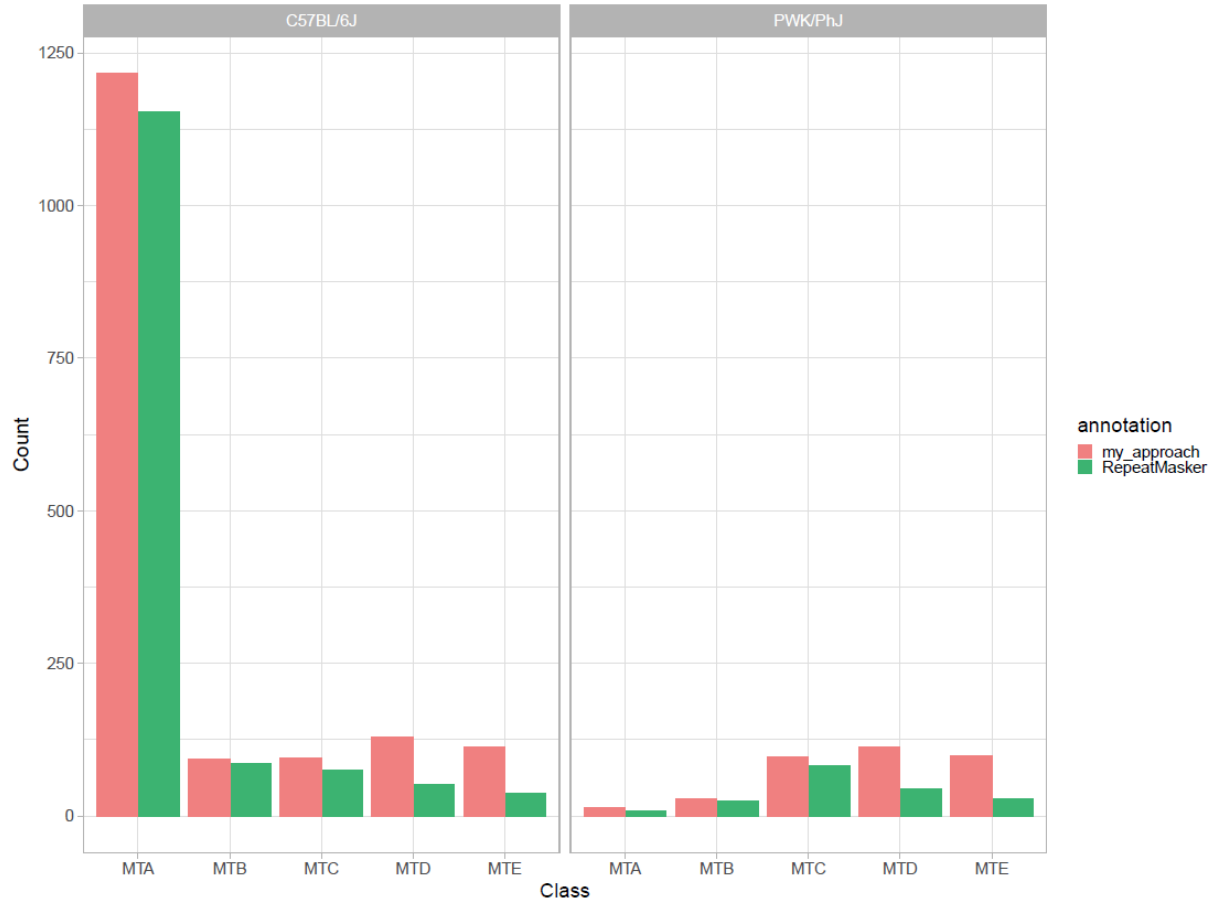

Figure 17. Distribution of numbers of annotated full-length MT elements using my approach and of full-length MT elements from RepeatMasker data with the same filtering criteria on the C57BL/6J mouse genome (left) and PWK/PhJ mouse genome (right).

4.4 Analysis and validation of BLAT results for mapped full-length MT elements from C57BL/6J mouse on to PWK/PhJ mouse

To identify homologous full-length MT elements between mouse strains C57BL/6J and PWK/PhJ, I had performed the search of C57BL/6J annotated full-length MT elements with my approach in PWK/PhJ genome. The result of that mapping is shown in Figure 18. MT elements that are on the diagonal have a higher percent identity of mapping than majority of elements outside the diagonal. Moreover, most of the elements that are outside the diagonal had originated from a different chromosome in the C57BL/6J genome. MTA elements have the most elements that originate from different chromosomes and have lower percent identity unlike MTC, MTD and MTE elements that have none.

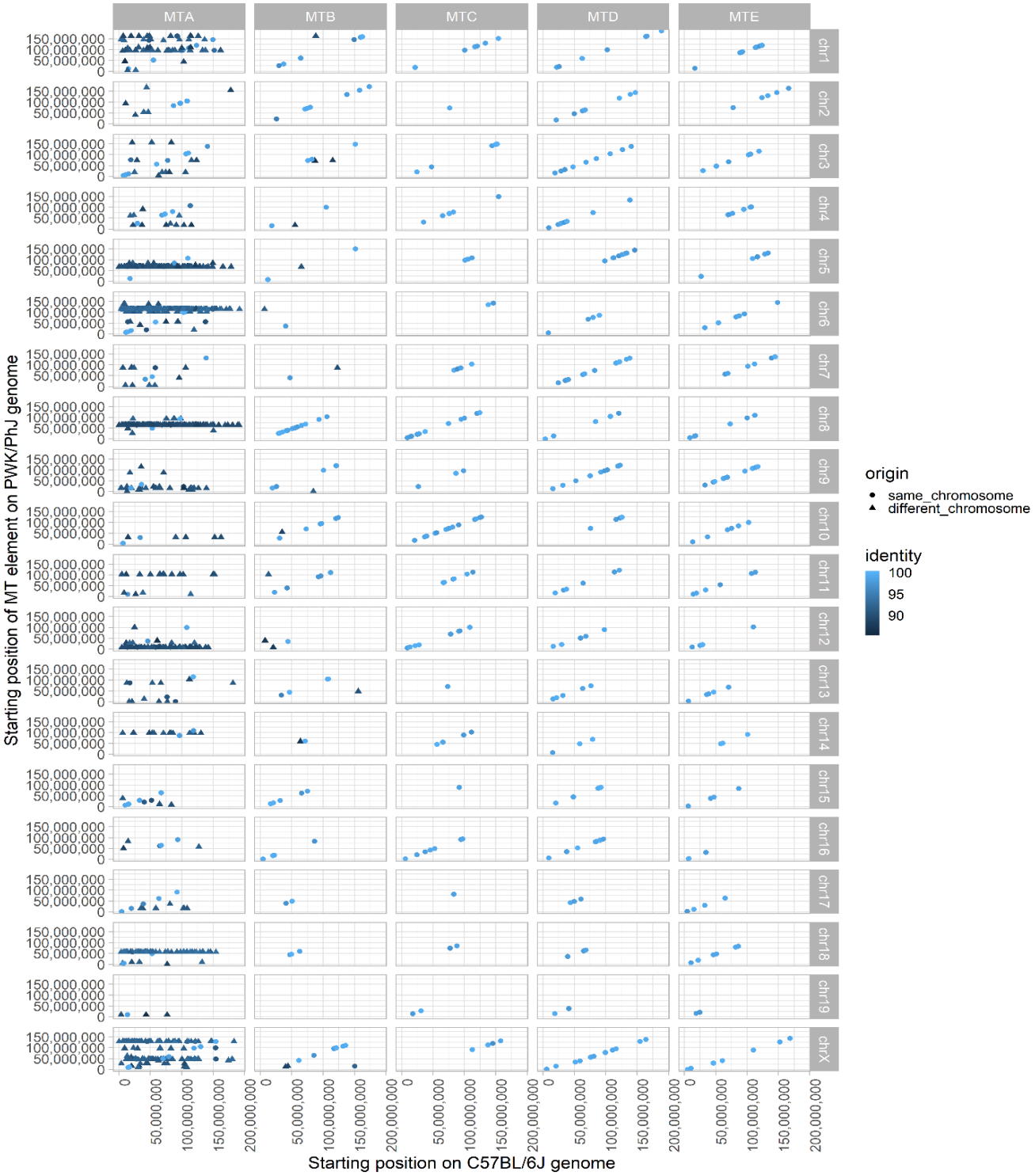

Figure 18. BLAT result of mapping annotated full-length MT elements from C57BL/6J onto PWK/PhJ genome. Starting position of the annotated full-length MT element that originated from the C57BL/6J genome is shown on the x-axis. While on the y-axis, the mapped starting position of the annotated full-length MT element from the C57BL/6J genome PWK/PhJ genome is shown. Different shapes of points represent whether the MT element is mapped onto the same chromosome on PWK/PhJ as the chromosome which it originates from in C57BL/6J.

4.4.1 Annotation of mapped full-length MT elements from C57BL/6J on to PWK/PhJ genome

To validate the BLAT results of mapping annotated full-length MT elements from C57BL/6J onto PWK/PhJ genome, I had examined if those mapped MT element map to the same group of MT element on PWK/PhJ. Over 90% of MT elements were correctly annotated in the PWK/PhJ genome. Most of the wrongly annotated full-length MTA elements from C57BL/6J are annotated as MTB elements in PWK/PhJ (Figure 19). Also, the number of wrongly annotated MTB elements from C57BL/6J as MTA elements in PWK/PhJ is quite high when compared to others. MTC, MTD and MTE elements have over 95% correctly annotated elements.

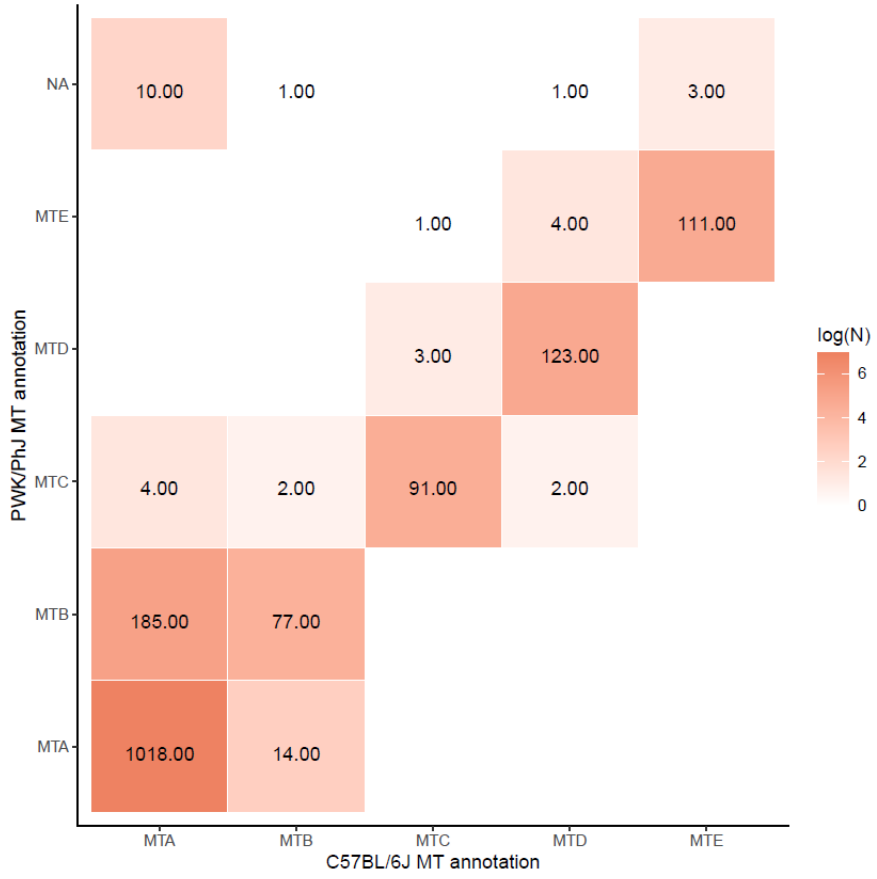

Figure 19. Display of correctly and wrongly annotated full-length MT elements from C57BL/6J in PWK/PhJ. On the x-axis is the annotation of the MT element in C57BL/6J and on the y-axis is the annotation of that element mapped in the PWK/PhJ genome. The color range is shown by a logarithmic scale.

4.4.2 Identification of conserved full-length MT elements

To identify conserved full-length MT elements between C57BL/6J and PWK/PhJ and also to assess the quality of BLAT results, I had examined if the homologous protein-coding genes surrounding the original and mapped MT element are the same. A total of 55 homologous protein-encoding genes on different chromosomes were removed, leaving 19 318 genes for further analysis. MTC, MTD and MTE have for all the elements the same surrounding genes (Figure 20). Majority of MTB elements have passed these criteria and only a small percentage of 6.7% MTA elements have also passed. There is a significant difference, present in MTA and MTB group, in the percent of identity of mapping between conserved MT elements and not conserved (Figure 21). Conserved means those MT elements that have the same surrounding genes and same annotation in both mouse strains. Only for the group of MTB elements, there is a statistical significance in the alignment length of conserved and not conserved MT elements.

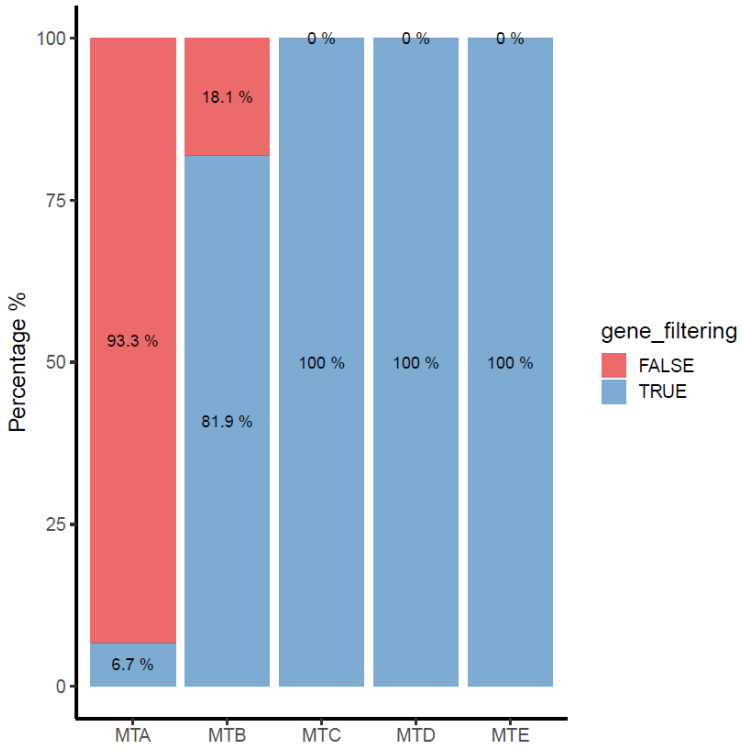

Figure 20. Percentage of MT elements from C57BL/6J which have passed (TRUE) and failed (FALSE) gene filtering step. Genes surrounding a certain mapped MT element on the PWK/PhJ genome had to be the same as the genes surrounding that same element in C57BL/6J in order for an MT element to pass this filtering step.

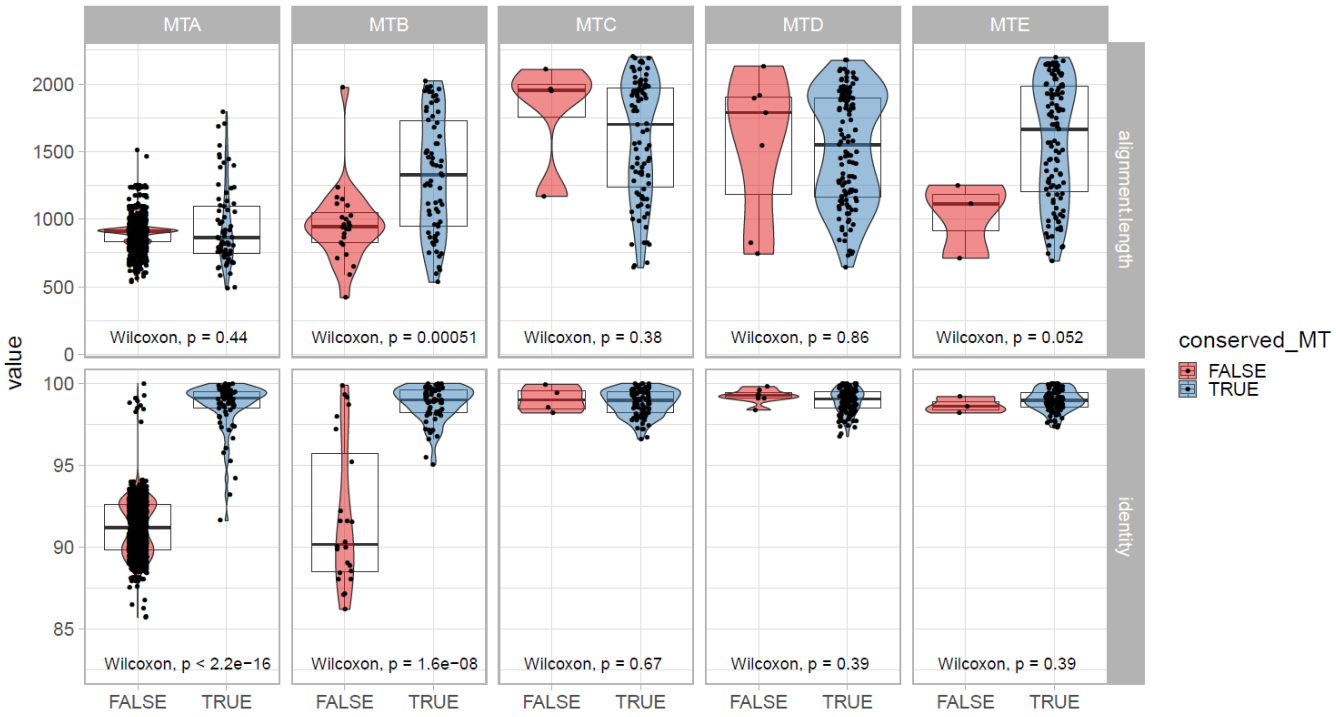

Figure 21. Distribution of alignment lengths (top row) and identity (bottom row) of mapped conserved (TRUE) and non-conserved MT (FALSE) elements from C57BL/6J onto PWK/PhJ genome. There is a significant difference in alignment length between conserved and non-conserved MTB elements (Wilcoxon test, p=0.00051). Also, there is a significant difference in identity between conserved and non-conserved MTA and MTB elements (Wilcoxon test, p<0.05).

I had done an additional mapping of annotated full-length MT elements from PWK/PhJ onto C57BL/6J to further validate the identification conserved full-length MT elements, that are annotated with my approach, between strains. The number of shared conserved full-length MT elements that appear as pairs in both of the mapping and those MT elements that only pass or not the above-mentioned filtering criteria are in Table 4. Around 82% of full-length MT elements from PWK/PhJ have been paired with full-length MT elements from C57BL/6J. The highest number of strain-specific full-length MT elements is in the group of MTA elements from C57BL/6J, followed by MTB elements from the same strain. By manually examining some elements in UCSC browser I have found that strain specific-elements do not have appropriate homologous elements on the same location in the other mouse strain genome. However, some uncertain elements do have an appropriate MT element in homologous regions. For example, in Figure 22A the identified homologous MTA element in PWK/PhJ strain has a length of LTRs around 350 which is below the filtering value in my method. Furthermore, in Figure 22B an uncertain MTA element in C57BL/6J has a homologous MTA that is lacking one LTR.

Table 4. Shared conserved, strain-specific and uncertain full-length MT elements, that are based on my annotations approach, between mouse strains C57BL/6J and PWK/PhJ. Conserved elements are determined by the analysis of reciprocal mapping of annotated MT elements from one strain onto the other, and vice versa, with BLAT. Strain-specific MT elements do not have the same surrounding protein-coding genes and annotation in both strains whereas uncertain MT elements have.

Category of MT element	MTA	MTB	MTC	MTD	MTE
Conserved	9	24	75	98	82
C57BL/6J specific	1 146	47	0	4	1
PWK/PhJ specific	4	6	21	16	17
Uncertain	62	23	20	28	31

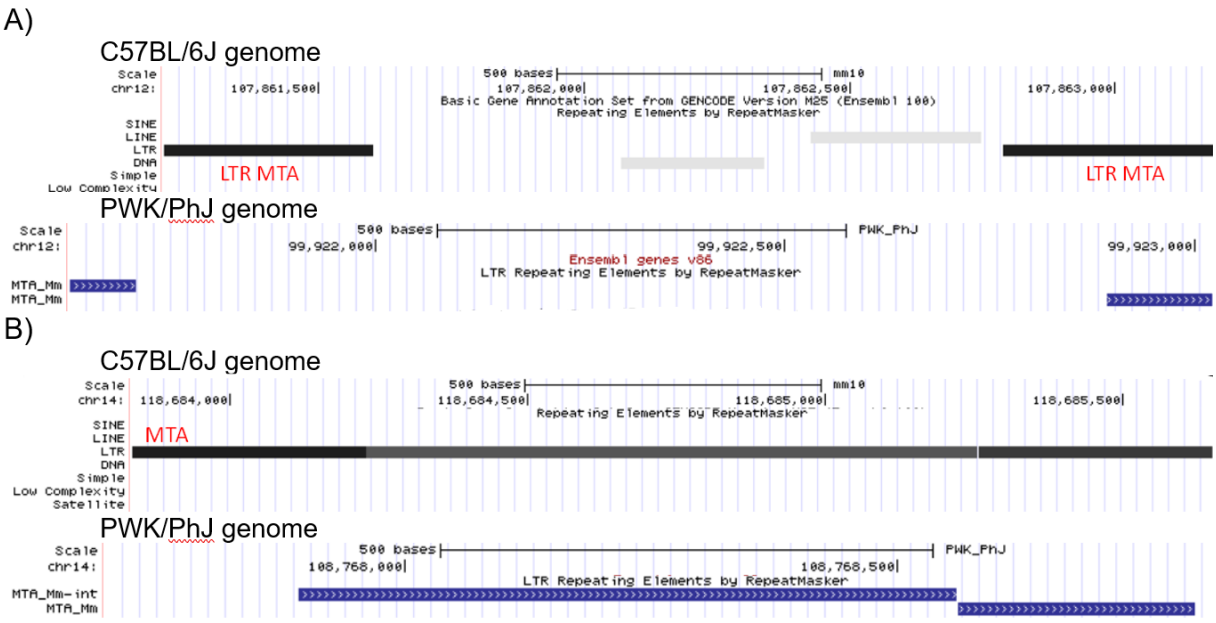

Figure 22. A) Uncertain MTA elements on chromosome 12 in C57BL/6J genome (upper) and PWK/PhJ genome (bottom). MTA element in C57BL/6J was annotated as full-length element and its found homologous MTA element was not. B) Uncertain MTA elements on chromosome 14 in C57BL/6J genome (upper) and PWK/PhJ genome (bottom). The MTA element in PWK/PhJ is fragmented. Screenshots were taken from UCSC Genome Browser for C57BL/6J genome (assembly mm10) and PWK/PhJ genome (PWK_PhJ_v1 Assembly) on 11.06.2020. Track represents the RepeatMasker annotation of repetitive elements.

4.5 Integration of shared conserved and strain-specific full-length MT elements into genomic regions

Analysis of the integration of defined conserved and strain-specific MT into C57BL/6J and PWK/PhJ genomes in antisense and sense direction in genes is shown in Figure 23A. Orientation bias between conserved and strain-specific MT elements is not significantly different between conserved and strain-specific MT elements in both strains as it was shown with the Chi-square test ($p>0.05$). In both mouse strains, strain-specific full-length MT elements are integrated mostly in intergenic, and then in intragenic regions (intron) (Figure 23B). Also, conserved full-length MT elements have the same observed trend of integration as strain-specific (image not shown). The majority of conserved and strain-specific MT integrations are into intronic parts of protein-coding genes (Figure 24). The highest number of strain-specific MT integrations into exons is for C57BL/6J specific full-length MT elements, around 34, integrated into non-coding genes (ncRNA).

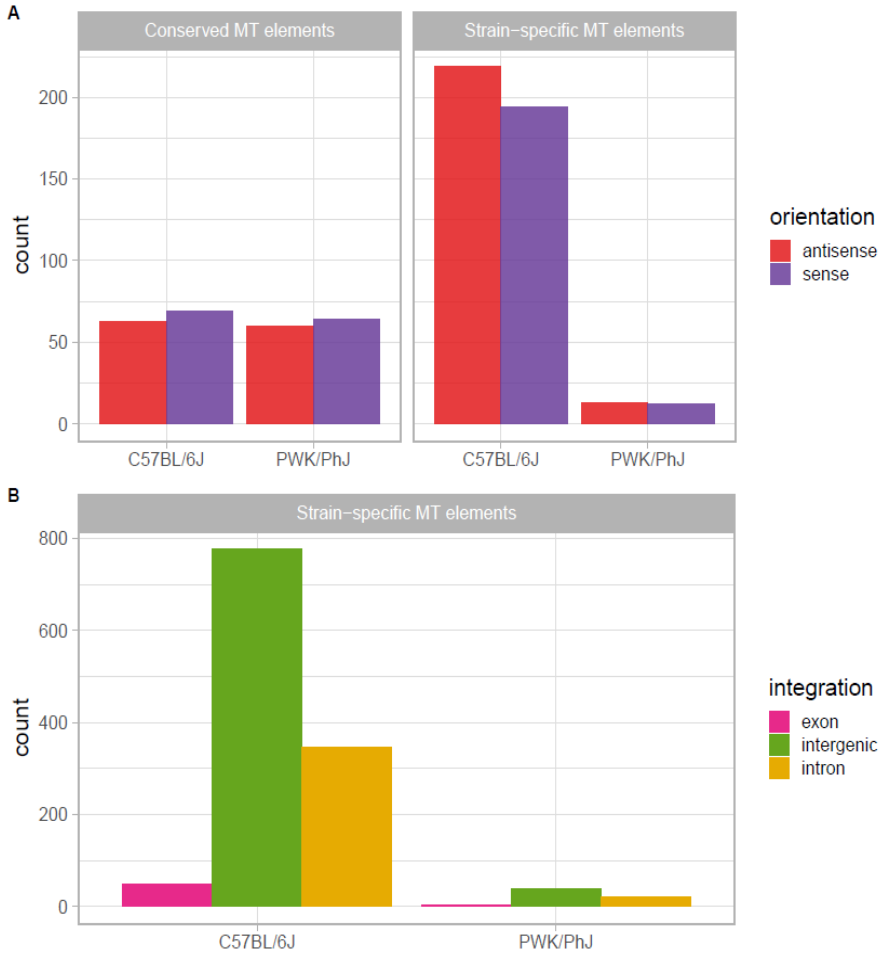

Figure 23. A) Integration of shared conserved and strain-specific full-length MT elements into genes in sense and antisense direction. Orientation biases are not significantly different between conserved and strain-specific MT elements (Chi-square, $p>0.05$) B) Integration of conserved and strain-specific full-length MT elements into exons, introns and intergenic regions.

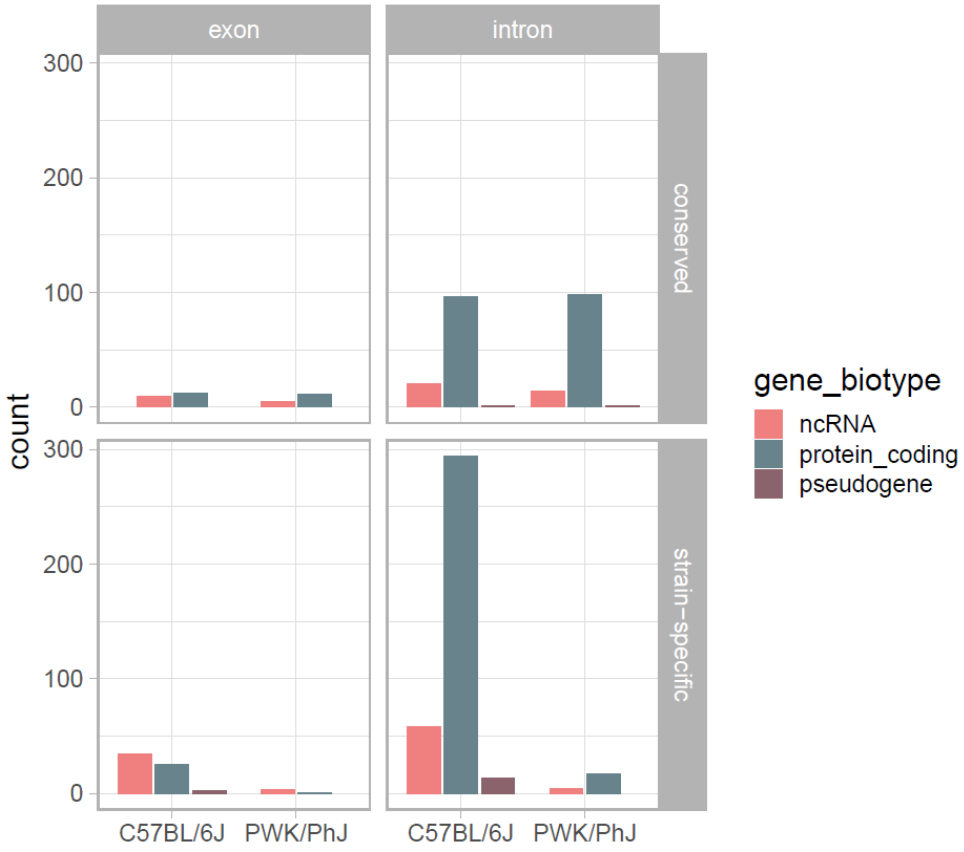

Figure 24. Integration of conserved and strain-specific full-length MT elements into exons and introns of protein-coding genes, pseudogenes and non-coding RNA genes (ncRNA) in C57BL/6J and PWK/PhJ genomes.

4.5.1 Examples of full-length MT element integrations into genomic regions

Olfactory receptor gene Olfr374 in C57BL/6J has an integrated MTB element as a part of the last exon whereas the homologous gene in PWK/PhJ does not have an integrated MT element (Figure 25A). Another protein-coding gene, Vmn2r79, has a full-length MTA element integrated into its last exon in C57BL/6J and not in PWK/PhJ (Figure 25B).

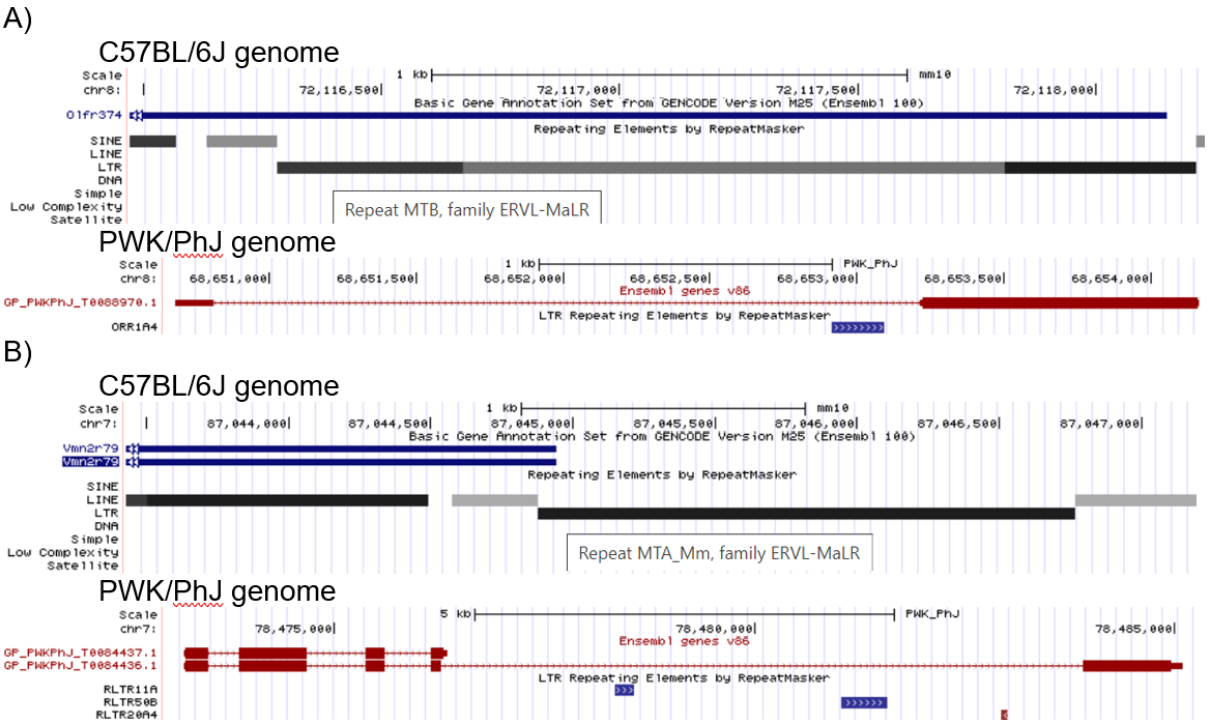

Figure 25. A) Strain-specific full-length MTB element integrated in the last exon of olfactory receptor gene Olfr374 (ENSMUSG00000046881) on chromosome 8 in the C57BL/6J genome. Homologous olfactory gene Olfr374 (MGP_PWKPhJ_G0032492) on chromosome 8 in the PWK/PhJ genome without an integration of any MT element. B) Strain-specific full-length MTA element integrated in the last exon of gene Vmn2r79 (ENSMUSG00000090362) on chromosome 7 in the C57BL/6J genome containing an integrated MTA element. Homologous gene Vmn2r79 (MGP_PWKPhJ_G0031233) on chromosome 7 in the PWK/PhJ genome without an integration of any MT element. Screenshots were taken from UCSC Genome Browser for C57BL/6J genome (assembly mm10) and PWK/PhJ genome (PWK_PhJ_v1 Assembly) on 11.06.2020. Upper track represents gene annotation for the mouse genome and the bottom track represents the RepeatMasker annotation of repetitive elements.

5 Discussion

One of the main obstacles in comparative genomic research is the quality of the assembled genomes. When comparing poorly assembled genomes of species, many important biological features that are shared or related among species may be overlooked, such as genetic variation, repetitive elements or differential gene expression. Therefore, a lot of effort has been made for the assembled mice genomes to be of the highest quality. The first assembled reference mouse genome, belonging to the classical mouse strain C57BL/6J, has been improved through the years, and therefore it has a more completed genome with fewer gaps than the more recently sequenced genome of wild-derived strain PWK/PhJ. Another important information when performing any comparative analysis of species, or in this case mouse strains C57BL/6J and PWK/PhJ, is how similar their genomes are. The fastest and the easiest way to assess their similarity is to make a dotplot of their genomic sequences. Mouse strains C57BL/6J and PWK/PhJ have diverged approximately 0.5 million years ago (Mya) (Morgan and Welsh 2015) and have homologous regions with tremendously high similarity, as it is shown by dotplot graph. Mouse strain PWK/PhJ genome is lacking assembled chromosome Y. The reason for this may be because chromosome Y is highly repetitive thus it involves different approaches and it is more difficult to assemble. Even in the first sequencing of the mouse genome, the chromosome Y was omitted from the main approach (whole-genome sequencing) and an older approach with hierarchical shotgun was used (Waterston et al. 2002). Moreover, the PWK/PhJ genome has more poorly assembled chromosome X with more gaps hence the homologous regions on chromosome X between C57BL/6J and PWK/PhJ are less similar.

The analysis of repeat content between C57BL/6J and PWK/PhJ annotated repeats by program RepeatMasker showed that both mouse strain genomes have similar distribution of repeats. Consistent with previous findings on the older version of C57BL/6J genome assembly (Franke 2016), over 85% of repeats in the C57BL/6J genome belong to retrotransposons class. I have found that this is also the case for the repetitive content in the PWK/PhJ genome. The most abundant class of retrotransposons in both strains are LINE elements followed by LTR elements. Both strains have the highest LTR retrotransposons composition of ERVL-MALR elements. Although there are slight differences in the abundance of certain classes between strains, these differences are not statistically significant. This means that there was not enough time and selective pressure for these closely related mouse strains to significantly diverge in repeat content. Similar logic is also applied to the analysis of rodent-specific MT elements between C57BL/6J and PWK/PhJ. It is important to emphasize that the older the MT element is, the more does it contribute to the retrotransposons content of the genome in both strains. However, this does not apply to the two phylogenetically youngest groups of MT elements; MTA and MTB. MTA elements are more

abundant than MTB in the C57BL/6J genome, whereas in PWK/PhJ these elements have approximately the same abundance. MTA elements are phylogenetically the youngest and are more abundant than the older MTB elements in the C57BL/6J genome. Their increased abundance may be connected to the fact that MTA elements are still retrotranspositionally active and thus are accumulating in the C57BL/6J mice as was experimentally shown with cDNA sequencing (Peaston et al. 2007). Lower abundance of MTA in PWK/PhJ genome, when compared to MTA in C57BL/6J, may indicate that MTA elements in PWK/PhJ are not retrotranspositionally active. However, there is still no evidence for the activity of rodent-specific MT elements in PWK/PhJ.

To identify full-length MT elements in C57BL/6J and PWK/PhJ genomes, I have developed a method that revolves around RepeatMasker annotation of long-terminal repeats (LTRs). LTRs of full-length elements exhibit relatively conserved length, so potential full-length elements can be identified by filtering LTRs according to their length. In both mouse strains, all MT elements have the highest peak, meaning the highest number of LTRs length, around 380 to 400 bp. The largest peak should correspond to the original LTR length upon insertion whereas the shorter length of LTR corresponds to fragmented LTR due to incomplete integration or/and mutations. Also, the LTRs of younger MT elements that are still active in the mouse genome, should contribute to the larger peak. This is in correspondence with one larger peak around 400 bp in active younger MTA of C57BL/6J mouse strain while the MTA in PWK/PhJ genome do not contain that peak. Instead the MTA elements in PWK/PhJ have more fragmented and shorter LTRs which may indicate their inactivity in the genome. Although not as strong as MTA elements, a similar trend is observed in MTB elements indicating their increased activity in the C57BL/6J genome and lower activity in PWK/PhJ genome. Not only are the length of LTRs conserved and contribute to the identification of full-length elements but the distance between two closest LTRs also serves as characteristics for annotation of full-length elements. Because full-length MT elements contain an internal sequence with regulatory signals (Smit 1996) of a certain length, it is reasonable to infer that recently integrated full-length MT elements would still have their length around 1 800 bp as I have shown. Active MTA elements in C57BL/6J have the highest peak of distances between LTRs, meaning that they are recently integrated into the C57BL/6J genome and that there wasn't enough time for ectopic recombination between the 5' and 3' LTRs. The opposite can be said for the MTA elements in PWK/PhJ genome which can also serve as additional evidence of their inactivity. Again, MTB elements have similar evidence as MTA for their increased activity in the C57BL/6J genome and lower activity in PWK/PhJ genome. However, these results and interpretations heavily depend on the quality of the assembly. As I have already mentioned, the PWK/PhJ genome assembly is of lower quality and contains more gaps. Those gaps can be present in the LTRs or internal sequences thus preventing the RepatMasker program to correctly annotate LTRs

and can later on negatively affect my method of annotation of full-length elements. More detailed analysis on how gaps influence the identification of full-length MT elements is required to get more reliable annotation, especially in the PWK/PhJ genome. This may have slightly contributed to higher numbers of annotated full-length MT element groups in the C57BL/6J genome than in the PWK/PhJ genome. Although the higher number of annotated full-length MT elements, especially MTA and MTB, is in accordance with their activity in the genome based on these analyses and therefore gaps should not influence the outcome of annotation by much.

The genomic distribution of transposable elements (TEs) differs in different types of TEs. Most of the TE are not uniformly distributed across the genome and tend to be enriched in constitutive euchromatin (Franke 2016) while some other TEs are much more frequent in heterochromatin (Jedlicka et al. 2019). Even though it was shown that MaLR LTRs have a relatively uniform chromosome distribution (Franke et al. 2017; van de Lagemaat et al. 2006), the analysis of the distribution of annotated full-length MT elements shows that they are not uniformly distributed alongside chromosomes in C57BL/6J and PWK/PhJ genomes. However, only for full-length MTA elements, that are present in higher numbers C57BL/6J genome, can certain regions (hotspots), in which they have accumulated in higher concentrations, be detected. It is plausible that in these hotspots there are retrotranspositionally active MTA elements which are then spreading to other nearby locations in the C57BL/6J genome.

To examine whether my annotation using filtering conditions of LTRs length was too strict for more fragmented MTA and MTB elements in PWK/PhJ genome, I have done less strict filtering of their LTRs lengths. Although there were more annotated full-length MTA and MTB elements with less rigid filtering, their two LTRs had a lower similarity detected with pairwise alignment. For a full-length MT element, it is important that the similarities and lengths of both conserved LTRs are as similar as possible. Therefore, for further analysis, I have selected only those MTA and MTB elements in the PWK/PhJ strain that were annotated the same way as their homologous elements in the C57BL/6J strain which also had higher similarity and longer LTRs lengths. By doing so, I could have lost some of the full-length MT elements in the PWK/PhJ strain which could later on affect my analysis of homologous MT elements between strains. Another step for evaluating the quality of method for annotating full-length MT elements with my approach involved the analysis of uniquely assigned IDs to certain elements by program RepeatMasker. In my annotation method, I have disregarded those IDs because certain integration of other elements or the presence of gaps in the sequence can lead the RepeatMasker program to annotate that one element as two separate elements. So, I have done the same filtering of LTRs lengths as in my method but on the uniquely assigned IDs of MT elements. My method has resulted in a greater number of annotated full-length MT elements. Pearson and colleagues (2007) have identified 1218 full-length MTA elements in the C57BL/6J genome using a slightly different length filtering

approach and also by ignoring uniquely assigned IDs. Whereas I have identified 1217 full-length MTA elements which is almost the same number as they have. This indicates a high accuracy for my method of detection and annotation of full-length MT elements.

Seeing as C57BL/6J and PWK/PhJ genomes exhibit highly similar homologous regions, I have expected that the homologous MT elements can be found on analogous positions in these mouse strain genomes. Those full-length MT elements that have emerged after the separations of these mouse strains can be found off the main diagonal and have lower similarity, as I have demonstrated with the analysis of mapped MT elements from C57BL/6J onto PWK/PhJ genome. Most of the elements that have evolved after the separation of strains are MTA elements, followed by MTB, in the C57BL/6J strain which is in line with their still ongoing active retrotransposition in the genome. To be certain that I have correctly identified homologous MT elements between mouse strains, I analysed whether the MT elements from C57BL/6J are mapped to the same MT group on PWK/PhJ genome. As well as if the surrounding genes of MT elements are the same homologous genes found in both strains. The MT element is deemed conserved whether it passes these two criteria. Most wrongly annotated full-length MT elements from C57BL/6J onto PWK/PhJ are MTA elements followed by MTB elements. This could occur because these elements are incorrectly identified by the RepeatMasker program in either of strain genomes or because new MTA and MTB elements that have emerged after speciation of strains do not have appropriate homologous MT element on PWK/PhJ genome thus they mapped onto the first most similar thing. Furthermore, because MTA and MTB elements are then mapped to very different positions between strains, they do not have the same surrounding homologous genes. This all points out that the older MT elements; MTC, MTD and MTE, are more conserved between strains whereas younger MTA and MTB elements are not so conserved between strains and most of them are new C57BL/6J specific full-length MT elements. As more time passes since the separation of species or strains, new retrotransposons emerge and older one's decay which leads to species/strain-specific retrotransposons (Gardner 2019). I have shown there is a statistically significant difference between the identity of mapped conserved and not conserved MT elements which is in accordance with previously mentioned statements. However, to obtain even more confident identified conserved and strain-specific MT elements, I had mapped annotated full-length MT elements from PWK/PhJ onto C57BL/6J genomes and looked if the same pairs of MT elements occur in both mappings. It is important to mention that the number of identified conserved MT elements is much lower than the number of conserved MT elements obtained with the previously described method. All the MT elements that were labeled conserved in previous steps but have not found their appropriate pair in the second mapping were now labeled as uncertain. Uncertain full-length MT elements in C57BL/6J have homologous MT elements in PWK/PhJ genome that are full-length but these MT elements in PWK/PhJ genome were not annotated and identified as ones due to

their shorter lengths of LTRs. This is one of the drawbacks of my designed method for annotating full-length MT elements and it acquires additional improvements in order to reduce the false negative conserved MT elements between strains. Nevertheless, some uncertain full-length MT elements from C57BL/6J strain have fragmented homologous MT elements in the PWK/PhJ genome which indicates incomplete integration of these elements in the PWK/PhJ genome. Further analysis is needed to identify homologous full-length MT elements that may have undergone complete or incomplete ectopic recombination.

Full-length MT elements integrated into genes show no orientation bias conserved and new strain-specific MT elements which was also shown for relatively young (little divergence) or old TEs (Nellåker et al. 2012). Usually retrotransposons insertions within genes have a strong orientation bias but it has been shown that strand bias is significantly lower for MT elements than for the rest of elements belonging to the MaLR family (Peaston et al. 2004). This would imply that sense-oriented full-length MT elements in C57BL/6J and PWK/PhJ genomes were not removed by selection. The removal of sense-oriented insertions is important to prevent the disruption of gene transcription due to polyA signals within MT elements. Moreover, I have identified that conserved and strain-specific full-length MT elements are more prevalent in intergenic regions compared to gene introns and exons which is a characteristic for all retrotransposon (Gardner 2019). The majority of conserved and strain-specific MT element integrations are in introns of protein-coding genes followed by integrations into non-coding RNA (ncRNA) genes and pseudogenes. A more detailed analysis of these integrations is required to distinguish whether some new strain-specific MT elements contribute to transcriptional regulation genes and/or give rise to secondary structures of certain non-coding genes. However, when manually examining some cases of MT element integrations in protein-coding genes, I have discovered that two C57BL/6J-specific MT elements, MTA and MTB, are integrated into the last exons of olfactory receptor genes. The integrations may be important for contributing to the phenotype differences between C57BL/6J and PWK/PhJ mouse strains, such as in diet and behavior, by affecting the gene expression. The phenotypic differences have already been linked to distinct olfactory receptors (Saraiva et al. 2016) as well as substantial variations in these receptors among inbred strains (Lilue et al. 2018). However, future more detailed analysis of identification of active retrotransposons between C57BL/6J and PWK/PhJ MT retrotransposons and differentially expressed genes using RNA-seq data is needed to validate these findings.

6 Conclusion

Computational analysis of MT retrotransposons in mouse strains C57BL/6J and PWK/PhJ led to the following conclusions:

1. C57BL/6J and PWK/PhJ genomes have large highly similar homologous regions
2. Composition of repetitive elements, including rodent-specific MT elements, is similar between C57BL/6J and PWK/PhJ genomes
3. Developed computational method for identification of full-length MT based on the RepeatMasker annotation of long-terminal repeats (LTRs) yields higher number of annotated full-length MT elements in both mouse strains
4. Phylogenetically younger annotated full-length MT elements, MTA and MTB, are significantly more abundant and have more conserved lengths of LTRs in C57BL/6J than PWK/PhJ genome
5. Annotated full-length MT elements are not uniformly distributed across both mouse strain genomes and retrotranspositionally active MTA elements in C57BL/6J genome may contribute to formation of hotspots from which they expand across C57BL/6J genome
6. Conserved homologous MT elements can be found on analogous positions in C57BL/6J and PWK/PhJ genomes
7. New strain-specific full-length MTA elements are more frequent in C57BL/6J genome than in PWK/PhJ genome
8. Conserved and strain-specific MT elements show no orientation bias when integrated into genes and majority of integrations occur in introns of protein-coding genes in both mouse strains.
9. Two C57BL/6J-specific MT elements, MTA and MTB, are integrated into last exons of olfactory receptor genes thus potentially may contribute to phenotypic difference in mouse strains C57BL/6J and PWK/PhJ.

7 Literature

Adams D.J., Doran A.G., Lilue J., Keane T.M. (2015): The Mouse Genomes Project: a repository of inbred laboratory mouse strain genomes. Mamm. Genome **26**: 403–412.

Altschul S.F., Gish W., Miller W., Myers E.W., Lipman D.J. (1990): Basic local alignment search tool. J. Mol. Biol. **215**: 403–410

Anisimova M. (2019). *Evolutionary Genomics Statistical and Computational Methods Second Edition Methods*. at <http://www.springer.com/series/7651>.

Birney E. (2004): An Overview of Ensembl. Genome Res. **14**: 925–928.

Britten, R.J. and Kohne D.E. (1964): Repeated sequence in DNA. **146**: 529-540.

Casacuberta E., González J. (2013): The impact of transposable elements in environmental adaptation. Mol. Ecol. **22**: 1503–1517.

Chuong E.B., Elde N.C., Feschotte C. (2017): Regulatory activities of transposable elements: From conflicts to benefits. Nat. Rev. Genet. **18**: 71–86.

Crichton J.H., Dunican D.S., MacLennan M., Meehan R.R., Adams I.R. (2014): Defending the genome from the enemy within: Mechanisms of retrotransposon suppression in the mouse germline. Cell. Mol. Life Sci. **71**: 1581–1605.

de Parseval N., Lazar V., Casella J.-F., Benit L., Heidmann T. (2003): Survey of Human Genes of Retroviral Origin: Identification and Transcriptome of the Genes with Coding Capacity for Complete Envelope Proteins. J. Virol. **77**: 10414–10422.

Dewannieux M., Esnault C., Heidmann T. (2003): LINE-mediated retrotransposition of marked Alu sequences. Nat. Genet. **35**: 41–48.

Dolinoy D.C. (2008): The agouti mouse model: An epigenetic biosensor for nutritional and environmental alterations on the fetal epigenome. Nutr. Rev. **66**: 7–11.

Dolinoy D.C., Das R., Weidman J.R., Jirtle R.L. (2007): Metastable epialleles, imprinting, and the fetal origins of adult diseases. Pediatr. Res. **61**: 30–37.

Finnegan D.J. (1989): Eukaryotic transposable elements and genome evolution. Trends Genet. **5**: 103–107.

Flemr M., Malik R., Franke V., Nejepinska J., Sedlacek R., Vlahovicek K., Svoboda P. (2013): A retrotransposon-driven dicer isoform directs endogenous small interfering rna production in mouse oocytes. Cell **155**: 807–816.

Franke V. (2016): Evolucija i funkcija traspozona MT, specifičnih za glodavce. Dissertation, Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet, cited: 05.06.2020., https://urn.nsk.hr/urn:nbn:hr:217:258467.

Franke V., Ganesh S., Karlic R., Malik R., Pasulka J., Horvat F., Kuzman M., Fulka H., Cernohorska M., Urbanova J., Svobodova E., Ma J., Suzuki Y., Aoki F., Schultz R.M., Vlahovicek K., Svoboda P. (2017): Long terminal repeats power evolution of genes and gene expression programs in mammalian oocytes and zygotes. Genome Res. **27**: 1384–1394.

Frazer K.A., Eskin E., Kang H.M., Bogue M.A., Hinds D.A., Beilharz E.J., Gupta R. V., Montgomery J., Morenzoni M.M., Nilsen G.B., Pethiyagoda C.L., Stuve L.L., Johnson F.M., Daly M.J., Wade C.M., Cox D.R. (2007): A sequence-based variation map of 8.27 million SNPs in inbred mouse strains. Nature **448**: 1050–1053.

Gagnier L., Belancio V.P., Mager D.L. (2019): Mouse germ line mutations due to retrotransposon insertions. Mob. DNA **10**: 1–22.

Gardner, J. M. (2019). The Contribution of Retrotransposons to the Transcriptomes of Murine Somatic Cells (Doctoral thesis). cited: 03.06.2020, https://doi.org/10.17863/CAM.41133

Goerner-Potvin P., Bourque G. (2018): Computational tools to unmask transposable elements. Nat. Rev. Genet. **19**: 688–704.

Gogvadze E., Buzdin A. (2009): Retroelements and their impact on genome evolution and functioning. Cell. Mol. Life Sci. **66**: 3727–3742.

Hartl D.L., Lozovskaya E.R., Lawrence J.G. (1992): Nonautonomous transposable elements in prokaryotes and eukaryotes. Genetica **86**: 47–53.

Huber W., Carey V.J., Gentleman R., Anders S., Carlson M., Carvalho B.S., Bravo H.C., Davis S., Gatto L., Girke T., Gottardo R., Hahne F., Hansen K.D., Irizarry R.A., Lawrence M., Love M.I., MaCdonald J., Obenchain V., Oleš A.K., Pagès H., Reyes A., Shannon P., Smyth G.K., Tenenbaum D., Waldron L., Morgan M. (2015): Orchestrating high-throughput genomic analysis with Bioconductor. Nat. Methods **12**: 115–121.

Jedlicka P., Lexa M., Vanat I., Hobza R., Kejnovsky E. (2019): Correction to: Nested plant LTR retrotransposons target specific regions of other elements, while all LTR retrotransposons often target palindromes and nucleosome-occupied regions: In silico study. Mob. DNA **11**: 1–14.

Jurka J., Kapitonov V. V., Pavlicek A., Klonowski P., Kohany O., Walichiewicz J. (2005): Repbase Update, a database of eukaryotic repetitive elements. Cytogenet. Genome Res. **110**: 462–

467.

Karolchik D., Hinrichs A.S., Kent W.J. (2009). *The UCSC genome browser*. Curr. Protoc. Bioinforma.

Kazazian H.H. (2004): Mobile Elements: Drivers of Genome Evolution. Science. **303**: 1626–1632.

Kent W.J. (2002): BLAT---The BLAST-Like Alignment Tool. Genome Res. **12**: 656–664.

Lambowitz A.M., Belfort M. (2014) Mobile bacterial group II introns at the crux of eukaryotic evolution. Microbiol Spectrum 3(1):MDNA3-0050-2014

Lander E.S., Linton L.M., Birren B., Nusbaum C., Zody M.C., Baldwin J., Devon K., Dewar K., … Morgan M.J. (2001): Initial sequencing and analysis of the human genome. Nature **409**: 860–921.

Lilue J., Doran A.G., Fiddes I.T., Abrudan M., Armstrong J., Bennett R., Chow W., Collins J., … Keane T.M. (2018): Sixteen diverse laboratory mouse reference genomes define strain-specific haplotypes and novel functional loci. Nat. Genet. **50**: 1574–1583.

Maksakova I.A., Romanish M.T., Gagnier L., Dunn C.A., van de Lagemaat L.N., Mager D.L. (2006): Retroviral elements and their hosts: Insertional mutagenesis in the mouse germ line. PLoS Genet. **2**: 1–10.

Martin S.L., Li W.L.P., Furano A. V., Boissinot S. (2005): The structures of mouse and human L1 elements reflect their insertion mechanism. Cytogenet. Genome Res. **110**: 223–228.

McClintock B. (1950): The origin and behavior of mutable loci in maize. Proc Natl Acad Sci USA. **36**: 344–355.

Mills R.E., Bennett E.A., Iskow R.C., Devine S.E. (2007): Which transposable elements are active in the human genome? Trends Genet. **23**: 183–191.

Molaro A., Malik H.S. (2016): Hide and seek: How chromatin-based pathways silence retroelements in the mammalian germline. Curr. Opin. Genet. Dev. **37**: 51–58.

Morgan A.P., Welsh C.E. (2015): Informatics resources for the Collaborative Cross and related mouse populations. Mamm. Genome **26**: 521–539.

Nellåker C., Keane T.M., Yalcin B., Wong K., Agam A., Belgard T.G., Flint J., Adams D.J., Frankel W.N., Ponting C.P. (2012): The genomic landscape shaped by selection on transposable elements across 18 mouse strains. Genome Biol. **13**: R45.

Padeken J., Zeller P., Gasser S.M. (2015): Repeat DNA in genome organization and stability.

Curr. Opin. Genet. Dev. **31**: 12–19.

Paigen K. (2003): One hundred years of mouse genetics: An intellectual history. I. The classical period (1902-1980). Genetics **163**: 1–7.

Peaston A.E., Evsikov A. V., Graber J.H., Vries W.N. de, Holbrook A.E., Solter D., Knowles B.B. (2004): Retrotransposons regulate host genes in mouse oocytes and preimplantation embryos. Dev. Cell **7**: 597–606.

Peaston A.E., Knowles B.B., Hutchison K.W. (2007): Genome plasticity in the mouse oocyte and early embryo. Biochem. Soc. Trans. **35**: 618–622.

Platt R.N., Vandewege M.W., Ray D.A. (2018): Mammalian transposable elements and their impacts on genome evolution. Chromosom. Res. **26**: 25–43.

R Core Team (2019). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Saraiva L.R., Kondoh K., Ye X., Yoon K.H., Hernandez M., Buck L.B. (2016): Combinatorial effects of odorants on mouse behavior. Proc. Natl. Acad. Sci. U. S. A. **113**: E3300–E3306.

Smit A.F.A.S. (1996): Structure and Evolution of Mammalian Interspersed Repeats. **6**: 274.

Smit A.F.A, Hubley R., Green P. (2013-2015): RepeatMasker Open-4.0. <http://www.repeatmasker.org>.

Steward C.A., Gonzalez J.M., Trevanion S., Sheppard D., Kerry G., Gilbert J.G.R., Wicker L.S., Rogers J., Harrow J.L. (2013): The non-obese diabetic mouse sequence, annotation and variation resource: An aid for investigating type 1 diabetes. Database **2013**: 1–12.

Suckow M.A., Danneman P., Brayton C. (2001). *The Laboratory Mouse*. CRC Press Boca Raton

Sul J.H., Martin L.S., Eskin E. (2018): Population structure in genetic studies: Confounding factors and mixed models. PLoS Genet. **14**: 1–22.

Szak S.T., Pickeral O.K., Makalowski W., Boguski M.S., Landsman D., Boeke J.D. (2002): Molecular archeology of L1 insertions in the human genome. Genome Biol. **3**: 1–18.

Thomas J., Perron H., Feschotte C. (2018): Variation in proviral content among human genomes mediated by LTR recombination. Mob. DNA **9**: 1–15.

van de Lagemaat L.N., Medstrand P., Mager D.L. (2006): Multiple effects govern endogenous retrovirus survival patterns in human gene introns. Genome Biol. **7**: 1-14.

Waterston R.H., Lindblad-Toh K., Birney E., Rogers J., Abril J.F., Agarwal P., Agarwala R.,

Ainscough R., … Lander E.S. (2002): Initial sequencing and comparative analysis of the mouse genome. Nature **420**: 520–562.

Werren J.H., Nur U., Wu C.I. (1988): Selfish genetic elements. Trends Ecol. Evol. **3**: 297–302.

8 Supplementary

Supplementary 1. List and usage description of used R packages

Package	Usage description
data.table	fast manipulation of vast data in table format
stringr	tools for string manipulation with the use of regular expressions
magrittr	provides a “pipe” operator, %>%, that enables easier writing and reading of the code
rtracklayer	interfacing with genome browsers and easier import/export of different formats, like GFF or GTF
GenomicRanges, GenomicAlignment	manipulation of genomic annotations, alignments, short genomic intervals
Biostrings	quick manipulation and analysis of large biological sequences or sets of sequences
BSgenome	infrastructure for representing genome sequences
ggplot2	simple graphical language for creating elegant and complex plot
ggbio	different graphical representations of genomic data
cowplot	combines multiple plots into one figure
ggpubr	computes and adds statistical results to plots

Supplementary 2. Rscript

```
#Links to data
MOUSE_STRAIN_RMSK="/PATH/TO/REPEATMASKER/ANNOTATION/FILE"
MOUSE_STRAIN_ANNOTFILE="/PATH/TO/ENSEMBL/ANNOTATION/FILE"
MOUSE_MM10_RMSK="/PATH/TO/REPEATMASKER/ANNOTATION/FILE"
MOUSE_MM10_ANNOTFILE="/PATH/TO/ENSEMBL/ANNOTATION/FILE"

#Import files
strain.rmsk <- fread(MOUSE_STRAIN_RMSK)
strain.annotation <- rtracklayer::import(MOUSE_STRAIN_ANNOTFILE)
mm10.rmsk <- fread(MOUSE_MM10_RMSK)
mm10.annotation <- rtracklayer::import(MOUSE_MM10_ANNOTFILE)

#Retrotransposon content comparison
#Percentage of base pairs belonging to each of the repetitive classes in the mouse ge
nome.
my.mouse.palette <- setNames(object = scales::hue_pal()(2), nm = c("C57BL/6J","PWK/Ph
J"))
mouse.rmsk <- rbind(strain.rmsk, mm10.rmsk)
class <- c("LINE","SINE","LTR","Simple_repeat","DNA","Low_complexity","Sattelite","sc
RNA","Unknown","Other")

assignClassRetrotransposons <- function(table, class) {
  table$category <- "Other"
  table[, ID:=paste(repClass, repFamily,repName, sep="|")]
  lapply(class, function(i) table[, category:= ifelse(str_detect(ID, i)==TRUE, i, cat
egory)])
  tb <- as.data.table(table)
  tb[, ":="(first=min(start), last=max(end)), by=rmsk_ID]
  tb[, length:=(last-first), by=rmsk_ID]
  tb.mm10 <- tb[, .SD[1], by=rmsk_ID][, sum(length), by=category]
  tb.mm10[,percentage:=(V1/sum(V1)*100)]
```

```
  return(tb.mm10)
}
#Whole content
mm10.tb.rep <- assignClassRetrotransposons(mm10.rmsk, class)
strain.tb.rep <- assignClassRetrotransposons(strain.rmsk, class)

mouse.tb.rep <- rbind(mm10.tb.rep[,mouse_strain:="C57BL/6J"],
                      strain.tb.rep[,mouse_strain:="PWK/PhJ"])
mouse.tb.rep$category <- factor(mouse.tb.rep$category,
                                levels = c("LINE","LTR","SINE","Simple_repeat","DNA",
"Low_complexity","Other", "Unknown","scRNA" ))

p.rep <- ggplot(mouse.tb.rep, aes(x=category, y=percentage, fill=mouse_strain)) +
  geom_col(stat="identity",position = position_dodge2(preserve = "single")) +
  theme_light() +
  labs(y="Percentage of base pairs %", x=NULL)
p.rep2 <- ggplot(mouse.tb.rep, aes(fill=reorder(category,percentage), y=percentage, x
=mouse_strain)) +
    geom_bar(position="stack", stat="identity") +
  theme_light() +
  labs(y="Percentage of base pairs %", x=NULL)

##LTR
class <- c("ERVK", "ERV1", "ERVL", "ERVL-MaLR", "Gypsy", "Other")
mm10.tb.ltr <- assignClassRetrotransposons(mm10.rmsk[grepl("LTR",repClass)], class)
strain.tb.ltr <- assignClassRetrotransposons(strain.rmsk[grepl("LTR",repClass)], clas
s)

mouse.tb.ltr <- rbind(mm10.tb.ltr[,mouse_strain:="C57BL/6J"],
                      strain.tb.ltr[,mouse_strain:="PWK/PhJ"])
mouse.tb.ltr$category <- factor(mouse.tb.ltr$category,
                                levels = c("ERVL-MaLR","ERVK","ERVL","ERV1","Gypsy",
"Other"))

p.ltr<-ggplot(mouse.tb.ltr, aes(x=category, y=percentage, fill=mouse_strain)) +
  geom_col(stat="identity",position = position_dodge2(preserve = "single")) +
  theme_light() +
  labs(y="Percentage of base pairs %", x=NULL)
p.ltr2 <- ggplot(mouse.tb.rep, aes(x=category, y=percentage, fill=mouse_strain)) +
    geom_bar(stat="identity",position = position_dodge2(preserve = "single")) +
  theme_light() +
  labs(y="Percentage of base pairs %", x=NULL)
#Plot
pdf("", width = 10)
cowplot::plot_grid(p.rep, p.ltr, labels = "AUTO",
          nrow = 2, align = "v", rel_heights=2, rel_widths=c(0.1,0.1))
cowplot::plot_grid(p.rep2, p.ltr2, labels = "AUTO",
          nrow = 2, align = "v", rel_heights=2, rel_widths=c(0.1,0.1))
dev.off()

##MT elements
class <- c("MTA", "MTB", "MTC", "MTD", "MTE")
mm10.tb.mt <- assignClassRetrotransposons(mm10.rmsk[grepl("MT",repName)][repFamily=="
ERVL-MaLR"], class)
strain.tb.mt <- assignClassRetrotransposons(strain.rmsk[grepl("MT",repName)][repFamil
y=="ERVL-MaLR"], class)

mouse.tb.mt <- rbind(mm10.tb.mt[,mouse_strain:="C57BL/6J"],
```

```
                        strain.tb.mt[,mouse_strain:="PWK/PhJ"])
#Plot
pdf("", width = 10)
ggplot(mouse.tb.mt, aes(x=reorder(category,-percentage), y=percentage, fill=mouse_str
ain)) +
  geom_col(stat="identity",position = position_dodge2(preserve = "single")) +
  theme_light(base_size = 18) +
  labs(y="Percentage of base pairs %", x=NULL)

ggplot(mouse.tb.mt, aes(fill=reorder(category,percentage), y=percentage, x=mouse_stra
in)) +
    geom_bar(position="stack", stat="identity") +
  theme_light(base_size = 18) +
  labs(y="Percentage of base pairs %", x=NULL)
dev.off()

#Annotation of full-length MT elements
mm10.rmsk.LTR <- mm10.rmsk[grep("MT", repName)][!(grep("int", repName))]
strain.rmsk.LTR <- strain.rmsk[grep("MT", repName)][!(grep("int", repName))]
mouse.rmsk.LTR <- rbind(mm10.rmsk.LTR, strain.rmsk.LTR)

##### Annotate full length elements
######Look at the distribution of LTRs and filter them
mouse.rmsk.LTR[,group:="Other"]
lapply(c("MTE", "MTD","MTC", "MTA", "MTB"), function(i) mouse.rmsk.LTR[, group:=ifels
e(str_detect(string = repName, pattern = i), i, group)])
#Plot
MTelements <- c("MTA", "MTB", "MTC", "MTD", "MTE")
paletteMT = setNames(object = scales::hue_pal()(length(MTelements)), nm = MTelements)
paletteMT[names(paletteMT)=="MTD"] <- "#00B0F6"
paletteMT[names(paletteMT)=="MTE"] <- "#E76BF3"

pdf("",  width = 20, height = 15)
ggplot(mouse.rmsk.LTR[group!="Other"], aes(x=(as.numeric(end)-start), fill=group)) +
  geom_histogram(binwidth=10,bins=40) +
  theme_light(base_size = 24) +
  labs(x="LTR length of MT element / bp") +
  facet_grid(group~mouse) +
  xlim(0,700)+
  scale_fill_manual(values = paletteMT)
dev.off()

mouse.rmsk.LTR[,LTR_length:=end-start]

annotateFullLengthMT <- function(strain.MTA.rmsk, min.length=250, max.length=500, add
.group="MTA",add.mouse="C57BL/6J") {
  strain.MTA.rmsk <- copy(strain.MTA.rmsk[abs(end-start)>=min.length & abs(end-start)
<=max.length])#filter LTR
  index <- nearest(GRanges(strain.MTA.rmsk))
  new.tb <- rbindlist(lapply(index, function(x) {  return(strain.MTA.rmsk[x,])    }))
  ltr.MTA.pairs <- na.omit(cbind(strain.MTA.rmsk[,.(seqnames, start, end, strand)],
      new.tb[,.(seqnames2=seqnames, start2=start, end2=end, strand2=strand)]))
  ltr.MTA.pairs[, grp2:=.GRP,by=.(start2, end2)][, grp1:=.GRP,by=.(start, end)]
  ltr.MTA.pairs[start>start2, ":="(start=start2, start2=start)]
  ltr.MTA.pairs[end2<end, ":="(end2=end, end=end2)][, dist:=abs(start-end2)][order(gr
p2, dist), .SD[1], by=grp2]
  return(unique(ltr.MTA.pairs[order(grp2, dist), .SD[1], by=grp2][,-c("grp1","grp2")]
)[,":="(group=add.group,
```

```
mouse=add.mouse)])
}

vector.filter.LTR.old <- c("C57BL/6J:MTE:300:410","C57BL/6J:MTD:300:410","C57BL/6J:MT
C:300:410",
                      "C57BL/6J:MTA:390:400", "C57BL/6J:MTB:380:410",
                      "PWK/PhJ:MTE:300:410", "PWK/PhJ:MTD:300:410", "PWK/PhJ:MTC:300:4
10",
                       "PWK/PhJ:MTA:250:500", "PWK/PhJ:MTB:320:450")
vector.filter.LTR.strict <- c("C57BL/6J:MTE:300:410","C57BL/6J:MTD:300:410","C57BL/6J
:MTC:300:410",
                      "C57BL/6J:MTA:390:400", "C57BL/6J:MTB:380:410",
                      "PWK/PhJ:MTE:300:410", "PWK/PhJ:MTD:300:410", "PWK/PhJ:MTC:300:4
10",
                      "PWK/PhJ:MTA:390:400", "PWK/PhJ:MTB:380:410")
mouse.rmsk.MT.annot <- rbindlist(lapply(vector.filter.LTR,
       function(x) {annotateFullLengthMT(mouse.rmsk.LTR[mouse==unlist(strsplit(x, ":"
))[1] &
                                                         group==unlist(strsplit(x, "
:"))[2]],
                                          min.length = as.numeric(unlist(strsplit(x, "
:")))[3],
                                          max.length = as.numeric(unlist(strsplit(x, "
:")))[4],
                                          add.group = unlist(strsplit(x, ":"))[2],
                                          add.mouse = unlist(strsplit(x, ":"))[1] )}))
#Plot
p1.mt.ltr <- ggplot(mouse.rmsk.MT.annot[group=="MTA"][dist>=1500 & dist<=3200], aes(x
=dist, fill=group)) +
  geom_histogram(binwidth=35,bins=60) +
  theme_light(base_size = 22) +
  labs(x="Distance between LTR / bp") +
  facet_grid(~mouse) +
  scale_fill_manual(values = paletteMT)
p2.mt.ltr <- ggplot(mouse.rmsk.MT.annot[group!="MTA"][dist>=1500 & dist<=3200], aes(x
=dist, fill=group)) +
  geom_histogram(binwidth=35,bins=60) +
  theme_light(base_size = 22) +
  labs(x="Distance between LTR / bp") +
  facet_grid(group~mouse)+
  scale_fill_manual(values = paletteMT)

pdf("", width = 20, height = 18)
cowplot::plot_grid(p1.mt.ltr, p2.mt.ltr, labels = "AUTO",
           nrow = 2, align = "v", rel_heights=3, rel_widths=c(0.1,0.1))
p1.mt.ltr
p2.mt.ltr
dev.off()

########Distribution of annotated MT elements on mouse genomes
#mm10
mouse.rmsk.MT.annot.clean <- mouse.rmsk.MT.annot[dist>=1800 & dist<=2200]
mouse.rmsk.MT.annot.clean <- fread("/common/WORK/pstancl/results/projects/integration
_sites_mus_strains/masters/strict_filtering/AnnotatedMTelements210420.csv")
gr.mm10.MT <- makeGRangesFromDataFrame(df = mouse.rmsk.MT.annot.clean[mouse=="C57BL/6
J"][!(grep("random",seqnames))],
                         seqnames.field = "seqnames",
                         start.field = "start",
                         end.field = "end2",
```

```
                          strand.field = "strand",
                          keep.extra.columns = F)
mcols(gr.mm10.MT)$group <- mouse.rmsk.MT.annot.clean[mouse=="C57BL/6J"][!(grep("rando
m",seqnames))]$group
seqlengths(gr.mm10.MT) <- seqlengths(genome_mm10)[names(seqlengths(gr.mm10.MT))]
#pwk
gr.pwk.MT <- makeGRangesFromDataFrame(df = mouse.rmsk.MT.annot.clean[mouse=="PWK/PhJ"
][!(grep("LVX",seqnames))],
                          seqnames.field = "seqnames",
                          start.field = "start",
                          end.field = "end2",
                          strand.field = "strand",
                          keep.extra.columns = F)

mcols(gr.pwk.MT)$group <- mouse.rmsk.MT.annot.clean[mouse=="PWK/PhJ"][!(grep("LVX",se
qnames))]$group
seqlengths(gr.pwk.MT) <- seqlengths(genome_pwk)[names(seqlengths(gr.pwk.MT))]
###Autoplot from ggbio
MTelements <- c("MTA", "MTB", "MTC", "MTD", "MTE")
paletteMT = setNames(object = scales::hue_pal()(length(MTelements)), nm = MTelements)
paletteMT[names(paletteMT)=="MTD"] <- "#00B0F6"
paletteMT[names(paletteMT)=="MTE"] <- "#E76BF3"

pdf("", width=10)
lapply(MTelements, function(i) autoplot(seqinfo(gr.mm10.MT),legend = FALSE) +
  layout_karyogram(gr.mm10.MT[mcols(gr.mm10.MT)[,"group"]==i],
                   aes(fill=group, color=as.character(paletteMT[names(paletteMT)==i])
), alpha=0.5) +
    scale_color_manual(values=as.character(paletteMT[names(paletteMT)==i])))
lapply(MTelements, function(i) autoplot(seqinfo(gr.pwk.MT)) +
  layout_karyogram(gr.pwk.MT[mcols(gr.pwk.MT)[,"group"]==i],
                   aes(fill=group, color=as.character(paletteMT[names(paletteMT)==i])
), alpha=0.5) +
    scale_color_manual(values=as.character(paletteMT[names(paletteMT)==i])))
dev.off()

#Similarity of defined LTR pairs
mouse.rmsk.MT.annot.clean <- mouse.rmsk.MT.annot[dist>=1800 & dist<=2200]

pairwiseAlignemntSimilarity <- function(ltr.MTA, add.group="MTE", add.genome=genome_p
wk, add.mouse="C57BL/6J") {
  ltr.MTA.pairwise <- data.table(LTR.left=BSgenome::getSeq(add.genome, names=ltr.MTA$
seqnames, start=ltr.MTA$start,
                                                           end=ltr.MTA$end, strand=lt
r.MTA$strand, as.character=TRUE),
                                 LTR.right=BSgenome::getSeq(add.genome, names=ltr.MTA$s
eqnames, start=ltr.MTA$start2,
                                                            end=ltr.MTA$end2, strand=lt
r.MTA$strand, as.character=TRUE))
  ltr.MTA.pairwise[, ":=" (score=score(pairwiseAlignment(LTR.left, LTR.right)),
                           perc_match=pid(pairwiseAlignment(LTR.left, LTR.right)),
                           distance=ltr.MTA$dist,
                  start_align=start(subject(pairwiseAlignment(LTR.left, LTR.right))),
                  end_align=end(subject(pairwiseAlignment(LTR.left, LTR.right))),
                  group=add.group,
                  mouse=add.mouse)]
  ltr.MTA.pairwise[,alig_len:=end_align-start_align]
  return(ltr.MTA.pairwise)
}
```

```
vector.similar.LTR <- c("C57BL/6J:MTE:genome_mm10","C57BL/6J:MTD:genome_mm10","C57BL/
6J:MTC:genome_mm10",
                        "C57BL/6J:MTA:genome_mm10", "C57BL/6J:MTB:genome_mm10",
                        "PWK/PhJ:MTE:genome_pwk", "PWK/PhJ:MTD:genome_pwk", "PWK/PhJ:MTC
:genome_pwk",
                        "PWK/PhJ:MTA:genome_pwk", "PWK/PhJ:MTB:genome_pwk")
mm10.rmsk.MT.annot.similarity <- rbindlist(lapply(vector.similar.LTR[1:5],
       function(x) {pairwiseAlignemntSimilarity(mouse.rmsk.MT.annot.clean[mouse==unli
st(strsplit(x, ":"))[1] &
                                                                  group==unlist(strsplit(x, "
:"))[2]],
                                            add.group = unlist(strsplit(x, ":"))[2],
                                            add.genome = genome_mm10,
                                            add.mouse = unlist(strsplit(x, ":"))[1])}))
pwk.rmsk.MT.annot.similarity <- rbindlist(lapply(vector.similar.LTR[6:10],
       function(x) {pairwiseAlignemntSimilarity(mouse.rmsk.MT.annot.clean[mouse==unli
st(strsplit(x, ":"))[1] &
                                                                  group==unlist(strsplit(x, "
:"))[2]],
                                            add.group = unlist(strsplit(x, ":"))[2],
                                            add.genome = genome_pwk,
                                            add.mouse = unlist(strsplit(x, ":"))[1])}))

mouse.rmsk.MT.annot.similarity <- rbind(mm10.rmsk.MT.annot.similarity, pwk.rmsk.MT.an
not.similarity)
##Taking less strict filtering condition (fill it grey and new plot where you color t
he new one)
old.mouse.rmsk.MT.annot.similarity[LTR.right%in%mouse.rmsk.MT.annot.similarity$LTR.ri
ght & LTR.left%in%mouse.rmsk.MT.annot.similarity$LTR.left, group_color:=group]
#Plot pairwise similarity result
pdf("",  width = 20, height = 15)
ggplot(old.mouse.rmsk.MT.annot.similarity, aes(x=alig_len, y=perc_match, color=group_
color)) +
  geom_point(size=2.5) +
  labs(x="Alignment length", y="Percent identity") +
  theme_light(base_size = 24) +
  facet_grid(group~mouse)
dev.off()
#####Comparison of my annotated full-length MT element and RepeatMasker annotation of
elements
annotateFullLengthMT_RMSK <- function(strain.MTA.rmsk, min.length=250, max.length=500
, add.group="MTA",add.mouse="C57BL/6J") {
  tb1 <- strain.MTA.rmsk[mouse==add.mouse & group==add.group & LTR_length>min.length
& LTR_length<max.length]
  tb2 <- tb1[rmsk_ID%in%tb1[,.N,rmsk_ID][N==2]$rmsk_ID]
  tb3 <- tb2[, inner_distance:=max(end)-min(start),rmsk_ID][inner_distance>=1800 & in
ner_distance<=2200]
  return(tb3)
}

mouse.RMSK.MT.annot <- rbindlist(lapply(vector.filter.LTR,
       function(x) {annotateFullLengthMT_RMSK(mouse.rmsk.LTR[mouse==unlist(strsplit(x
, ":"))[1] &
                                                                  group==unlist(strsplit(x, "
:"))[2]],
                                            min.length = as.numeric(unlist(strsplit(x, "
:")))[3],
                                            max.length = as.numeric(unlist(strsplit(x, "
```

```
:")))[4],
                                          add.group = unlist(strsplit(x, ":"))[2],
                                          add.mouse = unlist(strsplit(x, ":"))[1] )}))

rmskAnnot.myAnnot <-rbind(mouse.RMSK.MT.annot[,.N,.(mouse,group,rmsk_ID)][,.N,.(mouse
,group)][,annotation:="RepeatMasker"],
                          mouse.rmsk.MT.annot.clean[,.N,.(mouse,group)][,annotation:
="my_approach"])

pdf("", width = 20, height = 15)
ggplot(rmskAnnot.myAnnot, aes(x=group, y=N, fill=annotation)) +
  geom_col(position = "dodge")+
  labs(x="Class", y="Count") +
  theme_light(base_size = 24) +
  facet_grid(~mouse) +
  scale_fill_manual(values=c("#F08080", "#3CB371"))
dev.off()

###Extract the sequences of annotated full-length MT elements
extractSequencesGenome <- function(add.genome, tb.extract.seq) {
  strain.MT.fa <-  BSgenome::getSeq(add.genome,
                                    names=tb.extract.seq$seqnames,
                                    start=tb.extract.seq$start,
                                    end=tb.extract.seq$end2,
                                    strand=tb.extract.seq$strand)
  names(strain.MT.fa) <- paste(names(strain.MT.fa),
                               tb.extract.seq$start,
                               tb.extract.seq$end2,
                               tb.extract.seq$group,
                               tb.extract.seq$mouse, sep=":")
  return(strain.MT.fa)
}

strain.MT.fa <- extractSequencesGenome(add.genome=genome_pwk, tb.extract.seq=mouse.rm
sk.MT.annot.clean[mouse=="PWK/PhJ"])
mm10.MT.fa <- extractSequencesGenome(add.genome=genome_mm10, tb.extract.seq=mouse.rms
k.MT.annot.clean[mouse!="PWK/PhJ"])

writeXStringSet(strain.MT.fa, "MTanotatedPWK.fasta")
writeXStringSet(mm10.MT.fa, "MTanotatedmm10.fasta")

#Analyse BLAT results
tb.blat <- fread("PATH/TO/BLAT/RESULTS")

#Fixing column names
fixblast8name <- function(table){
  name <- c("query.id","subject.id","identity","alignment.length","mismatches","gap.o
penings","q.start","q.end","s.start","s.end","e-value","bit score")
  names(table) <- name
  return(table)
}
tb.blat <- fixblast8name(tb.blat)

tb.blat[, c("pwk.chr","pwk.start","pwk.end","group", "mouse") := tstrsplit(query.id,
":", fixed=TRUE)]
tb.blat[, ":="(pwk.start=as.numeric(pwk.start),
               pwk.end=as.numeric(pwk.end))]

#Take first best hit
```

```
bestAllMTAHit <- tb.blat[,.SD[1],by=query.id] #tb.blat[,.SD[1],by=.(query.id,subject.
id)][pwk.chr==subject.id]
bestAllMTAHit[s.start>s.end, ":="(s.start=s.end, s.end=s.start)]
bestAllMTAHit[, origin:=ifelse(pwk.chr==subject.id, "same_chromosome",
                               "different_chromosome")]
bestAllMTAHit$subject.id <- factor(bestAllMTAHit.manual$subject.id,
                      levels = c("chr1","chr2","chr3","chr4","chr5", "chr6","chr7",
                              "chr8","chr9","chr10","chr11","chr12","chr13","chr14",
                                 "chr15","chr16","chr17","chr18","chr19","chrX",
                                 "chrUn_LVXU01028580v1"))

#Plot
pdf("",  width = 20, height = 10)
ggplot(bestAllMTAHit.manual[!(grepl("_",subject.id))][!(grepl("_",pwk.start))][!(is.n
a(subject.id))], aes(y=s.start, x=pwk.start, color=identity, shape=origin)) +
  geom_point(size=2.8,show.legend = T)+
  facet_grid(subject.id~group) +
  labs(x="Starting position on C57BL/6J genome", y="Starting position of MT element o
n PWK/PhJ genome")+#,subtitle = x +
  theme_light(base_size = 22) +
  scale_x_continuous(labels = scales::comma)+
  scale_y_continuous(labels = scales::comma)+
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
dev.off()

###Annotate your findings of PWK MTA on mm10 genome and annotate overlaps of exons an
d defined retrotransposons (MT) in PWK and mm10
bestAllMTAHit.manual <- copy(bestAllMTAHit)
bestAllMTAHit.manual[s.start>s.end, ":="(s.start=s.end, s.end=s.start)]
gr.MT.BLAT <- makeGRangesFromDataFrame(df = bestAllMTAHit.manual,
                         seqnames.field = "subject.id",
                         start.field = "s.start",
                         end.field = "s.end",
                         strand.field = "strand",
                         keep.extra.columns = T)

makeTableFromOverlaps <- function(gr.H3K27ac_chr17, gr.genes){
  hits <- findOverlaps(gr.H3K27ac_chr17, gr.genes, ignore.strand=TRUE)
  MTexons.overlaps <- cbind(as.data.table(gr.H3K27ac_chr17[queryHits(hits)])[, setnam
es(.SD, "strand", "strand1")],
                            as.data.table(gr.genes[subjectHits(hits)])[, setnames(.SD,
"strand", "strand")])
  return(MTexons.overlaps)
}

#length
strain.rmsk[,pwk.length:=max(end)-min(start), rmsk_ID]
####make overlap
overlap.MT.pwk <- makeTableFromOverlaps(gr.MT.BLAT, GRanges(strain.rmsk))
##assign appropriate MT category to each mm10 hit on pwk
lapply(c("MTA","MTB","MTC","MTD","MTE"),function(x) {
  add.category <- overlap.MT.pwk[repName %like% x][,.N,query.id]$query.id
  overlap.MT.pwk[query.id%in%add.category, pwk.group:=x]
})

bestAllMTAHit.manual[!(query.id%in%overlap.MT.pwk$query.id)]
overlap.MT.pwk <- rbind(overlap.MT.pwk,bestAllMTAHit.manual[!(query.id%in%overlap.MT.
pwk$query.id)], fill=TRUE)
```

```
hm.plot.tb<-unique(overlap.MT.pwk[,c("query.id","group","pwk.group")])[,.N,.(group,pwk.group)]
#Depend on the mapping result be careful about x and y-axis
pdf("")
ggplot(data = hm.plot.tb, aes(x = group, y = pwk.group)) +
  geom_tile(aes(fill = log(N)), colour = "white") +
  geom_text(aes(label = sprintf("%1.2f", N)), vjust = 1) +
  scale_fill_gradient(low = "white", high ="salmon2" ) +
  labs(y = "PWK/PhJ MT annotation", x = "C57BL/6J MT annotation") +
  theme_classic()
dev.off()

##Look at the genes
mm10.genes <- as.data.table(mm10.annotation)[type=="gene" & gene_biotype=="protein_coding"]
strain.genes <- as.data.table(strain.annotation)[type=="gene" & gene_biotype=="protein_coding"]
strain.genes[,projection_parent_gene_id:=str_extract(projection_parent_gene,"ENSMUSG\\d+")]

#Same annotation of protein coding genes for mm10 and pwk
mm10.same.genes <- GRanges(mm10.genes[gene_id%in%strain.genes$projection_parent_gene_id])
strain.same.genes <- GRanges(strain.genes[projection_parent_gene_id%in%mm10.genes$gene_id])

###Plot genes
mouse.same.genes <- merge(as.data.table(mm10.same.genes), as.data.table(strain.same.genes), by.x = "gene_id", by.y = "projection_parent_gene_id") %>%
  .[gene_biotype.x=="protein_coding" | gene_biotype.y=="protein_coding"]
mouse.same.genes[, origin:=ifelse(as.character(seqnames.y)==as.character(seqnames.x), "same_chromosome", "different_chromosome")]

#Remove genes that are on different chromosome between mouse strains
blacklist.genes <- unique(mouse.same.genes[origin!="same_chromosome"]$gene_id)
mm10.same.genes <- GRanges(mm10.genes[gene_id%in%strain.genes$projection_parent_gene_id][!(gene_id%in%blacklist.genes)])
strain.same.genes <- GRanges(strain.genes[projection_parent_gene_id%in%mm10.genes$gene_id][!(projection_parent_gene_id%in%blacklist.genes)])
#remove
mm10.same.genes <- GRanges(mm10.genes[gene_id%in%strain.genes$projection_parent_gene_id][gene_biotype=="protein_coding"])
strain.same.genes <- GRanges(strain.genes[projection_parent_gene_id%in%mm10.genes$gene_id][gene_biotype=="protein_coding"])
#####Border genes on mm10
gr.MT.mm10 <-  makeGRangesFromDataFrame(df = bestAllMTAHit.manual,
                         seqnames.field = "pwk.chr",
                         start.field = "pwk.start",
                         end.field = "pwk.end",
                         keep.extra.columns = F)
#####Border genes on pwk
gr.MT.strain <-  makeGRangesFromDataFrame(df = bestAllMTAHit.manual,
                         seqnames.field = "subject.id",
                         start.field = "s.start",
                         end.field = "s.end",
                         keep.extra.columns = F)

findBorderGenes <- function(add.gr, add.table, add.genes, add.mouse="mm10") {
  ##precede() and follow()=overlapping ranges are excluded. Ignore the strand.
```

```
  up <- precede(add.gr, add.genes, ignore.strand=TRUE)
  down <- follow(add.gr, add.genes, ignore.strand=TRUE)
  ##remembering those with NA so we can change that
  correction <- data.table(upstream=up, downstream=down)
  ##replacing NA with 1. Important to include the corection table later and replace g
enes with NA.
  up[is.na(up)] <- 1
  down[is.na(down)] <- 1
  mm10.gene.MT <- rbind(as.data.table(add.genes[up])[,category:="up_mm10"],
                        as.data.table(add.genes[down])[,category:="down_mm10"])
  index <- nearest(add.gr, add.genes, ignore.strand=TRUE)
  index[is.na(index)] <- 1
  if (add.mouse=="mm10") {
    add.table[,paste("down_gene",add.mouse, sep="_"):=as.data.table(add.genes[down])$
gene_id]
    add.table[,paste("up_gene",add.mouse, sep="_"):=as.data.table(add.genes[up])$gene
_id]
    add.table[,paste("nearest_gene",add.mouse, sep="_"):=as.data.table(add.genes[inde
x])$gene_id]
  } else {
    add.table[,paste("down_gene",add.mouse, sep="_"):=as.data.table(add.genes[down])$
projection_parent_gene_id]
    add.table[,paste("up_gene",add.mouse, sep="_"):=as.data.table(add.genes[up])$proj
ection_parent_gene_id]
    add.table[,paste("nearest_gene",add.mouse, sep="_"):=as.data.table(add.genes[inde
x])$projection_parent_gene_id]
  }
  return(add.table)
}

findBorderGenes(add.gr=gr.MT.mm10,
                        add.table=bestAllMTAHit.manual,
                        add.genes= mm10.same.genes,
                        add.mouse="mm10")
findBorderGenes(add.gr=gr.MT.strain,
                        add.table=bestAllMTAHit.manual,
                        add.genes= strain.same.genes,
                        add.mouse="pwk")

#Finding same surrounding genes
 bestAllMTAHit.manual[down_gene_pwk==down_gene_mm10 | up_gene_pwk==up_gene_mm10 | nea
rest_gene_pwk==nearest_gene_mm10 |  down_gene_pwk==up_gene_mm10 | up_gene_pwk==down_g
ene_mm10 | nearest_gene_pwk==up_gene_mm10 | nearest_gene_pwk==down_gene_mm10 | neares
t_gene_mm10==down_gene_pwk |
                        nearest_gene_mm10==up_gene_pwk, gene_filter:=TRUE]

pass.border.criteria <- bestAllMTAHit.manual[down_gene_pwk==down_gene_mm10 | up_gene_
pwk==up_gene_mm10 | nearest_gene_pwk==nearest_gene_mm10 |  down_gene_pwk==up_gene_mm1
0 | up_gene_pwk==down_gene_mm10 | nearest_gene_pwk==up_gene_mm10 |nearest_gene_pwk==d
own_gene_mm10 | nearest_gene_mm10==down_gene_pwk |
                        nearest_gene_mm10==up_gene_pwk][,.(N_pass=.N),group]
total.number.annotated.MT <- bestAllMTAHit.manual[,.(N_total=.N),group]

summary.border.filtering <- cbind(pass.border.criteria[order(group)], total.number.an
notated.MT[order(group),-c("group")])[,N_fail:=N_total-N_pass][,":="(percentage_pass=
N_pass/N_total,percentage_fail=N_fail/N_total)]%>%
  melt(., id.vars = c("group"),
               measure.vars = c("percentage_fail", "percentage_pass")) %>%
  .[,gene_filtering:=ifelse(variable=="percentage_fail","FALSE","TRUE")]
```

X

```
#Plot
pdf("")
ggplot(summary.border.filtering, aes(x=factor(group),y=value*100, fill=gene_filtering
))+
    geom_col(position="stack", alpha=0.65) +
  theme_classic(base_size = 15)+
  labs(x=NULL, y="Percentage %")+
  geom_text(data=summary.border.filtering, aes(label=paste(round(value*100,1), "%")),
              position=position_stack(vjust=0.5)) +
  scale_fill_brewer(palette = "Set1")
dev.off()

##Annotation and gene border criteria
bestAllMTAHit.manual.clean <- merge(bestAllMTAHit.manual,
                                    unique(overlap.MT.pwk[,c("query.id","group","pwk.
group")]), by="query.id")
bestAllMTAHit.manual.clean[, annotation_filter:=ifelse(group.x==pwk.group,TRUE,FALSE)
] %>%
  .[,gene_filter:=ifelse(is.na(annotation_filter),FALSE,annotation_filter) ] %>%
  .[, conserved_MT:=ifelse(annotation_filter==TRUE & gene_filter==TRUE, TRUE, FALSE)]
%>%
  .[is.na(conserved_MT), conserved_MT:=FALSE]

###Plot + statistics
stat.plot <- bestAllMTAHit.manual.clean[, c("alignment.length", "identity","group.x",
"conserved_MT")] %>%
  melt(., id=c("conserved_MT","group.x"))

pdf("", width=18, height = 10)
ggplot(stat.plot, aes(x=conserved_MT,y=value,fill=conserved_MT))+
  geom_boxplot(alpha=0) +
  geom_violin(alpha=0.5)+
  geom_jitter(position = position_jitter(width = .1))+
  theme_light(base_size = 22) +
  facet_grid(variable~group.x, scales = "free_y") +
  stat_compare_means(method = "wilcox.test", label.y = 82.5,label.x = 1, size=5.5) +
  scale_fill_brewer(palette = "Set1")
dev.off()

##Reciprocal BLAT analysis and identification of conserved (unique) homologous
#B6 MT on PWK
B6blat <- fread("B6_BLAT_on_PWK")
names(B6blat)[13:15] <- c("mm10.chr", "mm10.start", "mm10.end")
names(B6blat)[-c(13:15)] <- paste("mm10",names(B6blat)[-c(13:15)], sep="_")

#PWK MT on B6
PWKblat <- fread("PATH/TO/PWK/BLAT/on/B6")

#Overlapping BLAT results from B6 on PWK
gr.B6blat<- makeGRangesFromDataFrame(df = B6blat,
                         seqnames.field = "subject.id",
                         start.field = "s.start",
                         end.field = "s.end",
                         strand.field = "strand",
                         keep.extra.columns = T)
gr.PWK <- makeGRangesFromDataFrame(df = PWKblat,
                         seqnames.field = "pwk.chr",
                         start.field = "pwk.start",
```

```
                          end.field = "pwk.end",
                          strand.field = "strand",
                          keep.extra.columns = T)
res.overlap <- makeTableFromOverlaps(gr.B6blat, gr.PWK)
#Overlapping BLAT results from PWK on B6
gr.PWK2<- makeGRangesFromDataFrame(df = PWKblat,
                          seqnames.field = "subject.id",
                          start.field = "s.start",
                          end.field = "s.end",
                          strand.field = "strand",
                          keep.extra.columns = T)
gr.B6blat2 <- makeGRangesFromDataFrame(df = B6blat,
                          seqnames.field = "mm10.chr",
                          start.field = "mm10.start",
                          end.field = "mm10.end",
                          strand.field = "strand",
                          keep.extra.columns = T)

res.overlap2 <- makeTableFromOverlaps(gr.B6blat2, gr.PWK2)
#Defining unique homologuous and assigning conserveed, strain-specific and uncertain
categories to MT elements
unique.homologs <- merge(res.overlap[,c("mm10_query.id","query.id","group","mm10_grou
p.x")],
      res.overlap2[,c("mm10_query.id","query.id")], by="mm10_query.id", all=TRUE)[que
ry.id.x==query.id.y]
#adding strand for the annotated element
LTRmt <- fread("annotated_full_length_LTRs_in_mice")
LTRmt[, query.id:=paste(seqnames,start,end2,group,mouse,sep=":")]
#B6
integrationMT <- bestAllMTAHit.manual.clean[query.id%in%unique.homologs$mm10_query.id
, unique_homolog:="conserved"] %>%
  .[unique_homolog=="conserved" | conserved_MT==FALSE] %>%
  .[,category:=ifelse(is.na(unique_homolog),"strain-specific", unique_homolog)]

genomic.annotation <- merge(integrationMT, LTRmt[,c("query.id","strand")]) %>%
  .[, strand_MT:=strand]
#PWK
PWKblat <- fread("/PATH/TO/BLAT/PWK/ON/B6")
names(unique.homologs)[1:2] <- c("mm10_query.id","query.id")
PWKblat[query.id%in%unique.homologs$query.id, unique_homolog:="conserved"]
genomic.annotationMT.pwk <- merge(merge(PWKblat, unique.homologs, by="query.id",all=T
RUE), LTRmt[,c("query.id","strand")]) %>%
  .[, strand_MT:=strand]

###Analysis of conserved and strain-specific MT element integrations into genomic reg
ions in both strains
makeDifferentOverlaps <- function(mm10.annotation, gr.MT.BLAT.mm10) {
  gene.range <- as.data.table(mm10.annotation)[type=="gene"] %>%
  GRanges(.)
exons <- as.data.table(mm10.annotation)[type=="exon"] %>%
  GRanges(.)
MTgene.overlaps <- makeTableFromOverlaps(gr.MT.BLAT.mm10, gene.range)

MTgene.overlaps[strand1==strand][,.N,query.id]
MTgene.overlaps[strand1!=strand][,.N,query.id]
MTexons.overlaps <- makeTableFromOverlaps(gr.MT.BLAT.mm10, exons) %>%
  .[, annotation:="exon"]
#GRange object for non_overlapping sets of annotated exons.
non.overlaping.exons.tb <- unique(makeTableFromOverlaps(GenomicRanges::reduce(exons),
```

```
                                                    exons))[strand1==strand]
non.overlaping.exons.tb[, ":="(strand_old=strand,
                               strand=NULL)]
names(non.overlaping.exons.tb)[c(5,6:9)] <- c("strand", "ori_seqnames", "ori_start",
"ori_end", "ori_width")
non.overlaping.exons <- GRanges(non.overlaping.exons.tb[,.SD[1], by=.(gene_id, seqnam
es, start, end)])
#GRangeList for each gene_id
(g.nonOver.exon.list <- split(non.overlaping.exons, non.overlaping.exons$gene_id))
#How many are in intronic reagions?
introns <- unlist(endoapply(g.nonOver.exon.list, gaps))
introns$gene_id <- names(introns)
introns[start(introns)!=1] #check it with data.table #as.data.table(introns)[, .SD[-1
], by=gene_id]

MTintron.overlap <- unique(makeTableFromOverlaps(gr.MT.BLAT.mm10, introns[start(intro
ns)!=1]))
###Intergenic region
strand(gene.range) <- "*"
intergenicRegion <- gaps(gene.range)

MTintergenic.overlap <- makeTableFromOverlaps(gr.MT.BLAT.mm10, intergenicRegion)
names(MTintergenic.overlap)[11:12] <- c("intergenic_start", "intergenic_end")

MTintergenic.overlap[, ":=" (LTR_downstream_gene=ifelse(strand_MT=="+",
                                                        intergenic_end-end,
                                                        start-intergenic_start))]
MTintergenic.overlap[, ":=" (LTR_upstream_gene=ifelse(strand_MT=="+",
                                                      start-intergenic_start,
                                                      intergenic_end-end))]
return(list(MTgene.overlaps, MTintron.overlap, MTintergenic.overlap, MTexons.overlaps
))
}

##B6 analysis
gr.MT.BLAT.mm10 <- makeGRangesFromDataFrame(df = genomic.annotation,
                         seqnames.field = "pwk.chr",
                         start.field = "pwk.start",
                         end.field = "pwk.end",
                         strand.field = "strand",
                         keep.extra.columns = T)
B6.tb.overlaps <- makeDifferentOverlaps(mm10.annotation=mm10.annotation,
                                        gr.MT.BLAT.mm10=gr.MT.BLAT.mm10)

#PWK analysis
gr.MT.pwk <- makeGRangesFromDataFrame(df = genomic.annotationMT.pwk,
                         seqnames.field = "pwk.chr",
                         start.field = "pwk.start",
                         end.field = "pwk.end",
                         strand.field = "strand",
                         keep.extra.columns = T)
PWK.tb.overlaps <- makeDifferentOverlaps(mm10.annotation=strain.annotation,
                                         gr.MT.BLAT.mm10=gr.MT.pwk)
###Divede the results
B6.genes <- B6.tb.overlaps[[1]] %>%
  .[, integration:="gene"]
B6.intron <- B6.tb.overlaps[[2]] %>%
  .[, integration:="intron"]
```

```
B6.intergenic <- B6.tb.overlaps[[3]] %>%
  .[,integration:="intergenic"]
B6.exon <- B6.tb.overlaps[[4]] %>%
  .[, integration:="exon"]
#
PWK.genes <- PWK.tb.overlaps[[1]] %>%
  .[, integration:="gene"]
PWK.intron <- PWK.tb.overlaps[[2]]%>%
  .[, integration:="intron"]
PWK.intergenic <- PWK.tb.overlaps[[3]]%>%
  .[, integration:="intergenic"]
PWK.exon <- PWK.tb.overlaps[[4]]%>%
  .[, integration:="exon"]
##
B6.genes[,c("query.id", "query.id.x", "unique_homolog", "gene_id")][!(is.na(query.id.
x))]
PWK.genes[,c("query.id", "mm10_query.id", "unique_homolog", "projection_parent_gene")
]
B6.genes[query.id%in%PWK.genes$mm10_query.id]
#Sense i antisense
B6.genes[, orientation:=ifelse(strand==strand_MT, "sense", "antisense")]
PWK.genes[, orientation:=ifelse(strand==strand_MT, "sense", "antisense")]
#
gene1 <- B6.genes[,.N,.(query.id,unique_homolog,orientation)] %>%
  .[,mouse:="C57BL/6J"]
gene2 <- PWK.genes[,.N,.(query.id,unique_homolog,orientation)]%>%
  .[,mouse:="PWK/PhJ"]
#Plot
plot.gene.tb <- rbind(gene1,gene2) %>%
  .[, unique_homolog:=ifelse(is.na(unique_homolog), "Strain-specific MT elements", "C
onserved MT elements")]
pdf("", height = 10, width = 10)
ggplot(plot.gene.tb, aes(x=mouse, fill=orientation))+
  geom_bar(position="dodge", alpha=0.85)+
  theme_light(base_size = 23)+
  facet_grid(~unique_homolog) +
  scale_fill_manual(values=brewer.pal(10, "Paired")[c(6,10)])
dev.off()
###Filter and assign the term hierarchically (exon>intron>intergenic)
B6.summary <- rbind(B6.exon[, c("query.id","unique_homolog", "integration")],
                    B6.intron[, c("query.id","unique_homolog", "integration")],
                    B6.intergenic[, c("query.id","unique_homolog", "integration")]) %
>%
  unique(.) %>%
  .[,.SD[1], query.id]%>%
  .[,mouse:="C57BL/6J"]

PWK.summary <- rbind(PWK.exon[, c("query.id","unique_homolog", "integration")],
                    PWK.intron[, c("query.id","unique_homolog", "integration")],
                    PWK.intergenic[, c("query.id","unique_homolog", "integration")])
%>%
  unique(.)%>%
  .[,.SD[1], query.id] %>%
  .[,mouse:="PWK/PhJ"]

integration.sites <- rbind(B6.summary, PWK.summary) %>%
  .[, unique_homolog:=ifelse(is.na(unique_homolog), "Strain-specific MT elements", un
ique_homolog)]
names(integration.sites)[2] <- "full_length_MT_element"
```

```
###Integration of MT elements into exonin and intrnic parts of different genes
B6.genes[, gene_biotype:=ifelse(grepl("pseudogene", gene_biotype)==TRUE,"pseudogene",
gene_biotype)]
B6.genes[, gene_biotype:=ifelse(gene_biotype=="protein_coding",gene_biotype,
                                ifelse(gene_biotype=="pseudogene", gene_biotype,
                                       "ncRNA"))]

PWK.genes[, gene_biotype:=ifelse(grepl("pseudogene", gene_biotype)==TRUE,"pseudogene"
,gene_biotype)]
PWK.genes[, gene_biotype:=ifelse(gene_biotype=="protein_coding",gene_biotype,
                                ifelse(gene_biotype=="pseudogene", gene_biotype,
                                       "ncRNA"))]

p1 <- merge(B6.genes[,c("query.id","gene_biotype")],B6.summary, by="query.id")[,.N,.(
gene_biotype,integration,unique_homolog)] %>%
  .[, mouse:="C57BL/6J"]
p2 <-merge(PWK.genes[,c("query.id","gene_biotype")],PWK.summary, by="query.id")[,.N,.
(gene_biotype,integration,unique_homolog)] %>%
  .[, mouse:="PWK/PhJ"]

gene.types <- rbind(p1,p2) %>%
  .[, unique_homolog:=ifelse(is.na(unique_homolog), "strain-specific", unique_homolog
)]
names(gene.types)[3] <- "full_length_MT_element"
##
g1<-ggplot(plot.gene.tb, aes(x=mouse, fill=orientation))+
  geom_bar(position="dodge", alpha=0.85)+
  theme_light(base_size = 18)+
  facet_grid(~unique_homolog) +
  scale_fill_manual(values=brewer.pal(10, "Paired")[c(6,10)]) +
  labs(x=NULL)
g2 <- ggplot(integration.sites[full_length_MT_element!="conserved"], aes(x=mouse, fil
l=integration))+
  geom_bar(position="dodge")+
  theme_light(base_size = 18)+
  facet_grid(~full_length_MT_element) +
  scale_fill_manual(values=brewer.pal(8, "Dark2")[4:6])+
  labs(x=NULL)
#Plot
pdf("PLOT1.pdf", width=9, height=8)
cowplot::plot_grid(g1,g2, labels = "AUTO",
          nrow = 2, align = "v", rel_heights=2, rel_widths=c(0.1,0.1))
dev.off()

pdf("plot2.pdf", width = 8)
ggplot(gene.types,
       aes(x=mouse, y=N, fill=gene_biotype))+
  geom_col(position = position_dodge2(preserve = "single"))+
  facet_grid(full_length_MT_element~integration) +
  theme_light(base_size = 18) +
  scale_fill_manual(values = c("lightcoral", "lightblue4", "pink4")) +
  labs(x=NULL, y="count")
dev.off()
```

CURRICULUM VITAE

Paula Štancl

paula.stancl@gmail.com

Born: 22.06.1996, Zagreb, Croatia

EDUCATION

2015.-2018.	Bachelor of Science in Molecular Biology Faculty of Science, Division of Biology, Zagreb
2018.-2020.	University Graduate Programme in Molecular Biology Faculty of Science, University of Zagreb (Croatia)

WORK AND RESEARCH EXPERIENCE

2019.	Summer internship at Laboratory of Epigenetic Regulations, Prague Petr Svoboda group
2018.-2020.	Bioinformatics group, Faculty of Science, Zagreb Intern with prof. dr.sc. Kristian Vlahoviček
2018.-2020.	Co-leader of Bioinformatics section at Biology Students Association (BIUS)
2015.-2019.	Student Mentor for national competition in Biology at XV.gymnasium, Zagreb, Croatia

CONFERENCES

2018. and 2019.	International medical conference ZIMS, Zagreb, Croatia Workshop leader: Mutations and disease, a bioinformatics approach Tracking down Alzheimer's
2017.	Science Festival, Technical Museum, Zagreb, Croatia Workshop leader: GLOBE-Global learning and observation for the benefit of nature
2016.	Summer School of Science Požega, Croatia Workshop leader: Phylogenetic tree

AWARDS

2019.	Rector's award for individual scientific work (one/two author) in natural science
2017.	Zagreb scholarship for excellence
2015.	Bronze award at International Environment & Sustainability Project Olympiad, Amsterdam

SKILLS

Languages	Croatian (native proficiency), English (full professional proficiency)
Software	R, Linux