# Supplementary material for ZIMMEROME Part 12 (HERV-K analysis) - Methods

Aurélie Kapusta & Cédric Feschotte

University of Utah School of Medicine

***Associated data files:***

**HERVKloci.bed (bed file)**

**HERV-K\_analysis.xlsx = Details of the loci (excel file)**

### I. Building the known loci coordinates file from references [1,2]

We used the data from references [1,2] and labeled them (“no\_ref”, “ref\_variable” and “ref\_polym” based on ref. [2] Dataset\_S03, ] and “MacFarlane” for the loci listed in ref. [1]), see the file HERVKloci.bed

The coordinates in that file correspond to the empty site for the “no\_ref” ones as well as the locus 19p12c [1] (see case 1 from **SubjectZ\_HERVK\_schema.pptx**), and otherwise to the 5’ end junction, meaning 5’ flanking DNA + HERV-K LTR DNA sequences (see case 2 from **SubjectZ\_HERVK\_schema.pptx**). Exact coordinates can be found in the file HERVKloci.bed, and more details in the excel file

The bam file has numbers without the label chr for the chromosome numbers, replace:

sed –i 's/chr//' HERVKloci.bed > HERVKloci.bed

### II. Intersect with the coordinates of the genomic reads of SubjectZ

We used bedtools v2.25.0 [3].

Just to be safe, sort the loci file:

sortBed -i HERVKloci.bed > HERVKloci.sorted.bed

First, to avoid some errors (“…has inconsistent naming convention for record”), we converted the bam file to bed file:

bamToBed -i PG0004515-BLD.final.bam > PG0004515-BLD.final.bed &

Additionally, to avoid some kill errors, we extracted subsets of reads prior to intersection:

nohup grep "^X" PG0004515-BLD.final.bed > PG0004515-BLD.final.chrX.bed &

nohup grep "^Y" PG0004515-BLD.final.bed > PG0004515-BLD.final.chrY.bed &

nohup grep "^1" PG0004515-BLD.final.bed | grep -v -E "^1[0-9]" > PG0004515-BLD.final.chr1.bed &

nohup grep "^2" PG0004515-BLD.final.bed | grep -v -E "^2[0-9]" > PG0004515-BLD.final.chr2.bed &

nohup grep -E "^1[0-4]" PG0004515-BLD.final.bed > PG0004515-BLD.final.chr10-14.bed &

nohup grep -E "^1[5-9]" PG0004515-BLD.final.bed > PG0004515-BLD.final.chr15-19.bed &

nohup grep -E "^2[0-2]" PG0004515-BLD.final.bed > PG0004515-BLD.final.chr20-22.bed &

nohup grep -E "^[3-4]" PG0004515-BLD.final.bed > PG0004515-BLD.final.chr3-4.bed &

nohup grep -E "^[5-6]" PG0004515-BLD.final.bed > PG0004515-BLD.final.chr5-6.bed &

nohup grep -E "^[7-9]" PG0004515-BLD.final.bed > PG0004515-BLD.final.chr7-9.bed &

Then we ran the intersections:

nohup intersectBed -a HERVKloci.sorted.bed -b PG0004515-BLD.final.chrX.bed -wo > HERVKloci.CZbed.chrX.bed &

nohup intersectBed -a HERVKloci.sorted.bed -b PG0004515-BLD.final.chrY.bed -wo > HERVKloci.CZbed.chrY.bed &

nohup intersectBed -a HERVKloci.sorted.bed -b PG0004515-BLD.final.chr1.bed -wo > HERVKloci.CZbed.chr1.bed &

nohup intersectBed -a HERVKloci.sorted.bed -b PG0004515-BLD.final.chr2.bed -wo > HERVKloci.CZbed.chr2.bed &

nohup intersectBed -a HERVKloci.sorted.bed -b PG0004515-BLD.final.chr10-14.bed -wo > HERVKloci.CZbed.chr10-14.bed &

nohup intersectBed -a HERVKloci.sorted.bed -b PG0004515-BLD.final.chr15-19.bed -wo > HERVKloci.CZbed.chr15-19.bed &

nohup intersectBed -a HERVKloci.sorted.bed -b PG0004515-BLD.final.chr20-22.bed -wo > HERVKloci.CZbed.chr20-22.bed &

nohup intersectBed -a HERVKloci.sorted.bed -b PG0004515-BLD.final.chr3-4.bed -wo > HERVKloci.CZbed.chr3-4.bed &

nohup intersectBed -a HERVKloci.sorted.bed -b PG0004515-BLD.final.chr7-9.bed -wo > HERVKloci.CZbed.chr7-9.bed &

nohup intersectBed -a HERVKloci.sorted.bed -b PG0004515-BLD.final.chr5-6.bed -wo > HERVKloci.CZbed.chr5-6.bed &

### III. Visualize the genomic reads of SubjectZ in the UCSC genome browser

Thanks: Edward B. Chuong

We used bedtools v2.25.0 [3] and samtools v.1-3 [4]

To visualize the reads in UCSC genome browser, we needed to add the chromosome numbers in the bam file:

<http://seqanswers.com/forums/showthread.php?t=22504>

nohup samtools view -h PG0004515-BLD.final.bam | awk 'BEGIN{FS=OFS="\t"} (/^@/ && !/@SQ/){print $0} $2~/^SN:[1-9]|^SN:X|^SN:Y|^SN:MT/{print $0}  $3~/^[1-9]|X|Y|MT/{$3="chr"$3; print $0} ' | sed 's/SN:/SN:chr/g' | sed 's/chrMT/chrM/g' | samtools view -bS - > PG0004515-BLD.final.chr.bam &

Generate the associated .bai:

nohup samtools index PG0004515-BLD.final.chr.bam > PG0004515-BLD.final.chr.bam.bai.log &

Generate the required files:

nohup bedtools genomecov -bg -ibam PG0004515-BLD.final.chr.bam -g hg19.chrom.sizes > PG0004515-BLD.final.chr.bdg &

wget <http://hgdownload.cse.ucsc.edu/admin/exe/linux.x86_64/bedGraphToBigWig>

nohup ./bedGraphToBigWig PG0004515-BLD.final.chr.bdg hg19.chrom.sizes PG0004515-BLD.final.chr.bw > PG0004515-BLD.final.chr.bw.log &

Then these files were uploaded by Ed Chuong to his own Amazon account and loaded as a track in the UCSC genome browser. This allowed us to check specific loci and take the screen shots showed the associated file in results (**SubjectZ\_HERV-K\_screenshots.pptx**).

### IV. Check the loci in the HuRef (Craig Venter’s genome) assembly

We used blast 2.2.29+ [5]

#### a. Loci from ref [1]

Fasta files were downloaded from:

<ftp://ftp.ncbi.nih.gov/genomes/Homo_sapiens/ARCHIVE/BUILD.37.3>

wget ftp://ftp.ncbi.nih.gov/genomes/Homo\_sapiens/ARCHIVE/BUILD.37.3/CHR\_02/hs\_alt\_HuRef\_chr2.fa.gz .

wget ftp://ftp.ncbi.nih.gov/genomes/Homo\_sapiens/ARCHIVE/BUILD.37.3/CHR\_03/hs\_alt\_HuRef\_chr3.fa.gz .

wget ftp://ftp.ncbi.nih.gov/genomes/Homo\_sapiens/ARCHIVE/BUILD.37.3/CHR\_04/hs\_alt\_HuRef\_chr4.fa.gz .

wget ftp://ftp.ncbi.nih.gov/genomes/Homo\_sapiens/ARCHIVE/BUILD.37.3/CHR\_05/hs\_alt\_HuRef\_chr5.fa.gz .

wget ftp://ftp.ncbi.nih.gov/genomes/Homo\_sapiens/ARCHIVE/BUILD.37.3/CHR\_06/hs\_alt\_HuRef\_chr6.fa.gz .

wget ftp://ftp.ncbi.nih.gov/genomes/Homo\_sapiens/ARCHIVE/BUILD.37.3/CHR\_07/hs\_alt\_HuRef\_chr7.fa.gz .

wget ftp://ftp.ncbi.nih.gov/genomes/Homo\_sapiens/ARCHIVE/BUILD.37.3/CHR\_08/hs\_alt\_HuRef\_chr8.fa.gz .

wget ftp://ftp.ncbi.nih.gov/genomes/Homo\_sapiens/ARCHIVE/BUILD.37.3/CHR\_12/hs\_alt\_HuRef\_chr12.fa.gz .

wget ftp://ftp.ncbi.nih.gov/genomes/Homo\_sapiens/ARCHIVE/BUILD.37.3/CHR\_19/hs\_alt\_HuRef\_chr19.fa.gz .

Concatenate the ones with annotated loci:

cat hs\_alt\_HuRef\_chr2.fa hs\_alt\_HuRef\_chr3.fa hs\_alt\_HuRef\_chr4.fa hs\_alt\_HuRef\_chr5.fa hs\_alt\_HuRef\_chr6.fa hs\_alt\_HuRef\_chr7.fa hs\_alt\_HuRef\_chr8.fa hs\_alt\_HuRef\_chr12.fa hs\_alt\_HuRef\_chr19.fa > hs\_alt\_HuRef.somechr.fa

Build the blast db:

/home/software/ncbi-blast-2.2.29+/bin/makeblastdb -dbtype nucl -in hs\_alt\_HuRef.somechr.fa

Building a new DB, current time: 03/14/2016 16:50:31

New DB name:   hs\_alt\_HuRef.somechr.fa

New DB title:  hs\_alt\_HuRef.somechr.fa

Sequence type: Nucleotide

Keep Linkouts: T

Keep MBits: T

Maximum file size: 1000000000B

Adding sequences from FASTA; added 434 sequences in 43.7227 seconds.

Extract the sequences of the junctions:

bedtools getfasta -fi /data/genomes/Homo\_sapiens/hg19/fa/hg19.fa -bed MacFarlan.bed -fo MacFarlan.fa

Make the junctions file:

MacFarlan.junctions.fa

Blast fasta files against the HuRef assembly:

/home/software/ncbi-blast-2.2.29+/bin/blastn -db /data/genomes/Homo\_sapiens/HuRef/chr/hs\_alt\_HuRef.somechr.fa -query MacFarlan.junctions.fa -out MacFarlan.junctions\_HuRef.blast &

/home/software/ncbi-blast-2.2.29+/bin/blastn -db /data/genomes/Homo\_sapiens/HuRef/chr/hs\_alt\_HuRef.somechr.fa -query MacFarlan.fa -out MacFarlan\_HuRef.blast &

Parse:

perl ~/bin/my/parseblast-simple\_ak.pl MacFarlan\_HuRef.blast &

perl ~/bin/my/parseblast-simple\_ak.pl MacFarlan.junctions\_HuRef.blast &

Resulting observations are detailed in the excel file: **HERV-K\_analysis.xls**

#### b. Loci from ref [2]

Fasta files were downloaded from:

http://www.ncbi.nlm.nih.gov/Traces/wgs/?val=ABBA01#contigs

Then were decompressed, renamed, concatenated

make blast db

/home/software/ncbi-blast-2.2.29+/bin/makeblastdb -dbtype nucl -in HuRef.wgsABBA01.fa

Building a new DB, current time: 03/14/2016 16:22:09

New DB name:   HuRef.wgsABBA01.fa

New DB title:  HuRef.wgsABBA01.fa

Sequence type: Nucleotide

Keep Linkouts: T

Keep MBits: T

Maximum file size: 1000000000B

Adding sequences from FASTA; added 254535 sequences in 92.5486 seconds.

Extract the sequences of the junctions

bedtools getfasta -fi /data/genomes/Homo\_sapiens/hg19/fa/hg19.fa -bed Wildschutte.ref.bed -fo Wildschutte.ref.fa

bedtools getfasta -fi /data/genomes/Homo\_sapiens/hg19/fa/hg19.fa -bed Wildschutte.non-ref.bed -fo Wildschutte.non-ref.fa

Blast:

/home/software/ncbi-blast-2.2.29+/bin/blastn -db /data/genomes/Homo\_sapiens/HuRef/HuRef.wgsABBA01.fa -query Wildschutte.non-ref.fa -out Wildschutte.non-ref.fa\_HuRef.blast &

/home/software/ncbi-blast-2.2.29+/bin/blastn -db /data/genomes/Homo\_sapiens/HuRef/HuRef.wgsABBA01.fa -query Wildschutte.ref.fa -out Wildschutte.ref.fa\_HuRef.blast &

Parse:

perl ~/bin/my/parseblast-simple\_ak.pl Wildschutte.non-ref.fa\_HuRef.blast &

perl ~/bin/my/parseblast-simple\_ak.pl Wildschutte.ref.fa\_HuRef.blast &

Resulting observations are detailed in the excel file: **HERV-K\_analysis.xls**

### *References :*

[1] Macfarlane, CM and Badge, RM (2015) Genome-wide amplification of proviral sequences reveals new polymorphic HERV-K(HML-2) proviruses in humans and chimpanzees that are absent from genome assemblies. Retrovirology, Apr 28;12:35.

[2] Wildschutte, JH et al. (2016) Discovery of unfixed endogenous retrovirus insertions in diverse human populations. PNAS, vol.113 no.16.

[3] Quinlan, AR and Hall, IM (2010) BEDTools: a flexible suite of utilities for comparing genomic features. Bioinformatics. 26, 6, pp. 841–842.

[4] Li, H, Handsaker, B et al. (2009) The Sequence alignment/map (SAM) format and SAMtools. Bioinformatics, 25, 2078-9

[5] Camacho, C et al. (2008) BLAST+: architecture and applications. BMC Bioinformatics 10:421