hg38\_all.fa

Download link: <https://hgdownload.soe.ucsc.edu/goldenPath/hg38/bigZips/analysisSet/hg38.analysisSet.chroms.tar.gz>

Generating/Processing: tar -xzvf hg38.analysisSet.chroms.tar.gz

cat ./hg38.analysisSet.chroms/\*.fa > hg38\_all.fa

hg38\_all.fa.fai

Generating/Processing: samtools faidx hg38\_all.fa (samtools version: 0.1.18 (r982:295))

chrNameLength.txt

Generating/Processing: STAR --runMode genomeGenerate --genomeDir ./ --genomeFastaFiles hg38\_all.fa (STAR version:

dbSNP150\_hg38\_single\_genomic.bed

Download link: <https://hgdownload.soe.ucsc.edu/goldenPath/hg38/database/snp150Common.txt.gz>

Generating/Processing: zcat snp150Common.txt.gz | awk 'BEGIN{FS=OFS="\t"}{if (($4-$3==1) && ($11=="genomic") && ($12=="single")) print $2,$3,$4,$5,"0","+"}' > dbSNP150\_hg38\_single\_genomic.bed

dbSNP147\_hg38\_single\_genomic.bed

Download link: <https://hgdownload.soe.ucsc.edu/goldenPath/hg38/database/snp147Common.txt.gz>

Generating/Processing: zcat snp147Common.txt.gz | awk 'BEGIN{FS=OFS="\t"}{if (($4-$3==1) && ($11=="genomic") && ($12=="single")) print $2,$3,$4,$5,"0","+"}' > dbSNP147\_hg38\_single\_genomic.bed

dbSNP150\_hg38\_deletion\_in-del\_insertion\_genomic\_50nt.bed

Generating/Processing: zcat snp150Common.txt.gz | awk 'BEGIN{FS=OFS="\t"}{if (($11=="genomic") && (($12=="deletion") || ($12=="in-del") || ($12=="insertion"))) print $2,$3,$4,$5,"0","+"}' > dbSNP150\_hg38\_deletion\_in-del\_insertion\_genomic.bed

bedtools slop -i dbSNP150\_hg38\_deletion\_in-del\_insertion\_genomic.bed -b 50 -g chrNameLength.txt > dbSNP150\_hg38\_deletion\_in-del\_insertion\_genomic\_50nt.bed (Bedtools version: v2.27.0)

dbSNP147\_hg38\_deletion\_in-del\_insertion\_genomic\_50nt.bed

Generating/Processing: zcat snp147Common.txt.gz | awk 'BEGIN{FS=OFS="\t"}{if (($11=="genomic") && (($12=="deletion") || ($12=="in-del") || ($12=="insertion"))) print $2,$3,$4,$5,"0","+"}' > dbSNP147\_hg38\_deletion\_in-del\_insertion\_genomic.bed

bedtools slop -i dbSNP147\_hg38\_deletion\_in-del\_insertion\_genomic.bed -b 50 -g chrNameLength.txt > dbSNP147\_hg38\_deletion\_in-del\_insertion\_genomic\_50nt.bed

b150\_GMAF\_nonRefAllele\_TGP\_TOPMED\_single\_genomic.bed

Download links: <https://ftp.ncbi.nih.gov/snp/organisms/human_9606_b150_GRCh38p7/known_issues/b150_GMAF_nonRefAllele_TGP.txt.gz>

<https://ftp.ncbi.nih.gov/snp/organisms/human_9606_b150_GRCh38p7/known_issues/b150_GMAF_nonRefAllele_TOPMED.txt.gz>

Generating/Processing: cat b150\_GMAF\_nonRefAllele\_T\*.txt | cut -f 1 | sort | uniq | awk '{print "rs"$0}' | grep -w -f /dev/stdin <(zcat snp150.txt.gz | cut -f 2-7) > b150\_GMAF\_nonRefAllele\_TGP\_TOPMED.bed (grep version: 3.1)

bedtools intersect -a b150\_GMAF\_nonRefAllele\_TGP\_TOPMED.bed -b <(zcat snp150.txt.gz | cut -f 2-7,11,12) -wa -wb -sorted | awk '$4==$10 && $3-$2==1 && $13=="genomic" && $14=="single"' | cut -f 1-6 > b150\_GMAF\_nonRefAllele\_TGP\_TOPMED\_single\_genomic.bed

b150\_GMAF\_nonRefAllele\_TGP\_TOPMED\_deletion\_in-del\_insertion\_genomic\_50nt.bed

Generating/Processing: bedtools intersect -a b150\_GMAF\_nonRefAllele\_TGP\_TOPMED.bed -b <(zcat snp150.txt.gz | cut -f 2-7,11,12) -wa -wb -sorted | awk '$4==$10 && ($13=="genomic") && (($14=="deletion") || ($14=="in-del") || ($14=="insertion"))' | cut -f 1-6 > b150\_GMAF\_nonRefAllele\_TGP\_TOPMED\_deletion\_in-del\_insertion\_genomic.bed

dbSNP150\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED\_revAlleles\_revStrand\_wrongRefAllele.bed

Generating/Processing: cat dbSNP150\_hg38\_single\_genomic.bed | awk -F '\t' '{printf("%s",$1); for (i=2;i<=NF;i++){if ($i=="") $i="NA"; printf("\t%s",$i);} printf "\n"}' | sort --parallel=24 -k1,1 -k2,2n -k3,3n > \_hg38\_single\_genomic\_sorted.bed (sort version: GNU coreutils 8.25)

cat dbSNP150\_hg38\_single\_genomic\_sorted.bed | cut -f 8 | tr "ACGTacgt" "TGCAtgca" | paste dbSNP150\_hg38\_single\_genomic\_sorted.bed /dev/stdin > dbSNP150\_hg38\_single\_genomic\_sorted\_refByStrand.bed

cat dbSNP150\_hg38\_single\_genomic\_sorted\_refByStrand.bed | grep -v -P '^chr.+\_.+\_alt\t' > dbSNP150\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr.bed

cat dbSNP150\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr.bed | awk -F'\t' '$8 ~ /[ACGTacgt]/' > dbSNP150\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT.bed

cat dbSNP150\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT.bed | grep -E '1000GENOMES|TOPMED' > dbSNP150\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED.bed

cat dbSNP150\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED.bed | cut -f 22 | tr "ACGT" "TGCA" | paste dbSNP150\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED.bed /dev/stdin > dbSNP150\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED\_revAlleles.bed

cat dbSNP150\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED\_revAlleles.bed | cut -f 6 | tr "+-" "-+" | paste dbSNP150\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED\_revAlleles.bed /dev/stdin > dbSNP150\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED\_revAlleles\_revStrand.bed

cat dbSNP150\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED\_revAlleles\_revStrand.bed | awk -F'\t' '{if ($21>0) {strand=$6; if ($18 ~ /ObservedMismatch/) strand=$28; ref=$8; if (strand=="-") ref=$26; alleles=$22; n\_ann\_snp=split($9,a,"/"); if (n\_ann\_snp==$21 && $18 ~ /InconsistentAlleles/ && alleles !~ /N/) alleles=$27; alleles=substr(alleles,1,length(alleles)-1); split(alleles,arr\_alleles,","); split($24,freq,",");max\_allele="";max\_freq=0;for (i in arr\_alleles) {if (freq[i]>max\_freq) {max\_freq=freq[i];max\_allele=arr\_alleles[i]}} if (max\_allele!=ref) print $0}}' > dbSNP150\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED\_revAlleles\_revStrand\_wrongRefAllele.bed

dbSNP146\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED\_revAlleles\_revStrand\_wrongRefAllele.bed

Download link: <https://hgdownload.soe.ucsc.edu/goldenPath/hg38/database/snp146Common.txt.gz>

Generating/Processing: zcat snp146Common.txt.gz | awk 'BEGIN{FS=OFS="\t"}{if (($4-$3==1) && ($11=="genomic") &&($12=="single")) print $2,$3,$4,$5,$6,$7}' > dbSNP146\_hg38\_single\_genomic.bed

cat dbSNP146\_hg38\_single\_genomic.bed | awk -F '\t' '{printf("%s",$1); for (i=2;i<=NF;i++){if ($i=="") $i="NA"; printf("\t%s",$i);} printf "\n"}' | sort --parallel=24 -k1,1 -k2,2n -k3,3n > \_hg38\_single\_genomic\_sorted.bed (sort version: GNU coreutils 8.25)

cat dbSNP146\_hg38\_single\_genomic\_sorted.bed | cut -f 8 | tr "ACGTacgt" "TGCAtgca" | paste dbSNP146\_hg38\_single\_genomic\_sorted.bed /dev/stdin > dbSNP146\_hg38\_single\_genomic\_sorted\_refByStrand.bed

cat dbSNP146\_hg38\_single\_genomic\_sorted\_refByStrand.bed | grep -v -P '^chr.+\_.+\_alt\t' > dbSNP146\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr.bed

cat dbSNP146\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr.bed | awk -F'\t' '$8 ~ /[ACGTacgt]/' > dbSNP146\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT.bed

cat dbSNP146\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT.bed | grep -E '1000GENOMES|TOPMED' > dbSNP146\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED.bed

cat dbSNP146\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED.bed | cut -f 22 | tr "ACGT" "TGCA" | paste dbSNP146\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED.bed /dev/stdin > dbSNP146\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED\_revAlleles.bed

cat dbSNP146\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED\_revAlleles.bed | cut -f 6 | tr "+-" "-+" | paste dbSNP146\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED\_revAlleles.bed /dev/stdin > dbSNP146\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED\_revAlleles\_revStrand.bed

cat dbSNP146\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED\_revAlleles\_revStrand.bed | awk -F'\t' '{if ($21>0) {strand=$6; if ($18 ~ /ObservedMismatch/) strand=$28; ref=$8; if (strand=="-") ref=$26; alleles=$22; n\_ann\_snp=split($9,a,"/"); if (n\_ann\_snp==$21 && $18 ~ /InconsistentAlleles/ && alleles !~ /N/) alleles=$27; alleles=substr(alleles,1,length(alleles)-1); split(alleles,arr\_alleles,","); split($24,freq,",");max\_allele="";max\_freq=0;for (i in arr\_alleles) {if (freq[i]>max\_freq) {max\_freq=freq[i];max\_allele=arr\_alleles[i]}} if (max\_allele!=ref) print $0}}' > dbSNP146\_hg38\_single\_genomic\_sorted\_refByStrand\_noAltChr\_onlyACGT\_TGP\_TOPMED\_revAlleles\_revStrand\_wrongRefAllele.bed

Alu\_hg38\_160118\_sorted.bed

Download link: https://hgdownload.soe.ucsc.edu/goldenPath/hg38/database/rmsk.txt.gz

Generating/Processing: zcat rmsk.txt.gz | awk 'BEGIN{FS=OFS="\t"}{if ($13=="Alu") print $6,$7,$8,$11,"0",$10}' | sort -k1,1 -k2,2n > Alu\_hg38\_160118\_sorted.bed

RefSeq\_Curated\_NM\_CDS\_010817\_spliced\_noNs\_chopped\_76\_19\_mms.bed

Retrieval: RefSeq\_Curated\_NM\_CDS\_010817.bed input file was retrieved using UCSC table browser (<http://genome.ucsc.edu/cgi-bin/hgTables>). We used the RefSeq Curated table from the NCBI RefSeq track and filtered only records with name=”NM\_\*”. The filtration results were retrieved in BED format requiring only coding exons in the final output file.

Generating/Processing: bedtools getfasta -bed RefSeq\_Curated\_NM\_CDS\_010817.bed -fi hg38\_all.fa -s -bedOut 2> /dev/null | awk 'BEGIN{FS=OFS="\t"}{split($4,a,"\_");print $1,$2,$3,a[1]"\_"a[2],$5,$6,$7}' | grep -i -v "NN" | perl sequence\_splicer\_gene\_coords.pl /dev/stdin RefSeq\_Curated\_NM\_CDS\_010817\_spliced\_noNs.fa

perl sequence\_chopper\_with\_coords.pl RefSeq\_Curated\_NM\_CDS\_010817\_spliced\_noNs.fa 76 19 RefSeq\_Curated\_NM\_CDS\_010817\_spliced\_noNs\_chopped\_76\_19.fa

blat hg38\_all.fa RefSeq\_Curated\_NM\_CDS\_010817\_spliced\_noNs\_chopped\_76\_19.fa -stepSize=5 -repMatch=2253 -minScore=0 -minIdentity=0 -out=pslx -noHead RefSeq\_Curated\_NM\_CDS\_010817\_spliced\_noNs\_chopped\_76\_19.pslx (BLAT version: 36x2)

cat RefSeq\_Curated\_NM\_CDS\_010817\_spliced\_noNs\_chopped\_76\_19.pslx | awk '$1>=68 && $2>0 && $5==0 {aln\_blocks\_sum=0; split($19,aln\_blocks,","); for (i=1;i<=$18;i++) aln\_blocks\_sum+=aln\_blocks[i]; if ($1/aln\_blocks\_sum >= 0.96) print $0}' | awk 'BEGIN{FS=OFS="\t"; rev["A"]="T";rev["C"]="G";rev["G"]="C";rev["T"]="A"}{aln\_length=$1; aln\_strand=$9; n=split($10,query\_name,"\_"); query\_chr=query\_name[4]; query\_strand=query\_name[3]; if (query\_strand=="-") query\_strand\_flag=1; else query\_strand\_flag=0; j=1; for (i=5; i<=n; i++) {split(query\_name[i],query\_intervals,"-"); query\_intervals[1]+=1; for (w=query\_intervals[1+(1\*query\_strand\_flag)]; w!=query\_intervals[2-(1\*query\_strand\_flag)]+(1-(2\*query\_strand\_flag));w=w+(1-(2\*query\_strand\_flag))){query\_coords[j]=w;j++}} n\_aln=$18; split($19,aln\_blocks,","); split($20,query\_blocks,","); split($21,target\_blocks,","); split($22,query\_seqs,","); split($23,target\_seqs,","); if (aln\_strand=="-") aln\_strand\_flag=1; else aln\_strand\_flag=0; for (cur\_block=n\_aln-(n\_aln+1)\*aln\_strand\_flag;cur\_block!=0+(n\_aln+1)\*aln\_strand\_flag; cur\_block=cur\_block-1+2\*aln\_strand\_flag){cur\_aln\_length=aln\_blocks[cur\_block]; for (pos=1; pos<=cur\_aln\_length;pos++) {query\_nuc=toupper(substr(query\_seqs[cur\_block],pos,1)); target\_nuc=toupper(substr(target\_seqs[cur\_block],pos,1)); if (query\_nuc!=target\_nuc) {if (aln\_strand=="-"){query\_nuc=rev[query\_nuc];target\_nuc=rev[target\_nuc]} query\_pos=pos+query\_blocks[cur\_block]; query\_coord=query\_coords[query\_pos+(j-1-2\*query\_pos+1)\*aln\_strand\_flag]; target\_pos=pos+target\_blocks[cur\_block]-1; print query\_chr,query\_coord-1,query\_coord, query\_nuc target\_nuc,"0", query\_strand, $14 "\_" target\_pos "\_" target\_pos+1 "\_" aln\_strand}}}}' > RefSeq\_Curated\_NM\_CDS\_010817\_spliced\_noNs\_chopped\_76\_19\_mms.bed