**Turning the LBD data from GRCh38 to GRCh37**

Firstly, download the LBD dataset from here <http://ftp.ebi.ac.uk/pub/databases/gwas/summary_statistics/GCST90001001-GCST90002000/GCST90001390/>

Used CrossMap 0.6.4. and the “CrossMap vcf” command within python 3.11.4. The environment was created using conda 23.7.2. Steps were to make a vcf file from the GWAS summary statistics data. This was done by taking the chromosome, base pair position, variant ID, effect allele, and other allele (in that order) and adding 2 columns consisting of only a dot for QUAL and FILTER, and a number identifier identical to the row number in the GWAS data in INFO so as to make it easier to then convert the GWAS summary statistics data into GRCh37 from the CrossMap output (doesn’t affect the conversion because CrossMap doesn’t look at those three columns for conversion). I called this vcf file “LBD\_GRCh38.vcf”. The full code on how this was created is in “LBD\_GRCh38\_to\_GRCh37\_prep.R”.

**Of note:** After creating “LBD\_GRCh38.vcf”, I manually added a “#” on the first line to the left of “CHROM” to emulate a vcf file better. Without this, the code will not run properly.

The full CrossMap command looked like this:

CrossMap vcf hg38ToHg19.over.chain.gz LBD\_GRCh38.vcf hg19.fa out\_LBD\_hg19.vcf

Vcf: Means that you are converting a vcf file

hg38ToHg19.over.chain.gz: **input chain** File provided by the makers of CrossMap that has instructions on switching from GRCh38 to GRCh37. Describes pairwise alignments between two genomes. Format described in greater detail (<https://genome.ucsc.edu/goldenPath/help/chain.html>) Details for the one used (<http://hgdownload.soe.ucsc.edu/goldenPath/hg38/liftOver/>), install link for the one used (<http://hgdownload.soe.ucsc.edu/goldenPath/hg38/liftOver/hg38ToHg19.over.chain.gz>)

LBD\_GRCh38.vcf: Described above.

hg19.fa **refgenome.fa** Chromosome sequences of target assembly in FASTA. My target assembly was GRCh37/hg19. More information on all available hg19 fastas (<https://hgdownload.soe.ucsc.edu/goldenPath/hg19/bigZips/>) , the one I downloaded (<https://hgdownload.soe.ucsc.edu/goldenPath/hg19/bigZips/latest/hg19.fa.gz>)

out\_LBD\_hg19.vcf Name of converted output.

At first, of the 7,843,595 SNPs present, CrossMap was only able to map 1,904,477 to hg19. The good thing about CrossMap is that it also outputs the unmapped SNPs in a file called “out\_LBD\_hg19.vcf.unmap” and gives an explanation as to why each SNP wasn’t mapped to the target genome. They all said that (REF==ALT), so I switched the order of the alleles and created a separate file for them because I knew that afterwards I would have the switch the alleles again. I called this new vcf “LBD\_switch\_alleles\_GRCh38.vcf”. How I did this is documented in “LBD\_GRCh38\_to\_GRCh37\_switch\_alleles.R”. I then I ran CrossMap again

**Of note:** After creating “LBD\_switch\_alleles\_GRCh38.vcf”, I manually added a “#” on the first line to the left of “CHROM” to emulate a vcf file better. Without this, the code will not run properly.

CrossMap vcf hg38ToHg19.over.chain.gz LBD\_switch\_alleles\_GRCh38.vcf hg19.fa out\_LBD\_switch\_alleles\_hg19.vcf

Of the 5939118 million variants in the new vcf file, only 14,694 went unmapped, saved in “out\_ LBD\_switch\_alleles\_hg19.vcf.unmap”. When checking for the reason why these 15k variants went unmapped, the program just said “fail (unmap)”. Tried to do it again using only those 14,694 variants and adding extra parameters, but they couldn’t map. Not much I could do about it; those 14,694 variants will be removed from the dataset. I checked and none of them reach genome-wide significance in the LBD GWAS anyways (lowest pval = 0.000224420). I then, on R 4.3.3, harmonised the data, adding the GRCh37 values and taking into account the changes in allele order, thus successfully switching over to GRCh37. How I did this is documented in “Convert\_LBD\_hg38\_to\_hg19.R”.