**Regenie Step 1**

**Plink**

for anc in ALL; do plink2 \

  --bfile /project\_root/UKBB/merged\_genotyping\_files/ukb22418\_allchr\_hg38 \

  --mac 100 --maf 0.01 --geno 0.1 --hwe 1e-15 \

  --keep /project\_root/ML\_GWAS/Input/ids\_${anc}.csv  \

  --no-pheno --write-snplist --write-samples --no-id-header \

  --out /project\_root/ML\_GWAS/Plink/qc\_pass\_${anc}; done

for anc in afr amr sas eas eur mid; do plink2 \

  --bfile /project\_root/UKBB/merged\_genotyping\_files/ukb22418\_allchr\_hg38 \

  --mac 100 --maf 0.01 --geno 0.1 --hwe 1e-15 \

  --keep /project\_root/NAFLD/Genetics/Input/Ancestry/ids\_${anc}.csv  \

  --no-pheno --write-snplist --write-samples --no-id-header \

  --out /project\_root/ML\_GWAS/Plink/qc\_pass\_${anc}; done

**Cases - pooled analysis**

for anc in ALL; do bsub -q premium -P acc\_GENECAD -W 29:59 -n 24 -o run.out -e run.error -R span[hosts=1] -R rusage[mem=1000] /project\_root/regenie/regenie --step 1 --extract /project\_root/ML\_GWAS/Plink/qc\_pass\_${anc}.snplist --bed /project\_root/UKBB/merged\_genotyping\_files/ukb22418\_allchr\_hg38 --covarFile /project\_root/ML\_GWAS/Input/cov\_${anc}.csv --phenoFile /project\_root/ML\_GWAS/Input/cases.csv --bsize 2000 --lowmem --bt --out /project\_root/ML\_GWAS/Step1/fit\_l1\_cases; done

**Cases - ancestry-specific analyses**

for anc in afr amr sas eas mid eur; do bsub -q premium -P acc\_GENECAD -W 29:59 -n 20 -o run.out -e run.error -R span[hosts=1] -R rusage[mem=1000] /project\_root/regenie/regenie --step 1 --extract /project\_root/ML\_GWAS/Plink/qc\_pass\_${anc}.snplist --bed /project\_root/UKBB/merged\_genotyping\_files/ukb22418\_allchr\_hg38 --covarFile /project\_root/ML\_GWAS/Input/cov\_ALL.csv --phenoFile /project\_root/ML\_GWAS/Input/Ancestry/cases\_${anc}.csv --bsize 2000 --lowmem --bt --out /project\_root/ML\_GWAS/Step1\_Anc/fit\_l1\_cases\_${anc}; done

**ML predictions - pooled analysis**

for anc in ALL; do for dx in afib cad celiac gallstone polyp t2d varicose vte; do for cond in A B; do bsub -q premium -P acc\_GENECAD -W 39:59 -n 16 -o run.out -e run.error -R span[hosts=1] -R rusage[mem=1500] /project\_root/regenie/regenie --step 1 --extract /project\_root/ML\_GWAS/Plink/qc\_pass\_${anc}.snplist --bed /project\_root/UKBB/merged\_genotyping\_files/ukb22418\_allchr\_hg38 --covarFile /project\_root/ML\_GWAS/Input/cov\_${anc}.csv --phenoFile /project\_root/ML\_GWAS/Input/${dx}\_scores\_${cond}.csv --bsize 2000 --lowmem --out /project\_root/ML\_GWAS/Step1/fit\_l1\_${dx}\_${cond}; done; done; done

**ML predictions - ancestry-specific analyses**

for anc in afr amr sas eas mid eur; do bsub -q premium -P acc\_GENECAD -W 29:59 -n 20 -o run.out -e run.error -R span[hosts=1] -R rusage[mem=1000] /project\_root/regenie/regenie --step 1 --extract /project\_root/ML\_GWAS/Plink/qc\_pass\_${anc}.snplist --bed /project\_root/UKBB/merged\_genotyping\_files/ukb22418\_allchr\_hg38 --covarFile /project\_root/ML\_GWAS/Input/cov\_ALL.csv --phenoFile /project\_root/ML\_GWAS/Input/Ancestry/preds\_${anc}.csv --bsize 2000 --lowmem --out /project\_root/ML\_GWAS/Step1\_Anc/fit\_l1\_preds\_${anc}; done

**Regenie Step 2**

**Plink**

for anc in ALL; do for chr in {1..22} X; do plink2 \

  --bfile /project\_root/Phenotype/GWAS/filtered/ukb\_imp\_chr${chr} \

  --keep /project\_root/ML\_GWAS/Input/ids\_${anc}.csv --mac 100 --maf 0.001 --geno 0.1 --hwe 1e-10 \

  --no-pheno --write-snplist --write-samples --no-id-header \

  --out /project\_root/ML\_GWAS/Plink/qc\_pass\_chr${chr}\_${anc}; done; done

for anc in afr amr sas eas eur mid; do for chr in {1..22} X; do bsub -q premium -P acc\_GENECAD -W 29:59 -n 6 -o run.out -e run.error -R span[hosts=1] -R rusage[mem=3GB] plink2 \

  --bfile /project\_root/Phenotype/GWAS/filtered/ukb\_imp\_chr${chr} \

  --keep /project\_root/NAFLD/Genetics/Input/Ancestry/ids\_${anc}.csv --mac 100 --maf 0.001 --geno 0.1 --hwe 1e-10 \

  --no-pheno --write-snplist --write-samples --no-id-header \

  --out /project\_root/ML\_GWAS/Plink/qc\_pass\_chr${chr}\_${anc}; done; done

**Cases - Pooled analysis**

for anc in ALL; do for i in {1..22} X; do bsub -q premium -P acc\_GENECAD -W 29:59 -n 12 -o run.out -e run.error -R span[hosts=1] -R rusage[mem=1GB] /project\_root/regenie/regenie3.4 \

  --step 2 \

  --bed /project\_root/Phenotype/GWAS/filtered/ukb\_imp\_chr${i} \

  --phenoFile /project\_root/ML\_GWAS/Input/cases.csv \

  --extract /project\_root/ML\_GWAS/Plink/qc\_pass\_chr${i}\_${anc}.snplist \

  --covarFile /project\_root/ML\_GWAS/Input/cov\_${anc}.csv \

  --bt \

  --firth --approx --pThresh 0.01 \

  --pred /project\_root/ML\_GWAS/Step1/fit\_l1\_cases\_pred.list \

  --bsize 500 \

  --out /project\_root/ML\_GWAS/Step2/cases\_chr${i}; done; done

**Cases - Ancestry-specific analyses**

for anc in afr amr sas eas eur mid; do for i in {1..22} X; do bsub -q premium -P acc\_GENECAD -W 29:59 -n 12 -o run.out -e run.error -R span[hosts=1] -R rusage[mem=1GB] /project\_root/regenie/regenie3.4 \

  --step 2 \

  --bed /project\_root/Phenotype/GWAS/filtered/ukb\_imp\_chr${i} \

  --phenoFile /project\_root/ML\_GWAS/Input/Ancestry/cases\_${anc}.csv \

  --extract /project\_root/ML\_GWAS/Plink/qc\_pass\_chr${i}\_${anc}.snplist \

  --covarFile /project\_root/ML\_GWAS/Input/cov\_ALL.csv \

  --bt \

  --firth --approx --pThresh 0.01 \

  --pred /project\_root/ML\_GWAS/Step1\_Anc/fit\_l1\_cases\_${anc}\_pred.list \

  --bsize 500 \

  --out /project\_root/ML\_GWAS/Step2\_Anc/cases\_${anc}\_chr${i}; done; done

**ML predictions - pooled analysis**

for anc in ALL; do for dx in afib cad celiac gallstone polyp t2d varicose vte; do for cond in A B; do for i in {1..22} X; do bsub -q premium -P acc\_GENECAD -W 29:59 -n 12 -o run.out -e run.error -R span[hosts=1] -R rusage[mem=1GB] /project\_root/regenie/regenie3.4 \

  --step 2 \

  --bed /project\_root/Phenotype/GWAS/filtered/ukb\_imp\_chr${i} \

  --phenoFile /project\_root/ML\_GWAS/Input/${dx}\_scores\_${cond}.csv \

  --extract /project\_root/ML\_GWAS/Plink/qc\_pass\_chr${i}\_${anc}.snplist \

  --covarFile /project\_root/ML\_GWAS/Input/cov\_${anc}.csv \

  --pred /project\_root/ML\_GWAS/Step1/fit\_l1\_${dx}\_${cond}\_pred.list \

  --bsize 500 \

  --out /project\_root/ML\_GWAS/Step2/${dx}\_${cond}\_chr${i}; done; done; done; done

**ML predictions - ancestry-specific analysis**

for anc in afr amr sas eas eur mid; do for i in {1..22} X; do bsub -q premium -P acc\_GENECAD -W 29:59 -n 12 -o run.out -e run.error -R span[hosts=1] -R rusage[mem=1GB] /project\_root/regenie/regenie3.4 \

  --step 2 \

  --bed /project\_root/Phenotype/GWAS/filtered/ukb\_imp\_chr${i} \

  --phenoFile /project\_root/ML\_GWAS/Input/Ancestry/preds\_${anc}.csv \

  --extract /project\_root/ML\_GWAS/Plink/qc\_pass\_chr${i}\_${anc}.snplist \

  --covarFile /project\_root/ML\_GWAS/Input/cov\_ALL.csv \

  --pred /project\_root/ML\_GWAS/Step1\_Anc/fit\_l1\_preds\_${anc}\_pred.list \

  --bsize 500 \

  --out /project\_root/ML\_GWAS/Step2\_Anc/preds\_${anc}\_chr${i}; done; done

**Combine files**

cd /project\_root/ML\_GWAS

# Remove A B C in file names

for file in ./Step2/\*\_A\_\* ./Step2/\*\_B\_\*; do

  mv "$file" "$(echo "$file" | sed 's/\_A\_/\_/g; s/\_B\_/\_/g')"

done

input\_dir="./Step2"

output\_dir="./Step2\_Comb"

# Define the list of conditions

conditions=("afib" "cad" "celiac" "gallstone" "polyp" "t2d" "varicose" "vte")

# Loop through each condition

for condition in "${conditions[@]}"; do

  # Initialize the output file

  output\_file="$output\_dir/cases\_${condition}.regenie"

  > "$output\_file"

  # Loop through each chromosome

  for chr in {1..22} X; do

    input\_file="$input\_dir/cases\_chr${chr}\_${condition}.regenie"

    # If the chromosome file exists, append it to the output file

    if [[ -f "$input\_file" ]]; then

      if [[ "$chr" == "1" ]]; then

        # Copy the header and the data for the first file

        cat "$input\_file" >> "$output\_file"

      else

        # Skip the header and append the data for subsequent files

        tail -n +2 "$input\_file" >> "$output\_file"

      fi

    fi

  done

done

combinations=$(find ./Step2 -type f -name "\*.regenie" | sed -E 's#.\*/([^\_]+)\_chr[0-9X]+\_(.+)\.regenie#\1\_\2#' | sort | uniq)

# Loop through each unique combination

for combination in $combinations; do

    # Extract A and C from the combination

    A=$(echo $combination | cut -d'\_' -f1)

    C=$(echo $combination | cut -d'\_' -f2-)

    # Initialize a variable to store the header

    header=""

    # Initialize an output file

    output\_file="./Step2\_Comb/${A}\_${C}.regenie"

    > "$output\_file"

    # Loop through each chromosome file

    for B in {1..22} X; do

        input\_file="./Step2/${A}\_chr${B}\_${C}.regenie"

        if [ -f "$input\_file" ]; then

            # Get the header from the first file only

            if [ -z "$header" ]; then

                header=$(head -n 1 "$input\_file")

                echo "$header" > "$output\_file"

            fi

            # Append the content excluding the header

            tail -n +2 "$input\_file" >> "$output\_file"

        fi

    done

done

**Combine files - ancestry**

rm -rf /project\_root/ML\_GWAS/Step2\_Anc\_Comb/preds\*

rm -rf /project\_root/ML\_GWAS/Step2\_Anc\_Comb/cases\*

input\_dir="/project\_root/ML\_GWAS/Step2\_Anc"

output\_dir="/project\_root/ML\_GWAS/Step2\_Anc\_Comb"

# Define the list of conditions

conditions=("afib" "cad" "celiac" "gallstone" "polyp" "t2d" "varicose" "vte")

# Loop through each condition

for anc in afr amr sas eas mid eur; do for condition in "${conditions[@]}"; do

  # Initialize the output file

  output\_file="$output\_dir/preds\_${anc}\_${condition}.regenie"

  > "$output\_file"

  # Loop through each chromosome

  for chr in {1..22} X; do

    input\_file="$input\_dir/preds\_${anc}\_chr${chr}\_${condition}\_af.regenie"

    # If the chromosome file exists, append it to the output file

    if [[ -f "$input\_file" ]]; then

      if [[ "$chr" == "1" ]]; then

        # Copy the header and the data for the first file

        cat "$input\_file" >> "$output\_file"

      else

        # Skip the header and append the data for subsequent files

        tail -n +2 "$input\_file" >> "$output\_file"

      fi

    fi

  done

done; done

input\_dir="/project\_root/ML\_GWAS/Step2\_Anc"

output\_dir="/project\_root/ML\_GWAS/Step2\_Anc\_Comb"

conditions=("afib" "cad" "celiac" "gallstone" "polyp" "t2d" "varicose" "vte")

# Loop through each condition

for anc in eur; do for condition in "${conditions[@]}"; do

  # Initialize the output file

  output\_file="$output\_dir/cases\_${anc}\_${condition}.regenie"

  > "$output\_file"

  # Loop through each chromosome

  for chr in {1..22} X; do

    input\_file="$input\_dir/cases\_${anc}\_chr${chr}\_${condition}.regenie"

    # If the chromosome file exists, append it to the output file

    if [[ -f "$input\_file" ]]; then

      if [[ "$chr" == "1" ]]; then

        # Copy the header and the data for the first file

        cat "$input\_file" >> "$output\_file"

      else

        # Skip the header and append the data for subsequent files

        tail -n +2 "$input\_file" >> "$output\_file"

      fi

    fi

  done

done; done

**Create cleaned regenie files for clumping**

# Set directories

input\_dir="/project\_root/ML\_GWAS/Step2\_Comb"

output\_dir="/project\_root/ML\_GWAS/Sig"

# Iterate over each .regenie file in the input directory

for file in "$input\_dir"/\*.regenie; do

    # Define output filename

    output\_file="$output\_dir/$(basename "$file" .regenie).txt"

    # Process the file

    awk 'BEGIN {OFS="\t"}

    NR == 1 {print "snp", "chr", "pos", "ref", "alt", "beta", "se", "pval"}

    NR > 1 && $12 > 7.3010 {

        snp = $1":"$2"\_"$4"\_"$5

        pval = 10^(-$12)

        print snp, $1, $2, $4, $5, $9, $10, pval

    }' "$file" > "$output\_file"

done

for file in /project\_root/ML\_GWAS/Sig/\*; do

    sed -i 's/23:/X:/g' "$file"

done

**Re-generate split files for clumping job submission**

rm -rf /project\_root/MR/Resources/Splits

mkdir /project\_root/MR/Resources/Splits

input\_dir="/project\_root/ML\_GWAS/Sig\_no\_MHC/"

output\_dir="/project\_root/MR/Resources/Splits/"

# Find all unique file names, read them into an array

readarray -t filenames < <(find "$input\_dir" -type f -exec basename {} \; | sort -u)

# Create a master file that contains all unique file names

printf "%s\n" "${filenames[@]}" > "/project\_root/MR/Resources/split.txt"

# Split filenames into chunks

for (( i=0; i<${#filenames[@]}; i+=40 )); do

  # Determine the split file number

  split\_num=$((i/40+1))

  # Create each split file containing up to 40 filenames

  printf "%s\n" "${filenames[@]:i:40}" > "${output\_dir}split\_${split\_num}.txt"

done

**Run clumping jobs**

for i in {1..4}; do

    bsub -q premium -P acc\_GENECAD -W 11:59 -n 8 -o run.out -e run.error -R "span[hosts=1]" -R "rusage[mem=1GB]" \

    "while read filename; do

        plink2 --pfile /project\_root/ML\_GWAS/Imputed/ukb\_imp\_allchr --clump /project\_root/ML\_GWAS/Sig\_no\_MHC/\$filename --clump-p1 5e-8 --clump-p2 5e-8 --clump-r2 0.1 --clump-kb 1000 --clump-field \"pval\" --clump-snp-field \"snp\" --out /project\_root/ML\_GWAS/Clump/\$filename

    done < /project\_root/MR/Resources/Splits/split\_${i}.txt"

done