GWAS Analyses utilising the CARTaGENE database

Bastien CHASSAGNOL and Marie-Pier SCOT-KELLY and Clément and Fabrice

2025-04-23

Contents

# Welcome

This is the online book for running and reproducing: **GWAS Analyses utilising the CARTaGENE database**.

# 1. Main GWAS pipeline

## 1.1 Reproduce the pipeline and questions to address

### 1.1.1 Two-step pipeline

Assuming PCA components have already been computed (report to [Section 1.3.2.1](#sec-white-selection) and [Section 1.3.2.2](#sec-preprocessing-redundancy) for details), follow this two-step protocol:

1. Retrieve all known variants associated with your genes of interest in [Section 1.3.4](#sec-variant-extraction), and generate the corresponding VCF file.
2. Change the response variable to predict in [Listing 1.5](#lst-glm-bash) (with option --glm, the model applied should adjust automatically whether the outcome is binary or continuous).

### 1.1.2 Reproducibility

Two command-lines tools are needed for reproducing the analyses and/or delivering new outcomes:

* [plink2](https://s3.amazonaws.com/plink2-assets/alpha6/plink2_linux_x86_64_20250129.zip) for running most of the SNPs analyses.
* [bcftools](https://github.com/samtools/bcftools/releases/download/1.21/bcftools-1.21.tar.bz2) for extracting variants of interest in a given genomic region.
* Details for installing from scratch these two CLI tools are reported in [Section A.1](#sec-tools-install).

The code, main results (figures and tables), along with reports, are versioned in an unique place, on GitHub:

* [GitHub project: cartagene-gwas](https://github.com/ArnaudDroitLab/cartagene-gwas) to version the code
* R project + renv() to create a reproducible and standalone computational environment.

### 1.1.3 Important Questions to Address!!

* Check Human reference genome assembly, see Section 1.3.4.1. Indeed, among the five genome arrays assembled in Section 1.3.2.1, most have been mapped against GRCh37 (hg19), but at least one has been mapped against the recent hg38 (GRCh38, 2013) version, report to page 38 in Pelletier (2022).
* We definitely do not have the most recent version of CarTaGene (in the paper, around samples mentioned, against in my genotyping arrays database)
* Why starting from BAM/BED/BIM pre-processed files in Section 1.3.2.1, instead of FASTQ files available in /mnt/projects\_tn01/Cartagene/data/merged? Related question: we use the BAM/BED/BIM files stored in folder /mnt/projects\_tn01/Cartagene/analyses/QC for PCA computation in Section 1.3.2.2, but VCF files stored in /mnt/projects\_tn01/Cartagene/analyses/variants\_extraction for running the GWAS analyses, for which reason?
* Why use a biased gene-centric approach, instead of true GWAS (further discussed in Section 1.4.4)?. If ou really want to use a *candidate-gene approach*, report to [A current guide to candidate gene association studies](https://www.cell.com/trends/genetics/abstract/S0168-9525), from David ([2021](#ref-david2021tig)).
* Run sample size experiences prior to gene candidate studies, see Section 2.3.

## 1.2 Introduction

### 1.2.1 GWAS Litterature Review

* Fundamental paper, aka [Genome-wide association studies: theoretical and practical concerns](https://www.nature.com/articles/nrg1522), from Wang et al. ([2005](#ref-wang2005nrg)).
* Book reviewing pros and cons of statistical GWAS approaches, from [Overview of Statistical Methods for Genome-Wide Association Studies](https://doi.org/10.1007/978-1-62703-447-0_6), from Hayes ([2013](#ref-hayes2013gasagp)).
* [Genome-wide association studies](https://www.nature.com/articles/s43586-021-00056-9), from Uffelmann et al. ([2021](#ref-uffelmann2021nrmp)).

### 1.2.2 Design of the CARTaGENE study

* [Cohort profile of the CARTaGENE study: Quebec’s population-based biobank for public health and personalized genomics](https://doi.org/10.1093/ije/dys160), from Awadalla et al. ([2013](#ref-awadalla2013ijoe)):
  + Includes both Single nucleotide polymorphisms (SNPs) and insertion-deletion (InDels).

#### 1.2.2.1 Impute unobserved GWAS in CARTaGENE

* Most of the CARTaGENE samples are *WES* (whole exome sequencing), instead of *WGS* (whole genome sequencing).
* [Évaluation de l’imputation des données génétiques Canadiennes-Françaises](https://umontreal.scholaris.ca/bitstreams/83d64aa1-da20-4a6f-8ae0-9dcb0165417d/download), from Pelletier ([2022](#ref-pelletier2022)). **Key points**[[1]](#footnote-35):
  + **Chapter 3**, named “Evaluation of genetic imputation in the French-Canadian founder population”, pages 54-93. Details the underlying imputation strategy, derived from TOPMed (for Trans-Omics for Precision Medicine).
  + **Page 45/61**: Describes imputation strategies, along with the preprocessing steps. Code for reproducing the analyses is [FC-imputation](https://github.com/JustinPelletier/FC-imputation). **Imputation performance** is much lower for SNPs exhibiting MAF<0.01, strongly promoting early removal of rare variants:
  + **Page 83/99**: Strong polymorphism in some regions, such as HLA.

|  |
| --- |
| Figure 1.1: **GWAS of CarTaGene SNPs using TOPMed Merge-Impute strategy.** The [TOPMed Imputation Server](https://imputation.biodatacatalyst.nhlbi.nih.gov/) has been used. Manhattan plots depict the genome-wide association study (GWAS) results for variants, before and after imputation The figure is reproduced from ([Pelletier 2022, 97–98](#ref-pelletier2022)). The -axis represents the -value association of each variant. Variants with -values equal to zero are highlighted in green, while the horizontal red line indicates the genome-wide significance threshold.. The total sample consisted of individuals, with the **GWAS focus being comparing ground-truth SNPs versus imputed SNPs.** |

## 1.3 Analyses

### 1.3.1 Setup enviroment

#### 1.3.1.1 Bash configuration:

We use *symbolic links* to encapsulate the whole project, see [ln -s Post](https://www.linkedin.com/posts/%F0%9F%8E%AF-ming-tommy-tang-40650014_the-most-underrated-unix-command-for-bioinformatics-activity-7287842704753848321-qhe) for details:

## Render executive files findable  
export PATH=./bin/bcftools/bin:$PATH  
export PATH=./bin/plink2:$PATH  
  
## Check versions  
plink2 --version  
echo -e "\n"  
bcftools --version  
  
## Create symbolic links to organise everything within the same folder  
## warning: if original content is deleted, everything broke!!!!!!  
echo ln -s ./data/genotypages/ /mnt/projects\_tn01/Cartagene/genotypage/imputation/imputation\_merged

## PLINK v2.0.0-a.6.9LM 64-bit Intel (29 Jan 2025)  
##   
##   
## bcftools 1.21  
## Using htslib 1.21  
## Copyright (C) 2024 Genome Research Ltd.  
## License Expat: The MIT/Expat license  
## This is free software: you are free to change and redistribute it.  
## There is NO WARRANTY, to the extent permitted by law.  
## ln -s ./data/genotypages/ /mnt/projects\_tn01/Cartagene/genotypage/imputation/imputation\_merged

#### 1.3.1.2 R configuration:

## data wrangling and visualisations  
library(haven)  
library(flextable)  
library(dplyr)  
library(ggplot2)  
## Required for code linking  
library(downlit)  
library(xml2)  
  
# Avoid warning "Replace previous import ‘utils::findMatches’ by ‘S4Vectors::findMatches’ when loading ‘AnnotationDbi’"  
library(conflicted)  
conflicted::conflict\_prefer("findMatches", "S4Vectors")  
conflicted::conflicts\_prefer(base::setdiff, base::intersect,   
 base::setequal, base::union)  
conflicted::conflicts\_prefer(dplyr::select, dplyr::filter)  
  
  
## Retrieve gene positions  
library(GenomicFeatures) # Pre-load the package for mapping gene IDs  
library(org.Hs.eg.db) ## Provides gene symbol to Entrez ID mapping  
  
## for generating nice visualisations  
source("R/gwas\_plots.R")

### 1.3.2 Compute PCA **loadings**

#### 1.3.2.1 Step 1: Merge genotype arrays

Genotypes were split into five different genotyping arrays. Arrays were merged together using Bash script [merge\_datasets.sh](./mnt/projects_tn01/Cartagene/analyses/QC/merge_datasets.sh), and command --bmerge of tool plink2. This command only keeps matching **SNPs** and **alleles** across the 5 arrays.

The resulting BAM files (sequence alignments), BED files (genomic regions of interest, for viewers), BIM (SNP information) and FAM (phenotype data, such as individuals and pedigree), are listed in folder /mnt/projects\_tn01/Cartagene/analyses/QC, with prefix merge\_5\_\* (for 5 genotypes concatenated).

#### 1.3.2.2 Step 2: Preprocessing for Removing Correlated Features

**Remark**: the prefix eur\_only stands for Caucasian phenotypes.

##### 1.3.2.2.1 i) Trim missing SNPs

* **Inputs**:
  + **Phenotype Ids of white individuals**: eur\_only/cartagene\_self\_reported\_EUR.plink\_format.txt
  + **BIM/BED/FAM/hh** folder generated with --make-bed command: analyses/QC/merge\_5\_datasets. It seems that among the 5 genotype arrays concatenated, at least one genotype, namely gsa.17k.final.hg19.bim, has been mapped thanks to GRCh37 (hg19) reference instead of more recent hg38 (GRCh38, 2013) version.$$
* **Objective**: Keep white individuals and remove SNPs variants with genotyping rate . The **genotyping rate** measures the proportion of successfully genotyped markers and indicates how complete the genotype data is. Removing SNPs with low Genotyping Rate increases Missing Data imputation performance, and overall maintains higher statistical power.
* Remark: Also advised to exclude samples with low Genotyping Rate, below 0.98.
* **Bash command**:

plink2 \  
 --bfile /mnt/projects\_tn01/Cartagene/analyses/QC/merge\_5\_datasets \  
 --geno 0.05 \  
 --keep /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/cartagene\_self\_reported\_EUR.plink\_format.txt \  
 --make-bed \  
 --out /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/merge\_5\_datasets.eur\_only.geno095

##### 1.3.2.2.2 ii) Hardy-Weinberg Disequilibrium

* **Inputs**:
  + **BIM/BED/FAM/hh (Homozygous-Haplotype)** files generated from previous SNP missing removal, see [Section 1.3.2.2.1](#sec-missing-SNPs).
* **Objective**: Remove SNPs variants with **Hardy-Weinberg disequilibrium** . Indeed, if a SNP is in HWD, it might reflect hidden sub-population structure or poor-quality rather than true genetic associations.
* Remark: If a SNP is under strong natural selection, such as SNPs involved in the HLA genes, they are likely to deviate from HWE due to balancing selection. If the SNP is biologically important, don’t exclude it blindly!! Alternative: compute the SNP score.
* **Bash command**:

plink2 \  
 --bfile /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/merge\_5\_datasets.eur\_only.geno095 \  
 --hwe 1e-6 \  
 --make-bed \  
 --out /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/merge\_5\_datasets.eur\_only.geno095.hwe1e06

##### 1.3.2.2.3 iii) Linkage Disequilibrium

###### 1.3.2.2.3.1 Local LD analysis

* **Inputs**:
  + **BIM/BED/FAM/hh (Homozygous-Haplotype)** files /merge\_5\_datasets.eur\_only.geno095.hwe1e0 generated from previous HWD SNP trimming, see [Section 1.3.2.2.2](#sec-HWD).
* **Objective**: Remove SNPs variants associated with strong **linkage disequilibrium**. In an ideal population under random mating, allele combinations should be independent (). However, due to factors such as genetic drift, or physical proximity, certain alleles tend to be inherited together more often than expected. In GWAS, LD pruning avoids **overfitting** (SNPs in high LD carry redundant information), avoiding spurious inflation of GWAS association signals.
* Remark: Report the Number of SNPs Before and After Pruning, usually more stringent R2 is considered.
* **Tool**: plink --indep-pairwise 50 5 0.5 will discard SNPs with a correlation coefficient above (given that a score of 1 indicates a perfect correlation), on a *rolling window* of SNPs (focus on local zones) and step by .
* **Bash command**:

## identify SNPs with low LD  
plink2 \  
 --bfile /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/merge\_5\_datasets.eur\_only.geno095.hwe1e06 \  
 --indep-pairwise 50 5 0.5 \  
 --out /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/merge\_5\_datasets.eur\_only.geno095.hwe1e06.pruned\_snps  
   
## save pruned SNPs associated with low LD  
plink2 \  
 --bfile /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/merge\_5\_datasets.eur\_only.geno095.hwe1e06 \  
 --extract /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/merge\_5\_datasets.eur\_only.geno095.hwe1e06.pruned\_snps.prune.in \  
 --make-bed \  
 --out /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/merge\_5\_datasets.eur\_only.geno095.hwe1e06.pruned\_snps\_bfile

###### 1.3.2.2.3.2 Global LD analysis from prior expert knowledge

* **Inputs**:
  + **BIM/BED/FAM/hh (Homozygous-Haplotype)** files merge\_5\_datasets.eur\_only.geno095.hwe1e06.pruned\_snps\_bfile generated from previous local LD trimming, see [Section 1.3.2.2.3.1](#sec-local-LD).
* **Objective**: Remove SNPs variants associated with strong **linkage disequilibrium** using prior expert knowledge, see [here](http://dougspeed.com/wp-content/uploads/highld.txt) for details.[[2]](#footnote-55)
* **Bash command**:

plink2 \  
 --bfile /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/merge\_5\_datasets.eur\_only.geno095.hwe1e06.pruned\_snps\_bfile \  
 --exclude range /mnt/projects\_tn01/Cartagene/analyses/QC/high\_ld\_regions.plink\_format.txt \  
 --make-bed \  
 --out /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/merge\_5\_datasets.eur\_only.geno095.hwe1e06.pruned\_snps.high\_LD\_excluded

##### 1.3.2.2.4 iv) Exclude affiliated individuals based on high IBD scores

* **Inputs**:
  + **BIM/BED/FAM/hh (Homozygous-Haplotype)** files merge\_5\_datasets.eur\_only.geno095.hwe1e06.pruned\_snps\_bfile generated from previous local LD trimming, see [Section 1.3.2.2.3.1](#sec-local-LD).
* **Objective**: Exclude related individuals computing **identity by descent** (IBD), a genetic metric of the relatedness between two individuals, and exclude one individual by pairs of individuals with a score above . Indeed, strongly associated patients increase bias in GWAS, raising a stronger score than expected in the general population, and prevents quality controls to detect duplicates or sample mix-ups.
* **Details**: A two-step, more stringent IBD filtering strategy has been chosen, eliminating *first* the most problematic individuals, and the *second* ensuring the remaining related individuals are properly filtered (for each correlated pair, prune randomly one of them). The Bash instructions are reported [here](./mnt/projects_tn01/Cartagene/analyses/QC/eur_only/related_ids/Readme.txt).
* Remarks: Choice of a heuristic threshold of n=68 affiliated patients removal to be further discussed. Current score of PI\_HAT score of 0.2 is surprising, as common thresholds are either 0.25 for discarding grandparent-grandchild, or 0.125 or lower for only keeping the most distant relatives. Starting from PLINK 2.0, the recommended approach is now utilizing the --king-cutoff command, over the older --rel-cutoff and --genome --min + aggregates all subsequent steps simultaneously, especially in heterogeneous populations[[3]](#footnote-59).
* **Bash command**:

## Calculate IBD, select individuals with IBD above 0.2 for further pruning  
  
plink2 \  
 --bfile /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/merge\_5\_datasets.eur\_only.geno095.hwe1e06.pruned\_snps\_bfile \  
 --genome \  
 --memory 12006 \  
 --min 0.2 \  
 --out /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/merge\_5\_datasets.eur\_only.geno095.hwe1e06.pruned\_snps.IBD  
  
  
## Remove related individuals, in a two-stage process  
  
plink2 \  
 --bfile /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/merge\_5\_datasets.eur\_only.geno095.hwe1e06.pruned\_snps\_bfile \  
 --make-bed \  
 --out /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/merge\_5\_datasets.eur\_only.geno095.hwe1e06.pruned\_snps\_bfile.rem\_rel\_ind\_round\_1\_of\_2 \  
 --remove /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/related\_ids/IBD.genome.iids.merged.sorted.count.reverse.ids\_related\_to\_2\_individuals.ids\_only.FID\_IID\_format  
  
  
plink2 \  
 --bfile /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/merge\_5\_datasets.eur\_only.geno095.hwe1e06.pruned\_snps\_bfile.rem\_rel\_ind\_round\_1\_of\_2 \  
 --make-bed \  
 --out /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/merge\_5\_datasets.eur\_only.geno095.hwe1e06.pruned\_snps\_bfile.rem\_rel\_ind\_round\_2\_of\_2 \  
 --remove /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/related\_ids/IBD.genome2.related\_individuals\_to\_remove.txt

#### 1.3.2.3 Step 3: PCA Computation and Population Genetic Evaluation

* **Inputs**:
  + **BIM/BED/FAM/hh (Homozygous-Haplotype)** files merge\_5\_datasets.eur\_only.geno095.hwe1e06.pruned\_snps.high\_LD\_excluded resulting from the pre-processing operations reported in [Section 1.3.2.2](#sec-preprocessing-redundancy) (LD, HWD and missing SNPs trimming)
  + **phenotype IDs** listed in merge\_5\_datasets.eur\_only.geno095.hwe1e06.pruned\_snps\_bfile.rem\_rel\_ind\_round\_2\_of\_2, where all affiliated patients have been removed.
* **Objective**: Compute PCA and keep the 10 first principal components.
* **Bash command**:

plink2 \  
 --bfile /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/merge\_5\_datasets.eur\_only.geno095.hwe1e06.pruned\_snps.high\_LD\_excluded \   
 --keep merge\_5\_datasets.eur\_only.geno095.hwe1e06.pruned\_snps\_bfile.rem\_rel\_ind\_round\_2\_of\_2.fam \  
 --memory 12006 \  
 --out /mnt/projects\_tn01/Cartagene/analyses/QC/eur\_only/merge\_5\_datasets.eur\_only.geno095.hwe1e06.pruned\_snps.high\_LD\_excluded.unrelated\_ind.PCA \  
 --pca 10

* **Remarks**:
  + Remark 1: use visualizations, such as scree plots, elbow point or/and Tracy-Widom Test to select the final number of PCs, instead of hard threshold, and on the other hand, scatter plots to identify latent structures. See details here and there.
  + **Remark 2**: The [popgen *‘Calculating Basic Population Genetic Statistics from SNP Data’*](https://popgen.nescent.org/StartSNP.html#genetic-diversity-observed-and-expected-heterozygosity) vignette details how to generate statistics quantifying patterns of genetic diversity, such as LD and Hardy-Weinberg, in a given population. The [popgen *‘Calculating genetic differentiation and clustering methods from SNP data’* vignette](https://popgen.nescent.org/DifferentiationSNP.html) vignette describes both unsupervised, PCA-like and discriminatory approaches to identify latent genetic sub-populations.  
    Finally, the [popgen *‘Individual Based Genetic Distance for SNP Data’*](https://popgen.nescent.org/2015-05-18-Dist-SNP.html) vignette reports different individual genetic distances, enabling to spot and trim individual outliers with a significant distinct genetic distribution. See also paper [‘A benchmark study on current GWAS models in admixed populations’](https://doi.org/10.1093/bib/bbad437), from Yang et al. ([2024](#ref-yang2024bib)) for an unbiased and up-to-date comparison of statistical approaches to address within-patient heterogeneity.
  + **Remark 3**: To evaluate within-population diversity, geneticists typically report the level of *heterozygosity* as a general metric, see [vcfR‘Genetic differentiation’ Vignette](https://knausb.github.io/vcfR_documentation/genetic_differentiation.html).

### 1.3.3 Run GWAS analyses

#### 1.3.3.1 Step 1: Phenotype feature extraction

Original phenotype annotations are available in folder /mnt/projects\_tn01/Cartagene/analyses/phenotypes. Variables of interest are reported [here](./data/phenotypes/cartagene_response_variables.xlsx):

cartagene\_response\_variables <- readxl::read\_excel("data/phenotypes/cartagene\_response\_variables.xlsx")

##### 1.3.3.1.1 i) Merge genotype IDs, phenotypes and PCAs values

General phenotype features are provided in

cartagene\_phenodata <- haven::read\_sas("data/phenotypes/cart\_mars2025.sas7bdat")  
flextable(head(cartagene\_phenodata)) |>   
 bold(bold = TRUE, part = "header")

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| | **PROJECT\_CODE** | **ASA** | **STATINE** | **Age (decimal)** | **CKD\_EPI** | **Body Mass Index (BMI)** | **T-Score** | **BPS** | **SRAAI** | **GENDER01** | **ETHNICITY6M** | **TABACCOURANT\_M** | **ROHSEMAINE** | **DIABETEGLOBAL** | **DLPGLOBAL** | **HTAGLOBAL** | **ASTHME\_MPOCAUTO\_RX** | **CIRRHOSEAUTO\_M** | **OSTEOAUTO\_M** | **OSTEORX** | **NEOALL\_AUTO\_M** | **NEOCHIMIO\_AUTO\_M** | **NEORADIO\_AUTO\_M** | **OSTEOSECONDAIRE** | **CVMACE\_POST** | **CVMACE\_PROCEDURE\_POST** | **CVALL\_POST** | **CVALL\_POSTDATE** | **CVMACE\_POSTDATE** | **CVMACE\_PROCEDURE\_POSTDATE** | **DECES** | **DATEDECES1216** | **FXALL\_POST** | **FXALL\_PRE1** | **FXALL\_PRE5** | **FXMOF\_POST** | **FXALL\_POSTDATE** | **FXMOF\_POSTDATE** | **AVCGLOBAL\_PRE\_M** | **ATCDFAMCV** | **HRT** | **IRC** | **PWV1** | **MCASGLOBAL\_PRE\_M** | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 78,295,800,001 | 1 | 1 | 62.08 | 56.60608 |  | 1.7 | 0 | 0 | 0 | Blanc | 0 | 0.25 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2,228 | 2,228 | 2,228 | 0 |  | 0 | 0 | 0 | 0 | 2,228 | 2,228 | 0 | 0 | 0 | 1 | 10.293434 | 1 | | 78,295,800,002 | 0 | 0 | 43.84 | 77.58291 | 35.4 | 3.2 | 0 | 0 | 1 | Latin | 0 | 0.00 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 236 | 2,059 | 236 | 0 |  | 0 | 0 | 0 | 0 | 2,059 | 2,059 | 0 | 0 | 0 | 0 | 6.110750 | 0 | | 78,295,800,003 | 0 | 0 | 59.27 | 86.25898 | 22.7 | -2.3 | 0 | 0 | 1 | Blanc | 0 | 8.00 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2,205 | 2,205 | 2,205 | 0 |  | 0 | 0 | 0 | 0 | 2,205 | 2,205 | 0 | 0 | 0 | 0 | 8.271314 | 0 | | 78,295,800,004 | 1 | 0 | 58.52 | 85.17528 | 23.5 | -1.8 | 1 | 0 | 1 | Blanc | 0 | 0.00 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2,184 | 2,184 | 2,184 | 0 |  | 0 | 0 | 0 | 0 | 2,184 | 2,184 | 0 | 0 | 1 | 0 | 8.810156 | 0 | | 78,295,800,005 | 0 | 0 | 54.92 | 105.51846 | 30.3 | 1.1 | 0 | 0 | 0 | Blanc | 0 | 4.50 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2,103 | 2,103 | 2,103 | 0 |  | 0 | 0 | 0 | 0 | 2,103 | 2,103 | 0 | 1 | 0 | 0 | 8.310378 | 0 | | 78,295,800,006 | 0 | 0 | 45.06 | 113.30294 | 28.2 | -1.2 | 0 | 0 | 1 | Arabe | 0 | 0.00 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2,000 | 2,000 | 2,000 | 0 |  | 0 | 0 | 0 | 0 | 2,000 | 2,000 | 0 | 0 | 0 | 0 |  | 0 |   Table 1.1: Read SAS table describing CarTaGene phenotypes. |

Number of patients overall is 19990, and number of phenotype variables is 44.

We then need to map each individual patient ID (IID) with its corresponding genotype array [Table 1.2](#tbl-cartage-genotypes-IDs), as done in [Listing 1.1](#lst-join-phenotype-genotype).

cartagene\_genotypes\_ID <- readr::read\_csv2("./data/phenotypes/cartagene\_genotype\_IDs.csv",  
 show\_col\_types = FALSE,   
 col\_types = c("d", "c","c")) |>   
 dplyr::rename(PROJECT\_CODE = "project\_code", geno\_id = "file\_111", batch="batch")  
   
  
flextable(head(cartagene\_genotypes\_ID)) |>   
 bold(bold = TRUE, part = "header")

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| | **PROJECT\_CODE** | **geno\_id** | **batch** | | --- | --- | --- | | 78,295,800,001 | 11,118,538 | gsa.17k | | 78,295,800,002 | 11,135,721 | gsa.17k | | 78,295,800,003 | 11,129,497 | gsa.archi.withIBS | | 78,295,800,004 | 11,107,817 | gsa.17k | | 78,295,800,005 | 11,131,634 | gsa.archi.withIBS | | 78,295,800,006 | 11,126,455 | gsa.17k |   Table 1.2: Read SAS table describing CarTaGene phenotypes. |

|  |
| --- |
| Listing 1.1: Inner join between phenotypes IDs and genotypes, while constraing the remaining individuals to belong to white ethnicity.  cartagene\_phenodata <- cartagene\_phenodata |>   dplyr::inner\_join(cartagene\_genotypes\_ID, by="PROJECT\_CODE") |>   dplyr::filter(ETHNICITY6M=="Blanc") |

The resulting phenotype table, after joining with genotypes IDs and restraining to Eurasian phenotypes, stores 17147 individuals.

PCA vectors computed in [Section 1.3.2.3](#sec-PCA) are subsequently merged with phenotype data in [Listing 1.2](#lst-join-phenotype-PCA).

|  |
| --- |
| Listing 1.2: Inner join between phenotypes IDs and first 10 PCA eigen vectors.  PCs <- readr::read\_delim("data/PCAs/PCA\_eigenvec",  col\_names = c("FID", "geno\_id", "PC1", "PC2", "PC3", "PC4", "PC5", "PC6", "PC7", "PC8", "PC9", "PC10"),   delim = " ",  show\_col\_types = FALSE) cartagene\_phenodata <- PCs |>   dplyr::inner\_join(cartagene\_phenodata, by="geno\_id") |>   dplyr::rename(IID = geno\_id) |

* Remark: Avoid using French CSV settings, switch to universal convention, where delimiter is a comma: ,

##### 1.3.3.1.2 ii) Bone damage: osteoporosis and fractures

* The response variables OSTEOPONIA and OSTEOPOROSIS have been computed following these rules [Listing 1.3](#lst-OSTEOPONIA-computation), with the resulting contingency tables reported in [Table 1.3](#tbl-OSTEOPONIA-computation).
  1. Exclude patients with osteosecondaire==1.
  2. OSTEOPONIA is case: 1 if DMOTSCORE\_mod <= -1.5 and control elsewhere[[4]](#footnote-78).
  3. OSTEOPOROSIS is case: 1 if DMOTSCORE\_mod < -2.5 and control elsewhere. DMOTSCORE\_mod has been likely Gaussian standardised + lacks of evidence supporting clear multi-modal distributions. Is there a reporting of the pre-processing operations?.

|  |
| --- |
| Listing 1.3: Generate scores of interest.  cartagene\_phenodata <- cartagene\_phenodata |>   dplyr::mutate(DMOTSCORE\_mod = dplyr::if\_else(OSTEOSECONDAIRE ==1, NA, DMOTSCORE),   OSTEOPONIA = dplyr::case\_when(DMOTSCORE\_mod <= -1.5 ~ 1,   DMOTSCORE\_mod > -1.5 ~ 0,   .default = NA),  OSTEOPOROSIS = dplyr::case\_when(DMOTSCORE\_mod < -2.5 ~ 1,   DMOTSCORE\_mod >= -2.5 ~ 0,   .default = NA))     flextable::proc\_freq(cartagene\_phenodata,   "OSTEOPONIA", "OSTEOPOROSIS") |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| | OSTEOPONIA |  | OSTEOPOROSIS | | | | | --- | --- | --- | --- | --- | --- | | 0 | 1 | Missing | Total | | 0 | Count | 12,463 (74.8%) |  |  | 12,463 (74.8%) | | Mar. pct (1) | 94.3% ; 100.0% |  |  |  | | 1 | Count | 753 (4.5%) | 48 (0.3%) |  | 801 (4.8%) | | Mar. pct | 5.7% ; 94.0% | 100.0% ; 6.0% |  |  | | Missing | Count |  |  | 3,393 (20.4%) | 3,393 (20.4%) | | Mar. pct |  |  | 100.0% ; 100.0% |  | | Total | Count | 13,216 (79.3%) | 48 (0.3%) | 3,393 (20.4%) | 16,657 (100.0%) | | (1) Columns and rows percentages | | | | | |   Table 1.3: OSTEOPOROSIS is considered as more severe than OSTEOPONIA, hence the striclty lower number of individuals affected by the disease. |

* **Fracture events**:
  1. We merge FXALL\_PRE5 (any fracture occurring within the five years prior to recruitment) with FXALL\_POST (all fractures occurring during follow-up) into variable FXALL, being positive if an event fracture occured prior or posterior. Which variable between FXALL\_PRE5 and FXALL\_PRE1 should be considered? Variables differ by 471 % overall!!
  2. FXMOF\_POST (only osteoporotic fractures during follow-up)[[5]](#footnote-81) In contrast with labelling, osteoporotic events, as stored in FXMOF\_POST, contain both pre-, and post- fracture events, as shown by Figure 1.2.

cartagene\_phenodata <- cartagene\_phenodata |>   
 mutate(FXALL = dplyr::if\_else(FXALL\_POST == 1L | FXALL\_PRE1 ==1L, 1L, 0L))  
  
flextable::proc\_freq(cartagene\_phenodata,   
 "FXALL", "FXMOF\_POST")  
flextable::proc\_freq(cartagene\_phenodata,   
 "FXALL\_POST", "FXALL\_PRE1")

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | | FXALL |  | FXMOF\_POST | | | | --- | --- | --- | --- | --- | | 0 | 1 | Total | | 0 | Count | 15,824 (95.0%) |  | 15,824 (95.0%) | | Mar. pct (1) | 96.8% ; 100.0% |  |  | | 1 | Count | 529 (3.2%) | 304 (1.8%) | 833 (5.0%) | | Mar. pct | 3.2% ; 63.5% | 100.0% ; 36.5% |  | | Total | Count | 16,353 (98.2%) | 304 (1.8%) | 16,657 (100.0%) | | (1) Columns and rows percentages | | | | |   (a) All fractures vs osteporotic fractures | |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | | FXALL\_POST |  | FXALL\_PRE1 | | | | --- | --- | --- | --- | --- | | 0 | 1 | Total | | 0 | Count | 15,824 (95.0%) | 92 (0.6%) | 15,916 (95.6%) | | Mar. pct (1) | 95.6% ; 99.4% | 82.9% ; 0.6% |  | | 1 | Count | 722 (4.3%) | 19 (0.1%) | 741 (4.4%) | | Mar. pct | 4.4% ; 97.4% | 17.1% ; 2.6% |  | | Total | Count | 16,546 (99.3%) | 111 (0.7%) | 16,657 (100.0%) | | (1) Columns and rows percentages | | | | |   (b) Pre vs post-fractures. | |

Table 1.4: Description of fracture events.

|  |  |  |  |
| --- | --- | --- | --- |
| |  | | --- | | (a) Generated by VennDiagram | | |  | | --- | | (b) Generated by ggVennDiagram | |

Figure 1.2: Venn Diagrams of fracture events, before and after diagnosis control.

##### 1.3.3.1.3 iii) Cardiovascular diseases

* The variables CVALL\_POST and CVMACE\_POST correspond to the occurrence of any cardiovascular event and **major adverse cardiovascular events** (MACE), including myocardial infarction, stroke, or cardiovascular death events, is the cognate subset of the most life-threatening heart conditions. An extended follow-up dataset is available for CVALL\_POST; however, it is not currently accessible.
* The variable MCASGLOBAL\_PRE\_M captures ischemic heart disease events, at baseline.
* The variable AVCGLOBAL\_PRE\_M represents the history of stroke events.

Contingency tables of both pre-, and post-, cardiovascular events are reported in [Table 1.5](#tbl-cardiovascular-diseases).

flextable::proc\_freq(cartagene\_phenodata,   
 "CVALL\_POST", "CVMACE\_POST")  
flextable::proc\_freq(cartagene\_phenodata,   
 "MCASGLOBAL\_PRE\_M", "AVCGLOBAL\_PRE\_M")

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | | CVALL\_POST |  | CVMACE\_POST | | | | --- | --- | --- | --- | --- | | 0 | 1 | Total | | 0 | Count | 16,215 (97.3%) |  | 16,215 (97.3%) | | Mar. pct (1) | 98.7% ; 100.0% |  |  | | 1 | Count | 221 (1.3%) | 221 (1.3%) | 442 (2.7%) | | Mar. pct | 1.3% ; 50.0% | 100.0% ; 50.0% |  | | Total | Count | 16,436 (98.7%) | 221 (1.3%) | 16,657 (100.0%) | | (1) Columns and rows percentages | | | | |   (a) CVALL\_POST vs CVMACE\_POST | |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | | MCASGLOBAL\_PRE\_M |  | AVCGLOBAL\_PRE\_M | | | | --- | --- | --- | --- | --- | | 0 | 1 | Total | | 0 | Count | 15,520 (93.2%) | 203 (1.2%) | 15,723 (94.4%) | | Mar. pct (1) | 94.8% ; 98.7% | 72.0% ; 1.3% |  | | 1 | Count | 855 (5.1%) | 79 (0.5%) | 934 (5.6%) | | Mar. pct | 5.2% ; 91.5% | 28.0% ; 8.5% |  | | Total | Count | 16,375 (98.3%) | 282 (1.7%) | 16,657 (100.0%) | | (1) Columns and rows percentages | | | | |   (b) MCASGLOBAL\_PRE\_M vs AVCGLOBAL\_PRE\_M | |

Table 1.5: Cardiovascular disease events

Finally, we save the updated phenotype dataset as a Tab-separated file, using readr::write\_tsv, see **?@tbl-cartage-pheno-features** for details:

## define Boolean categorical variables  
categorical\_variables <- cartagene\_response\_variables$Features[  
 cartagene\_response\_variables$Type == "Boolean"]  
  
cartagene\_phenodata <- cartagene\_phenodata |>  
 dplyr::mutate(dplyr::across(dplyr::all\_of(categorical\_variables),  
 \(x) dplyr::if\_else(x == 0L, 1L, 2L, missing = NA)))  
readr::write\_tsv(cartagene\_phenodata,   
 file="data/phenotypes/merge\_phenos\_PCs.txt")  
  
  
# Report overall summaries  
cartagene\_phenodata\_summary <- cartagene\_phenodata |>   
 dplyr::select(dplyr::all\_of(c("IID", "GENDER01", categorical\_variables))) |>   
 tidyr::pivot\_longer(-dplyr::all\_of(c("IID", "GENDER01")),   
 names\_to = "Features", values\_to = "Value") |>   
 dplyr::mutate(Value = tidyr::replace\_na(as.character(Value), "osteosecondaire")) |>   
 dplyr::count(Features, GENDER01, Value) |>   
 dplyr::rename(Counts=n) |>   
 dplyr::inner\_join(cartagene\_response\_variables, by = "Features")  
  
  
cartagene\_phenodata\_summary |>   
 dplyr::filter(Disease == "Bone Damage") |>   
 dplyr::select(Features, Description, GENDER01, Value, Counts)  
cartagene\_phenodata\_summary |>   
 dplyr::filter(Disease == "Cardiovascular events") |>   
 dplyr::select(Features, Description, GENDER01, Value, Counts)

|  |  |
| --- | --- |
| ## # A tibble: 20 × 5 ## Features Description GENDER01 Value Counts ## <chr> <chr> <dbl> <chr> <int> ## 1 FXALL All fracture events 0 1 7640 ## 2 FXALL All fracture events 0 2 324 ## 3 FXALL All fracture events 1 1 8184 ## 4 FXALL All fracture events 1 2 509 ## 5 FXMOF\_POST Restriction to osteoporisis events 0 1 7857 ## 6 FXMOF\_POST Restriction to osteoporisis events 0 2 107 ## # ℹ 14 more rows | ## # A tibble: 16 × 5 ## Features Description GENDER01 Value Counts ## <chr> <chr> <dbl> <chr> <int> ## 1 AVCGLOBAL\_PRE\_M Pre General Critical Cardio event 0 1 7805 ## 2 AVCGLOBAL\_PRE\_M Pre General Critical Cardio event 0 2 159 ## 3 AVCGLOBAL\_PRE\_M Pre General Critical Cardio event 1 1 8570 ## 4 AVCGLOBAL\_PRE\_M Pre General Critical Cardio event 1 2 123 ## 5 CVALL\_POST Post General Cardio event 0 1 7636 ## 6 CVALL\_POST Post General Cardio event 0 2 328 ## # ℹ 10 more rows |

Table 1.6: Features of interest to regress on SNPs in CarTaGene. Besides, recent versions of plink2 impose explicit encoding of categorical variables, encoding 1 as ‘controls’, and 2 as ‘cases’.

### 1.3.4 Step 2: Variants extraction

All the curated VCF files have been downloaded and processed by [Email Cartagene](mailto:access@cartagene.qc.ca). Preprocessing details are reported [here](https://cartagene.qc.ca/files/documents/other/Info_GeneticData3juillet2023.pdf)[[6]](#footnote-104)

#### 1.3.4.1 a) HDAC-family

##### 1.3.4.1.1 i) Retrieve HDAC positions

* **Objective**: extract the annotated VCF, for the 6 HDAC genes identified of interest, on chromosome 7 (see [Section 1.3.4.1](#sec-HDAC9) for Bash commands).
* **Methods**:

1. We retrieve positions of the HDAC genes (HDAC 4, 5, 6, 7, 9 and 10) on the latest Hg38 Genome Build with script [Listing 1.4](#lst-hdac-positions). HDAC positions are reported in [Table 1.7](#tbl-hdac-positions)[[7]](#footnote-105)
2. When several start and end positions were reported for the same gene, we consider the overall min starting position, and the max end position, respectively.

txdb <- GenomicFeatures::makeTxDbFromUCSC(genome = "hg38", tablename = "refGene")  
AnnotationDbi::saveDb(txdb, "./data/genome\_builds/human\_gencode\_v42.sqlite")

|  |
| --- |
| Listing 1.4: Use AnnotationDBI to fetch and retrieve automatically start and end positions of the HDAC family.  hdac\_genes <- c("HDAC4", "HDAC5", "HDAC6", "HDAC7", "HDAC9", "HDAC10")  ## Convert HGNC symbols to Entrez IDs with org.Hs.eg.db ## 1-1 mapping, great!! hdac\_entrez\_ids <- AnnotationDbi::mapIds(org.Hs.eg.db, keys = hdac\_genes,  column = "ENTREZID", keytype = "SYMBOL", multiVals = "first") hdac\_entrez\_ids <- tibble::tibble(  HGNC\_SYMBOL = names(hdac\_entrez\_ids),  GENEID = hdac\_entrez\_ids)  ## retrieve start and end positions of hdac chromosoms txdb <- AnnotationDbi::loadDb("data/genome\_builds/human\_gencode\_v42.sqlite") hdac\_coords <- AnnotationDbi::select(txdb,  keys = hdac\_entrez\_ids$GENEID,  columns = c("TXCHROM", "TXSTART", "TXEND"),  keytype = "GENEID")  ## take the min and max positions for all chromosomes hdac\_coords <- hdac\_coords |>  dplyr::group\_by(GENEID, TXCHROM) |>  dplyr::summarise(TXSTART=min(TXSTART), TXEND=max(TXEND)) |>  dplyr::ungroup() |>  dplyr::inner\_join(hdac\_entrez\_ids, by = "GENEID") |>  dplyr::arrange(GENEID) |>  relocate(HGNC\_SYMBOL) readr::write\_csv(hdac\_coords,  "data/gene\_positions/hdac\_coords.csv")  ## hdac\_coords <- readr::read\_csv("data/gene\_positions/hdac\_coords.csv",  ## show\_col\_types = FALSE) flextable(hdac\_coords) |>   bold(bold = TRUE, part = "header") |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| | **HGNC\_SYMBOL** | **GENEID** | **TXCHROM** | **TXSTART** | **TXEND** | | --- | --- | --- | --- | --- | | HDAC6 | 10013 | chrX | 48,801,398 | 48,824,982 | | HDAC5 | 10014 | chr17 | 44,076,753 | 44,123,641 | | HDAC7 | 51564 | chr12 | 47,782,724 | 47,819,903 | | HDAC10 | 83933 | chr22 | 50,245,184 | 50,251,265 | | HDAC9 | 9734 | chr7 | 18,086,825 | 19,002,414 | | HDAC4 | 9759 | chr2 | 239,048,168 | 239,401,649 |   Table 1.7: HDAC Gene Positions |

* Remark: Instead of fetching start and end locations automatically, add an INFO/GENE field directly within the VCF files.

##### 1.3.4.1.2 ii) Extract SNPs corresponding to provided HDAC gene

* **Inputs**:
  + Cleaned **VCF** file of the chromosome 7 (where HDAC-9 is present)[[8]](#footnote-110).
* **Outputs**:
  + chr7.HDAC9.vcf: VCF file, restrained to HDAC9 region with affiliated SNPs.
* **Objective**: Extract annotated SNPs within HDAC9 boundaries, as defined by the hg38 reference genome. **Details**:
  + Discard variants with **Minor Allele Frequency (MAF)** (in other words, the SNP must be present in at least of the samples).
  + SNPs were extracted using bcftools view, with shell script [extract\_variants.sh](./shell/extract_variants.sh)[[9]](#footnote-112):

nohup ./shell/extract\_variants.sh > shell/extract\_variants.log 2>&1 &

* **Remark:** --maf guarantees removal of highly recessive SNPs, present in less than of cases with respect to the dominant form: infrequent SNPs are indeed associated with lower statistical power.

### 1.3.5 Step 3: GWAS Analyses

#### 1.3.5.1 i) GLM and GWAS

* **Objective**: Use of PLINK2 + glm with the first 10 principal components as covariates, see [Listing 1.5](#lst-glm-bash), using logistic regression for categorical variables, and lm for continuous outcomes.
* **Input**:
  + VCF file on the region/gene of interest
  + Phenotypes, with individual patient IDs.
  + Explanatory variable to predict, provided with --pheno-name <response\_variable>.

|  |
| --- |
| Listing 1.5  **Template GLM instruction for GWAS studies:**  plink2 \  --double-id \ #  --pheno ./phenotypes/merge\_phenos\_PCs.txt \ #  --pheno-name OSTEOPONIA \ #  --vcf /mnt/projects\_tn01/Cartagene/analyses/variants\_extraction/chr7.merged.clean.noMono.extracted\_variants.HDAC9.vcf dosage=HDS #  --glm hide-covar \ #  --covar /mnt/projects\_tn01/Cartagene/analyses/phenotypes/merge\_phenos\_PCs.txt \ #  --covar-name PC1,PC2,PC3,PC4,PC5,PC6,PC7,PC8,PC9,PC10 \ #  --out HDAC9\_OSTEOPONIA #  Line 3  The --double-id option ensures independent GWAS analyses per individual, aka the *single-sample mode*, stating explicitly that the VCF file follows a *family-based format* (FID/IID pairs).  Lines 4-6  **GWAS inputs**: We need the *phenotype information* (provided with --pheno and --keep commands), the *variable to predict* (provided with --pheno-name) and the *VCF file* (command --vcf), here using the SNPs annotations for the HDAC9 gene[[10]](#footnote-117).  Lines 7-9  GWAS model options: --glm is the general linear model, which uses by default a *logistic regression* for categorical variables, and a *standard linear Gaussian model*, equivalent to lm for continuous variables[[11]](#footnote-118). --covar is the covariate file, with relevant explanatory variables to integrate reported with --covar-name (see @ for details). PCAs are used to describe the population structure in an unsupervised manner; and avoid and detect latent subgroups.  Line 10  The output GWAS folder, with --out command. Stored for now in /mnt/projects\_tn01/Cartagene/analyses/association. |

In practice, run the [plink2\_gwas.sh](./shell/plink2_gwas.sh) script in the background (possibility to customise gene name and feature variables to regress on):

## Run the GWAS pipeline in the background,  
## persist after logout, and log everything.  
  
nohup ./shell/plink2\_gwas.sh > shell/plink2\_gwas.log 2>&1 &

* **Remark 1**: you may come up with Error: Cannot proceed with --glm regression on phenotype 'TACAIX', since variance inflation factor for covariate 'PC2' is too high. In this case, you may try removing completely covariates (with --glm allow-no-covars), or/and increase variance inflation threshold (--vif number\_vif option)[[12]](#footnote-121)
* Remark 2: All the pre-processing operations detailed in Section 1.3.2.2 are only used for the computation of the PCA components, but not subsequently used in the glm regression for trimming strongly correlated features.
* Remark 3: Add phenotype covariates, such as SEX or AGE in the regression framework, which can play a strong leverage on the impact of SNPs. GENESIS is a R package mixing environmental and genetic factors in a fixed linear model approach.

#### 1.3.5.2 ii) GWAS Visualisations

For each gene-phenotype pair, we generate cognate **Manhattan plots**[[13]](#footnote-125), and -values distributions, using *histograms*[[14]](#footnote-127), and **QQplots**[[15]](#footnote-129). Core function for generating these 3 GWAS visualisations is [generate\_gwas\_per\_phenotype](./R/gwas_plots.R). An example for gene HDAC-9 is provided in [Figure 1.3](#fig-GWAS-HDAC9).

## p-value adjustment  
## genename <- "HDAC-9"; pheno\_label <- "OSTEOPONIA"; filedate <- "2025-04-07"  
num\_variants <- readr::read\_csv("./tables/HDAC-9/HDAC-9\_AVCGLOBAL\_PRE\_M\_2025-04-07.AVCGLOBAL\_PRE\_M.glm.logistic.hybrid", show\_col\_types = FALSE) |> nrow()  
pval\_threshold <- 0.05/(num\_variants\*length(cartagene\_response\_variables$Features))  
## pval\_threshold <- 0.01/num\_variants  
  
gwas\_plots\_hdac <- lapply(cartagene\_response\_variables$Features,   
 function(feature\_label) generate\_gwas\_per\_phenotype (genename = "HDAC-9",  
 pheno\_label = feature\_label, filedate = "2025-04-07", pval\_threshold = pval\_threshold))   
  
gwas\_plots\_hdac <- gridExtra::marrangeGrob(gwas\_plots\_hdac, nrow=1, ncol=1)  
  
ggsave("figures/gwas\_HDAC9.pdf",  
 gwas\_plots\_hdac, dpi = 600,  
 width = 8, height = 12)

|  |
| --- |
| Figure 1.3 |

#### 1.3.5.3 iii) Multiple test correction

To compute the adjusted *significance threshold*, we applied a FWER-like approach, conventional in the GWAS field: {r} 0.05/(num\_variants\*length(cartagene\_response\_variables$Features)), in which you divide the pre-defined threshold (standard: 0.05) by the total number of pairwise tests carried out (number of variants times number of regression variables times). Final threshold is accordingly: 2.1188236^{-6}, as 2622 variants were extracted, and 9 phenotype variables were predicted.

The FWER approach is particularly conservative, in other words, only the most significant differences are detected.

* **Perspectives**:
  + [Recent Adjusted -value correction method](https://www.linkedin.com/posts/adrianolszewski_statistic-datascience-research-activity-7301605962820284417-ho3Z).
  + [Multiple Comparisons Using R](https://www.taylorfrancis.com/books/mono/10.1201/9781420010909/multiple-comparisons-using-frank-bretz-torsten-hothorn-peter-westfall), comprehensive resource of distinct R package strategies for correcting for multiplicity of -values, from Bretz, Hothorn, and Westfall ([2016](#ref-bretz2016)). Report also to [Note 1.1](#nte-MTC) for an overview of existing approaches.
  + [Multiple Testing: Methods Overview](https://yasenov.com/2022/10/multiple-hypotheses-testing-a-methods-overview/) reviews the most popular approaches for controlling the false-error rate, while providing practical R code snippets to implement them. The blog concludes on the prevalance of the *Benjamini-Yekuiteli approach*, offering the best compromise between statistical power and interpretability.

|  |
| --- |
| Note 1.1: Classes of Multiple Test correction approaches |
| **Multiple comparison procedures** (MCPs) can be classified into the following adjustment methods:   1. *Fixed sequential*: no adjustment as long as H0s are rejected. One non-rejection and it stops testing with all others non-rejected. 2. *Callback*: significance level (e.g. 0.05) is split between all hypotheses (can be further weighted). With rejection of the subsequent H0s, their fractions of significance level is accumulated and passed to the next comparison. 3. *Gatekeeper*: serial, parallel & combined trees, collecting hypotheses into families. Groups of testing can be interpreted as *gatekeepers*: only if any in a family is rejected that the significance is propagated to the next family. The approach is the most versatile, and can be flexibly implemented in R package [gMCPLite](https://merck.github.io/gMCPLite). See also [Graphical Approaches to Multiple Test Problems](https://baselbiometrics.github.io/home/docs/talks/20220329/2_Glimm_Bretz_Xi.pdf), for additional details. |

## 1.4 Perspectives

### 1.4.1 Post-hoc GWAS

* [Collect information on variants with LLM-like VarChat app](https://varchat.engenome.com/)

### 1.4.2 Explore non-coding and regulatory regions (from WES to WGS)

* Paper [Systematic differences in discovery of genetic effects on gene expression and complex traits](https://www.nature.com/articles/s41588-023-01529-1), from Mostafavi et al. ([2023](#ref-mostafavi2023ng)), suggest extending extraction of genomic regions of interest beyond coding sequences (report to [Section 1.3.4](#sec-variant-extraction) for current approach). Indeed, the article demonstrates that most of identified signals originate from nearby gene regulatory sites.
* [EPInformer: a scalable deep learning framework for gene expression prediction by integrating promoter-enhancer sequences with multimodal epigenomic data](https://www.biorxiv.org/content/10.1101/2024.08.01.606099v1), from Lin, Luo, and Pinello ([2024](#ref-lin2024)). Avalaible as a [GH package](https://github.com/pinellolab/EPInformer). Can infer and predict gene expression from promoter and enhancer sequences paired with epigenomic signals; identify the most significant enhancers; and identify *regulatory sequences* and *transcription factor binding motifs*.
* Automatically extract and annotate variants, based on their locations, using [VariantAnnotation::locateVariants()](https://www.bioconductor.org/packages/release/bioc/vignettes/VariantAnnotation/inst/doc/VariantAnnotation.html#locating-variants-in-and-around-genes) function. Variants annotations notably encompass coding, UTRs, introns and promoters.

### 1.4.3 Post-GWAS: Explore other types of genetic variations

**The most typical mutations included in GWAS studies are *Single Nucleotide Polymorphisms (SNPs)*, and occasionally small indels.** Beyond them, we could have considered *Structural Variants (SVs): large insertions, deletions, duplications, inversions, translocations*, *Copy Number Variations (CNVs)*: gains or losses of large DNA segments, Microsatellites and Epigenetic changes in regulatory regions.

* [Genome-wide association testing beyond SNPs](https://www.nature.com/articles/s41576-024-00778-y), from Harris et al. ([2025](#ref-harris2025nrg))
* [Benchmarking post-GWAS analysis tools in major depression: Challenges and implications](https://www.frontiersin.org/journals/genetics/articles/10.3389/fgene.2022.1006903/full), from Pérez-Granado, Piñero, and Furlong ([2022](#ref-perez-granado2022fg)).
* The [VariantAnnotation::predictCoding](https://www.bioconductor.org/packages/release/bioc/vignettes/VariantAnnotation/inst/doc/VariantAnnotation.html#amino-acid-coding-changes) computes amino acid coding changes for non-synonymous variants, while the [SIFT and PolyPhen](https://www.bioconductor.org/packages/release/bioc/vignettes/VariantAnnotation/inst/doc/VariantAnnotation.html#sift-and-polyphen-databases) web databases evaluate the impact of SNP substitution, based on the predicted level of changes of the 3D-protein reconfiguration.

### 1.4.4 Post-GWAS: Retrieve automatically variants of interest

This analysis is not formally a GWAS study, but rather a *gene-centric* study, pre-selecting genes based on prior knowledge and/or intuition.

Alternatively, we could consider a data-driven, and more agnostic approach, to refine the selection of gene candidates. Several approaches to that end have been implemented:

* Gnanaolivu et al. ([2025](#ref-gnanaolivu2025bb)) proposed in [A clinical knowledge graph-based framework to prioritize candidate genes for facilitating diagnosis of Mendelian diseases and rare genetic conditions](https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-025-06096-2), the phenotype prioritization and analysis for rare diseases, PPAR algorithm to rank genes based on *human phenotype ontology (HPO)* terms.[[16]](#footnote-158). In details, PPAR combines embeddings from the *human knowledge graph*, incorporating genes, HPO terms, and gene ontology annotations connections. For each input HPO term, a prioritized list of genes is returned based on their relevance and similarity to the HPO term.
* Bridges et al. ([2025](#ref-bridges2025bb)) developed in [Towards a standard benchmark for phenotype-driven variant and gene prioritisation algorithms: PhEval, Phenotypic inference Evaluation framework](https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-025-06105-4) a unified benchmarking platform to clean inputs for *phenotype-driven VGPAs*. Variant and Gene Prioritisation Algorithms integrate complex and multi-modal datasets, such as ontologies and gene-to-phenotype associations, to predict the most influential and promising gene targets controlling the evolution of rare diseases. Model is available as GitHub Repo [PhEval](https://github.com/monarch-initiative/pheval).

### 1.4.5 Post-GWAS: Incorporate familial pedigrees

An alternative to *candidate gene* association studies relies on *linkage analysis*, which uses familial pedigrees to map genetic variants underlying common human diseases. **This strategy requires both genetic data and detailed family pedigrees.**

* [Genome-wide association study of fat content and fatty acid composition of shea tree.](https://bmcgenomics.biomedcentral.com/articles/10.1186/s12864-025-11344-z), from Attikora et al. ([2025](#ref-attikora2025bg)). Under the hood, relies on the mrMLM R package to include both the *population structure matrix* and the *kinship matrix*. Provides a complete GWAS pipeline, including a number of meaningful illustrations.
* [BHCox: Bayesian heredity-constrained Cox proportional hazards models for detecting gene-environment interactions](https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-025-06077-5), from Sun et al. ([2025](#ref-sun2025bb)). Under the hood, relies on the brms R package.

### 1.4.6 Post-GWAS: Polygenic risk scores

* [Optimizing and benchmarking polygenic risk scores with GWAS summary statistics](https://doi.org/10.1186/s13059-024-03400-w), from Z. Zhao et al. ([2024](#ref-zhao2024gb)).
* [XPRS: A Tool for Interpretable and Explainable Polygenic Risk Score](10.1093/bioinformatics/btaf143), from Kim and Lee ([2025](#ref-kim2025b)), is a R package for inferring the *polygenic risk score* for assessing genetic susceptibility to diseases, providing additional interpretation and visualisation tools. PRSs are further split and classified into genes and single nucleotide polymorphism (SNP) contribution scores via *Shapley additive explanations* (SHAPs), visualised as **Manhattan plots**, **LocusZoom-like** plots and tables at the population and individual levels.
* [PNL: a software to build polygenic risk scores using a Super Learner approach based on PairNet, a Convolutional Neural Network](https://academic.oup.com/bioinformatics/advance-article/doi/10.1093/bioinformatics/btaf071/8015614), from Chen et al. ([2025](#ref-chen2025b)).

### 1.4.7 Post-GWAS: Orthogonal integration

#### 1.4.7.1 Epigenomic

* G. Zhao and Lai ([2025](#ref-zhao2025bib)) implements [SC-VAR: a computational tool for interpreting polygenic disease risks using single-cell epigenomic data](https://academic.oup.com/bib/article/26/2/bbaf123/8092303?searchresult=1&login=false), a novel computational tool available as a GitHub repo and Python application: [SC-VAR GitHub](https://github.com/gefeiZ/SC_VAR). SC-VAR uses single-cell *epigenomic* data to predict functional outcomes of the identified disease-associated *GWAS variants*, enhancing their interpretability. Under the hood, relies on [MAGMA linear modelling framework](https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1004219), developed by Leeuw et al. ([2015](#ref-leeuw2015pcb)).

#### 1.4.7.2 eQTL (quantitative expression)

* [FIVEx](https://github.com/statgen/fivex) is a web-based interface to explore CarTaGene to visualize and query eQTL (quantitative trait loci) data in various ways. **Would need a stringent maintenance!!**

## 1.5 Appendix

### 1.5.1 Scalability with Nextflow

Lots of intermediate files are not required for downstream analyses, hence it would be relevant to rely on an existing *Nextflow* or *Snakemake* DSL workflows:

* [nf-GWAS ‘Nextflow pipeline’](https://genepi.github.io/nf-gwas/), from Schönherr and Forer ([[2021] 2024](#ref-schonherr2024)), is actively maintained by Curie Bioinformatics Team, and includes the latest plink2 facilities. Complementary tools include:
* The Nextflow Seqera [multiqc](https://seqera.io/multiqc/) initiative collects within the same repository a variety of command-line tools, such as bcftools and plink2, with notably the [vcftools](https://github.com/MultiQC/MultiQC/tree/main/multiqc/modules/vcftools) and [multivcfanalyzer](https://github.com/MultiQC/MultiQC/tree/main/multiqc/modules/multivcfanalyzer) modules. Note however that these tools are optimised in Python.
* [PopGLen ‘Snakemake’ pipeline](https://academic.oup.com/bioinformatics/advance-article/doi/10.1093/bioinformatics/btaf105/8069456), from Nolen ([2025](#ref-nolen2025b)). Not for running GWAS analyses, but rather for evaluating the quality and impact of preprocessing and quality mapping at the population-level genome starting from FASTQ files.

### 1.5.2 FASTQ and FASTA Quality Controls

* The 7th chapter entitled [“Quality Check, Processing and Alignment of High-throughput Sequencing Reads”](https://compgenomr.github.io/book/processingReads.html) introduces the fundamentals, using R packages, to process FASTA reads, and evaluate their quality with a variety of metrics (-mer over-representation, sequence quality, percentage of duplicated reads, ..).

# 2. CartaGene Sample Size estimation

## 2.1 Setup for statistical power analyses

## data wrangling and visualisations  
library(dplyr)  
library(ggplot2)  
library(flextable)  
## Required for code linking  
library(downlit)  
library(xml2)  
  
## Report automated QCs for VCF files  
library(vcfR)  
  
# Sample size estimation  
library(genpwr)

## 2.2 VCF files Reporting with vcfR

### 2.2.1 Load VCF files

# vcf\_file <- "./data/variants/chr17.HDAC5.vcf"  
vcf\_file <- "./data/variants/chr7.HDAC9.vcf"  
vcf <- vcfR::read.vcfR(vcf\_file)

## Scanning file to determine attributes.  
## File attributes:  
## meta lines: 24  
## header\_line: 25  
## variant count: 2622  
## column count: 29346  
##   
Meta line 24 read in.  
## All meta lines processed.  
## gt matrix initialized.  
## Character matrix gt created.  
## Character matrix gt rows: 2622  
## Character matrix gt cols: 29346  
## skip: 0  
## nrows: 2622  
## row\_num: 0  
##   
Processed variant 1000  
Processed variant 2000  
Processed variant: 2622  
## All variants processed

chrom <- vcfR::create.chromR(name = "Supercontig", vcf = vcf) |>   
 # removing low quality variants, none of these metrics is reported  
 # vcfR::masker(min\_QUAL = 0, min\_DP = 350, max\_DP = 650,   
 # min\_MQ = 59.5, max\_MQ = 60.5) |>   
 vcfR::proc.chromR(win.size = 1e4)

## Names in vcf:

## chr7

## Initializing var.info slot.

## var.info slot initialized.

## Warning in vcfR::proc.chromR(vcfR::create.chromR(name = "Supercontig", vcf =  
## vcf), : seq slot is NULL.

## Warning in vcfR::proc.chromR(vcfR::create.chromR(name = "Supercontig", vcf =  
## vcf), : annotation slot has no rows.

## Warning in vcfR::proc.chromR(vcfR::create.chromR(name = "Supercontig", vcf =  
## vcf), : seq slot is NULL, chromosome representation not made (seq2rects).

## Warning in vcfR::proc.chromR(vcfR::create.chromR(name = "Supercontig", vcf =  
## vcf), : seq slot is NULL, chromosome representation not made (seq2rects,  
## chars=n).

## Population summary complete.

## elapsed time: 83.522

## window\_init complete.

## elapsed time: 0.002

## Warning in vcfR::proc.chromR(vcfR::create.chromR(name = "Supercontig", vcf =  
## vcf), : seq slot is NULL, windowize\_fasta not run.

## Warning in vcfR::proc.chromR(vcfR::create.chromR(name = "Supercontig", vcf =  
## vcf), : ann slot has zero rows.

## windowize\_variants complete.  
## elapsed time: 0.002

### 2.2.2 VCF files QC

vcfR::chromoqc(chrom, dp.alpha = 22,  
 xlim = c(min(chrom@var.info$POS), max(chrom@var.info$POS)))  
  
plot(chrom)

|  |
| --- |
| Figure 2.1: Quality control per chrosomsome, for a given variant |

|  |
| --- |
| Figure 2.2: Quality control per chrosomsome, for a given variant |

**?@fig-plot-qc-per-chrom** is a **chromo plot**:

* **Annotation data** on the lowest plot. Features in the annotation file (e.g., gene models, exons, etc.) are represented as dark red rectangles.
* Above is a **sequence track**, with nucleotides in green while uncalled nucleotides are in red.
* Last track summarizes the number of variants per window. Above that we see dot plots for quality, mapping quality and read depth.
* Generated with [Visualizing VCF data 1](https://knausb.github.io/vcfR_documentation/visualization_1.html)

### 2.2.3 MAF histogram distribution

Main [tidy vcfR](https://knausb.github.io/vcfR_documentation/tidy_vcfR.html) instructions.

vcf\_tidy <- vcfR::vcfR2tidy(vcf)  
vcfR::vcf\_field\_names(vcf, tag = "INFO")  
vcfR::vcf\_field\_names(vcf, tag = "FORMAT")

|  |  |
| --- | --- |
| ## # A tibble: 3 × 5 ## Tag ID Number Type Description  ## <chr> <chr> <chr> <chr> <chr>  ## 1 INFO IMPUTED 0 Flag Marker was imputed but NOT genotyped ## 2 INFO TYPED 0 Flag Marker was genotyped AND imputed  ## 3 INFO TYPED\_ONLY 0 Flag Marker was genotyped but NOT imputed | ## # A tibble: 4 × 5 ## Tag ID Number Type Description  ## <chr> <chr> <chr> <chr> <chr>  ## 1 FORMAT GT 1 String Genotype  ## 2 FORMAT DS 1 Float Estimated Alternate Allele Dosage : [P(0/1)+2\*P… ## 3 FORMAT HDS 2 Float Estimated Haploid Alternate Allele Dosage  ## 4 FORMAT GP 3 Float Estimated Posterior Probabilities for Genotypes… |

Table 2.1: VCF annotations and descriptions.

**Minor allele frequency (MAF)** for a given variant can be computed from **DS (Estimated Alternate Allele Dosage)**, **HDS (Haploid Dosage)**, or **GP** (Posterior Probabilities for Genotypes), as reported in [Table 2.2](#tbl-MAF-formula), and the corresponding MAF tables and abundancies are reported in [Table 2.3](#tbl-MAF-estimation).

|  |
| --- |
| Tip 2.1: Computation of the MAF score |
| |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | | Source | How to compute AF | Then MAF | | --- | --- | --- | | DS |  |  | | HDS |  |  | | GP |  |  |   Table 2.2: Infer MAF formulas | |

vcf\_variants\_counts <- vcf\_tidy$gt  
N\_indiv <- length(unique(vcf\_variants\_counts$Indiv))  
  
# Compute MAF from DS (Estimated Alternate Allele Dosage)  
vcf\_maf\_distributions <- vcf\_variants\_counts |>   
 dplyr::group\_by(POS) |>   
 dplyr::summarise(AF = sum(gt\_DS, na.rm = TRUE) / (2 \* n()), # alternate allele frequency  
 MAF = pmin(AF, 1 - AF), .groups = "drop") # minor allele frequency  
  
# alternative computation, similar outputs (by default, return the minor allele)  
vcf\_maf\_distributions\_alt <- vcfR::maf(vcf, element = 2) |>   
 tibble::as\_tibble(rownames = "SNP\_ID")  
  
flextable(head(vcf\_maf\_distributions\_alt)) |>   
 bold(part = "header")   
flextable(vcf\_maf\_distributions |>   
 select(POS, MAF) |>   
 arrange(-MAF) |>   
 head()) |>   
 flextable::bold(part = "header")

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | | **SNP\_ID** | **nAllele** | **NA** | **Count** | **Frequency** | | --- | --- | --- | --- | --- | | chr7:18087100:T:A | 58,674 | 0 | 1,024 | 0.01745236 | | chr7:18087117:T:A | 58,674 | 0 | 1,548 | 0.02638307 | | chr7:18087870:T:C | 58,674 | 0 | 867 | 0.01477656 | | chr7:18088625:A:G | 58,674 | 0 | 24,666 | 0.42039063 | | chr7:18089395:A:G | 58,674 | 0 | 626 | 0.01066912 | | chr7:18089923:TTTTG:T | 58,674 | 0 | 16,876 | 0.28762314 |   (a) Result from calling the vcfR::maf function on a vcfobject. The row.names report the ID of the mutation (chromosome:position:SNP), the first column the total number of alleles identified in the cohort, the third is the count and fourth the frequency (you have to divide by 2 the Counts with diploid organisms). | |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | | **POS** | **MAF** | | --- | --- | | 18,773,294 | 0.4989486 | | 18,769,554 | 0.4980236 | | 18,104,245 | 0.4958051 | | 18,092,439 | 0.4942534 | | 18,775,850 | 0.4941757 | | 18,770,620 | 0.4934447 |   (b) Header FORMAT and Description for the VCF file. | |

Table 2.3

ggplot(vcf\_maf\_distributions, aes(x = MAF)) +  
 geom\_histogram(binwidth = 0.02) +  
 geom\_density(adjust = 0.5, linewidth = 2, col = "red") +  
 scale\_x\_continuous(breaks = seq(0, 1, by = 0.05), limits = c(0, 0.5)) +  
 theme\_minimal()

|  |
| --- |
| Figure 2.3: Distribution of MAF (minor allele frequency values) |

## 2.3 Sample Size Calculation

* [Sample Size Calculation in Genetic Association Studies: A Practical Approach](https://pmc.ncbi.nlm.nih.gov/articles/PMC9863799/#sec5-life-13-00235), from Politi et al. ([2023](#ref-politi2023l)), notably listing the following tools:
  + [GENPWR: Power and Sample Size Calculations for Genetic Association Studies in the Presence of Genetic Model Mis-Specification](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7666027/), from Moore, Jacobso, and Fingerlin ([2019](#ref-moore2019hh)) and available as a [R package](https://cran.r-project.org/web/packages/genpwr/vignettes/vignette.html), is able to compute the sample size for dichotomous or continuous outcomes in gene-only studies, implementing additive, dominant and recessive models (see [Tip 2.2](#tip-genetic-model) for details). Report to [Section 2.3.1](#sec-GENPWR) for a practical illustration.
  + [PGA: Power for Genetic Association](https://pmc.ncbi.nlm.nih.gov/articles/PMC2387159/), from Menashe, Rosenberg, and Chen ([2008](#ref-menashe2008bg)) is able to adjust for **multiple comparisons**, in contrast with GENPWR, and LD between the marker and the disease allele. Seems not available anymore

|  |
| --- |
| Tip 2.2: Genetic models |
| * **Genotype**: At a given SNP with alleles **A** (major) and **a** (minor), the genotypes are:   + **AA** (homozygous *major*)   + **Aa** (*heterozygous*)   + **aa** (homozygous *minor*) * **Phenotype**: A continuous or binary outcome (e.g., height, disease status) * **Notations**: : phenotype, : genotype, : effect size, : covariates (e.g., age, sex, PCs) and : error term.   🧬 Genetic Models in GWAS   * **Additive model**: in [Equation 2.1](#eq-genetic-model), ‘gene cat’ is converted as AA = 0, Aa = 1 and aa = 2. Each additional copy of the minor allele increases the trait linearly. * **Dominant model**: in [Equation 2.1](#eq-genetic-model), ‘gene cat’ is converted as AA = 0, Aa = 1 and aa = 1. Presence of **at least one** minor allele confers full effect (no dose response). * **Recessive model**: in [Equation 2.1](#eq-genetic-model), ‘gene cat’ is converted as AA = 0, Aa = 0 and aa = 1. Only **homozygous minor** genotype has an effect.     Main Genetic Models. In conclusion, use **additive** by default (most comprehensive), and explore **dominant/recessive** in **post-GWAS** or when you have strong biological priors (e.g., loss-of-function){#tbl-genetic-model}   | Model | Coding | Assumes effect from | Pros | Cons | | --- | --- | --- | --- | --- | | Additive | 0 / 1 / 2 | Per additional allele | High power, interpretable | May miss non-linear effects | | Dominant | 0 / 1 / 1 | Any copy of minor allele | Fits some Mendelian patterns | Oversimplifies if dose-response | | Recessive | 0 / 0 / 1 | Two copies of minor allele | Fits true recessive traits | Low power with rare alleles |   Other less Frequent Models include **Overdominant**, in which the effect only occurs in **heterozygotes**. Rarely used in GWAS, but relevant for **heterozygous** scenarios (e.g., sickle cell and malaria). |

### 2.3.1 GENPWR Sample Size estimation

* Noted out some bugs for GENPWR in extreme scenarios, like negative estimation of sample sizes, or sample sizes not increasing monotonically with decreasing p− value.
* Fadista et al. ([2016](#ref-fadista2016ejhg)) estimates the following thresholds:
  + -value threshold for WGS for common genetic variations in the European population ().
  + for
  + for rare variants with .
  + From the **Conclusion section**, for WES, it turns out that these thresholds can be softened by several orders of magnitude.
* David ([2021](#ref-david2021tig)) determines that:
  + The odds-ratio was on average below 2 for frequent variants.
  + And around 4 for rare variants,
* 2dfstands for two degrees of freedom, in other words, when no inheritance model is assumed.

cartagene\_phenodata <- readr::read\_tsv("data/phenotypes/merge\_phenos\_PCs.txt",   
 show\_col\_types = FALSE)  
  
num\_features <- 9  
num\_variants <- nrow(vcf\_tidy$fix)  
   
case\_rate\_tbl <- table(cartagene\_phenodata$OSTEOPONIA)  
case\_rate <- stats::setNames(case\_rate\_tbl, names(case\_rate\_tbl))  
case\_rate <- case\_rate\_tbl["2"]/sum(case\_rate)  
case\_rate <- case\_rate[["2"]]

#### Without adjustement

pval\_free <- 0.05  
ss\_not\_adjusted <- genpwr.calc(calc = "ss",  
 model = "logistic",   
 ge.interaction = NULL,  
 OR = c(2, 3, 4),   
 Case.Rate = case\_rate,   
 Power = 0.8,  
 MAF = c(0.01, 0.02, 0.05, 0.10, 0.3),   
 Alpha = pval\_free,  
 True.Model = "All",   
 Test.Model = c("Additive"))  
  
ss.plot(ss\_not\_adjusted)  
  
ss\_not\_adjusted <- ss\_not\_adjusted |>   
 rename\_with( ~ "Pval",  
 .cols = starts\_with("N\_total\_at\_Alpha"))  
  
best\_scenario <- ss\_not\_adjusted |>   
 filter(Pval == min(Pval)) |>  
 pull(Pval) |> ceiling()  
  
intermediate\_scenario <- ss\_not\_adjusted |>   
 filter(True.Model == "Additive" & MAF == 0.05 & OR == 3) |>   
 pull(Pval) |> ceiling()  
  
worst\_scenario <- ss\_not\_adjusted |>   
 filter(Pval == max(Pval)) |>   
 pull(Pval) |> ceiling() |>   
 formatC(format = "d", big.mark = ",")  
  
message(paste("Most optimistic scenario (MAF: 0.3, OR: 4, Real:Recessive vs Tested:Additive) needs ", best\_scenario, "samples.\n",   
 "Intermediate scenario (MAF: 0.05, OR: 3, Real:Additive vs Tested:Additive) needs", intermediate\_scenario, "samples.\n",   
 "Finally, the worst scenario (MAF: 0.01, OR: 2, Real:Recessive vs Tested:Additive) needs", worst\_scenario, "samples."))

## Most optimistic scenario (MAF: 0.3, OR: 4, Real:Recessive vs Tested:Additive) needs 157 samples.  
## Intermediate scenario (MAF: 0.05, OR: 3, Real:Additive vs Tested:Additive) needs 725 samples.  
## Finally, the worst scenario (MAF: 0.01, OR: 2, Real:Recessive vs Tested:Additive) needs 86,593,663 samples.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  | | --- | | (a) Odds ratio of 2 | | |  | | --- | | (b) Odds ratio of 3 | | |  | | --- | | (c) Odds ratio of 4 | |

Figure 2.4: Without any adjustement, to be avoided.

#### Adjustement specific to the gene

pval\_adjusted\_gene\_specific <- 0.05/(num\_variants\*num\_features)  
ss\_gene <- genpwr.calc(calc = "ss",  
 model = "logistic",   
 ge.interaction = NULL,  
 OR = c(2, 3, 4),   
 Case.Rate = case\_rate,   
 Power = 0.8,  
 MAF = c(0.01, 0.02, 0.05, 0.10, 0.3),   
 Alpha = pval\_adjusted\_gene\_specific,  
 True.Model = "All",   
 Test.Model = c("Additive"))  
  
ss.plot(ss\_gene)  
  
ss\_gene <- ss\_gene |>   
 rename\_with( ~ "Pval",  
 .cols = starts\_with("N\_total\_at\_Alpha"))  
  
best\_scenario <- ss\_gene |>   
 filter(Pval == min(Pval)) |>  
 pull(Pval) |> ceiling()  
  
intermediate\_scenario <- ss\_gene |>   
 filter(True.Model == "Additive" & MAF == 0.05 & OR == 3) |>   
 pull(Pval) |> ceiling()  
  
worst\_scenario <- ss\_gene |>   
 filter(Pval == max(Pval)) |>   
 pull(Pval) |> ceiling() |>   
 formatC(format = "d", big.mark = ",")  
  
message(paste("Most optimistic scenario (MAF: 0.3, OR: 4, Real:Recessive vs Tested:Additive) needs ", best\_scenario, "samples.\n",   
 "Intermediate scenario (MAF: 0.05, OR: 3, Real:Additive vs Tested:Additive) needs", intermediate\_scenario, "samples.\n",   
 "Finally, the worst scenario (MAF: 0.01, OR: 2, Real:Recessive vs Tested:Additive) needs", worst\_scenario, "samples."))

## Most optimistic scenario (MAF: 0.3, OR: 4, Real:Recessive vs Tested:Additive) needs 622 samples.  
## Intermediate scenario (MAF: 0.05, OR: 3, Real:Additive vs Tested:Additive) needs 2877 samples.  
## Finally, the worst scenario (MAF: 0.01, OR: 2, Real:Recessive vs Tested:Additive) needs 343,931,663 samples.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  | | --- | | (a) Odds ratio of 2 | | |  | | --- | | (b) Odds ratio of 3 | | |  | | --- | | (c) Odds ratio of 4 | |

Figure 2.5: With an -threshold adjusted to the gene-candidate context.

#### Recommended GWAS adjustement

pval\_adjusted\_gwas <- 5 \* 10^-8  
ss\_GWAS <- genpwr.calc(calc = "ss",  
 model = "logistic",   
 ge.interaction = NULL,  
 OR = c(2, 3, 4),   
 Case.Rate = 0.1,   
 Power = 0.8,  
 MAF = c(0.01, 0.02, 0.05, 0.10, 0.3),   
 Alpha = pval\_adjusted\_gwas,  
 True.Model = "All",   
 Test.Model = c("Additive"))  
  
ss.plot(ss\_GWAS)  
  
ss\_GWAS <- ss\_GWAS |>   
 rename\_with( ~ "Pval",  
 .cols = starts\_with("N\_total\_at\_Alpha"))  
  
best\_scenario <- ss\_GWAS |>   
 filter(Pval == min(Pval)) |>  
 pull(Pval) |> ceiling()  
  
intermediate\_scenario <- ss\_GWAS |>   
 filter(True.Model == "Additive" & MAF == 0.05 & OR == 3) |>   
 pull(Pval) |> ceiling() |>   
 formatC(format = "d", big.mark = ",")  
  
worst\_scenario <- ss\_GWAS |>   
 filter(Pval == max(Pval)) |>   
 pull(Pval) |> ceiling() |>   
 formatC(format = "d", big.mark = ",")  
  
message(paste("Most optimistic scenario (MAF: 0.3, OR: 4, Real:Recessive vs Tested:Additive) needs ", best\_scenario, "samples.\n",   
 "Intermediate scenario (MAF: 0.05, OR: 3, Real:Additive vs Tested:Additive) needs", intermediate\_scenario, "samples.\n",   
 "Finally, the worst scenario (MAF: 0.01, OR: 2, Real:Recessive vs Tested:Additive) needs", worst\_scenario, "samples."))

## Most optimistic scenario (MAF: 0.3, OR: 4, Real:Recessive vs Tested:Additive) needs 517 samples.  
## Intermediate scenario (MAF: 0.05, OR: 3, Real:Additive vs Tested:Additive) needs 2,478 samples.  
## Finally, the worst scenario (MAF: 0.01, OR: 2, Real:Recessive vs Tested:Additive) needs 287,495,685 samples.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  | | --- | | (a) Odds ratio of 2 | | |  | | --- | | (b) Odds ratio of 3 | | |  | | --- | | (c) Odds ratio of 4 | |

Figure 2.6: With an -threshold of .

## 2.4 Additional resources

* [FreeBayes variant calling workflow for DNA-Seq](https://bioinformaticsworkbook.org/dataAnalysis/VariantCalling/freebayes-dnaseq-workflow.html) for detecting from scratch small polymorphisms, such as *SNPs* (single-nucleotide polymorphisms), *indels* (insertions and deletions), and *MNPs* (multi-nucleotide polymorphisms):
  + Provides as well a comprehensive tutorial for retrieving variants annotations.
  + QC plots with the vcfR package.

# Appendix A — Tools installation

## A.1 Build from scratch command-line GWAS tools

### A.1.1 Bash commands to install plink2

# Step 1: Download the latest PLINK2 binary  
wget -O ./bin/plink2.zip "https://s3.amazonaws.com/plink2-assets/alpha6/plink2\_linux\_x86\_64\_20250129.zip"   
  
unzip -d ./bin/plink2 ./bin/plink2.zip && rm ./bin/plink2.zip  
  
# Step 4: Ensure the binary is executable  
chmod 777 ./bin/plink2  
  
# Step 5: Verify installation  
./bin/plink2 --version

### A.1.2 Bash commands to install bcftools

wget -O ./bin/bcftools-1.21.tar.bz2 "https://github.com/samtools/bcftools/releases/download/1.21/bcftools-1.21.tar.bz2"   
  
tar -xjvf ./bin/bcftools-1.21.tar.bz2 -C ./bin/ && rm ./bin/bcftools-1.21.tar.bz2  
  
cd ./bin/bcftools-1.21  
./configure --prefix=/mnt/projects\_tn01/Cartagene/cartagene-gwas/bin/bcftools  
  
make  
make install  
  
rm -rf ./bin/bcftools-1.21

### A.1.3 Bash commands to save path executions

# Temporary (Current Shell Session Only)  
export PATH=./bin/bcftools/bin:$PATH   
  
# Semi temporary (for a given user)  
echo 'export PATH=./bin/bcftools/bin:$PATH' >> ~/.bashrc  
source ~/.bashrc

# Bibliographic References

Attikora, Affi Jean Paul, Kouakou Alfred Kouassi, Saraka Didier Martial Yao, Dougba Noel Dago, Souleymane Silué, Caroline De Clerck, Nafan Diarrassouba, et al. 2025. ‘Genome-Wide Association Study of Fat Content and Fatty Acid Composition of Shea Tree (Vitellaria Paradoxa C.F. Gaertn Subsp. Paradoxa)’. *BMC Genomics* 26 (1): 164. <https://doi.org/10.1186/s12864-025-11344-z>.

Awadalla, Philip, Catherine Boileau, Yves Payette, Youssef Idaghdour, Jean-Philippe Goulet, Bartha Knoppers, Pavel Hamet, Claude Laberge, and on behalf of the CARTaGENE Project. 2013. ‘Cohort Profile of the CARTaGENE Study: Quebec’s Population-Based Biobank for Public Health and Personalized Genomics’. *International Journal of Epidemiology* 42 (5): 1285–99. <https://doi.org/10.1093/ije/dys160>.

Bretz, Frank, Torsten Hothorn, and Peter Westfall. 2016. *Multiple Comparisons Using R*. New York: Chapman and Hall/CRC. <https://doi.org/10.1201/9781420010909>.

Bridges, Yasemin, Vinicius de Souza, Katherina G. Cortes, Melissa Haendel, Nomi L. Harris, Daniel R. Korn, Nikolaos M. Marinakis, et al. 2025. ‘Towards a Standard Benchmark for Phenotype-Driven Variant and Gene Prioritisation Algorithms: PhEval - Phenotypic Inference Evaluation Framework’. *BMC Bioinformatics* 26 (1): 87. <https://doi.org/10.1186/s12859-025-06105-4>.

Chen, Ting-Huei, Chia-Jung Lee, Syue-Pu Chen, Shang-Jung Wu, and Cathy S J Fann. 2025. ‘PNL: A Software to Build Polygenic Risk Scores Using a Super Learner Approach Based on PairNet, a Convolutional Neural Network’. *Bioinformatics* 41 (2): btaf071. <https://doi.org/10.1093/bioinformatics/btaf071>.

David, Susana. 2021. ‘A Current Guide to Candidate Gene Association Studies’. *Trends in Genetics* 37 (12): 1056–59. <https://doi.org/10.1016/j.tig.2021.07.009>.

Fadista, João, Alisa K. Manning, Jose C. Florez, and Leif Groop. 2016. ‘The (in)famous GWAS P-value Threshold Revisited and Updated for Low-Frequency Variants’. *European Journal of Human Genetics* 24 (8): 1202–5. <https://doi.org/10.1038/ejhg.2015.269>.

Gnanaolivu, Rohan, Gavin Oliver, Garrett Jenkinson, Emily Blake, Wenan Chen, Nicholas Chia, Eric W. Klee, and Chen Wang. 2025. ‘A Clinical Knowledge Graph-Based Framework to Prioritize Candidate Genes for Facilitating Diagnosis of Mendelian Diseases and Rare Genetic Conditions’. *BMC Bioinformatics* 26 (1): 82. <https://doi.org/10.1186/s12859-025-06096-2>.

Harris, Laura, Ellen M. McDonagh, Xiaolei Zhang, Katherine Fawcett, Amy Foreman, Petr Daneck, Panagiotis I. Sergouniotis, et al. 2025. ‘Genome-Wide Association Testing Beyond SNPs’. *Nature Reviews Genetics* 26 (3): 156–70. <https://doi.org/10.1038/s41576-024-00778-y>.

Hayes, Ben. 2013. ‘Overview of Statistical Methods for Genome-Wide Association Studies (GWAS)’. In *Genome-Wide Association Studies and Genomic Prediction*, edited by Cedric Gondro, Julius van der Werf, and Ben Hayes, 149–69. Totowa, NJ: Humana Press. <https://doi.org/10.1007/978-1-62703-447-0_6>.

Kim, Na Yeon, and Seunggeun Lee. 2025. ‘XPRS: A Tool for Interpretable and Explainable Polygenic Risk Score’. *Bioinformatics*, March, btaf143. <https://doi.org/10.1093/bioinformatics/btaf143>.

Leeuw, Christiaan A. de, Joris M. Mooij, Tom Heskes, and Danielle Posthuma. 2015. ‘MAGMA: Generalized Gene-Set Analysis of GWAS Data’. *PLOS Computational Biology* 11 (4): e1004219. <https://doi.org/10.1371/journal.pcbi.1004219>.

Lin, Jiecong, Ruibang Luo, and Luca Pinello. 2024. ‘EPInformer: A Scalable Deep Learning Framework for Gene Expression Prediction by Integrating Promoter-Enhancer Sequences with Multimodal Epigenomic Data’. 1 August 2024. <https://doi.org/10.1101/2024.08.01.606099>.

Menashe, Idan, Philip S Rosenberg, and Bingshu E Chen. 2008. ‘PGA: Power Calculator for Case-Control Genetic Association Analyses’. *BMC Genetics* 9 (May): 36. <https://doi.org/10.1186/1471-2156-9-36>.

Moore, Camille M., Sean A. Jacobso, and Tasha E. Fingerlin. 2019. ‘Power and Sample Size Calculations for Genetic Association Studies in the Presence of Genetic Model Mis-Specification’. *Human Heredity* 84 (6): 256–71. <https://doi.org/10.1159/000508558>.

Mostafavi, Hakhamanesh, Jeffrey P. Spence, Sahin Naqvi, and Jonathan K. Pritchard. 2023. ‘Systematic Differences in Discovery of Genetic Effects on Gene Expression and Complex Traits’. *Nature Genetics* 55 (11): 1866–75. <https://doi.org/10.1038/s41588-023-01529-1>.

Nolen, Zachary J. 2025. ‘PopGLen—A Snakemake Pipeline for Performing Population Genomic Analyses Using Genotype Likelihood-Based Methods’. *Bioinformatics*, March, btaf105. <https://doi.org/10.1093/bioinformatics/btaf105>.

Pelletier, Justin. 2022. ‘Évaluation de l’imputation Des Données Génétiques Canadiennes-Françaises’, April. <http://hdl.handle.net/1866/27582>.

Pérez-Granado, Judith, Janet Piñero, and Laura I. Furlong. 2022. ‘Benchmarking Post-GWAS Analysis Tools in Major Depression: Challenges and Implications’. *Frontiers in Genetics* 13 (October). <https://doi.org/10.3389/fgene.2022.1006903>.

Politi, Cristina, Stefanos Roumeliotis, Giovanni Tripepi, and Belinda Spoto. 2023. ‘Sample Size Calculation in Genetic Association Studies: A Practical Approach’. *Life* 13 (1): 235. <https://doi.org/10.3390/life13010235>.

Schönherr, Sebastian, and Lukas Forer. (2021) 2024. ‘Nf-Gwas - A Nextflow Pipeline to Perform GWAS.’ <https://github.com/genepi/nf-gwas>.

Sun, Na, Qiang Han, Yu Wang, Mengtong Sun, Ziqing Sun, Hongpeng Sun, and Yueping Shen. 2025. ‘BHCox: Bayesian Heredity-Constrained Cox Proportional Hazards Models for Detecting Gene-Environment Interactions’. *BMC Bioinformatics* 26 (1): 58. <https://doi.org/10.1186/s12859-025-06077-5>.

Uffelmann, Emil, Qin Qin Huang, Nchangwi Syntia Munung, Jantina de Vries, Yukinori Okada, Alicia R. Martin, Hilary C. Martin, Tuuli Lappalainen, and Danielle Posthuma. 2021. ‘Genome-Wide Association Studies’. *Nature Reviews Methods Primers* 1 (1): 1–21. <https://doi.org/10.1038/s43586-021-00056-9>.

Wang, William Y. S., Bryan J. Barratt, David G. Clayton, and John A. Todd. 2005. ‘Genome-Wide Association Studies: Theoretical and Practical Concerns’. *Nature Reviews Genetics* 6 (2): 109–18. <https://doi.org/10.1038/nrg1522>.

Yang, Zikun, Basilio Cieza, Dolly Reyes-Dumeyer, Rosa Montesinos, Marcio Soto-Añari, Nilton Custodio, and Giuseppe Tosto. 2024. ‘A Benchmark Study on Current GWAS Models in Admixed Populations’. *Briefings in Bioinformatics* 25 (1): bbad437. <https://doi.org/10.1093/bib/bbad437>.

Zhao, Gefei, and Binbin Lai. 2025. ‘SC-VAR: A Computational Tool for Interpreting Polygenic Disease Risks Using Single-Cell Epigenomic Data’. *Briefings in Bioinformatics* 26 (2): bbaf123. <https://doi.org/10.1093/bib/bbaf123>.

Zhao, Zijie, Tim Gruenloh, Meiyi Yan, Yixuan Wu, Zhongxuan Sun, Jiacheng Miao, Yuchang Wu, Jie Song, and Qiongshi Lu. 2024. ‘Optimizing and Benchmarking Polygenic Risk Scores with GWAS Summary Statistics’. *Genome Biology* 25 (1): 260. <https://doi.org/10.1186/s13059-024-03400-w>.

1. Comprehensive **Introduction section** reviewing the most critical challenges when coping with GWAS. [↑](#footnote-ref-35)
2. Notably includes MHC, lactase region, and known inversions 8p23 and 17q21.31. [↑](#footnote-ref-55)
3. --king-cutoff 0.0884 corresponds to PI\_HAT = 0.125, see details here [↑](#footnote-ref-59)
4. DMOTSCORE\_mod represents the measurement of bone mineral density at the calcaneus. While no universally accepted threshold exists for the diagnosis of osteopenia or osteoporosis, we adopt the conventional cut-off of -1.5. [↑](#footnote-ref-78)
5. For both FXMOF\_POST and FXALL\_POST, the dates of occurrence and censoring dates are available in FXALL\_POSTDATE and FXMOF\_POSTDATE respectively. [↑](#footnote-ref-81)
6. Use of dnaseq Genpipes pipelines, run on Compute Canada Clusters, version 3.1.5. [↑](#footnote-ref-104)
7. Avoid using BiomarT package a tall costs: it’s not anymore maintained, and leveraged out-of-date Hg37 Human Genome Build. Report to [Genome Builds Versions](https://www.linkedin.com/posts/%F0%9F%8E%AF-ming-tommy-tang-40650014_genome-builds-matter-avoid-costly-mistakes-activity-7293270985930141698-ZFkH) for details. [↑](#footnote-ref-105)
8. All curated VCF files per chromosome are stored in ./data/genotypages/ (linked symbolically with /mnt/projects\_tn01/Cartagene/genotypage/imputation/imputation\_merged). [↑](#footnote-ref-110)
9. nohup option combined with & character enables running variants extraction in the background, even with terminal disrupting and closing. [↑](#footnote-ref-112)
10. dosage=HDS, for Hard Dosage, provides the instruction describing genotype uncertainties for enhanced statistical power. It’s particularly useful when working as here with **imputed genotypes**, where part of the SNPs were inferred using reference panels. [↑](#footnote-ref-117)
11. The hide-covar option runs GLM, but doesn’t output covariate results. Besides, note that --covar passes *quantitative covariates* (e.g., age, PCs), while --ide-covar passes *categorical covariates* [↑](#footnote-ref-118)
12. **Note that a VIF factor above 10 is considered problematic as it can significantly distort regression estimates.** Discussion and visualisations on [*VIF influence*](https://www.linkedin.com/feed/update/urn:li:activity:7308463515432833026/). Briefly, VIF quantifies the degree of similarity across predictor variables, as the GLM-family usually assumes independence between explanatory variables. When VIF is significant, independence assumption is discarded, which might be explained by the fact that preprocessing in [Section 1.3.2.2](#sec-preprocessing-redundancy) for removing correlated SNps and individuals are not used in subsequent GWAS differential downstream analyses. [↑](#footnote-ref-121)
13. More customisation available in [Manhattan plot in R: a review](https://r-graph-gallery.com/101_Manhattan_plot.html) [↑](#footnote-ref-125)
14. [How to interpret a -value histogram](https://varianceexplained.org/statistics/interpreting-pvalue-histogram/) [↑](#footnote-ref-127)
15. [How I Make QQ Plots Using ggplot2?](https://danielroelfs.com/blog/how-i-make-qq-plots-using-ggplot/) [↑](#footnote-ref-129)
16. Note that relying on HPO terms, for example, HP:0004322 – Osteopenia, would alleviate the need for standardising phenotype labels, using universally acknowledged scientific terms instead. [↑](#footnote-ref-158)