Assignment 1

Maria Izabel Cavassim Alves

11/29/2021

# Assignment 1:

## GWAS Quality Control using PLINK

Software: We will be using plink 1.9 which can be downloaded here: <https://www.cog-genomics.org/plink/1.9/>

Plink is a comprehensive tool for handling and analyzing SNP data that can perform many different kinds of analyses. In the left side panel of the website you can see links to the documentation of the different plink data formats and functions.

If you want info about a specific command you can also use help command:

# shell  
plink --help <name\_of\_command>

We will also be using R and Rstudio to make plots and make simple calculations. R can be downloaded here: <http://mirrors.dotsrc.org/cran/> Rstudio here: <https://www.rstudio.com/products/rstudio/download/>

# Data:

In this practical, we will go through quality control (QC) steps of SNP/genotype data from a simulated genome-wide association study of 1000 cases and 1000 controls, typed for 317,503 autosomal and X chromosome SNPs.

The data set for this exercise can be downloaded from the [Dropbox website](https://www.dropbox.com/sh/l1htxivc3j56mh4/AABOiapxu9_cQnIf6vXZDelJa?dl=0)

The assignment is based on “Data quality control in genetic case-control association studies” (Anderson et al. 2010, Nature Protocols 5: 1564-73). This is the first article that you guys should read, and a pdf version is found [here](https://github.com/izabelcavassim/Analyses_of_genome_data/blob/main/Papers/Paper1_Monday.pdf).

# Exercise contents

We will begin by performing sample QC, including calculation of call rates, heterozygosity, and sex discordance. We will then perform SNP QC, including the calculation of call rates and deviation from Hardy-Weinberg equilibrium.

# Data Input and data output

Throughout this exercise we will be using both PLINK (in the cluster) and R/Rstudio in your computer. To make life easier in terms of transferring files, I would recommend that: Mac and windows users download the software cyberduck <https://cyberduck.io/> And set your email used to access the cluster and password using the SFTP mode. Alternatively you don’t use the cluster, but install the PLINK software in your local machine.

# Sample QC

## Identification of individuals with discordant sex information

At the shell prompt, type:

# shell  
./plink --bfile GWA-data --check-sex --out GWA-data

This command will infer the sex of the sample by looking at the mean homozygosity rate across X-chromosome markers and compare it the sex stated in the “.fam” file.

### Question 1

E1.1) What is outputted by the software?

### Question 2

E1.2) Take a look at the output file “GWA-data.sexcheck” by typing Bash cat GWA-data.sexcheck. How many problematic samples are there?

Problematic samples can be removed by copying the family ID (FID) and individual ID (IID) of the samples into a text file (e.g. wrong\_sex.txt) and using the remove command: (Hint: subset the data in R)

First let’s subset the GWA-data.sexcheck \*\*in R\*

# R  
  
# Set your own directory containing the exercise data:  
setwd('/Users/PM/Dropbox/PHD/GVHH/Exercise1\_data/')  
  
# Output of plink says:  
  
# --check-sex: 8921 Xchr and 0 Ychr variant(s) scanned, 3 problems detected.  
  
sex\_check = read.table('GWA-data.sexcheck', header = TRUE)  
  
# Copying the family ID and the individual ID of the samples where Status was not ok:  
wrong\_sex = sex\_check[sex\_check$STATUS != 'OK', c(1,2)]  
  
# Three observations did not pass the status   
View(wrong\_sex)  
  
write.table(wrong\_sex, file = 'wrong\_sex.txt', row.names = FALSE, col.names = TRUE, quote = FALSE)

Now remove the individuos that were assigned with the wrong sex by typing:

# shell  
./plink --bfile GWA-data --remove wrong\_sex.txt --make-bed --out GWA-data

The –out option in plink specifies the prefix of the output files that plink generates. And when we use the –make-bed command with the same prefix as the input **we are actually overwriting the input files**. This is OK for these exercises (so we avoid using to much space on your machines) but on a “real” data set you might not want to do that. Note that once you try the first command again you won’t see problems in the sex-check. Let’s try it again:

# shell  
./plink --bfile GWA-data --check-sex --out GWA-data

### Question 3

E1.3) Each time a plink command is run it writes a summary to a log file (the file name ends with “.log”). Look at the log file after removing the problematic individuals. How many cases and controls are left in the data set? *Hint:* use the cat command to see the file, or open it with you favorite text editor.

## Identification of individuals with elevated missing data rates or outlying heterozygosity rate

At the shell prompt, type:

# shell  
./plink --bfile GWA-data --missing --out GWA-data

This command will create the files *“GWA-data.imiss”* and *“GWA-data.lmiss”*. The fourth column in the file “GWA-data.imiss” (N\_MISS) denotes the number of missing SNPs and the sixth column (F\_MISS) denotes the proportion of missing SNPs per individual. What is the mean of the values of N\_MISS and F\_MISS across the data?

To compute this you can *use R* as following:

# R  
Imiss = read.table('GWA-data.imiss', header = TRUE)  
  
# Mean of the number of missing values:   
mean(Imiss$N\_MISS)  
  
# Mean of the frequency missing values  
mean(Imiss$F\_MISS)

At the shell prompt type:

# shell  
./plink --bfile GWA-data --het --out GWA-data

This command will create the file “GWA-data.het”, in which the third column denotes the observed number of homozygous genotypes [O(Hom)] and the fifth column denotes the number of non-missing genotypes [N(NM)] per individual.

You can calculate the observed heterozygosity rate per individual using the formula:

Het = (N(NM) − O(Hom))/N(NM)

*using R*, again reading the GWA-data.het file in your Rstudio:

# R  
# Calculating heterozigosity  
het = read.table('GWA-data.het', header = T)  
het$het\_cal = (het$N.NM. - het$O.HOM.)/ het$N.NM.

# Question 4

E1.4) What is the mean heterozygosity of the data? Hint: in R you can use the function *mean*. If you want to learn more about the function, and you want to get some help, you can type ?mean()in the R console.

### Question 5

E1.5) Use R to create a plot in which the observed heterozygosity rate per individual is plotted on the x axis and the proportion of missing SNPs per individuals is plotted on the y axis. To do so you can use the function plot in R. You are expected to create a figure similar to the Figure 1 from the paper (Anderson et al. 2010).

### Question 6

E1.6) Use R to make a file with the FID and IID of all individuals that have a genotype missing rate >=0.03 or a heterozygosity rate that is more than 3 s.d. from the mean. Then use plink to remove these individuals from the data set.

To subset individuals based on those criteria you can type *in R*

# R  
filtering = cbind(Imiss, het)  
right\_tail = mean\_het + 3\*sd(het$het\_cal)  
left\_tail = mean\_het - 3\*sd(het$het\_cal)  
  
# Individuals to be filtered  
filtering\_ind = subset(filtering, filtering$F\_MISS >= 0.03 | filtering$het\_cal > right\_tail | filtering$het\_cal < left\_tail)  
  
  
write.table(filtering\_ind[,c(1,2)], 'wrong\_het\_missing\_values.txt', col.names = FALSE, row.names = FALSE)

To subset the data with PLINK use the the feature --remove.

How many individuals did not pass the criteria above? How many individuals are left from case and control?

## Identification of duplicated or related individuals

To identify duplicated or related individuals we will calculate the identity by descent (IBD) matrix. This works best if it is done on a set of non-correlated SNPs. So first we will “prune” the data and create a list of SNPs where no pair (within a given genomic interval) has an r2 value greater than a given threshold, typically chosen to be 0.2.

This can be done by the “indep-pairwise” command:

plink --bfile GWA-data --indep-pairwise 500kb 5 0.2 --out GWA-data

It saves the list of independent SNPs as “GWA-data.prune.in”. This data set was simulated without LD so in this case there will not be a lot of variants removed.

To calculate IBD between each pair of individuals, type the following command at the shell prompt:

plink --bfile GWA-data --extract GWA-data.prune.in --genome --min 0.185 --out GWA-data

The “–min 0.185” option means that it will only print the calculated IBD if it is above 0.185. The PI\_HAT value in column 10 of the output file will be a number between 0 and 1 saying how much of the genome the two individuals share (1 for identical twins, 0.5 for siblings and so on).

### Question 7

E1.7) Remove a member from each of the pairs that are too closely related from the data set. To keep it simple you can just always remove the individual mentioned first. Remember to create the .txt file and delete those members as in the previous exercises.

To find out which individual you would want to keep, you can type the following *in R*:

# R  
ibd = read.table('GWA-data.genome', header = TRUE)  
# Eliminate the individuals mentioned first (IID1?)  
members = ibd[,3]  
members = unique(members)  
  
# IBD member to be removed:  
write.table(cbind(members,members), file = 'wrong\_ibd.txt', col.names = F, row.names = F)

How many individuals are found to share more than 0.185 of IBD?

Now use the file ‘wrong\_ibd.txt’ to remove out the individuals that did not pass our threshold.

## SNP QC

### SNPs with an excessive missing data rate

Run the “–missing” command again to generate the “GWA-data.lmiss” with the missing data rate for each SNP.

### Question 8

E1.8) Use R to make a histogram of the missing data rates. Hint: use the R function hist().

The “–test-missing” command tests all markers for differences in the call rate between cases and controls.

### Question 9

E1.9) Run the test-missing command and make a list of all the names of all SNPs where the differential missing-ness p-value is less than 10e-5.

plink --bfile GWA-data --test-missing --out test

Save the list as “fail-diffmiss-qc.txt”.

test\_missing = read.table('test.missing', header = TRUE)  
fail\_diffmiss\_qc = test\_missing[test\_missing$P < 10e-5, 2]  
write.table(fail\_diffmiss\_qc, file = 'fail-diffmiss-qc.txt', row.names = F, col.names = F)

To remove low-quality SNPs, type the following command at the shell prompt:

plink --bfile GWA-data --exclude fail-diffmiss-qc.txt --geno 0.05 --hwe 0.00001 --maf 0.01 --make-bed --out GWA-data

In addition to removing SNPs identified with differential call rates between cases and controls, this command removes SNPs with call rate less than 5% with –geno option and deviation from HWE (p<10-5) with the –hwe option. It also removes all SNPs with minor allele frequency less than a specified threshold using the –maf option.

### Question 10

E1.10) How many SNPs are left in the clean data set?