Genetic Association Studies with R (part II)

in

Master in Omic data analysis

Universitat Vic (Uvic)

Juan R Gonzalez

(juanr.gonzalez@isglobal.org)

BRGE - Bioinformatics Research Group in Epidemiology

http://brge.isglobal.org

Barcelona Institute for Global Health (ISGlobal)

Departament of Mathematics, Universidad Autonoma de Barcelona (UAB)

Genetic Association analysis using R

- **Part I:** Single association analysis: SNPassoc package
  - Descriptive analysis
  - HWE test
  - Association analysis
  - Haplotype analyses
- **Part II:** GWAS snpStats package
  - Quality control
  - Association analysis
  - Population stratification
  - Multiple comparisons
  - Manhantan plots

GWAS

Data format:

- CEGEN (Spanish Genotyping Center)
- BeadStudio (Illumina) and APT tools (Affymetrix)
- Plink

They provide information about genotypes (default). Other outputs can be generated upon request (e.g., plink format)

BeadStudio and APT tools

Information about: rs, chromosome, position, BAF, log2ratio, genotype

PLINK

- accesible at
  `http://pngu.mgh.harvard.edu/~purcell/plink`
- two files
  - **.ped** (genotypes with first 6 columns: family ID, individual ID, paternal ID, maternal ID, Sex, phenotype)
  - **.map** (annotation icluding: cromosoma, rs, genetic distance, base-pair position)
- binary format
  - **.bed** (genotype information in binary format)
  - **.fam** (individual and familiar information - 6 columns from .ped file)
  - **.bim** (annotation - .map including the name of the alleles)

(formerly `snpMatrix`) deal with GWAS data more efficiently than `SNPassoc`. Following analyses can be carried out:

- Descriptive analysis
- Association analysis (GWAS very fast)
- Population stratification
- TDT analysis
- Imputation

snpStats library

`snpStats` can be installed from Bioconductor

```
source("http://bioconductor.org/biocLite.R")
biocLite("snpStats")
```

- Data must be in a special format (PLINK o binary)
- Results are not so 'user-friendly' as in `SNPassoc`

snpStats library

Let us load the library

```
library(snpStats)
```

GWAS data in PLINK format would be loaded by executing

```
ob.plink <- read.plink("data/obesity.bed", "data/obesity.bim",
                       "data/obesity.fam")
```

or

```
ob.plink <- read.plink("data/obesity")
```

This object contains the following information

```
names(ob.plink)

## [1] "genotypes" "fam"        "map"
```

snpStats library

Genotypes are stored as a `SnpMatrix` object. NOTE: alleles are coded as 0,1,2.

```
ob.geno <- ob.plink$genotypes
ob.geno

## A SnpMatrix with  2312 rows and  100000 columns
## Row names:  100 ... 998
## Col names:  MitoC3993T ... rs28600179
```

Annotation is stored as a data.frame

```
annotation <- ob.plink$map
head(annotation)

##            chromosome   snp.name cM position allele.1 allele.2
## MitoC3993T         NA MitoC3993T NA     3993        T        C
## MitoG4821A         NA MitoG4821A NA     4821        A        G
## MitoG6027A         NA MitoG6027A NA     6027        A        G
## MitoT6153C         NA MitoT6153C NA     6153        C        T
## MitoC7275T         NA MitoC7275T NA     7275        T        C
## MitoT9699C         NA MitoT9699C NA     9699        C        T
```

Family structure is also available

```
family <- ob.plink$fam
head(family)

##      pedigree member father mother sex affected
## 100    FAM_OB    100     NA     NA   1        1
## 1001   FAM_OB   1001     NA     NA   1        1
## 1004   FAM_OB   1004     NA     NA   2        2
## 1005   FAM_OB   1005     NA     NA   1        2
## 1006   FAM_OB   1006     NA     NA   2        1
## 1008   FAM_OB   1008     NA     NA   1        1
```

snpStats library

Phenotypic data (e.g. clinical or epidemiological data) must be loaded in a different object (`data.frame`).

```
ob.pheno <- read.delim("data/obesity.txt")
head(ob.pheno)
```

```
##      id gender obese age   smoke country
## 1 4180   Male     1  41 Current      50
## 2 4880 Female    NA  35      Ex      51
## 3  435   Male     1  50      Ex      53
## 4 4938   Male     0  44 Current      53
## 5 2977   Male    NA  49   Never      53
## 6 1705   Male     0  40   Never      50
```

IMPORTANT: Individuals MUST be in the same order as in the genotype object

```
identical(rownames(ob.pheno), rownames(ob.geno))
```

```
## [1] FALSE
```

If FALSE, we cannot perform association analysis. Data have to be properly organized

First, be sure that 'ids' of phenotypic data are in the rownames

```
rownames(ob.pheno) <- ob.pheno$id
```

FALSE indicates that either there are different individuals in both objects or they are in different order. This can be fxed by selecting common individuals.

```
ids <- intersect(rownames(ob.pheno), rownames(ob.geno))
geno <- ob.geno[ids, ]
ob <- ob.pheno[ids, ]
identical(rownames(ob), rownames(geno))

## [1] TRUE

family <- family[ids, ]
```

Then association analyses will be performed using `geno` and `ob` data frames.

Quality control

The `geno` object is an object of class `SnpMatrix` and contains this information:

```
geno

## A SnpMatrix with  2312 rows and  100000 columns
## Row names:  4180 ... 277
## Col names:  MitoC3993T ... rs28600179
```

Quality control (QC) must be performed at both SNP and individual level before assessing association between SNPs and the trait of interest.

QC at SNP level

SNP information can be obtained by executing:

```
info.snps <- col.summary(geno)
head(info.snps)
```

```
##            Calls Call.rate Certain.calls       RAF           MAF
## MitoC3993T  2286 0.9887543             1 0.9851269 0.0148731409
## MitoG4821A  2282 0.9870242             1 0.9982472 0.0017528484
## MitoG6027A  2307 0.9978374             1 0.9956654 0.0043346337
## MitoT6153C  2308 0.9982699             1 0.9893847 0.0106152513
## MitoC7275T  2309 0.9987024             1 0.9991338 0.0008661758
## MitoT9699C  2302 0.9956747             1 0.9268028 0.0731972198
##                    P.AA         P.AB      P.BB     z.HWE
## MitoC3993T 0.0148731409 0.0000000000 0.9851269 -47.81213
## MitoG4821A 0.0017528484 0.0000000000 0.9982472 -47.77028
## MitoG6027A 0.0043346337 0.0000000000 0.9956654 -48.03124
## MitoT6153C 0.0103986135 0.0004332756 0.9891681 -47.05069
## MitoC7275T 0.0008661758 0.0000000000 0.9991338 -48.05206
## MitoT9699C 0.0729800174 0.0004344049 0.9265856 -47.82555
```

QC at SNP level

- Filter SNPs with a p-value lower than 0.001 (e.g. z-score of ± 3.3)
- HWE should only be tested in controls (e.g. obese = 0)
- Filter SNPs with a call rate of 95%
- Fliter SNPs with a call MAF of 5%

QC at SNP level

```
controls <- ob$obese ==0 & !is.na(ob$obese)
geno.controls <- geno[controls,]
info.controls <- col.summary(geno.controls)
use <- info.snps$Call.rate > 0.95 &
       info.snps$MAF > 0.05 &
       abs(info.controls$z.HWE < 3.3)
mask.snps <- use & !is.na(use)
geno.qc.snps <- geno[ , mask.snps]
annotation <- annotation[mask.snps, ]
```

QC at SNP level

```
geno

## A SnpMatrix with  2312 rows and  100000 columns
## Row names:  4180 ... 277
## Col names:  MitoC3993T ... rs28600179

geno.qc.snps

## A SnpMatrix with  2312 rows and  88723 columns
## Row names:  4180 ... 277
## Col names:  MitoT9699C ... rs28562204
```

QC at invididual level

A paper must described the number of SNPs that do not pass QC

```
# number of SNPs removed for bad call rate
sum(info.snps$Call.rate < 0.95)

## [1] 888

# number of SNPs removed for low MAF
sum(info.snps$MAF < 0.05, na.rm=TRUE)

## [1] 10461

#number of SNPs removed do not pass HWE
sum(abs(info.controls$z.HWE > 3.3), na.rm=TRUE)

## [1] 80

# The total number of SNPs removed for any reason
sum(!mask.snps)

## [1] 11277
```

QC at individual level

- Individuals with outlying missing genotype
- Individuals with discordant sex information
- Individuals with bad heterozygosity rate
- Duplicated or related individuals
- Individuals of divergent ancestry

Missing genotype

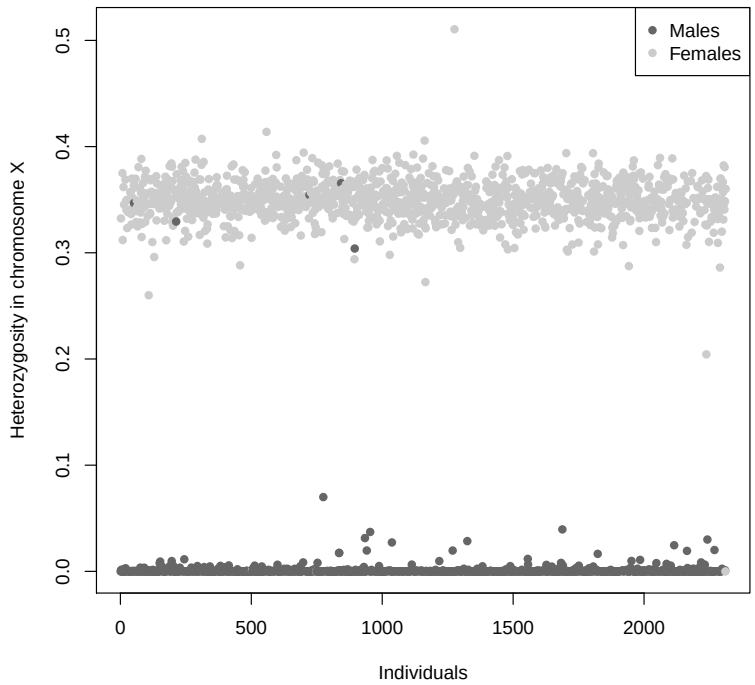
Basic information at individual level can be obtained from

```
info.indv <- row.summary(geno.qc.snps)
head(info.indv)

##      Call.rate Certain.calls Heterozygosity
## 4180 0.9998873             1      0.3426781
## 4880 0.9998197             1      0.3539180
## 435  0.9958297             1      0.3392188
## 4938 0.9994928             1      0.3411782
## 2977 0.9985348             1      0.3426004
## 1705 0.9936657             1      0.3357721
```

Sex discrepancies

```
geno.X <- geno.qc.snps[,annotation$chromosome=="23" &
                        !is.na(annotation$chromosome)]
info.X <- row.summary(geno.X)
mycol <- ifelse(ob$gender=="Male", "gray40", "gray80")
plot(info.X$Heterozygosity, col=mycol, pch=16, xlab="Individuals",
     ylab="Heterozygosity in chromosome X")
legend("topright", c("Males", "Females"), col=mycol,pch=16)
```



Sex discrepancies

```
sex.discrep <- (ob$gender=="Male" & info.X$Heterozygosity > 0.2) |
               (ob$gender=="Female" & info.X$Heterozygosity < 0.2)
```

Bad heterozygosity

F-statistic: $F = 1 - \frac{f(Aa)}{E(f(Aa))}$, where $f(Aa)$ corresponds to the observed proportion of heterozygous genotypes (Aa) for a given individual and $E(f(Aa))$ is the expected proportion of heterozygous genotypes for a given individual based on MAF across all non-missing SNPs for a given individual.

```
MAF <- col.summary(geno.qc.snps)$MAF
callmatrix <- !is.na(geno.qc.snps)
hetExp <- callmatrix %*% (2*MAF*(1-MAF))
hetObs <- with(info.indv,
               Heterozygosity*(ncol(geno.qc.snps))*Call.rate)
info.indv$hetF <- 1-(hetObs/hetExp)
```

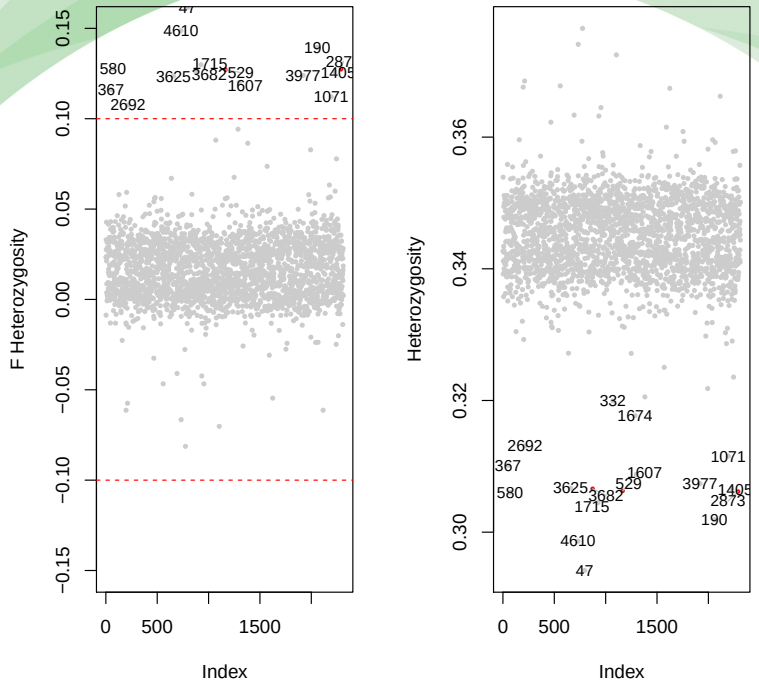
Bad heterozygosity

```
head(info.indv)

##      Call.rate Certain.calls Heterozygosity         hetF
## 4180 0.9998873             1      0.3426781  0.023324353
## 4880 0.9998197             1      0.3539180 -0.008701237
## 435  0.9958297             1      0.3392188  0.033025203
## 4938 0.9994928             1      0.3411782  0.027596273
## 2977 0.9985348             1      0.3426004  0.023487306
## 1705 0.9936657             1      0.3357721  0.042762824
```

A rule-of-thumb is to consider indiviuals who deviates their F statistic from ± 0.1

Bad heterozygosity



Duplicated and related individuals

Identity-by-descent (IBD) analysis provides kinship information (measure of relatedness). IBD analysis is performed by subsetting SNPs that are in LD. First PLINK data must be transformed into GDS format

```
library(SNPRelate)
snpgdsBED2GDS("data/obesity.bed",
              "data/obesity.fam",
              "data/obesity.bim",
              out="obGDS")

## Start snpgdsBED2GDS ...
##  BED file: "data/obesity.bed" in the SNP-major mode (Sample X SNP)
##  FAM file: "data/obesity.fam", DONE.
##  BIM file: "data/obesity.bim", DONE.
## Wed Dec 12 15:51:24 2018  store sample id, snp id, position, and ch
##  start writing: 2312 samples, 100000 SNPs ...
##   Wed Dec 12 15:51:24 2018 0%
##   Wed Dec 12 15:51:25 2018 100%
## Wed Dec 12 15:51:25 2018  Done.
## Optimize the access efficiency ...
## Clean up the fragments of GDS file:
##     open the file obGDS (55.7M)
##     # of fragments: 39
```

Duplicated and related individuals

SNPs are pruned for IBD analysis

```
genofile <- snpgdsOpen("obGDS")
set.seed(12345) #to guarantee reproducibility
snps.qc <- colnames(geno.qc.snps)
snp.prune <- snpgdsLDpruning(genofile, ld.threshold = 0.2,
                             snp.id = snps.qc)

## SNP pruning based on LD:
## Excluding 13,410 SNPs (non-autosomes or non-selection)
## Excluding 0 SNP (monomorphic: TRUE, MAF: NaN, missing rate: NaN)
## Working space: 2,312 samples, 86,590 SNPs
##     using 1 (CPU) core
##     sliding window: 500,000 basepairs, Inf SNPs
##     |LD| threshold: 0.2
##     method: composite
## Chromosome 1: 31.51%, 2,433/7,721
## Chromosome 2: 30.04%, 2,418/8,050
## Chromosome 3: 30.84%, 2,059/6,676
## Chromosome 4: 31.13%, 1,845/5,927
## Chromosome 5: 30.87%, 1,875/6,074
## Chromosome 6: 28.19%, 1,903/6,750
## Chromosome 7: 31.24%, 1,673/5,356
## Chromosome 8: 28.82%, 1,606/5,572
## Chromosome 9: 31.52%, 1,487/4,718
```

Duplicated and related individuals

IBD coefficients can be computed by using the method of moments that is implemented in the funcion `snpgdsIBDMoM`.

```
snps.ibd <- unlist(snp.prune, use.names=FALSE)
ibd <- snpgdsIBDMoM(genofile, kinship=TRUE,
                    snp.id = snps.ibd,
                    num.thread = 1)

## IBD analysis (PLINK method of moment) on genotypes:
## Excluding 69,258 SNPs (non-autosomes or non-selection)
## Excluding 0 SNP (monomorphic: TRUE, MAF: NaN, missing rate: NaN)
## Working space: 2,312 samples, 30,742 SNPs
##     using 1 (CPU) core
## PLINK IBD:    the sum of all selected genotypes (0,1,2) = 32844904
## Wed Dec 12 15:51:31 2018    (internal increment: 6656)
##
[.....................................................]  0%, ETC: ---
[=====================================================] 100%, completed i
## Wed Dec 12 15:51:42 2018    Done.
```

Duplicated and related individuals

This table indicates kinship among pairs of individuals that can be used to detect those who are related.

```
ibd.kin <- snpgdsIBDSelection(ibd)
head(ibd.kin)

##   ID1  ID2        k0         k1      kinship
## 1 100 1001 0.9926611 0.00191261 0.003191305
## 2 100 1004 1.0000000 0.00000000 0.000000000
## 3 100 1005 1.0000000 0.00000000 0.000000000
## 4 100 1006 1.0000000 0.00000000 0.000000000
## 5 100 1008 1.0000000 0.00000000 0.000000000
## 6 100 1013 1.0000000 0.00000000 0.000000000
```

Duplicated and related individuals

Let us check whether there are individuals who are candidate to be removed due to relatedness (e.g kinship > 0.1 can be considered as an adequate threshold)

```
ibd.kin.thres <- subset(ibd.kin, kinship > 0.1)
head(ibd.kin.thres)
```

```
##           ID1  ID2        k0        k1   kinship
## 46484    1049  188 0.2731024 0.5431008 0.2276736
## 232848   1202 1330 0.0000000 0.0000000 0.5000000
## 281069   1237  872 0.2747742 0.4556623 0.2486973
## 640474    155 1682 0.2410303 0.4506688 0.2668176
## 806337    170 2015 0.2548016 0.5399619 0.2376087
## 1158509 2055  825 0.0000000 0.0000000 0.5000000
```

Duplicated and related individuals

The ids of the individuals to be removed can be obtained by using a function that is called `related` and is available at `SNPassoc` package

```
ids.rel <-  SNPassoc::related(ibd.kin.thres)
ids.rel

##  [1] "4364" "3380" "2999" "2697" "2611" "2088" "1202" "872"  "825"
## [11] "188"  "170"  "155"  "2071"
```

QC individuals

```
use <- info.indv$Call.rate > 0.95 &
       abs(info.indv$hetF) < 0.1 &
       !sex.discrep &
       !rownames(info.indv)%in%ids.rel
mask.indiv <- use & !is.na(use)
geno.qc <- geno.qc.snps[mask.indiv, ]

ob.qc <- ob.pheno[mask.indiv, ]
identical(rownames(ob.qc), rownames(geno.qc))

## [1] TRUE
```

QC individuals

```
# number of individuals removed to bad call rate
sum(info.indv$Call.rate < 0.95)
```

```
## [1] 34
```

```
# number of individuals removed for heterozygosity problems
sum(abs(info.indv$hetF) > 0.1)
```

```
## [1] 15
```

```
# number of individuals removed for sex discrepancies
sum(sex.discrep)
```

```
## [1] 8
```

```
# number of individuals removed to be related with others
length(ids.rel)
```

```
## [1] 14
```

```
# The total number of individuals that do not pass QC
sum(!mask.indiv)
```

```
## [1] 70
```

Association analysis

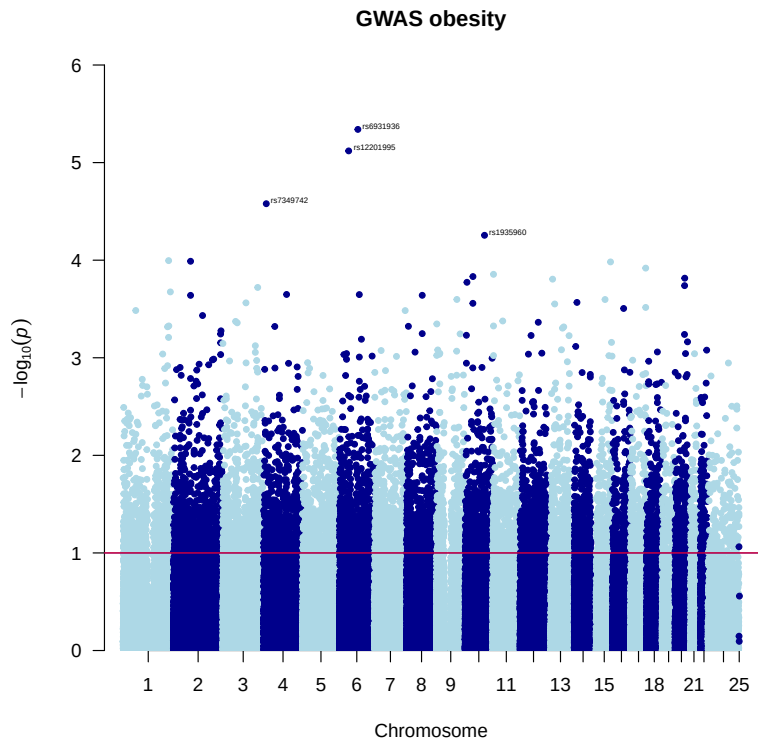
Association is carried out by executing

```
res <- single.snp.tests(obese, data=ob.qc,
                                snp.data=geno.qc)
res[1:5,]

##                N Chi.squared.1.df Chi.squared.2.df      P.1df      P
## MitoT9699C  2134         3.0263311                NA 0.08192307
## MitoA11252G 2090         0.3561812                NA 0.55063478
## MitoA12309G 2136         0.1776464                NA 0.67340371
## MitoG16130A 2069         2.4766387         3.9480296 0.11554896 0.138
## rs28705211  2125         0.7277258         0.7546827 0.39362135 0.685
```

Manhattan Plot

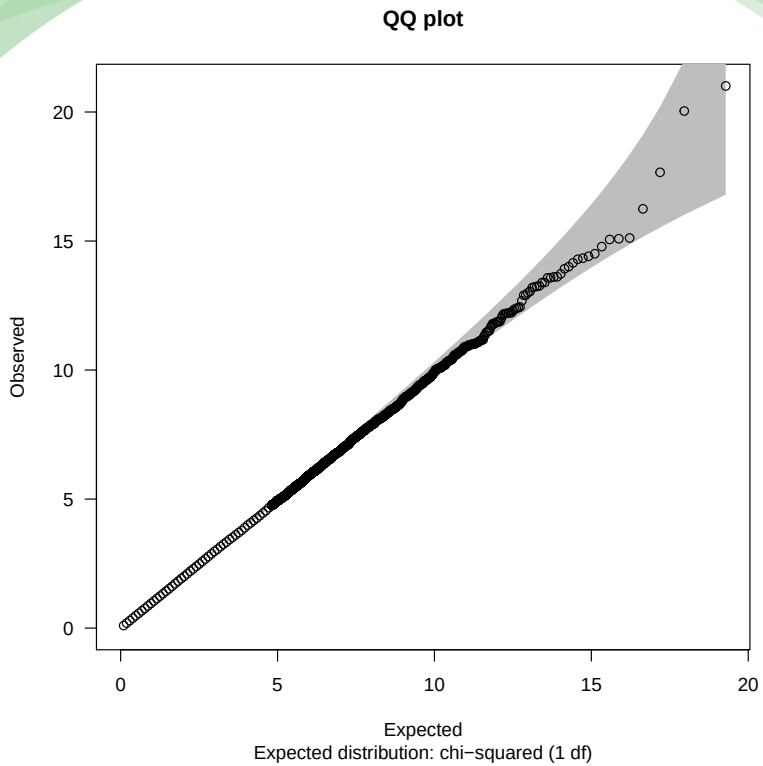
```
library(qqman)
pvals <- data.frame(SNP=annotation$snp.name, CHR=annotation$chromosome
                    BP=annotation$position, P=p.value(res, 1))
pvals <- subset(pvals, !is.na(CHR) & !is.na(P)) # missing data is not
manhattan(pvals, suggestiveline=TRUE, genomewideline=TRUE, annotatePva
          annotateTop = FALSE, main="GWAS obesity", col=c("lightblue",
```



Association analysis

We can also visualize Q-Q plot and to compute λ value to detect possible population stratification

```
chi2 <- chi.squared(res, df=1)
qq.chisq(chi2)
```




```
##              N       omitted        lambda
## 88723.000000      0.000000      1.003853
```

Association analysis

Association analysis can be stratified (gender, contry, region, ...)

```
res.stra <- single.snp.tests(obese,  data=ob.qc, snp.data=geno.qc,
                                     stratum=country)
head(res.stra)

##              N Chi.squared.1.df Chi.squared.2.df   P.1df P.2df
## MitoT9699C 2134         2.800754                NA 0.09422    NA
```

or adjusted for other covariates (NOTE: `snp.rhs.tests` function)

```
res.adj <- snp.rhs.tests(obese ~ smoke, data=ob.qc,
                                        snp.data=geno.qc)
head(res.adj)

##            Chi.squared Df    p.value
## MitoT9699C    3.014901  1 0.08250252
```

Association analysis

Quantitative trait

```
res.quant <- snp.rhs.tests(age ~ 1,  data=ob.qc, snp.data=geno.qc,
                           family="Gaussian")
head(res.quant)

##            Chi.squared Df  p.value
## MitoT9699C 0.003422591  1 0.953348
```

Population stratification

Principal componet analysis (PCA) can be carried out to control for population stratifcication

- PCA computes eigen vectors from a given correlation matrix
- We have information about X a matrix with n rows (samples) and p columns (SNPs)
- Genotype data is standardized (zero mean and unit variance) using:

$$\widetilde{X}_{ij} = \frac{X_{ij} - f_j}{\sqrt{f_j(1-f_j)/N}} \quad i = 1, \ldots, n \quad j = 1, \ldots, p$$

- Eigen vectors are found from $X^t X$ a pxp-matrix
- Hard dealing with a pxp-matrix!!!
- 'loadings' are then computed using $B = X^t S D^{-1/2}$

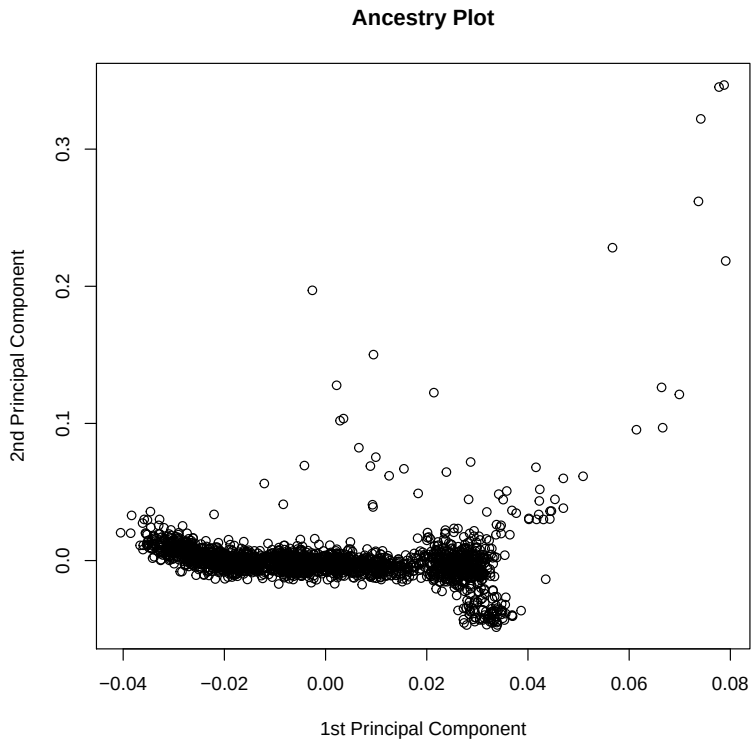
Population stratification

```
pca <- snpgdsPCA(genofile, sample.id = rownames(geno.qc),
                           snp.id = snps.ibd,
                           num.thread=1)

## Principal Component Analysis (PCA) on genotypes:
## Excluding 69,258 SNPs (non-autosomes or non-selection)
## Excluding 0 SNP (monomorphic: TRUE, MAF: NaN, missing rate: NaN)
## Working space: 2,242 samples, 30,742 SNPs
##     using 1 (CPU) core
## PCA:    the sum of all selected genotypes (0,1,2) = 31854924
## CPU capabilities: Double-Precision SSE2
## Wed Dec 12 15:51:55 2018    (internal increment: 216)
##
[..................................................]  0%, ETC: ---
[==================================================] 100%, completed i
## Wed Dec 12 15:52:37 2018    Begin (eigenvalues and eigenvectors)
## Wed Dec 12 15:52:42 2018    Done.
```

Population stratification

```
with(pca, plot(eigenvect[,1], eigenvect[,2],
               xlab="1st Principal Component",
               ylab="2nd Principal Component",
               main = "Ancestry Plot"))
```



Principal component analysis

PCs are added to phenotypic data

```
ob.qc <- data.frame(ob.qc, pca$eigenvect[, 1:5])
```

After performing QC, the GDS file can now be closed

```
closefn.gds(genofile)
```

Association analysis

Then, if population stratification is observed, association analysis is performed adjusting by the principal components

```
res.adj <- snp.rhs.tests(obese ~ X1 + X2 + X3 + X4,
                         data=ob.qc, snp.data=geno.qc)
head(res.adj)

##            Chi.squared Df    p.value
## MitoT9699C    3.183059  1 0.07440532
```