

# Introducción a la Genómica

## UNAL nov 2017

Alejandro Cáceres  
ISGlobal, Barcelona

October 17, 2017

# Genome-wide association studies

En los estudios de asociación genética se prueba la correlación (asociación) entre los SNPs y un fenotipo de interés. Siven para

- ▶ identificar variantes genéticas que den pistas sobre el desarrollo del fenotipo (enfermedad)
- ▶ medir la carga genética (heredabilidad) de un fenotipo
- ▶ crear modelos de riesgo genético que predigan la probabilidad de desarrollar una enfermedad.
- ▶ para ver los variantes funcionales que afectan endofenotipos como expresión genica (eQTIs)

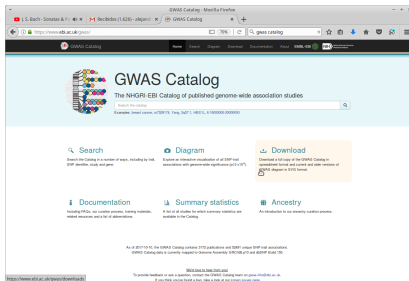
# Genome-wide association studies

Una Gran cantidad de GWAS se han hecho hasta el momento. Se pueden consultar en GWAS catalog



# Genome-wide association studies

Una Gran cantidad de GWAS se han hecho hasta el momento. Se pueden consultar en GWAS catalog



# Genome-wide association studies

Ejercicio:

- ▶ buscar en el GWAS catalog los resultados de estudios de Alzheimer's.
- ▶ buscar cuales GWAS han encontrado variantes geneticos en BRCA1

# Software

Han una gran cantidad de software para hacer estos análisis

- ▶ PLINK, rapido pero por linea de comandos. Da poca flexibilidad
- ▶ snpStats mas lento pero se pueden aprovechar los paquetes de R
- ▶ codigo en R. Se puede usar cualquier tipo de test. Lo ideal es paralelizar.
- ▶ snpAssoc. Tambien en R. Para estudios con número limitado de SNPs pero prueba todos lo modelos geneticos y ajusta por el número de tests.

PLINK tiene unos datos de prueba y unos ejemplos de analisis sobre ellos

Resources - PLINK 1.9 - Mozilla Firefox

https://www.cog-genomics.org/plink/1.9/resources

## Teaching materials and example dataset

These files were created by Shaun Purcell for PLINK 1.02 (+ gPLINK + Haploview), but everything except for the haplotypic analysis will still work with 1.90.

- **Tutorial data: [example.zip \(BWH mirror\)](#)**, which contains the following six files:
  - wgas1.ped (sample whole-genome .ped data file)
  - wgas1.map (corresponding .map file)
  - extra.ped (sample follow-up regional genotyping .ped file)
  - extra.map (corresponding .map file)
  - pop.cov (population membership variable)
  - command-list.txt (command list for 2nd part of practical)
  - The BWH mirror file also contains an old Windows plink.exe, and gPLINK/Haploview .jar files.
- **Teaching materials: [teaching.zip \(BWH mirror\)](#)**, which contains the following two files:
  - practical-1-slides.ppt
  - practical-2-notes.doc

Everything should be fairly self-explanatory after looking through the PowerPoint file and Word document.

## Gene range lists

These lists are valid input for flags such as `--make-set`, `--extract range`, `--annotate ranges`, and `--gene-report`.

- hg18: [glist-hg18](#) (older, ASCII-sorted BWH version)
- hg19: [glist-hg19](#)
- hg38: [glist-hg38](#)

They contain one gene per row, with the following four columns:

1. Chromosome code
2. Start of gene (base-pair units, 1-based)
3. End of gene (this position is included in the interval)
4. Gene ID

Veamos un ejemplo de análisis con snpStats  
Cargamos los genotipos

```
library("snpStats")

## Loading required package: survival
## Loading required package: Matrix

load("datos/snp.RData")
snp

## A SnpMatrix with 1500 rows and 439 columns
## Row names: 1 ... 1500
## Col names: 1 ... 439
```



cargamos los fenotipos

```
phenos<-read.table("datos/phenosCont.txt",header=TRUE)
head(phenos)
```

| ##   | pop  | caco |              | X1          | X2           | X3 |
|------|------|------|--------------|-------------|--------------|----|
| ## 1 | Pop1 | 1    | -0.045452947 | 0.02677151  | 0.028941220  |    |
| ## 2 | Pop1 | 0    | 0.018036264  | -0.03361612 | -0.048915896 |    |
| ## 3 | Pop1 | 1    | 0.001631087  | 0.03286500  | 0.004391887  |    |
| ## 4 | Pop1 | 0    | 0.021977998  | -0.05069528 | -0.092101747 |    |
| ## 5 | Pop1 | 0    | -0.003845641 | 0.02882352  | 0.048173898  |    |
| ## 6 | Pop1 | 0    | 0.017218303  | -0.01419022 | -0.011262438 |    |

## Análisis de asociación

```
results <- snp.rhs.tests(phenos$caco~1, snp.data=snp)
results[1:10]
```

| ##    |  | Chi.squared | Df | p.value     |
|-------|--|-------------|----|-------------|
| ## 1  |  | 10.4005458  | 1  | 0.001259781 |
| ## 2  |  | 8.3516844   | 1  | 0.003853300 |
| ## 3  |  | 2.5211014   | 1  | 0.112332103 |
| ## 4  |  | 12.1936022  | 1  | 0.000479537 |
| ## 5  |  | 0.8892616   | 1  | 0.345677499 |
| ## 6  |  | 2.2825407   | 1  | 0.130837388 |
| ## 7  |  | 0.2688703   | 1  | 0.604090611 |
| ## 8  |  | 0.4893109   | 1  | 0.484234857 |
| ## 9  |  | 4.7639765   | 1  | 0.029061334 |
| ## 10 |  | 3.2426563   | 1  | 0.071744229 |

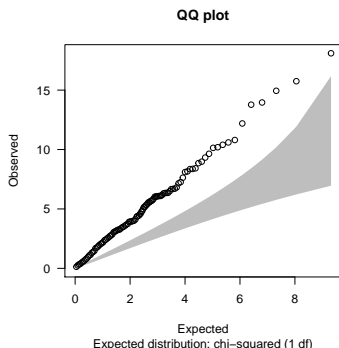
en la sintaxis se omite efecto de interes de los SNPs en la correlación.

## snpStats

El Q-Q plot compara los p-valores obtenidos con los p-valores que esperaríamos por azar

```
qq.chisq(chi.squared(results), 1)
```

```
##          N      omitted      lambda  
## 439.000000  0.000000  2.124386
```



vamos que los p-valores obtenidos son típicamente mas altos, están inflados.

# Corrección por estratificación poblacional

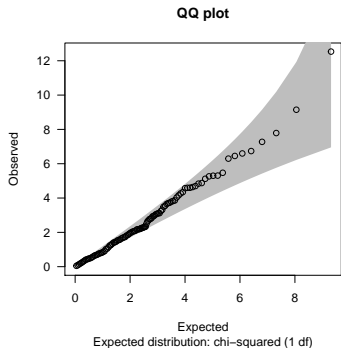
Si el fenotipo se correlaciona con la ancestría y esta es detectada por los SNPs, los valores de correlación entre fenotipo y SNPs están confundidos por la ancestría. Debemos corregir por la ancestría como una covariable.

# Corrección por estratificación poblacional

Incluimos la ancestría pop en la asociación

```
resultsAd <- snp.rhs.tests(phenos$caco~phenos$pop, snp.data=snp)  
qq.chisq(chi.squared(resultsAd), 1)
```

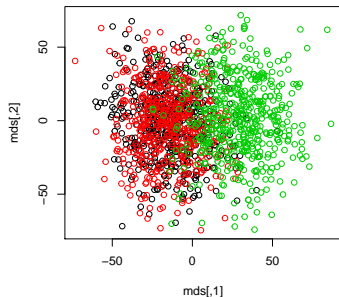
```
##           N      omitted      lambda  
## 439.000000    0.000000    1.008353
```



# Corrección por estratificación poblacional

Si no tenemos los datos de ancestría los podemos inferir de la PCA de los SNPs

```
snpnum<-matrix(as.numeric(snp),ncol=ncol(snp))  
d<-dist(snpnum, method="manhattan")  
mds<- cmdscale(d,eig=TRUE, k=2)$points  
plot(mds,col=phenos$pop)
```



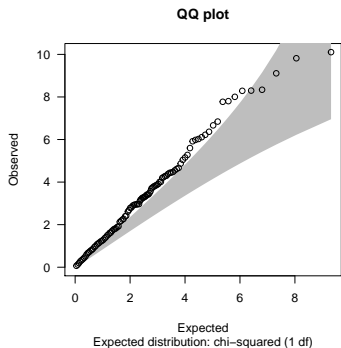
# Corrección por estratificación poblacional

Incluimos PCAs en la asociación

```
resultsAdPCA <- snp.rhs.tests(phenos$scaco~mds[,1]+mds[,2], snp.data=snp
```

```
qq.chisq(chi.squared(resultsAdPCA), 1)
```

```
##          N      omitted      lambda  
## 439.000000  0.000000  1.403764
```



## asociación en R

Para tener mas control en las asociaciones se pueden usar las funciones básicas de R como glm.

```
mod<-glm(phenos$caco~snpnum[,1]+mds[,1]+mds[,2], family="binomial")
summary(mod)
```

```
##
## Call:
## glm(formula = phenos$caco ~ snpnum[, 1] + mds[, 1] + mds[, 2],
##      family = "binomial")
##
## Deviance Residuals:
##      Min        1Q    Median        3Q        Max
## -2.3609  -1.0972   0.5874   0.9943   1.8875
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.626509   0.161892   3.870 0.000109 ***
## snpnum[, 1] -0.160589   0.077518  -2.072 0.038300 *
## mds[, 1]     0.029849   0.002320  12.864 < 2e-16 ***
## mds[, 2]     0.002336   0.002175   1.074 0.282769
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



# asociación en R

la información para la asociación del primer SNP se extrae como

```
summod<-summary(mod)
summod$coeff
```

| ## |              | Estimate     | Std. Error  | z value   | Pr(> z )     |
|----|--------------|--------------|-------------|-----------|--------------|
| ## | (Intercept)  | 0.626509309  | 0.161892007 | 3.869921  | 1.088705e-04 |
| ## | snppnum[, 1] | -0.160589175 | 0.077518112 | -2.071634 | 3.829956e-02 |
| ## | mds[, 1]     | 0.029849020  | 0.002320284 | 12.864380 | 7.141304e-38 |
| ## | mds[, 2]     | 0.002336155  | 0.002174946 | 1.074121  | 2.827686e-01 |

```
summod$coeff[2,c(1,4)]
```

| ## | Estimate    | Pr(> z )   |
|----|-------------|------------|
| ## | -0.16058917 | 0.03829956 |

# asociación en R

La información para todos los SNPs.

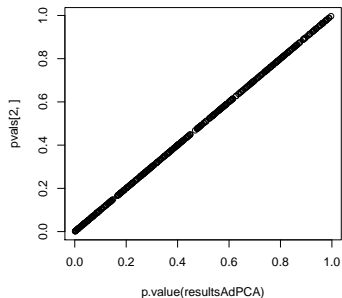
```
pvals<-sapply(1:ncol(snpnum), function(j)
{
  mod<-glm(phenos$caco~snpnum[,j]+mds[,1]+mds[,2], family="binomial")
  summod<-summary(mod)
  summod$coeff
  summod$coeff[2,c(1,4)]
})
head(t(pvals))
```

```
##           Estimate      Pr(>|z|)
## [1,] -0.16058917 0.038299564
## [2,] -0.11357896 0.139997189
## [3,] -0.09533903 0.203844762
## [4,] -0.21930524 0.004046846
## [5,]  0.03336892 0.669076533
## [6,]  0.09750542 0.225846032
```

# asociación en R

Los resultados son idénticos a los obtenidos con `snpStats`

```
plot(p.value(resultsAdPCA), pvals[2,])
```



# Genome-wide association studies

Elementos importantes a considerar.

- ▶ Usamos p-valores para identificar los SNPs que mas se correlacionan con el fenotipo
- ▶ la verdadera hipótesis del estudio es si existe *algún* SNP que se correlacione con el fenotipo
- ▶ si nos creemos SNPs con  $p < 0.05$  entonces siempre identificaríamos al rededor de 5% de asociaciones significativas, que son realmente puro azar.
- ▶ tememos que ajustar por el número de SNPs que probamos y creernos solo  $p = 0.05 / \text{numSNPs}$
- ▶ en GWAS de 1 millon de SNPs esto significa  $p < 10^{-8}$

# Ejercicio

- ▶ Estudio de asociación en código R si ajustamos por la ancestría dada por la variable `pop` en `phenos`.
- ▶ comparar con los pvalores que obtuvimos antes
- ▶ Qué podemos decir en términos de poder estadístico y falsos positivos?

# Ejercicio

```
pvalsPop<-sapply(1:ncol(snpnum), function(j)
{
  mod<-glm(phenos$caco~snpnum[,j]+phenos$pop, family="binomial")
  summod<-summary(mod)
  summod$coeff
  summod$coeff[2,c(1,4)]
})
```

# Ejercicio

```
plot(pvals[2,],pvalsPop[2,])
```

