Genetic association studies - Final exercise

Alvaro Ponce Cabrera

December 20, 2015

1 Perform a complete Genome-Wide Association Study (GWAS)

Write a little paragraph commenting the main results (1-5 lines for each point). Also include a disccusion about the findings indicating whether or not any positive association has any biological meaning.

1.1 Load the data

Firstly, the data was loaded and checked, genotypes and phenotypes individuals data had to be the same and in the same order to work with it.

```
> library(snpStats)
> # setwd("Data_for_exercises")  #set work directory
>
> geno <- read.plink("colon")   #Read the genotype data and save it in the variable geno.
> names(geno)                   #Check geno object

[1] "genotypes" "fam"       "map"

> genotypes <- geno$genotypes #Save the genotype SNP data.
> head(genotypes[,]) #Colum=SNPs  and Rows=individuals

A SnpMatrix with  6 rows and  100000 columns
Row names:  100 ... 1008
Col names:  MitoC464T ... rs7059911

> #phenotype
> feno <- read.delim("colon.txt") #Save the phenotype data
> head(feno)  #Check it

    id cascon age   smoke bmi     ev3    ev4
1  100      0  41 Current  31 -0.0007 0.0116
2 1001      0  35      Ex  NA -0.0026 0.0152
3 1004      0  50      Ex  31 -0.0007 0.0151
4 1005      1  44 Current  25  0.0002 0.0128
5 1006      1  49   Never  NA -0.0053 0.0132
6 1008      1  40   Never  24 -0.0020 0.0139

> #We need to check if the order of the individuals in genotypes and feno are the same
> identical (rownames(feno), rownames(genotypes)) #Rownames are not the same

[1] FALSE

> rownames(feno) <- feno$id #Rownames of genotypes are the IDs of individuals so lets do it for feno too
> identical (rownames(feno), rownames(genotypes)) #Now rownames are the same

[1] TRUE
```

```
> any(!rownames(feno)%in%rownames(genotypes))     #The order of individuals is the same too.

[1] FALSE

>
```

1.2 Quality control analysis

The data of the genotypes passed a quality control analysis. It's necessary to confirm that the minimum call rate and heterozygosity of individuals is not too small. The quality control of the SNPs is only checked in controls, because casos without HWE could be interesting in the following process.

```
> info.ind <- row.summary(genotypes) #QC of individuals #Save it in info
> head(info.ind)

     Call.rate Certain.calls Heterozygosity
100    0.99813             1      0.3075752
1001   0.99617             1      0.3100374
1004   0.99378             1      0.3170018
1005   0.99876             1      0.3058593
1006   0.99810             1      0.3114618
1008   0.99870             1      0.3134475

> plot(info.ind) #We see that the minimum call rate and heterozygositt is right
> info.snp <- col.summary(genotypes[feno$cascon==0,]) #QC of SNPs only taking account of controls.
> head(info.snp)

           Calls Call.rate Certain.calls       RAF          MAF         P.AA P.AB
MitoC464T   1124 0.9876977             1 0.9199288 0.0800711744 0.0800711744    0
MitoA829G   1137 0.9991213             1 0.9991205 0.0008795075 0.0008795075    0
MitoC1050T  1134 0.9964851             1 1.0000000 0.0000000000 0.0000000000    0
MitoA1738G  1132 0.9947276             1 1.0000000 0.0000000000 0.0000000000    0
MitoC2485T  1133 0.9956063             1 1.0000000 0.0000000000 0.0000000000    0
MitoC3993T  1123 0.9868190             1 0.9902048 0.0097951915 0.0097951915    0
                P.BB     z.HWE
MitoC464T  0.9199288 -33.52611
MitoA829G  0.9991205 -33.71943
MitoC1050T 1.0000000        NA
MitoA1738G 1.0000000        NA
MitoC2485T 1.0000000        NA
MitoC3993T 0.9902048 -33.51119

>
```

1.3 Association analysis, filtering and p.value calculate

In order to calculate the p-value of the association between Case-Control parameter and SNPs data, it was needed to create a filter to eliminate those controls individuals which SNPs weren't in HWE and had less than 0.01 MAF. Then, the p-value were calcualted and plotted.

```
> res<-single.snp.tests(cascon, data=feno, snp.data=genotypes) #Test of Case-Control & SNP association
> head(res) #Check res object

             N Chi.squared.1.df Chi.squared.2.df    P.1df P.2df
MitoC464T 2254        0.4500651               NA 0.502304    NA
```

```
> info.snp$pHWE <- 1 - pnorm(info.snp$z.HWE) #Transform z.value into p.value
> #Creating the filter using MAF and pHWE from controls (because Cases individuals with not HWE in
> #the SNPs could be interesting in the study). Controls without HWE are eliminated.
> filter <- info.snp$MAF > 0.01 & info.snp$pHWE > 0.001
> res.f<- res[filter,]
> head(res.f)  #Check the object

             N Chi.squared.1.df Chi.squared.2.df    P.1df P.2df
MitoC464T 2254        0.4500651               NA 0.502304    NA

> pval <- p.value(res.f, df=1) # Calculate of p.value
> head(pval) #Check the object

  MitoC464T  MitoA5657G  MitoT9717C MitoT10464C MitoC10874T MitoT12706C
  0.5023040   0.8427930   0.7651392   0.6447466   0.7675665   0.8221901

> plot(-log10(pval), col=ifelse(pval<0.0001, "red", "black")) #Plot of pvalues
```

1.4 Population stratification study. QQ-plot

Before continue, it was assesed if the population stratification were present. And it wasn't.

```
> chi<- chi.squared(res.f,df=1)
> qq.chisq(chi) #There is no population stratification

          N     omitted      lambda
9.45980e+04 0.00000e+00 9.93001e-01

>
```

1.5 Manhattan plot

Those significant SNPs that passed multiple comparisons were plotted in a Manhattan plot. 2 significant SNPs were found:”rs4733560” and ”rs10112382”. The multiple comparisons were done using Bonferroni and FDR method obtaining the same result. Then, in that moment, it was correct to say that 2 SNPs seemed to have associatiion with colon cancer.

```
> library(SNPassoc)
> library(GWASTools)
> #Bonferroni method
> p.adj.b<- p.adjust(pval,method="bonferroni") #Calculate of p.value
> #Because of the filtering,length of chromosome object (create a few lanes below)
> #is not equal of p.adj.b object, so we need to solve it filtering SNPs names in
> #chromosome map data too. We will do it again with the FDR method
> filter.chr.b<-match(names(p.adj.b),geno$map$snp.name)  #Create the filter
> chromosome <- geno$map$chromosome
> chromosome.ok<-chromosome[filter.chr.b]
> manhattanPlot(p.adj.b, chromosome.ok, signif=1e-7)
> #FDR method
> p.adj.fdr<-p.adjust(pval,method="fdr")
> filter.chr.fdr<-match(names(p.adj.fdr),geno$map$snp.name)
> chromosome <- geno$map$chromosome
> chromosome.ok<-chromosome[filter.chr.fdr]
> manhattanPlot(p.adj.fdr, chromosome.ok, signif=1e-7)
> #We can see that it's pretty similar
> head(order(p.adj.b))
```

```
[1] 48682 48681     1     2     3     4
```

```
> p.adj.b[48681]
```

```
   rs4733560
0.0002639158
```

```
> SNPs.imp2<- names(which(p.adj.fdr<0.05))
> SNPs.imp<- names(which(p.adj.b<0.05))
> SNPs.imp2
```

```
[1] "rs4733560"  "rs10112382"
```

```
> SNPs.imp
```

```
[1] "rs4733560"  "rs10112382"
```

```
> #The results using fdr and Bonferroni are the same, so we will use just one of this mehotds.
>
```

1.6 Annotation

The annotation of those significant SNPs was done using biomaRt package and Ensembl data base. The SNPs are in the 8 chromosome. The rest of the annotation information required to this practise were saved in snpInfo object.

```
> library(biomaRt)
> #Load the dataset of humaans SNPs from Ensembl.org
> mart <- useMart("ENSEMBL_MART_SNP", dataset = "hsapiens_snp", host="www.ensembl.org")
> snpInfo <- getBM(c("refsnp_id", "chr_name", "chrom_start", "allele"),
+                  filters = c("snp_filter"),
+                  values = SNPs.imp, mart = mart)
> (snpInfo)
```

```
   refsnp_id chr_name chrom_start allele
1 rs10112382        8   127772151    T/C
2  rs4733560        8   127766755    G/A
```

```
>
```

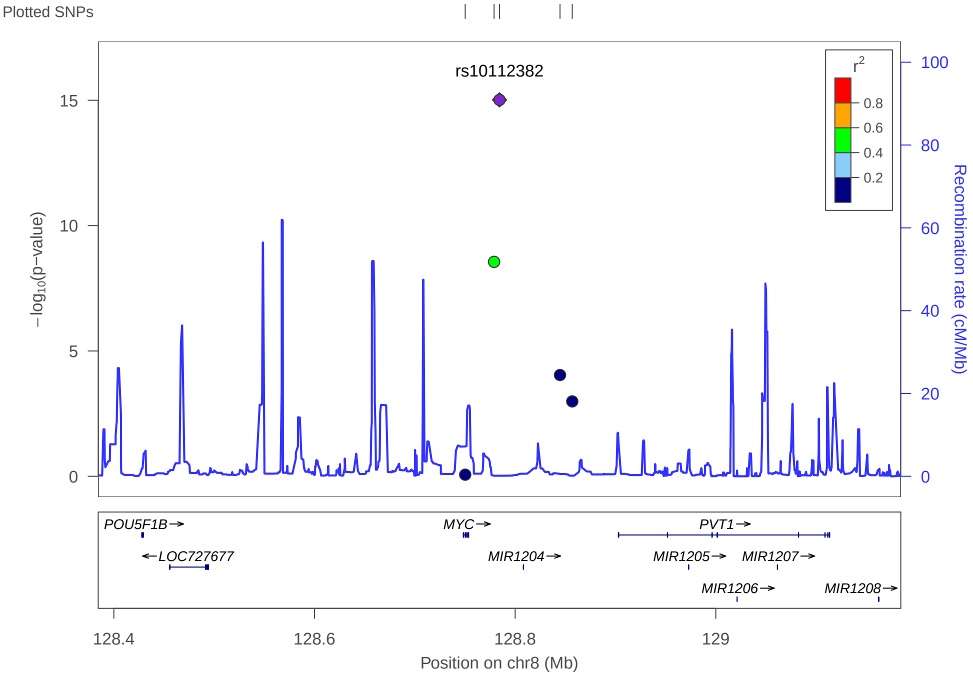
1.7 Create Locus Zoom plot

Locus zoom is a tool that allow the user to plot significant SNPs using the chromosome mapping. A candidate significant SNP is plotted into the chromosome map environment, it lets the user see close SNPs and Its possible relations. Those SNPs (the candidate and the close ones) are provide to the tool in a file exported from the data. The close SNPs are found by using a window with a determinate size.

```
> #Create a table with SNPs names, pvalues and p.adj values
> ans<- data.frame (SNP=names(res.f),
+                   pvalue=pval, bonferroni=p.adj.b)
> ans.o<-ans[order(ans$pvalue),] #Order the table
> head(ans.o)
```

```
             SNP       pvalue   bonferroni
48682 rs10112382 9.641204e-16 9.120386e-11
48681  rs4733560 2.789866e-09 2.639158e-04
22320 rs10027212 1.357062e-05 1.000000e+00
15910  rs6550962 2.462824e-05 1.000000e+00
84056 rs17769347 2.663669e-05 1.000000e+00
84055  rs5005414 3.479073e-05 1.000000e+00
```

```
> candidate <- as.character(ans.o$SNP[1]) #Save the most significant SNP in candidate
> annotation <- geno$map
> chr <- annotation[candidate, "chromosome"]
> pos <- annotation[candidate, "position"]
> size <- 100000
> #Saving the component of the mask we will use in the locus zoom plot
> mask <- annotation$chromosome == chr &
+   annotation$position > pos - size &
+   annotation$position < pos + size
> sum(mask)

[1] 6

> snps.sel <- annotation[mask, "snp.name"] #Use the mask to acces to the name of the SNPs found
> head(snps.sel)

[1] "rs4645956"  "rs4733560"  "rs10112382" "rs4733798"  "rs7815137"  "rs4332094"

> info.s<- ans[ans$SNP%in%snps.sel, 1:2] #Creation of the table we will use in the web page
> names(info.s) <- c("MarkerName", "P.value")
> head(info.s)

      MarkerName      P.value
48680  rs4645956 8.610971e-01
48681  rs4733560 2.789866e-09
48682 rs10112382 9.641204e-16
48683  rs4733798 9.106558e-05
48684  rs4332094 1.028684e-03

> write.table(info.s, file="final.snps.txt", sep="\t",
+                row.names=FALSE, quote=FALSE)  #Export the data by creating a file
> #This file should be use here
> #http://locuszoom.sph.umich.edu/locuszoom/genform.php?type=yourdata in order to
> #obtain the plot. The plot was saved as "Locus_Zoom.pdf".
>
> # openPDF("Locus_Zoom.pdf") #Look at help details in order to open it in Unix platforms
> #Anyway, the pdf with the resutl is shown below using latex package pdfpages.
>
```


Plotted SNPs
rs10112382
−log10(p−value)
0
5
10
15
Recombination rate (cM/Mb)
0
20
40
60
80
100
r2
0.8
0.6
0.4
0.2
POU5F1B→
←LOC727677
MYC→
MIR1204→
PVT1→
MIR1205→
MIR1207→
MIR1206→
MIR1208→
128.4
128.6
128.8
129
Position on chr8 (Mb)

date:	Sun Dec 20 16:58:13 2015
build:	hg19
display range:	chr8:128384397−129184397 [128384397−129184397]
hilite range:	0 − 0 [0 − 0]
reference SNP:	chr8:128784397
number of SNPs plotted:	6
max P.value:	9.64E−16 [chr8:128784397]
min P.value:	8.61E−1 [chr8:128750212]

Make more plots at http://csg.sph.umich.edu/locuszoom/

1.8 OR Stimation

About the rs10112382 SNP. It seems to be a protective allele. All the models are significants in the association study, and we can see how the appear of C allele decrease the OR value, which is 1 as maximum in T/T situation. The rs4733560 SNP has again significant p-values in all the model except the overdominant model. In this case, G/G situation has a OR value equal to 1 and the appear of A allele increase that value. In the codominant model it is 2.07.

```
> library(SNPassoc)
> SNPs.code<-as(genotypes[,SNPs.imp],"character") #To do the association study in order to see
> #the OR stimation we need the allele data of the interested SNPs
> SNPs.code.o=SNPs.code[,c(2,1)] #Reorder the data frame, most significant SNP first.
> head(SNPs.code.o)
```

```
     rs10112382 rs4733560
100  "B/B"      "B/B"
1001 "A/B"      "A/B"
1004 "A/B"      "A/B"
1005 "A/A"      "B/B"
1006 "A/B"      "A/B"
1008 "A/B"      "A/B"
```

```
> snpInfo #check the alle information
```

```
   refsnp_id chr_name chrom_start allele
1 rs10112382        8   127772151    T/C
2  rs4733560        8   127766755    G/A
```

```
> # help.search("snps") #We need information about the codification, because in the first moment
> #we dont know what means A and B
> # ?read.snps.long #This package is found searching about "snps" and inside it is found a nice
> #explication about SNPs coding. Now we know that A and B are the dirst and the second allele
> #following a alphabetic order, so we can change A and B using snpInfo information.
>
> SNPs.code.o[,1]<-gsub("A","C",SNPs.code.o[,1])
> SNPs.code.o[,1]<-gsub("B","T",SNPs.code.o[,1])
> SNPs.code.o[,2]<-gsub("A","A",SNPs.code.o[,2])
> SNPs.code.o[,2]<-gsub("B","G",SNPs.code.o[,2])
> head(SNPs.code.o)
```

```
     rs10112382 rs4733560
100  "T/T"      "G/G"
1001 "C/T"      "A/G"
1004 "C/T"      "A/G"
1005 "C/C"      "G/G"
1006 "C/T"      "A/G"
1008 "C/T"      "A/G"
```

```
> SNPs.decode=SNPs.code.o
> feno.snp<-cbind(feno,SNPs.decode ) #Put the SNPs allele data into the feno data frame
> head(feno.snp)
```

```
       id cascon age   smoke bmi     ev3    ev4 rs10112382 rs4733560
100   100      0  41 Current  31 -0.0007 0.0116        T/T       G/G
1001 1001      0  35      Ex  NA -0.0026 0.0152        C/T       A/G
1004 1004      0  50      Ex  31 -0.0007 0.0151        C/T       A/G
1005 1005      1  44 Current  25  0.0002 0.0128        C/C       G/G
1006 1006      1  49   Never  NA -0.0053 0.0132        C/T       A/G
1008 1008      1  40   Never  24 -0.0020 0.0139        C/T       A/G
```

```
> feno.s<-setupSNP(feno.snp,8:ncol(feno.snp)) #Treating of allele information before association study
> head(feno.s)

    id cascon age   smoke bmi     ev3    ev4 rs10112382 rs4733560
1  100      0  41 Current  31 -0.0007 0.0116        T/T       G/G
2 1001      0  35      Ex  NA -0.0026 0.0152        C/T       A/G
3 1004      0  50      Ex  31 -0.0007 0.0151        C/T       A/G
4 1005      1  44 Current  25  0.0002 0.0128        C/C       G/G
5 1006      1  49   Never  NA -0.0053 0.0132        C/T       A/G
6 1008      1  40   Never  24 -0.0020 0.0139        C/T       A/G

> ans <- WGassociation(cascon, feno.s) #Association study between SNPs data with cascon parameter
> WGstats(ans) #OR information

$rs10112382

SNP: rs10112382  adjusted by:
                 0    %    1    %   OR lower upper   p-value  AIC
Codominant
T/T            363 31.9  533 46.4 1.00             7.036e-15 3110
C/T            552 48.5  492 42.9 0.61  0.51  0.73
C/C            223 19.6  123 10.7 0.38  0.29  0.49
Dominant
T/T            363 31.9  533 46.4 1.00             9.893e-13 3122
C/T-C/C        775 68.1  615 53.6 0.54  0.46  0.64
Recessive
T/T-C/T        915 80.4 1025 89.3 1.00             2.531e-09 3138
C/C            223 19.6  123 10.7 0.49  0.39  0.62
Overdominant
T/T-C/C        586 51.5  656 57.1 1.00             6.692e-03 3166
C/T            552 48.5  492 42.9 0.80  0.68  0.94
log-Additive
0,1,2         1138 49.8 1148 50.2 0.61  0.54  0.69 6.892e-16 3108

$rs4733560

SNP: rs4733560  adjusted by:
                 0    %    1    %   OR lower upper   p-value  AIC
Codominant
G/G            457 40.4  342 30.1 1.00             1.922e-08 3110
A/G            525 46.5  564 49.7 1.44  1.19  1.73
A/A            148 13.1  229 20.2 2.07  1.61  2.65
Dominant
G/G            457 40.4  342 30.1 1.00             2.730e-07 3118
A/G-A/A        673 59.6  793 69.9 1.57  1.32  1.87
Recessive
G/G-A/G        982 86.9  906 79.8 1.00             5.693e-06 3123
A/A            148 13.1  229 20.2 1.68  1.34  2.10
Overdominant
G/G-A/A        605 53.5  571 50.3 1.00             1.238e-01 3142
A/G            525 46.5  564 49.7 1.14  0.97  1.34
log-Additive
0,1,2         1130 49.9 1135 50.1 1.44  1.27  1.62 2.505e-09 3108

attr(,"label.SNPs")
```

```
[1] "rs10112382" "rs4733560"
attr(,"models")
[1] 1 2 3 4 5
attr(,"quantitative")
[1] FALSE

>
```

1.9 Genetic score and evaluating of the predictive value using top-50 SNPs

Genetic score can be used for prediction of individual trait values. In this case, the genetic score of top-50 significant SNPs was calculated. The predictive value of the genecit score was evaluated by fitting a model and cheking it by a ROC curve. In this case AUC is 0.75, it means that the predictive power of the risk model is really good.

```
> head(ans.o)

             SNP       pvalue   bonferroni
48682 rs10112382 9.641204e-16 9.120386e-11
48681  rs4733560 2.789866e-09 2.639158e-04
22320 rs10027212 1.357062e-05 1.000000e+00
15910  rs6550962 2.462824e-05 1.000000e+00
84056 rs17769347 2.663669e-05 1.000000e+00
84055  rs5005414 3.479073e-05 1.000000e+00

> geno.sel<-genotypes[,as.character(ans.o$SNP[1:50])] #Here are the top 50 significant SNPs
> head(geno.sel)

A SnpMatrix with  6 rows and  50 columns
Row names:  100 ... 1008
Col names:  rs10112382 ... rs7459335

> geno.sel.df<-as.data.frame(geno.sel) #The information is saved as a data frame
> head(geno.sel.df)

     rs10112382 rs4733560 rs10027212 rs6550962 rs17769347 rs5005414 rs280768 rs6985894
100          03        03         02        02         03        03       03        02
1001         02        02         03        02         03        03       03        03
1004         02        02         02        02         02        03       03        03
1005         01        03         02        03         03        03       03        02
1006         02        02         03        03         03        03       02        01
1008         02        02         03        02         03        03       02        02
     rs10519732 rs7782875 rs4733798 rs12653807 rs9320236 rs325413 rs6806547 rs12912791
100          03        03        03         03        02       03        01         03
1001         03        03        02         03        01       02        02         02
1004         03        03        03         03        03       03        01         03
1005         03        03        01         03        02       03        03         02
1006         03        02        01         03        02       01        03         01
1008         03        03        02         03        03       02        02         02
     rs1861415 rs4358307 rs12508739 rs9965599 rs1951539 rs12918362 rs10951303 rs304343
100         02        01         02        01        02         03         03       03
1001        03        03         03        03        03         02         02       02
1004        03        02         02        01        03         03         02       03
1005        01        03         02        03        03         03         03       03
1006        02        03         03        02        03         03         03       03
1008        03        02         03        02        03         03         03       02
```

```
     rs6468379 rs984779 rs6804202 rs793891 rs139124 rs10829972 rs6545694 rs202855
100         02       03        02       03       01         01        02       03
1001        03       02        02       03       02         02        03       03
1004        03       02        03       03       02         02        03       03
1005        03       03        02       01       01         02        01       03
1006        03       01        02       03       02         02        02       02
1008        03       02        02       01       03         03        03       02
     rs4422383 rs17627811 rs12064728 rs10980253 rs3909307 rs6807414 rs1345148 rs2695674
100         02         03         01         03        03        01        02        03
1001        02         02         03         03        03        03        02        03
1004        02         02         02         03        03        03        02        03
1005        02         01         01         03        03        02        03        02
1006        03         01         01         01        03        02        03        03
1008        03         01         02         02        03        03        03        02
     rs1005066 rs1501790 rs13278529 rs4468469 rs7595749 rs20455 rs11145132 rs2651747
100         03        03         03        01        03      02         03        03
1001        03        03         03        03        03      03         03        03
1004        03        03         02        03        03      02         03        03
1005        03        03         03        01        02      02         03        01
1006        02        03         02        02        03      02         03        03
1008        02        03         02        02        02      02         03        02
     rs2649588 rs7459335
100         03        03
1001        03        02
1004        03        03
1005        03        03
1006        03        03
1008        03        03
```

```
> ff<-function(x)
+ {
+   xx<-as.numeric(x)-1
+   xx[xx<0]<-NA
+   return(xx)
+ }
> #This function is to save SNPs allele codification data corretly, it is: "0" (Minor homozigote),
> #"1" (Heterozygous), and "2" (Mayor homozigote.
>   geno.sel.numeric<- data.frame(lapply(geno.sel.df,ff)) #Apply of ff function to the top 50 SNPs data
>   head(geno.sel.numeric)
```

```
  rs10112382 rs4733560 rs10027212 rs6550962 rs17769347 rs5005414 rs280768 rs6985894
1          2         2          1         1          2         2        2         1
2          1         1          2         1          2         2        2         2
3          1         1          1         1          1         2        2         2
4          0         2          1         2          2         2        2         1
5          1         1          2         2          2         2        1         0
6          1         1          2         1          2         2        1         1
  rs10519732 rs7782875 rs4733798 rs12653807 rs9320236 rs325413 rs6806547 rs12912791
1          2         2         2          2         1        2         0          2
2          2         2         1          2         0        1         1          1
3          2         2         2          2         2        2         0          2
4          2         2         0          2         1        2         2          1
5          2         1         0          2         1        0         2          0
6          2         2         1          2         2        1         1          1
  rs1861415 rs4358307 rs12508739 rs9965599 rs1951539 rs12918362 rs10951303 rs304343
```

```
1         1         0          1         0         1          2          2        2
2         2         2          2         2         2          1          1        1
3         2         1          1         0         2          2          1        2
4         0         2          1         2         2          2          2        2
5         1         2          2         1         2          2          2        2
6         2         1          2         1         2          2          2        1
  rs6468379 rs984779 rs6804202 rs793891 rs139124 rs10829972 rs6545694 rs202855 rs4422383
1         1        2         1        2        0          0         1        2         1
2         2        1         1        2        1          1         2        2         1
3         2        1         2        2        1          1         2        2         1
4         2        2         1        0        0          1         0        2         1
5         2        0         1        2        1          1         1        1         2
6         2        1         1        0        2          2         2        1         2
  rs17627811 rs12064728 rs10980253 rs3909307 rs6807414 rs1345148 rs2695674 rs1005066
1          2          0          2         2         0         1         2         2
2          1          2          2         2         2         1         2         2
3          1          1          2         2         2         1         2         2
4          0          0          2         2         1         2         1         2
5          0          0          0         2         1         2         2         1
6          0          1          1         2         2         2         1         1
  rs1501790 rs13278529 rs4468469 rs7595749 rs20455 rs11145132 rs2651747 rs2649588
1         2          2         0         2       1          2         2         2
2         2          2         2         2       2          2         2         2
3         2          1         2         2       1          2         2         2
4         2          2         0         1       1          2         0         2
5         2          1         1         2       1          2         2         2
6         2          1         1         1       1          2         1         2
  rs7459335
1         2
2         1
3         2
4         2
5         2
6         2
```

```
>   attach(feno)
>   #Run stepwise model
>   library(MASS)
>   #Adding of cascon parameter from feno into top 50 SNPs data frame
>   geno.sel.cascon<-cbind(cascon,geno.sel.numeric)
>   # Save only no missing values
>   geno.sel.complete<-geno.sel.cascon[complete.cases(geno.sel.cascon),]
>   mod<- stepAIC(glm(cascon~.,geno.sel.complete,family = "binomial"),method="forward", trace=0)
>   summary(mod)
```

```
Call:
glm(formula = cascon ~ rs10112382 + rs10027212 + rs6550962 +
    rs17769347 + rs280768 + rs6985894 + rs7782875 + rs12653807 +
    rs9320236 + rs325413 + rs6806547 + rs1861415 + rs4358307 +
    rs9965599 + rs1951539 + rs12918362 + rs10951303 + rs304343 +
    rs6468379 + rs984779 + rs6804202 + rs793891 + rs139124 +
    rs10829972 + rs202855 + rs17627811 + rs12064728 + rs10980253 +
    rs3909307 + rs6807414 + rs1005066 + rs1501790 + rs13278529 +
    rs4468469 + rs7595749 + rs20455 + rs11145132 + rs2651747 +
    rs2649588 + rs7459335, family = "binomial", data = geno.sel.complete)
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2662  -0.9820   0.3300   0.9595   2.3512

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -4.02701    1.11922  -3.598 0.000321 ***
rs10112382   0.47008    0.07330   6.413 1.43e-10 ***
rs10027212  -0.29956    0.07887  -3.798 0.000146 ***
rs6550962    0.39796    0.11787   3.376 0.000735 ***
rs17769347   0.35313    0.10737   3.289 0.001005 **
rs280768    -0.24133    0.07830  -3.082 0.002056 **
rs6985894   -0.27716    0.07340  -3.776 0.000159 ***
rs7782875   -0.75511    0.23015  -3.281 0.001035 **
rs12653807   0.48618    0.13703   3.548 0.000388 ***
rs9320236    0.23406    0.07629   3.068 0.002155 **
rs325413     0.32399    0.08682   3.732 0.000190 ***
rs6806547   -0.28378    0.08348  -3.399 0.000675 ***
rs1861415    0.20343    0.07314   2.782 0.005411 **
rs4358307    0.30520    0.07478   4.081 4.48e-05 ***
rs9965599    0.23648    0.08577   2.757 0.005833 **
rs1951539   -0.31520    0.09874  -3.192 0.001412 **
rs12918362   0.59651    0.16514   3.612 0.000304 ***
rs10951303   0.22827    0.09078   2.515 0.011916 *
rs304343     0.47984    0.13562   3.538 0.000403 ***
rs6468379    0.19640    0.08348   2.353 0.018642 *
rs984779    -0.23723    0.07275  -3.261 0.001112 **
rs6804202    0.29220    0.07314   3.995 6.47e-05 ***
rs793891     0.25441    0.07578   3.357 0.000787 ***
rs139124     0.23224    0.07294   3.184 0.001453 **
rs10829972   0.27297    0.07185   3.799 0.000145 ***
rs202855    -0.26281    0.09681  -2.715 0.006634 **
rs17627811  -0.23083    0.07056  -3.272 0.001070 **
rs12064728   0.23322    0.07034   3.316 0.000914 ***
rs10980253   0.20367    0.08056   2.528 0.011465 *
rs3909307   -0.29650    0.10369  -2.860 0.004242 **
rs6807414    0.23109    0.07497   3.083 0.002052 **
rs1005066   -0.24980    0.09586  -2.606 0.009160 **
rs1501790    0.34106    0.15790   2.160 0.030773 *
rs13278529   0.19799    0.09841   2.012 0.044238 *
rs4468469   -0.17001    0.07375  -2.305 0.021147 *
rs7595749    0.27243    0.10676   2.552 0.010716 *
rs20455     -0.19244    0.07124  -2.701 0.006909 **
rs11145132  -0.37302    0.10552  -3.535 0.000408 ***
rs2651747    0.43369    0.12446   3.485 0.000493 ***
rs2649588   -0.33753    0.13540  -2.493 0.012675 *
rs7459335   -0.30298    0.12025  -2.520 0.011751 *
---
Signif. codes:  0 âĂŸ***âĂŹ 0.001 âĂŸ**âĂŹ 0.01 âĂŸ*âĂŹ 0.05 âĂŸ.âĂŹ 0.1 âĂŸ âĂŹ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2861.1  on 2063  degrees of freedom
```

```
Residual deviance: 2379.0  on 2023  degrees of freedom
AIC: 2461

Number of Fisher Scoring iterations: 4

>   snps.score <- names(coef(mod))[-1]
>   pos <- which(names(geno.sel.complete)%in%snps.score) #Colums of SNPs
>   library(PredictABEL)
>   score <- riskScore(mod, data=geno.sel.complete,   #Calculate of risk scores
+                      cGenPreds=pos,
+                      Type="unweighted")
>   table(score)

score
 31  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53
  1   7  16  15  32  40  71  89 102 139 196 181 229 199 178 155 130  87  82  46  34  15
 54  55  56
 13   6   1

>   mod.lin <- glm(cascon~score, geno.sel.complete, #Creation of the model to get predictive values.
+                  family="binomial")
> #Saved in mod.lin
>
> summary(mod.lin)

Call:
glm(formula = cascon ~ score, family = "binomial", data = geno.sel.complete)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.3533  -0.9544   0.3599   0.9603   2.2501

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -11.94051    0.66136  -18.05   <2e-16 ***
score         0.27120    0.01495   18.14   <2e-16 ***
---
Signif. codes:  0 âĂŸ***âĂŹ 0.001 âĂŸ**âĂŹ 0.01 âĂŸ*âĂŹ 0.05 âĂŸ.âĂŹ 0.1 âĂŸ âĂŹ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2861.1  on 2063  degrees of freedom
Residual deviance: 2415.5  on 2062  degrees of freedom
AIC: 2419.5

Number of Fisher Scoring iterations: 3

> coef(mod.lin)[2] #Odds ratio

    score
0.2712003

> predrisk <- predRisk(mod.lin, geno.sel.complete) #Predicted risk
> plotROC(data = geno.sel.complete, cOutcome = 1, predrisk = predrisk) #Roc Curve plot

AUC [95% CI] for the model 1 :  0.755 [ 0.734  -  0.775 ]
```

1.10 Pathway data analysis

In this case, after realise the pathway analysis in icsnpathway website there are no results as we can see below after the code lines.

```
> path.ann <- ans[,1:2] #Prepare the data
> head(path.ann)

           comments codominant
rs10112382        -          0
rs4733560         -          0

> write.table(path.ann, file="pvals.txt", sep="\t",  #Creation of the file for icsnpathway website.
+             row.names=FALSE, quote=FALSE,
+             col.names=FALSE)
>
>
>
```

The file .txt resulted of the pathway analysis contains the following:

```
>>Hypotheses

>>Candidate causal SNPs
Candidate causal SNP Functional class Gene Candidate causal pathway -log10(P) In LD with r2
D' -log10(P) in original GWAS

>>Candidate causal pathways
Index Candidate causal pathway  Gene set URL Description Nominal P FDR
```