GWAS practical

Andrea Manica

26 April 2018

A simple example of GWAS: genetic determinants of diabetes

We will use the library GenABEL to run GWAS (both simple, and more complex). Let's start by installing the package in your R environment. Type:

```
install.packages("GenABEL")
```

You will be asked to choose a mirror, select any that is close to you. After installation, we can load the library, so that we can use it (in the future, you won't need to reinstall the library, but you will need to load it every time you start R):

```
library("GenABEL")
```

```
## Warning: package 'GenABEL' was built under R version 3.4.4
```

```
## Loading required package: MASS
```

```
## Loading required package: GenABEL.data
```

```
## Warning: package 'GenABEL.data' was built under R version 3.4.1
```

For this tutorial, we will use data available with the library. They are simulated, keeping it simple in terms of ethical concerns for publishing data with phenotypes. We will immediately rename this dataset to something easy to remember (today, we will focus on diabetes, so let's use that as a name)

```
data(ge03d2ex)
ge03d2ex->diabetes
```

We can get a feel for the genetic data by typing:

```
head(summary(diabetes))
```

```
##           Chromosome Position Strand A1 A2 NoMeasured  CallRate        Q.2
## rs1646456          1      653      +  C  G        135 0.9926471 0.33333333
## rs4435802          1     5291      +  C  A        134 0.9852941 0.07462687
## rs946364           1     8533      -  T  C        134 0.9852941 0.27611940
## rs299251           1    10737      +  A  G        135 0.9926471 0.04444444
## rs2456488          1    11779      +  G  C        135 0.9926471 0.34814815
## rs3712159          1    12965      -  G  T        133 0.9779412 0.04511278
##           P.11 P.12 P.22    Pexact        Fmax      Plrt
## rs1646456   57   66   12 0.3323747 -0.10000000 0.2404314
## rs4435802  114   20    0 1.0000000 -0.08064516 0.2038385
## rs946364    68   58    8 0.3949055 -0.08275286 0.3302839
## rs299251   123   12    0 1.0000000 -0.04651163 0.4549295
## rs2456488   59   58   18 0.5698988  0.05343327 0.5360019
## rs3712159  122   10    1 0.2285004  0.12729659 0.2446682
```

For each SNP, we have: chromosome, map position, allele coding, number of observed genotypes, allelic frequency, genotypic distribution, P-value of the exact test for HWE, Fmax (estimate of deviation from HWE, allowing meta-analysis) and LRT P-value for HWE test.

We can summarise the genetic data with

```
descriptives.marker(diabetes)
```

```
## $`Minor allele frequency distribution`
##      X<=0.01 0.01<X<=0.05 0.05<X<=0.1 0.1<X<=0.2    X>0.2
## No   146.000      684.000     711.000    904.000 1555.000
## Prop   0.036        0.171       0.178      0.226    0.389
##
## $`Cumulative distr. of number of SNPs out of HWE, at different alpha`
##      X<=1e-04 X<=0.001 X<=0.01 X<=0.05 all X
## No     46.000   71.000 125.000 275.000  4000
## Prop    0.012    0.018   0.031   0.069     1
##
## $`Distribution of proportion of successful genotypes (per person)`
##      X<=0.9 0.9<X<=0.95 0.95<X<=0.98 0.98<X<=0.99 X>0.99
## No    1.000           0            0      135.000      0
## Prop  0.007           0            0        0.993      0
##
## $`Distribution of proportion of successful genotypes (per SNP)`
##      X<=0.9 0.9<X<=0.95 0.95<X<=0.98 0.98<X<=0.99   X>0.99
## No   37.000       6.000      996.000     1177.000 1784.000
## Prop  0.009       0.002        0.249        0.294    0.446
##
## $`Mean heterozygosity for a SNP`
## [1] 0.2582298
##
## $`Standard deviation of the mean heterozygosity for a SNP`
## [1] 0.1592255
##
## $`Mean heterozygosity for a person`
## [1] 0.2476507
##
## $`Standard deviation of mean heterozygosity for a person`
## [1] 0.04291038
```

Our data look pretty good in terms of missing data (successful genotypes by person and per SNP). However, note that the cumulative distribution of SNPs out of HWE is in excess of what we expect at the different alpha values; more specifically, we have too many sites out of HWE. This could be due to stratification, possibly as a consequence of problems with genotyping quality or population structure. We will deal with this issue in a little while.

Let's explore what phenotypic info is available in this dataset:

```
names(phdata(diabetes))
```

```
## [1] "id"     "sex"    "age"    "dm2"    "height" "weight" "diet"   "bmi"
```

(technically, we extracted the phenotypic data from the object, and listed the names)

We can now get some summaries with

```
descriptives.trait(diabetes)
```

```
##         No    Mean     SD
## id     136      NA     NA
## sex    136   0.529  0.501
## age    136  49.069 12.926
## dm2    136   0.632  0.484
```

```
## height 135 169.440  9.814
## weight 135  87.397 25.510
## diet   136   0.059  0.236
## bmi    135  30.301  8.082
```

We will focus on type 2 diabetes (dm2 in the list of traits). Let's check whether controls for that condition better match the expected null distribution. So, let's get a summary of our markers for control cases (dm2==0)

```
descriptives.marker(diabetes, ids=(phdata(diabetes)$dm2==0))
```

```
## $`Minor allele frequency distribution`
##      X<=0.01 0.01<X<=0.05 0.05<X<=0.1 0.1<X<=0.2   X>0.2
## No   233.000      676.000     671.000    898.000 1522.00
## Prop   0.058        0.169       0.168      0.224    0.38
##
## $`Cumulative distr. of number of SNPs out of HWE, at different alpha`
##      X<=1e-04 X<=0.001 X<=0.01 X<=0.05 all X
## No          0    3.000  14.000  98.000  4000
## Prop        0    0.001   0.003   0.024     1
##
## $`Distribution of proportion of successful genotypes (per person)`
##      X<=0.9 0.9<X<=0.95 0.95<X<=0.98 0.98<X<=0.99 X>0.99
## No        0           0            0           50      0
## Prop      0           0            0            1      0
##
## $`Distribution of proportion of successful genotypes (per SNP)`
##      X<=0.9 0.9<X<=0.95 0.95<X<=0.98 0.98<X<=0.99   X>0.99
## No   37.000      49.000     1523.000            0 2391.000
## Prop  0.009       0.012        0.381            0    0.598
##
## $`Mean heterozygosity for a SNP`
## [1] 0.2555009
##
## $`Standard deviation of the mean heterozygosity for a SNP`
## [1] 0.1618707
##
## $`Mean heterozygosity for a person`
## [1] 0.252572
##
## $`Standard deviation of mean heterozygosity for a person`
## [1] 0.04714886
```

This looks promising. The controls are at HWE, suggesting that the deviation was due to stratification within the cases. Let's quickly confirm that the cases are not at HWE. We can get just table 2 from the summary by subsetting the output:

```
descriptives.marker(diabetes, ids=(phdata(diabetes)$dm2==1)) [2]
```

```
## $`Cumulative distr. of number of SNPs out of HWE, at different alpha`
##      X<=1e-04 X<=0.001 X<=0.01 X<=0.05 all X
## No     45.000    79.00 136.000 268.000  4000
## Prop    0.011     0.02   0.034   0.067     1
```

Now that we have explored our data, let's run a first, very simple GWAS:

```
dm2.simple.gwas<- qtscore(dm2, diabetes, trait="binomial")
```

We can see the results of our analysis by calling the object;

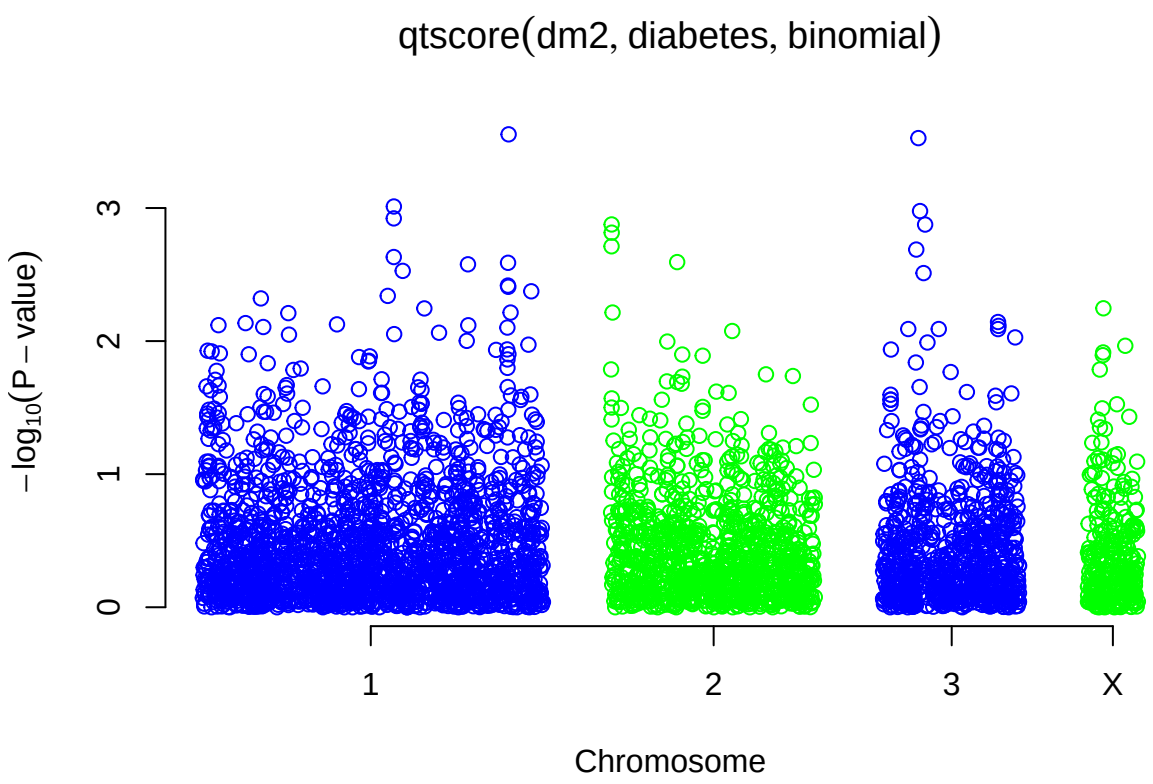
```
dm2.simple.gwas
```

```
## ***** 'scan.gwaa' object *****
## *** Produced with:
## qtscore(formula = dm2, data = diabetes, trait.type = "binomial")
## *** Test used: binomial
## *** no. IDs used: 136 ( id199 id287 id300 , ... )
## *** Lambda: 1.033102
## *** Results table contains 4000 rows and 9 columns
## *** Output for 10 first rows is:
##             N      effB    se_effB     chi2.1df      P1df     effAB
## rs1646456 135 0.9487666  5.0501959 0.0352941176 0.8509807 1.5558824
## rs4435802 134 2.6822601  1.6765631 2.5595403237 0.1096305 2.5142857
## rs946364  134 0.6376645  0.4012104 2.5260391714 0.1119810 0.7277883
## rs299251  135 0.5592122  0.5740215 0.9490674319 0.3299568 0.5569620
## rs2456488 135 0.8669860  1.5393112 0.3172278551 0.5732784 0.9736842
## rs3712159 133 0.8282737  2.4803134 0.1115153279 0.7384255 0.5641026
## rs4602970 136 1.5227297  2.2683336 0.4506421219 0.5020302 1.5131579
## rs175910  134 0.9949826 57.3970087 0.0003005055 0.9861693 0.5600000
## rs1919938 136 0.9303079  3.5515057 0.0686164762 0.7933619 0.1160000
## rs8892781 133 1.0953022 14.9055712 0.0053997133 0.9414220 1.0952381
##               effBB    chi2.2df       P2df
## rs1646456 0.4831933 3.885667042 0.14329734
## rs4435802        NA 2.559540324 0.10963046
## rs946364  0.2869565 3.020970009 0.22080286
## rs299251         NA 0.949067432 0.32995679
## rs2456488 0.6907895 0.493411146 0.78137072
## rs3712159       Inf 1.358996877 0.50687116
## rs4602970        NA 0.450642122 0.50203018
## rs175910  4.0727273 5.012993241 0.08155345
## rs1919938 0.1958333 6.001763944 0.04974318
## rs8892781        NA 0.005399713 0.94142198
##    ...
## ___ Use 'results(object)' to get complete results table ___
```

We have some information on the number of individual and SNPs, as well as a summary of results for the first ten SNPs. effB corresponds to the (approximate) Odds Ratio estimate for the SNP, and chi2.1df and P1df give the appropriate test. Further in the table, you have equivalent information for a test in which we consider specific genotypes (we won't worry about it in this practical, as in many cases we don't have enough instances of each possible genotype combination for proper testing).

So, let's plot our hits as a Manhattan plot

```
plot(dm2.simple.gwas)
```



Earlier on, we did a coarse check for stratification in the data by checking HWE in controls. A more formal test of the extent to which our data match the assumption of no stratification in the data is to estimate lambda, which give the degree of inflation (if lambda>1) or deflation (if lambda<1) in the distribution of deviations from the null distribution. Looking back at the output from the summary of our gwas object:

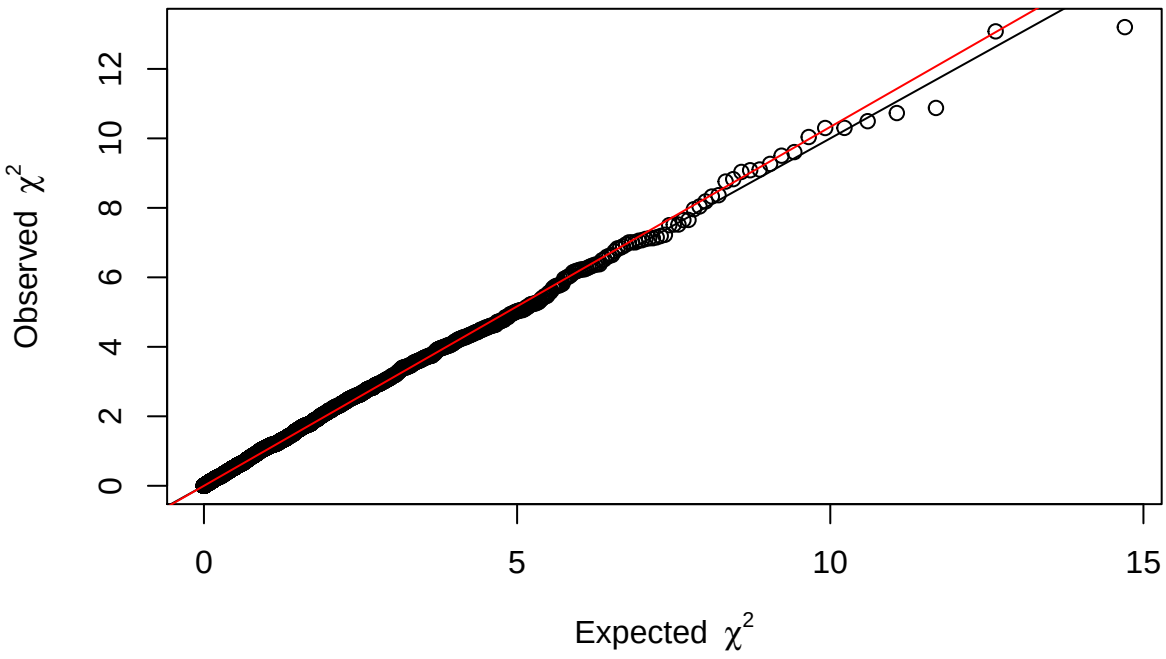
```
dm2.simple.gwas
```

```
## ***** 'scan.gwaa' object *****
## *** Produced with:
## qtscore(formula = dm2, data = diabetes, trait.type = "binomial")
## *** Test used: binomial
## *** no. IDs used: 136 ( id199 id287 id300 , ... )
## *** Lambda: 1.033102
## *** Results table contains 4000 rows and 9 columns
## *** Output for 10 first rows is:
##             N      effB    se_effB     chi2.1df      P1df     effAB
## rs1646456 135 0.9487666  5.0501959 0.0352941176 0.8509807 1.5558824
## rs4435802 134 2.6822601  1.6765631 2.5595403237 0.1096305 2.5142857
## rs946364  134 0.6376645  0.4012104 2.5260391714 0.1119810 0.7277883
## rs299251  135 0.5592122  0.5740215 0.9490674319 0.3299568 0.5569620
## rs2456488 135 0.8669860  1.5393112 0.3172278551 0.5732784 0.9736842
## rs3712159 133 0.8282737  2.4803134 0.1115153279 0.7384255 0.5641026
## rs4602970 136 1.5227297  2.2683336 0.4506421219 0.5020302 1.5131579
## rs175910  134 0.9949826 57.3970087 0.0003005055 0.9861693 0.5600000
## rs1919938 136 0.9303079  3.5515057 0.0686164762 0.7933619 0.1160000
## rs8892781 133 1.0953022 14.9055712 0.0053997133 0.9414220 1.0952381
##               effBB    chi2.2df       P2df
```

```
## rs1646456 0.4831933 3.885667042 0.14329734
## rs4435802        NA 2.559540324 0.10963046
## rs946364  0.2869565 3.020970009 0.22080286
## rs299251         NA 0.949067432 0.32995679
## rs2456488 0.6907895 0.493411146 0.78137072
## rs3712159       Inf 1.358996877 0.50687116
## rs4602970        NA 0.450642122 0.50203018
## rs175910  4.0727273 5.012993241 0.08155345
## rs1919938 0.1958333 6.001763944 0.04974318
## rs8892781        NA 0.005399713 0.94142198
##    ...
## ___ Use 'results(object)' to get complete results table ___
```

We can see that lambda has been estimated as 1.033. Whilst this number might seem small, lambda increases linearly with sample size, so given our small sample, the deviation is somewhat concerning. We can visualize that deviation by plotting

```
estlambda(dm2.simple.gwas[, "P1df"], plot=TRUE)
```




```
## $estimate
## [1] 1.033102
##
## $se
## [1] 0.0005639231
```

A lambda=1 would imply a 1:1 relationship between observed and expected chisquare values, but we can see that the relationship gets a bit messy at large values. It is possible to correct for this inflation, and that is what we have in the last column of the results section from the summary of the gwas object:

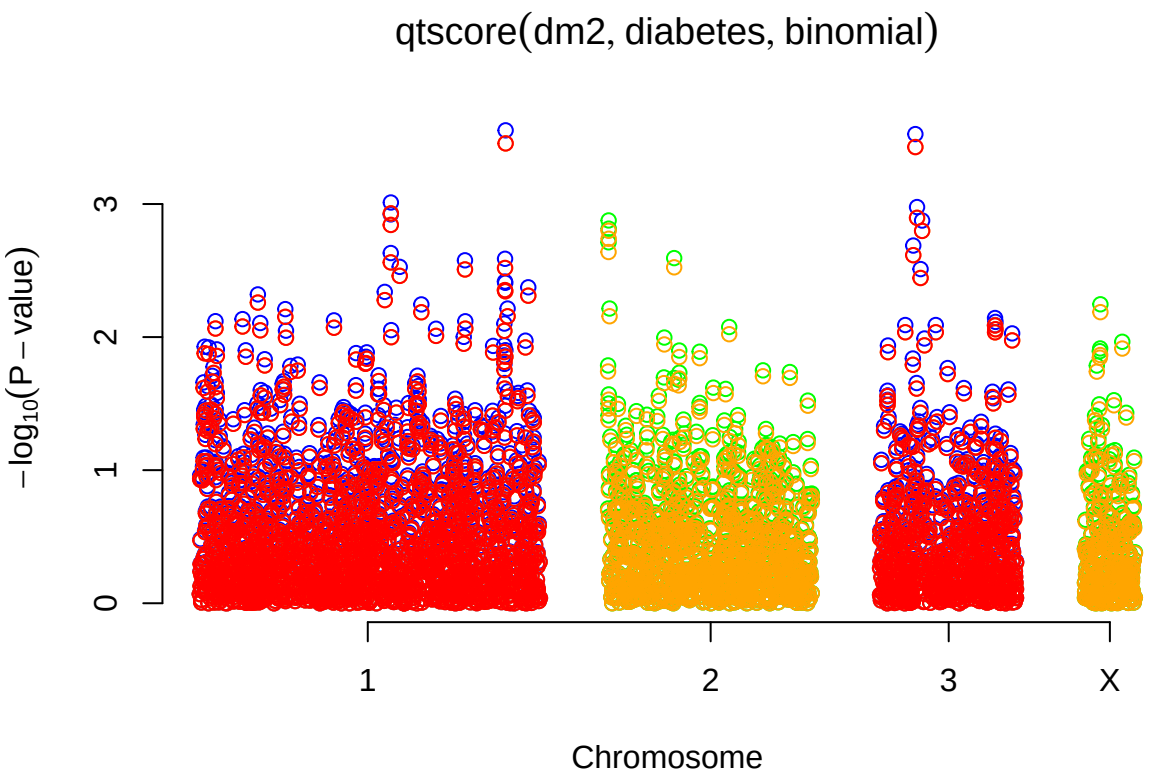
```
summary(dm2.simple.gwas)
```

```
## Summary for top 10 results, sorted by P1df
```

```
##           Chromosome Position Strand A1 A2   N       effB    se_effB
## rs1719133          1  4495479      +  T  A 136 0.33729339 0.09282784
## rs2975760          3 10518480      +  A  T 134 3.80380024 1.05172986
## rs7418878          1  2808520      +  A  T 136 3.08123060 0.93431795
## rs5308595          3 10543128      -  C  G 133 3.98254950 1.21582875
## rs4804634          1  2807417      +  C  G 132 0.43411456 0.13400290
## rs3224311          2  6009769      +  G  C 135 3.15831710 0.98401491
## rs26325            3 10617781      +  A  C 135 0.09742793 0.03035964
## rs8835506          2  6010852      +  A  T 132 3.17720829 1.00274087
## rs3925525          2  6008501      +  C  G 135 2.98416931 0.96286458
## rs2521089          3 10487652      -  T  C 135 2.50239493 0.81179595
##            chi2.1df         P1df     effAB     effBB  chi2.2df
## rs1719133 13.202591 0.0002795623 0.4004237  0.000000 14.729116
## rs2975760 13.080580 0.0002983731 3.4545455 10.000000 13.547345
## rs7418878 10.875745 0.0009743183 3.6051282  4.871795 12.181064
## rs5308595 10.729452 0.0010544366 3.3171429       Inf 10.766439
## rs4804634 10.494949 0.0011970132 0.5240642  0.173913 11.200767
## rs3224311 10.301681 0.0013290907 3.4151786  4.250000 11.658283
## rs26325   10.298495 0.0013313876 0.1097724        NA 10.298495
## rs8835506 10.039543 0.0015321522 3.4903846  4.125000 11.513206
## rs3925525  9.605423 0.0019400358 3.2380952  4.121212 10.782867
## rs2521089  9.502064 0.0020524092 2.5717703  4.772727  9.933387
##                   P2df        Pc1df
## rs1719133 0.0006333052 0.0003504258
## rs2975760 0.0011434877 0.0003732694
## rs7418878 0.0022642036 0.0011762545
## rs5308595 0.0045930101 0.0012699705
## rs4804634 0.0036964462 0.0014362332
## rs3224311 0.0029405999 0.0015897278
## rs26325   0.0013313876 0.0015923930
## rs8835506 0.0031618340 0.0018248521
## rs3925525 0.0045554384 0.0022944719
## rs2521089 0.0069661425 0.0024233145
```

Column Pc1df gives us the adjusted probabilities accounting for the measured lambda. Let's replot our hits and then superimpose the adjusted p values in a different colour

```
plot(dm2.simple.gwas)
add.plot(dm2.simple.gwas,df="Pc1df",col=c("red","orange"))
```



As you can see, the adjusted p-values are generally lower than the raw values (as one would expect), but not by a huge amount (again, as expected given the not overly large deviation from lambda equal to 1). Whilst this statistical correction attempts to reshape the distribution of chisquare2 values to the expected one, it is only effective at fixing small levels of stratification. We will see in the next section of this practical how to deal with more substantial stratification. We can now get a list of the top hits of our GWAS with

```
descriptives.scan(dm2.simple.gwas, sort="Pc1df")
```

```
## Summary for top 10 results, sorted by Pc1df

##           Chromosome Position Strand A1 A2   N       effB    se_effB
## rs1719133          1  4495479      +  T  A 136 0.33729339 0.09282784
## rs2975760          3 10518480      +  A  T 134 3.80380024 1.05172986
## rs7418878          1  2808520      +  A  T 136 3.08123060 0.93431795
## rs5308595          3 10543128      -  C  G 133 3.98254950 1.21582875
## rs4804634          1  2807417      +  C  G 132 0.43411456 0.13400290
## rs3224311          2  6009769      +  G  C 135 3.15831710 0.98401491
## rs26325            3 10617781      +  A  C 135 0.09742793 0.03035964
## rs8835506          2  6010852      +  A  T 132 3.17720829 1.00274087
## rs3925525          2  6008501      +  C  G 135 2.98416931 0.96286458
## rs2521089          3 10487652      -  T  C 135 2.50239493 0.81179595
##            chi2.1df         P1df     effAB     effBB  chi2.2df
## rs1719133 13.202591 0.0002795623 0.4004237  0.000000 14.729116
## rs2975760 13.080580 0.0002983731 3.4545455 10.000000 13.547345
## rs7418878 10.875745 0.0009743183 3.6051282  4.871795 12.181064
## rs5308595 10.729452 0.0010544366 3.3171429       Inf 10.766439
## rs4804634 10.494949 0.0011970132 0.5240642  0.173913 11.200767
```

```
## rs3224311 10.301681 0.0013290907 3.4151786  4.250000 11.658283
## rs26325   10.298495 0.0013313876 0.1097724        NA 10.298495
## rs8835506 10.039543 0.0015321522 3.4903846  4.125000 11.513206
## rs3925525  9.605423 0.0019400358 3.2380952  4.121212 10.782867
## rs2521089  9.502064 0.0020524092 2.5717703  4.772727  9.933387
##                  P2df        Pc1df
## rs1719133 0.0006333052 0.0003504258
## rs2975760 0.0011434877 0.0003732694
## rs7418878 0.0022642036 0.0011762545
## rs5308595 0.0045930101 0.0012699705
## rs4804634 0.0036964462 0.0014362332
## rs3224311 0.0029405999 0.0015897278
## rs26325   0.0013313876 0.0015923930
## rs8835506 0.0031618340 0.0018248521
## rs3925525 0.0045554384 0.0022944719
## rs2521089 0.0069661425 0.0024233145
```

Data quality can influence a lot your hits. The GenABEL package has a user friendly function that performs a few passes on the data until some basic quality checks all pass. There are a number of quality threshold switches that can be tweaked, and they should be tailored to the data that you are using (generally a function of the technology used to generate the data, Olivier covered these issues on the first day). In this case, let's run the quality control routine with defaults, except for ignoring HWE deviations (we want to explore what qc filtering does to it):

```
diabetes.qc<-check.marker(diabetes, p.level=0)
```

```
## Excluding people/markers with extremely low call rate...
## 4000 markers and 136 people in total
## 0 people excluded because of call rate < 0.1
## 6 markers excluded because of call rate < 0.1
## Passed: 3994 markers and 136 people
##
## Running sex chromosome checks...
## 197 heterozygous X-linked male genotypes found
## 1 X-linked markers are likely to be autosomal (odds > 1000 )
## 2 male are likely to be female (odds > 1000 )
## 0 female are likely to be male (odds > 1000 )
## 0 people have intermediate X-chromosome inbreeding (0.5 > F > 0.5)
## If these people/markers are removed, 0 heterozygous male genotypes are left
## Passed: 3993 markers and 134 people
##
## no X/Y/mtDNA-errors to fix
##
##
## RUN 1
## 3993 markers and 134 people in total
## 304 (7.613323%) markers excluded as having low (<1.865672%) minor allele frequency
## 36 (0.9015778%) markers excluded because of low (<95%) call rate
## 0 (0%) markers excluded because they are out of HWE (P <0)
## 1 (0.7462687%) people excluded because of low (<95%) call rate
## Mean autosomal HET is 0.2747262 (s.e. 0.03721277)
## 3 (2.238806%) people excluded because too high autosomal heterozygosity (FDR <1%)
## Excluded people had HET >= 0.4856887
## Mean IBS is 0.7730888 (s.e. 0.02031405), as based on 2000 autosomal markers
## 2 (1.492537%) people excluded because of too high IBS (>=0.95)
```

```
## In total, 3653 (91.4851%) markers passed all criteria
## In total, 128 (95.52239%) people passed all criteria
##
## RUN 2
## 3653 markers and 128 people in total
## 80 (2.189981%) markers excluded as having low (<1.953125%) minor allele frequency
## 0 (0%) markers excluded because of low (<95%) call rate
## 0 (0%) markers excluded because they are out of HWE (P <0)
## 0 (0%) people excluded because of low (<95%) call rate
## Mean autosomal HET is 0.2748341 (s.e. 0.01695461)
## 0 people excluded because too high autosomal heterozygosity (FDR <1%)
## Mean IBS is 0.7684512 (s.e. 0.01783107), as based on 2000 autosomal markers
## 0 (0%) people excluded because of too high IBS (>=0.95)
## In total, 3573 (97.81002%) markers passed all criteria
## In total, 128 (100%) people passed all criteria
##
## RUN 3
## 3573 markers and 128 people in total
## 0 (0%) markers excluded as having low (<1.953125%) minor allele frequency
## 0 (0%) markers excluded because of low (<95%) call rate
## 0 (0%) markers excluded because they are out of HWE (P <0)
## 0 (0%) people excluded because of low (<95%) call rate
## Mean autosomal HET is 0.2748341 (s.e. 0.01695461)
## 0 people excluded because too high autosomal heterozygosity (FDR <1%)
## Mean IBS is 0.7691005 (s.e. 0.01786321), as based on 2000 autosomal markers
## 0 (0%) people excluded because of too high IBS (>=0.95)
## In total, 3573 (100%) markers passed all criteria
## In total, 128 (100%) people passed all criteria
```

As you can see, each pass removed some markers/individuals, until the last pass had no further problematic instances. We can get a summary of our clean up with:

```
summary(diabetes.qc)
```

```
## $`Per-SNP fails statistics`
##           NoCall NoMAF NoHWE Redundant Xsnpfail
## NoCall        42     0     0         0        0
## NoMAF         NA   384     0         0        0
## NoHWE         NA    NA     0         0        0
## Redundant     NA    NA    NA         0        0
## Xsnpfail      NA    NA    NA        NA        1
##
## $`Per-person fails statistics`
##             IDnoCall HetFail IBSFail isfemale ismale isXXY otherSexErr
## IDnoCall           1       0       0        0      0     0           0
## HetFail           NA       3       0        0      0     0           0
## IBSFail           NA      NA       2        0      0     0           0
## isfemale          NA      NA      NA        2      0     0           0
## ismale            NA      NA      NA       NA      0     0           0
## isXXY             NA      NA      NA       NA     NA     0           0
## otherSexErr       NA      NA      NA       NA     NA    NA           0
```

We can now generate a clean dataset as

```
diabetes.clean<-diabetes[diabetes.qc$idok, diabetes.qc$snpok]
```

We can remove any cases of heterozygosity on the X chromosome in males with

```
diabetes.clean<-Xfix(diabetes.clean)
```

```
## no X/Y/mtDNA-errors to fix
```

Let's see how we are doing with HWE now:

```
descriptives.marker(diabetes.clean)[2]
```

```
## $`Cumulative distr. of number of SNPs out of HWE, at different alpha`
##      X<=1e-04 X<=0.001 X<=0.01 X<=0.05 all X
## No     44.000   66.000 117.000 239.000  3573
## Prop    0.012    0.018   0.033   0.067     1
```

We still have an excess of SNPs out of HWE, not much better than the original dataset:

```
descriptives.marker(diabetes)[2]
```

```
## $`Cumulative distr. of number of SNPs out of HWE, at different alpha`
##      X<=1e-04 X<=0.001 X<=0.01 X<=0.05 all X
## No     46.000   71.000 125.000 275.000  4000
## Prop    0.012    0.018   0.031   0.069     1
```

The next step is to investigate sub-structure in the dataset. We start by building a matrix of genome-wide IBD:

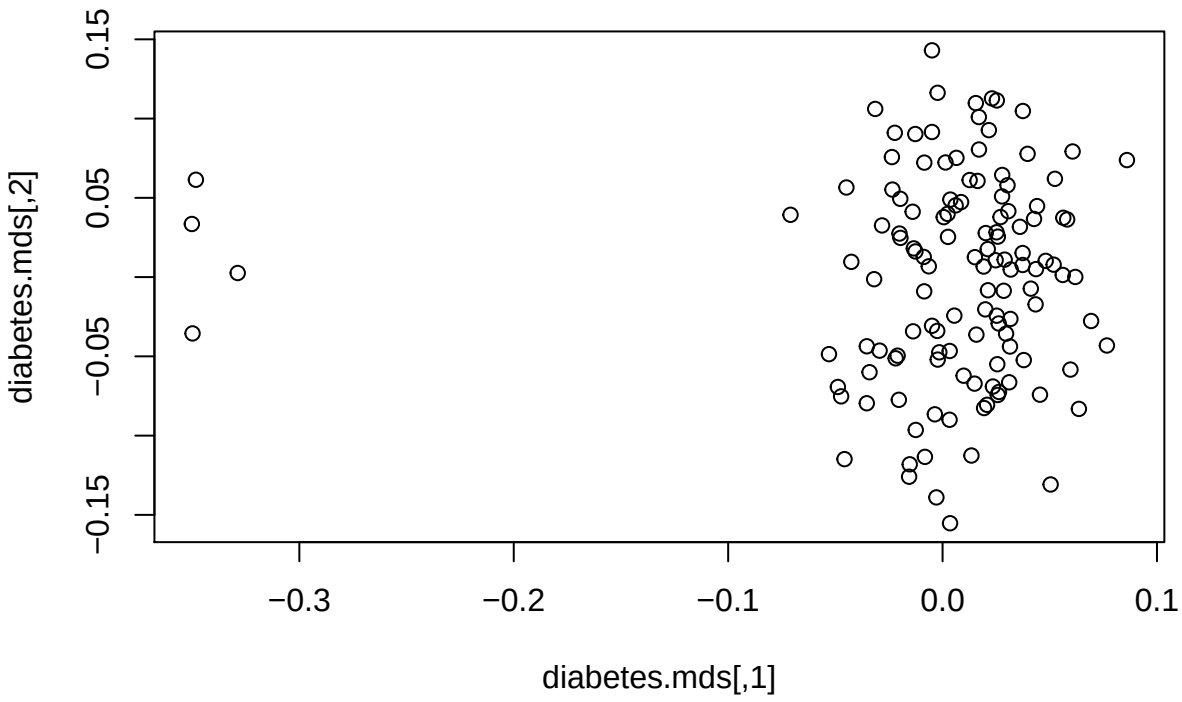
```
diabetes.ibd<-ibs(diabetes.clean[,autosomal(diabetes.clean)],weight="freq")
```

We can now use this IBD matrix to perform MDS (after transforming it into a distance matrix)

```
diabetes.mds<-cmdscale(as.dist(0.5-diabetes.ibd))
```

Plot the first two dimensions of our MDS:

```
plot(diabetes.mds)
```



We can spot that there are four individuals that form a cluster separated from the other data (as highlighted by the first dimension). We can formally identify these two clusters using a clustering algorithm:

```
diabetes.kclust<-kmeans(diabetes.mds,centers=2,nstart=100)
```

Let's figure out which cluster is the smaller one (i.e. the one with the outliers), and get the labels for those individuals:

```
outlier.cluster<- which.min(table(diabetes.kclust$cluster))
outliers<-names(which(diabetes.kclust$cluster==outlier.cluster))
```

We can now remove these outliers with

```
diabetes.clean2<-diabetes.clean[names(which(diabetes.kclust$cluster!=outlier.cluster)),]
```

Now,let's check that we have improved our HWE:

```
descriptives.marker(diabetes.clean2)[2]
```

```
## $`Cumulative distr. of number of SNPs out of HWE, at different alpha`
##      X<=1e-04 X<=0.001 X<=0.01 X<=0.05 all X
## No          1    2.000   2e+01 102.000  3573
## Prop        0    0.001   6e-03   0.029     1
```

Great, we now have decreased the SNPs out of HWE, showing that the excess was due to those outliers. For good measure, we should do a final clean up of the data, allowing for HWE checking

```
diabetes.qc2<-check.marker(diabetes.clean2)
```

```
## Excluding people/markers with extremely low call rate...
## 3573 markers and 124 people in total
```

```
## 0 people excluded because of call rate < 0.1
## 0 markers excluded because of call rate < 0.1
## Passed: 3573 markers and 124 people
##
## Running sex chromosome checks...
## 0 heterozygous X-linked male genotypes found
## 0 X-linked markers are likely to be autosomal (odds > 1000 )
## 0 male are likely to be female (odds > 1000 )
## 0 female are likely to be male (odds > 1000 )
## 0 people have intermediate X-chromosome inbreeding (0.5 > F > 0.5)
## If these people/markers are removed, 0 heterozygous male genotypes are left
## Passed: 3573 markers and 124 people
##
## no X/Y/mtDNA-errors to fix
##
##
## RUN 1
## 3573 markers and 124 people in total
## 40 (1.119507%) markers excluded as having low (<2.016129%) minor allele frequency
## 0 (0%) markers excluded because of low (<95%) call rate
## 1 (0.02798769%) markers excluded because they are out of HWE (FDR <0.2)
## 0 (0%) people excluded because of low (<95%) call rate
## Mean autosomal HET is 0.2780246 (s.e. 0.01642372)
## 0 people excluded because too high autosomal heterozygosity (FDR <1%)
## Mean IBS is 0.7697998 (s.e. 0.01256573), as based on 2000 autosomal markers
## 0 (0%) people excluded because of too high IBS (>=0.95)
## In total, 3532 (98.8525%) markers passed all criteria
## In total, 124 (100%) people passed all criteria
##
## RUN 2
## 3532 markers and 124 people in total
## 0 (0%) markers excluded as having low (<2.016129%) minor allele frequency
## 0 (0%) markers excluded because of low (<95%) call rate
## 0 (0%) markers excluded because they are out of HWE (FDR <0.2)
## 0 (0%) people excluded because of low (<95%) call rate
## Mean autosomal HET is 0.2780246 (s.e. 0.01642372)
## 0 people excluded because too high autosomal heterozygosity (FDR <1%)
## Mean IBS is 0.7706539 (s.e. 0.01253032), as based on 2000 autosomal markers
## 0 (0%) people excluded because of too high IBS (>=0.95)
## In total, 3532 (100%) markers passed all criteria
## In total, 124 (100%) people passed all criteria
```

```
diabetes.clean3<-diabetes.clean2[diabetes.qc2$idok, diabetes.qc2$snpok]
```

Now let's rerun the GWAS analysis with the clean dataset:

```
dm2.clean3.gwas<- qtscore(dm2, diabetes.clean3, trait="binomial")
```

And finally we can compare the top hits with this clean dataset compared to the original data:

```
descriptives.scan(dm2.clean3.gwas, sort="Pc1df")
```

```
## Summary for top 10 results, sorted by Pc1df
```

```
##           Chromosome Position Strand A1 A2   N      effB    se_effB
## rs1719133          1  4495479      +  T  A 124 0.3167801 0.08614528
## rs4804634          1  2807417      +  C  G 121 0.4119844 0.12480696
```

```
## rs8835506          2  6010852      +  A  T 121 3.5378209 1.08954331
## rs4534929          1  4474374      +  C  G 123 0.4547151 0.14160410
## rs1013473          1  4487262      +  A  T 124 2.7839368 0.86860745
## rs3925525          2  6008501      +  C  G 124 3.2807631 1.03380675
## rs3224311          2  6009769      +  G  C 124 3.2807631 1.03380675
## rs2975760          3 10518480      +  A  T 123 3.1802120 1.00916993
## rs2521089          3 10487652      -  T  C 123 2.7298775 0.87761175
## rs1048031          1  4485591      +  G  T 122 0.4510793 0.14548378
##            chi2.1df         P1df     effAB     effBB  chi2.2df
## rs1719133 13.522368 0.0002357368 0.3740771 0.0000000 14.677906
## rs4804634 10.896423 0.0009635013 0.6315789 0.1739130 12.375590
## rs8835506 10.543448 0.0011660066 4.0185185 4.0185185 12.605556
## rs4534929 10.311626 0.0013219476 0.4830918 0.1739130 10.510272
## rs1013473 10.272393 0.0013503553 3.0495868 5.8441558 10.926296
## rs3925525 10.070964 0.0015062424 3.6923077 4.0000000 11.765985
## rs3224311 10.070964 0.0015062424 3.6923077 4.0000000 11.765985
## rs2975760  9.930784 0.0016253728 3.0000000 8.0000000 10.172522
## rs2521089  9.675679 0.0018672326 3.0147059 5.0000000 10.543296
## rs1048031  9.613391 0.0019316360 0.4844720 0.1714286  9.965696
##                  P2df        Pc1df
## rs1719133 0.0006497303 0.0003051386
## rs4804634 0.0020543516 0.0011894974
## rs8835506 0.0018312105 0.0014303562
## rs4534929 0.0052206352 0.0016148745
## rs1013473 0.0042401869 0.0016484082
## rs3925525 0.0027864347 0.0018320223
## rs3224311 0.0027864347 0.0018320223
## rs2975760 0.0061810866 0.0019719163
## rs2521089 0.0051351403 0.0022549178
## rs1048031 0.0068545128 0.0023300667
```

```
descriptives.scan(dm2.simple.gwas, sort="Pc1df")
```

```
## Summary for top 10 results, sorted by Pc1df
```

```
##           Chromosome Position Strand A1 A2   N       effB    se_effB
## rs1719133          1  4495479      +  T  A 136 0.33729339 0.09282784
## rs2975760          3 10518480      +  A  T 134 3.80380024 1.05172986
## rs7418878          1  2808520      +  A  T 136 3.08123060 0.93431795
## rs5308595          3 10543128      -  C  G 133 3.98254950 1.21582875
## rs4804634          1  2807417      +  C  G 132 0.43411456 0.13400290
## rs3224311          2  6009769      +  G  C 135 3.15831710 0.98401491
## rs26325            3 10617781      +  A  C 135 0.09742793 0.03035964
## rs8835506          2  6010852      +  A  T 132 3.17720829 1.00274087
## rs3925525          2  6008501      +  C  G 135 2.98416931 0.96286458
## rs2521089          3 10487652      -  T  C 135 2.50239493 0.81179595
##            chi2.1df         P1df     effAB     effBB  chi2.2df
## rs1719133 13.202591 0.0002795623 0.4004237  0.000000 14.729116
## rs2975760 13.080580 0.0002983731 3.4545455 10.000000 13.547345
## rs7418878 10.875745 0.0009743183 3.6051282  4.871795 12.181064
## rs5308595 10.729452 0.0010544366 3.3171429       Inf 10.766439
## rs4804634 10.494949 0.0011970132 0.5240642  0.173913 11.200767
## rs3224311 10.301681 0.0013290907 3.4151786  4.250000 11.658283
## rs26325   10.298495 0.0013313876 0.1097724        NA 10.298495
## rs8835506 10.039543 0.0015321522 3.4903846  4.125000 11.513206
```

```
## rs3925525  9.605423 0.0019400358 3.2380952  4.121212 10.782867
## rs2521089  9.502064 0.0020524092 2.5717703  4.772727  9.933387
##                   P2df        Pc1df
## rs1719133 0.0006333052 0.0003504258
## rs2975760 0.0011434877 0.0003732694
## rs7418878 0.0022642036 0.0011762545
## rs5308595 0.0045930101 0.0012699705
## rs4804634 0.0036964462 0.0014362332
## rs3224311 0.0029405999 0.0015897278
## rs26325   0.0013313876 0.0015923930
## rs8835506 0.0031618340 0.0018248521
## rs3925525 0.0045554384 0.0022944719
## rs2521089 0.0069661425 0.0024233145
```

We can easily ask how many of our top 10 SNPs were in the original list

```
sum(rownames(descriptives.scan(dm2.clean3.gwas, sort="Pc1df")[2]) %in% rownames(descriptives.scan(dm2.s
```

```
## Summary for top 10 results, sorted by Pc1df
## Summary for top 10 results, sorted by Pc1df
```

```
## [1] 7
```

Only 7 out 10, so cleaning your data can change things quite a bit. The final step is to estimate genome-wide significance for the GWA scan (we can't use the p values from the function above, as they fail to account for multiple testing). To do so (using a randomisation approach), we run:

```
dm2.clean3.gwas.emp<- qtscore(dm2, diabetes.clean3, trait="binomial",times=200)
```

```
##
  |
  |                                                                      |   0%
  |
  |===                                                                   |   5%
  |
  |======                                                                |  10%
  |
  |==========                                                            |  15%
  |
  |=============                                                         |  20%
  |
  |================                                                      |  25%
  |
  |====================                                                  |  30%
  |
  |========================                                              |  35%
  |
  |============================                                          |  40%
  |
  |================================                                      |  45%
  |
  |====================================                                  |  50%
  |
  |========================================                              |  55%
  |
  |============================================                          |  60%
  |
```

```
  |=========================================                        |  65%
  |
  |=============================================                    |  70%
  |
  |=================================================                |  75%
  |
  |=====================================================            |  80%
  |
  |=========================================================        |  85%
  |
  |=============================================================    |  90%
  |
  |=================================================================|  95%
  |
  |=====================================================================| 100%
```

```
descriptives.scan(dm2.clean3.gwas.emp, sort="Pc1df")
```

```
## Summary for top 10 results, sorted by Pc1df

##           Chromosome Position Strand A1 A2   N      effB    se_effB
## rs1719133          1  4495479      +  T  A 124 0.3167801 0.08614528
## rs4804634          1  2807417      +  C  G 121 0.4119844 0.12480696
## rs8835506          2  6010852      +  A  T 121 3.5378209 1.08954331
## rs4534929          1  4474374      +  C  G 123 0.4547151 0.14160410
## rs1013473          1  4487262      +  A  T 124 2.7839368 0.86860745
## rs3925525          2  6008501      +  C  G 124 3.2807631 1.03380675
## rs3224311          2  6009769      +  G  C 124 3.2807631 1.03380675
## rs2975760          3 10518480      +  A  T 123 3.1802120 1.00916993
## rs1048031          1  4485591      +  G  T 122 0.4510793 0.14548378
## rs2521089          3 10487652      -  T  C 123 2.7298775 0.87761175
##            chi2.1df  P1df Pc1df     effAB     effBB  chi2.2df  P2df
## rs1719133 13.522368 0.330 0.385 0.3740771 0.0000000 14.677906 0.525
## rs4804634 10.896423 0.790 0.875 0.6315789 0.1739130 12.375590 0.950
## rs8835506 10.543448 0.860 0.930 4.0185185 4.0185185 12.605556 0.910
## rs4534929 10.311626 0.920 0.955 0.4830918 0.1739130 10.510272 1.000
## rs1013473 10.272393 0.920 0.960 3.0495868 5.8441558 10.926296 1.000
## rs3925525 10.070964 0.940 0.970 3.6923077 4.0000000 11.765985 0.975
## rs3224311 10.070964 0.940 0.970 3.6923077 4.0000000 11.765985 0.975
## rs2975760  9.930784 0.955 0.975 3.0000000 8.0000000 10.172522 1.000
## rs1048031  9.613391 0.975 0.980 0.4844720 0.1714286  9.965696 1.000
## rs2521089  9.675679 0.970 0.980 3.0147059 5.0000000 10.543296 1.000
```

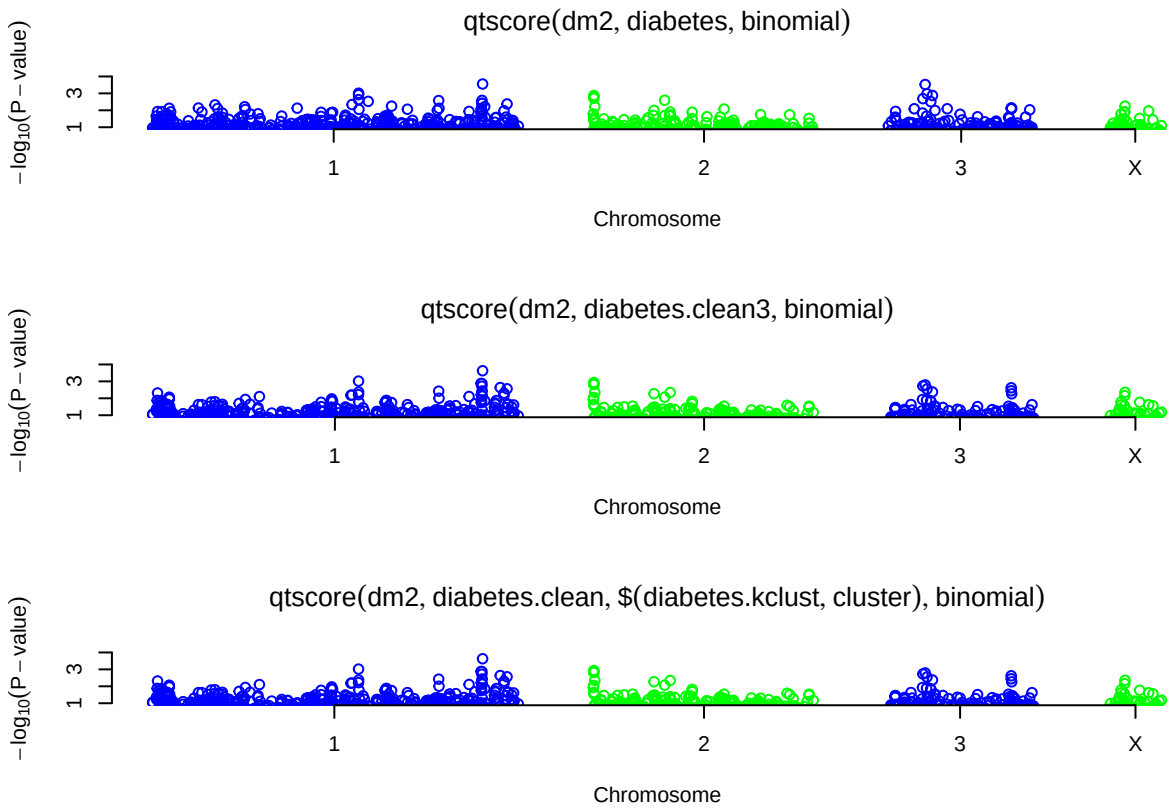
Note that none of our top hits are actually significant once we compute appropriate genome-wide p-values. In this simple example, given the small sample size, it made sense to remove one of the clusters as it was only composed of 4 individuals. But in larger datasets, we might have multiple large clusters (e.g. different ethnic groups), and removing any of them might not be desirable. There are a number of approaches to test for association whilst accounting for stratification. Let us go back to our dataset before we removed the outliers (diabetes.clean). We can run a stratified analysis simply specifying the argument strata in qtscore:

```
dm2.strat.gwas <- qtscore(dm2, diabetes.clean, trait="binomial",strata= diabetes.kclust$cluster)
```

Let's compare the hits for the original data, after removing outliers, and when accounting for stratification:

```
par(mfcol=c(3,1))
plot(dm2.simple.gwas,ylim=c(1,4))
plot(dm2.clean3.gwas,ylim=c(1,4))
```

```
plot(dm2.strat.gwas,ylim=c(1,4))
```



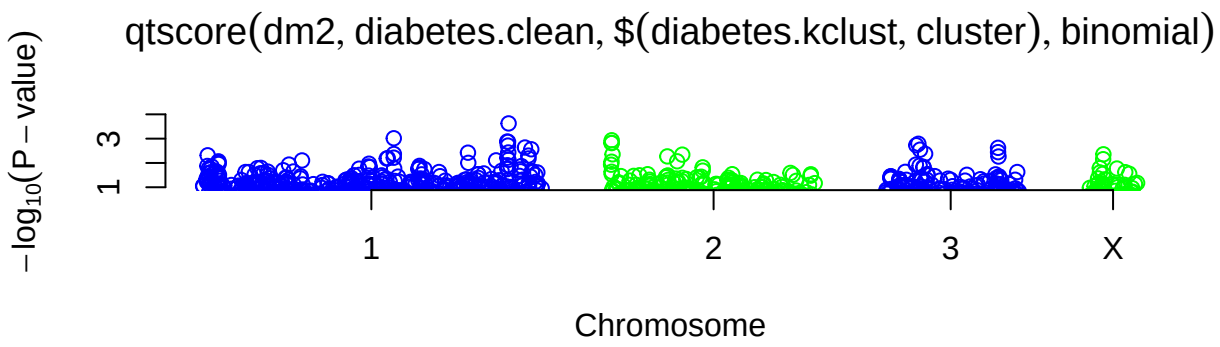

```
par(mfcol=c(1,1))
```

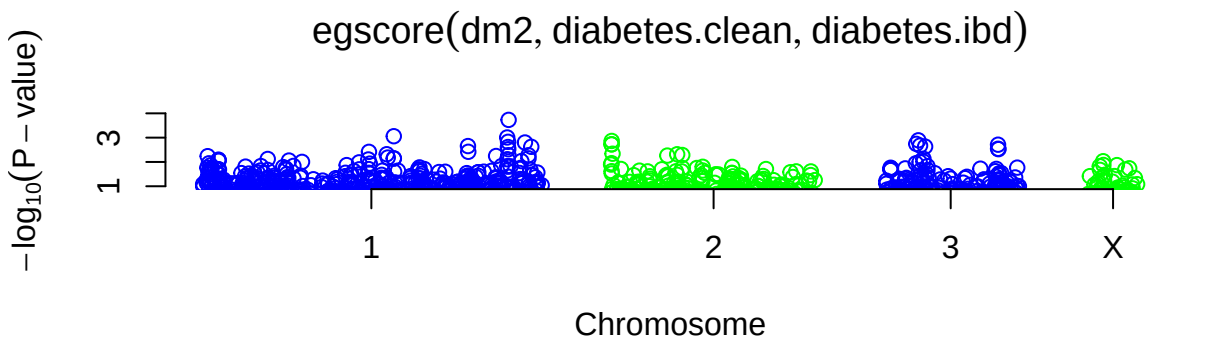
The differences are not overly large (some differences visible especially in the middle of chr 2 and in chr 3), but note that the results between the dataset without the outliers and the dataset analysed with stratification are virtually identical (in this case, since the removed cluster was very small, we don't get much additional information including it, but we get different results if we include it whilst ignoring stratification). Now let's see how we could correct for stratification using PCA of the kinship matrix.

```
dm2.pca.gwas<-egscore(dm2, data=diabetes.clean, kin= diabetes.ibd)
```

We can compare the results between stratifying the analysis by cluster and using the full kinship matrix:

```
par(mfcol=c(2,1))
plot(dm2.strat.gwas,ylim=c(1,4))
plot(dm2.pca.gwas,ylim=c(1,4))
```

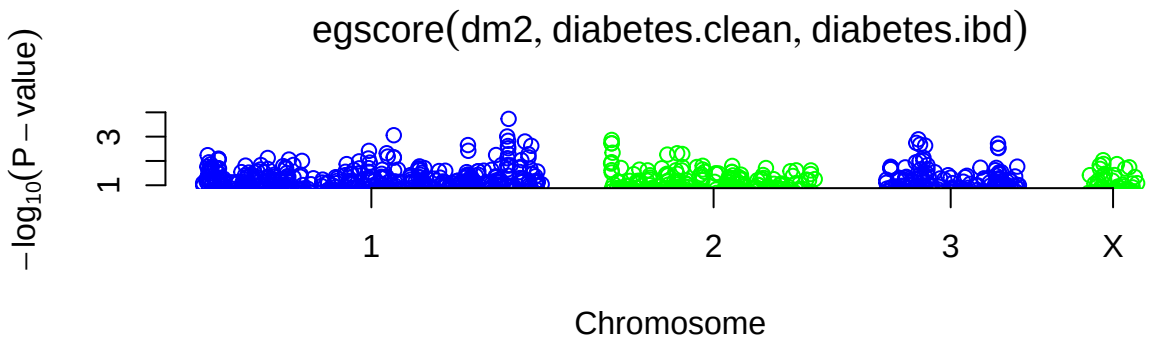


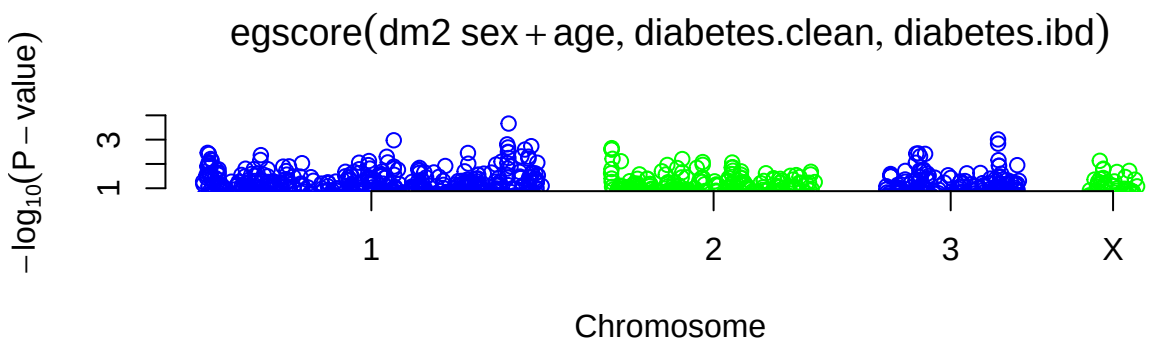



```
par(mfcol=c(1,1))
```

Whilst the results are broadly similar, note that, on the X chromosome, using the full kinship matrix removes some extreme values (which therefore were likely to be due to some cryptic stratification within the clusters). Finally, we can explore what would happen if we include covariates (sex and age), thus accounting for possible differences among the subjects:

```
dm2.cov.gwas<- egscore(dm2~sex+age, data=diabetes.clean, kin= diabetes.ibd)
par(mfcol=c(2,1))
plot(dm2.pca.gwas,ylim=c(1,4))
plot(dm2.cov.gwas,ylim=c(1,4))
```






```
par(mfcol=c(1,1))
```

In this dataset, the effect is limited, but note that the landscape of hits on chromosome 2 does change somewhat.

If you were interested in more advanced applications of GenABEL for GWAS analysis, there is a very extensive tutorial document available on the web: http://www.genabel.org/tutorials (you will see that this practical was heavily inspired by some of the sections in that document).