Exam_GWAS

1. What does it mean when we say that linear regression assumes no multicollinearity?

Ans: This means that the explanatory variable cannot have high correlation with each other. In statistics, the parsimonious model is often considered to be better so repeated information presented by the variable if kept then there will be alot of estimates which will cause redundency of information making our analysis corrupted.

2. Convert the following genotype table to the design matrix you would use under the Additive + Dominant model. (Remember the intercept.) Genotype AT AA AA TT AT AA AT

Ans:

```
add_dom<- matrix(c(1,1,1,1,1,1,1,1,2,2,0,1,2,1,1,0,0,0,1,0,1),nrow = 7)
```

3. Given the following markers, which one seems most likely to have the highest power for GWAS under the additive model?

Marker 1 Marker 2 Marker 3 # GG 146 276 70 # CG 6 27 44 # CC 120 8 67 # Missing 62 23 153

Gwas.

permutation -

Marker-1

Total = 394 → 62 missing

→ Here, 18.56% of data is missing

$$F(G) = \frac{2\times146+6}{2(146+6+120)}$$

$$= \frac{298}{544} = 0.5477$$

$$F(C) = 1-0.91$$

$$= 0.45 \text{ (Minor allele freq)}$$

Marker-2

Total = 334 → 23 missing

→ Here, 6.8% of data is missing

$$F(G) = \frac{2\times276+27}{2(276+27+8)}$$

$$= \frac{579}{622} = 0.93$$

$$F(C) = 1-0.93 = 0.06 \text{ (MAF)}$$

Marker-3

Total = 334 → 153 missing.

Here, 45.1% of data is missing

$$F(G) = \frac{2+70+44}{2(70+44+67)}$$

$$= \frac{184}{362} = 0.508$$

$$= 0.491 \text{ (MAF)}$$

40

Ans: After calculating the MAF of three markers the MAF are 0.45, 0.06, 0.491 for marker1, marker2, and marker3 respectively.Here, the highest MAF is in marker 3 making it more likely to have highest power, but we also need to consider the missing data. In marker 3 we have 45% of missing data which is almost half so even though the MAF is significantly high we will not have more power as the allele replication will be low. So marker 1 which has good MAF i.e 45% i.e the thumb rule of 20 sample at least carrying minor allele to be considered acceptable is followed and it also has less missing data i.e only 18% of data is missing.I would rely on marker 1 to be more on safe side but marker 3 also has potential.

4. What is the difference between basic linear regression and generalized linear regression, and what are the advantages of generalized linear regression?

Ans: Basic linear regression assume the response has a normal distribution while generalised linear regression assume and can be used for the data with non-normal response variable nature. generalised linear regression model is more flexible and robust. In linear regression the relation between the explanatory and response variable is assumed to be linear but in GLM it can be non-linear. Its advantage is that we can use the GLM for binary distribution data like disease presence or absence study or poission distribution following data and also it is more flexible in terms of assuming the residual variance distribution since it doesn't assume that the variance is constant across all independent variable.

5.What causes population structure? Give an example.

Ans: Population structure is the sub-grouping or clustering of genotype group based on their background or genetic variation within same species. It is caused by the different genetic background of the genotype, it can be the geographical location, of different genotype that has different genetic background or history i.e if we mix fingermillet from Nepal and the finger millet from Africa we can see different sub-group of fingermillet which is caused by the background of millet grown affected by the genetics of landraces in different place, teh climate, different enviornmental factors and also natural selection, genetic drift can cause the population structure within a species.

6. Take these matrices and multiply them (A * B * C).

```
A<- matrix(c(2,-1,4,-3,0,2),nrow = 3)
B<- matrix(c(7,1,-2,2,1,0,0,-4),nrow = 2)
C<- matrix(c(6,-2,0,1),nrow = 4)
#multiplication
result_matrix<- A %*% B %*% C
```

7. Load the data frame in "exam_strawberry.csv," which is looking at how nitrogen levels and variety affect strawberry flavor. Treating nitrogen as a quantitative variable (that is, not as a factor), calculate:
   - (XTX)-1

```
strawberry<- read.csv("exam_strawberry.csv")
strawberry_model<- lm(flavor ~ nitrogen+ variety, data=strawberry )
summary(strawberry_model)
```

```
##
## Call:
## lm(formula = flavor ~ nitrogen + variety, data = strawberry)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.1600 -0.8025  0.0150  0.6600  1.3550
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)         5.2750     0.8562   6.161 0.000271 ***
## nitrogen            0.0100     0.2707   0.037 0.971441
## varietyHokowase     2.3500     0.7415   3.169 0.013205 *
```

```
## varietySummerBloom   1.1750     0.7415   1.585 0.151687
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.049 on 8 degrees of freedom
## Multiple R-squared:  0.5567, Adjusted R-squared:  0.3905
## F-statistic: 3.349 on 3 and 8 DF,  p-value: 0.07623
```

```
#(XTX)-1
X<- matrix(c(1,1,1,1,1,1,1,1,1,1,1,1,1,1,2,2,2,3,3,3,4,4,4,1,0,0,0,0,1,0,0,1,1,0,0,0,1,0,1,0,0,1,0,0,
Y<- matrix(c(strawberry$flavor))
XtransX<- t(X)%*% X
XtransXinv<- solve(XtransX)

#beta
#computing beta i.e coefficient from the equation first we will calculate X^trans * Y
XtransY<- t(X)%*%Y
beta<- XtransXinv %*% XtransY
beta
```

```
##        [,1]
## [1,]  6.450
## [2,]  0.010
## [3,] -1.175
## [4,]  1.175
```

```
#residuals e
#calculating error
e<- Y- X%*%beta
e
```

```
##         [,1]
##  [1,] -0.385
##  [2,] -1.135
##  [3,] -1.160
##  [4,]  1.355
##  [5,]  1.130
##  [6,]  1.005
##  [7,]  0.545
##  [8,]  0.220
##  [9,]  0.295
## [10,] -0.915
## [11,] -0.765
## [12,] -0.190
```

8. The file "exam_models.csv" contains actual phenotype values and the predicted values from 4 different models. Which of these models has the best fit to the data? How do you know?

Ans: Model 3 is the best model in overall as the sums of sqaure of error is lowest among all for this model.

```
models<- read.csv("exam_models.csv")
sse_mod1 <- sum((models$actual - models$model1)^2)
sse_mod1
```

```
## [1] 674.5747
```

```
sse_mod2 <- sum((models$actual - models$model2)^2)
sse_mod2
```

```
## [1] 348.2165
```

```
sse_mod3 <- sum((models$actual - models$model3)^2)
sse_mod3
```

```
## [1] 299.2091
```

```
sse_mod4 <- sum((models$actual - models$model4)^2)
sse_mod4
```

```
## [1] 625.9735
```

9. You are running a GWAS for quantitative disease resistance in rice seedlings. You have a diversity panel of 580 inbred rice lines and microarray data on each of them at 120,000 sites. You filter the sites to remove any that have strong linkage between them, leaving you with 24,500 essentially independent sites. After running a GWAS, the most significant marker has a p-value of 6.82 * 10-34. What would its p-value be after performing a Bonferroni correction?

Ans: The p-value after bonferroni correction will be 1.6709e-29. Basically the p-value we get here is by multiplying the p-value (6.82e-34 * no.of test(24,500)).

10. You decide that a Bonferroni correction for the above question is too stringent and so do a false discovery rate correction instead. At FDR < 0.05, you get 124 SNPs significantly associated with your phenotypes. Assuming everything worked properly, how many of these are probably real?

Ans: The real significant snps will be around 117. As the FDR correction gives us 6 false hit i.e fdr < 0.05 means 5% of 124 are our false hit.

11. The file "exam_violets.csv" contains data for mapping purple color in violets (=sample name, phenotype, first 2 PCs, and genotype data at one specific marker). Using an additive model and both PCs as covariates, run this data as a basic linear regression. What is the p-value for this marker?

Ans:The p-value of this marker is 8.775e-05.

```
violets<- read.csv("exam_violets.csv")
violets$additive <- ifelse(violets$marker == "CC", 2,
                    ifelse(violets$marker == "CT", 1,
                    ifelse(violets$marker == "TT", 0, NA)))

violet_model<- lm(purple ~ additive + pc1 + pc2, data=violets)
summary(violet_model)
```

```
##
## Call:
## lm(formula = purple ~ additive + pc1 + pc2, data = violets)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.44475 -0.22526 -0.13205 -0.01501  1.03420
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.30209    0.08584   3.519 0.000538 ***
## additive    -0.10741    0.04575  -2.348 0.019894 *
## pc1          0.14802    0.08679   1.706 0.089681 .
## pc2          0.04540    0.01302   3.486 0.000604 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.3753 on 196 degrees of freedom
## Multiple R-squared:  0.1031, Adjusted R-squared:  0.08937
## F-statistic:  7.51 on 3 and 196 DF,  p-value: 8.775e-05
```

12. Given the marker data in "exam_barley.csv", calculate the genetic principal coordinates.

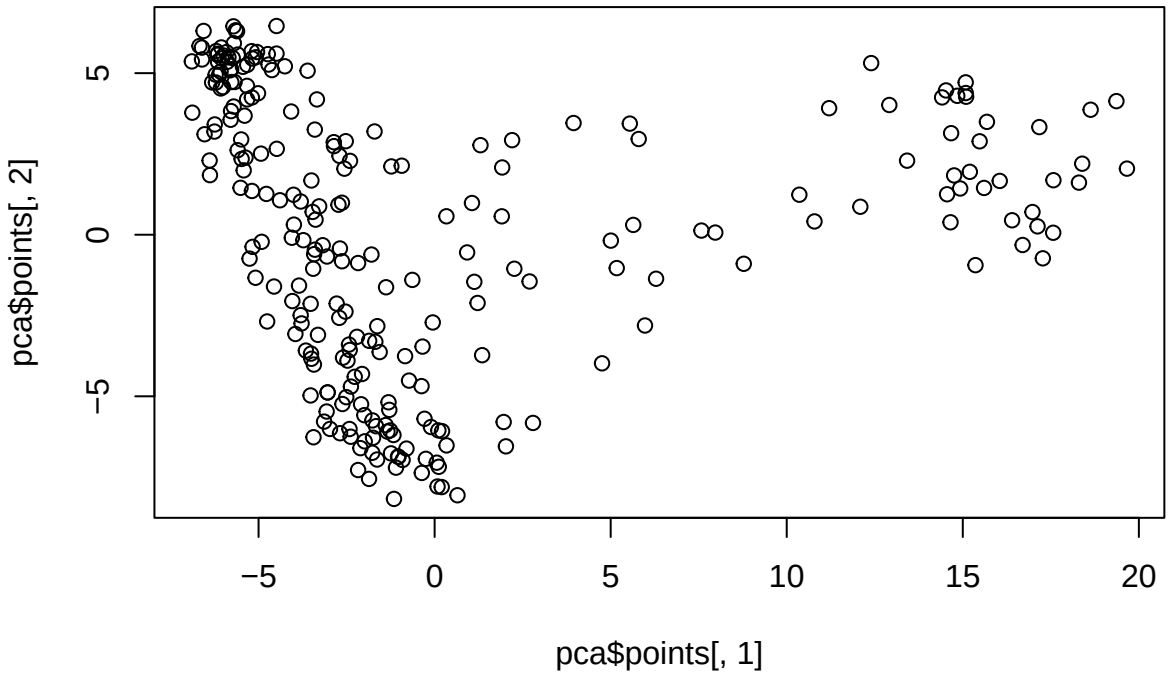
    a. Plot the first two PCs.

    b. Plot the scree plot. How many PCs should you include in your GWAS as covariates?

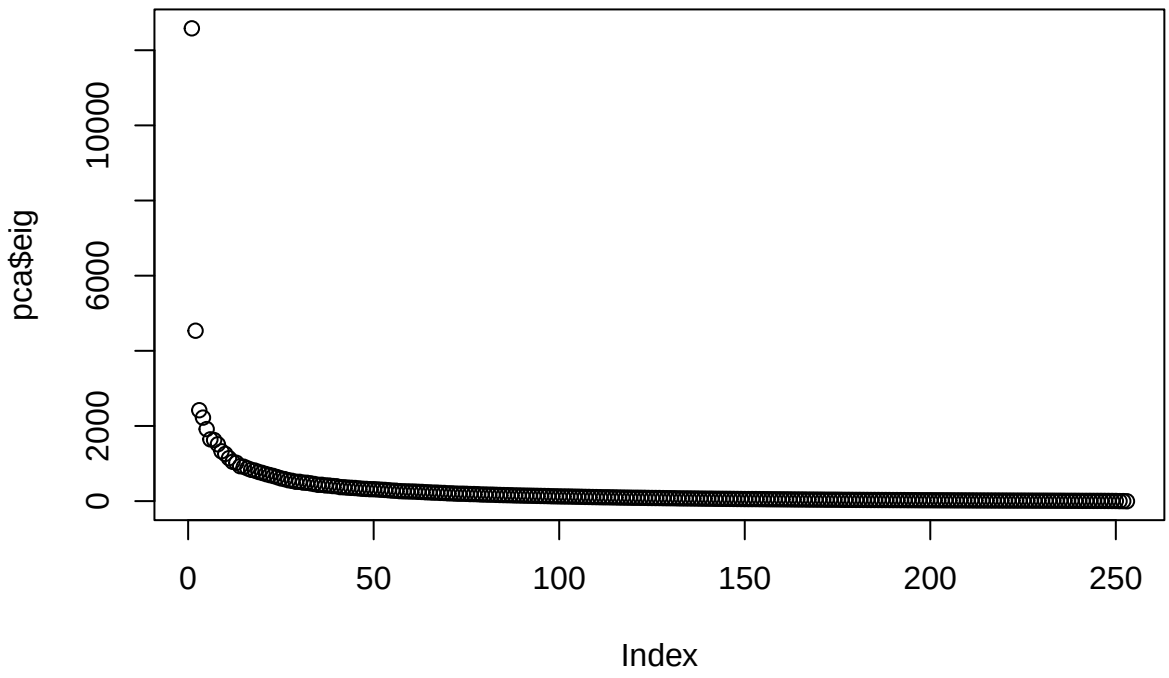
Ans: I will include first 5 PCs as a covariates in the GWAS study. c. How many would you need to include to capture 50% of the total genetic variation?

Ans: First 15 PCs need to be included to capture 50% of total genetic variation. i. Tip: The cumsum() function may be useful for this.

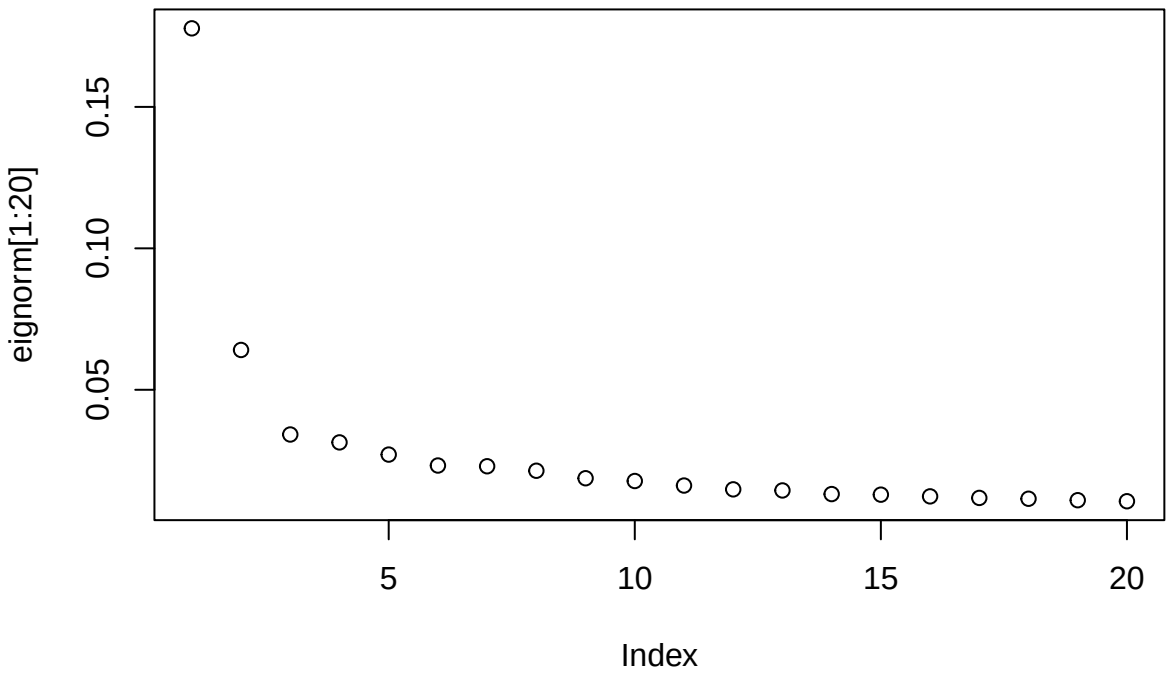
```
barley<- read.csv("exam_barley.csv", row.names = 1)
mydist= dist(barley)
pca= cmdscale(mydist, eig=TRUE, k=20)
plot(pca$points[,1], pca$points[,2]) #there are clusters of genotypes
```




```
#getting the eigen value
#plotting the screeplot
plot(pca$eig)
```




```
#normalising the eigen
eignorm<- pca$eig/sum(pca$eig)
plot(eignorm[1:20])
```




```
percents=(pca$eig[1:20] / sum(pca$eig))
cumsum(percents)
```

```
##  [1] 0.1777743 0.2418479 0.2760385 0.3074333 0.3345244 0.3577470 0.3807122
##  [8] 0.4020858 0.4208171 0.4385633 0.4547128 0.4695091 0.4839241 0.4970535
## [15] 0.5099777 0.5222949 0.5340654 0.5455403 0.5565007 0.5670967
```

13. Using rrBLUP and the genotype and phenotype data provided, run a GWAS. "exam_gwas_phenos.csv" contains phenotypes, "exam_gwas_genos_pca.csv" contains the genotypes formatted for calculating PCs, and "exam_gwas_genos_rrblup.csv" contains the same genotypes formatted for rrBLUP. (I didn't want to make you waste time doing reformatting.)

a. Plot the raw field data (in order and as a histogram).

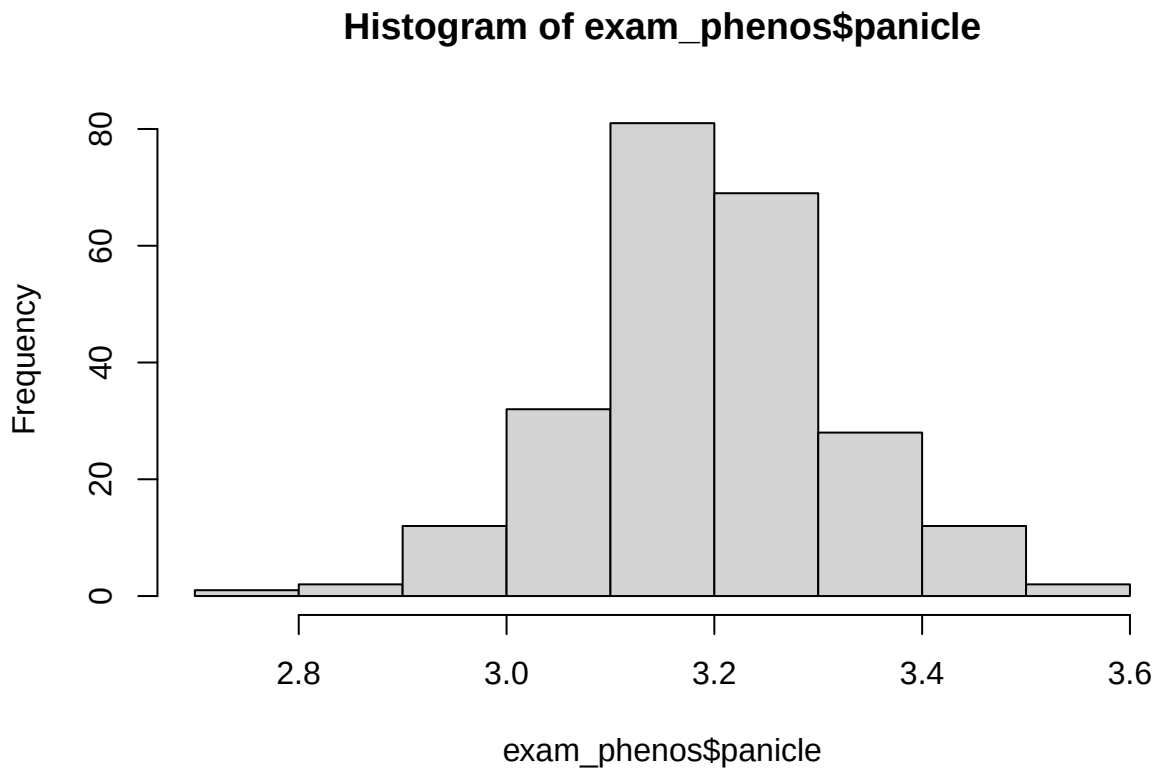
b. Plot the first two genetic principal coordinates and the scree plot of them. How many PCs do you think should be included in the model and why?

Ans: First three PCs should be included as they looks like they explain the most of variation and after the third one there is no significant breaking point.

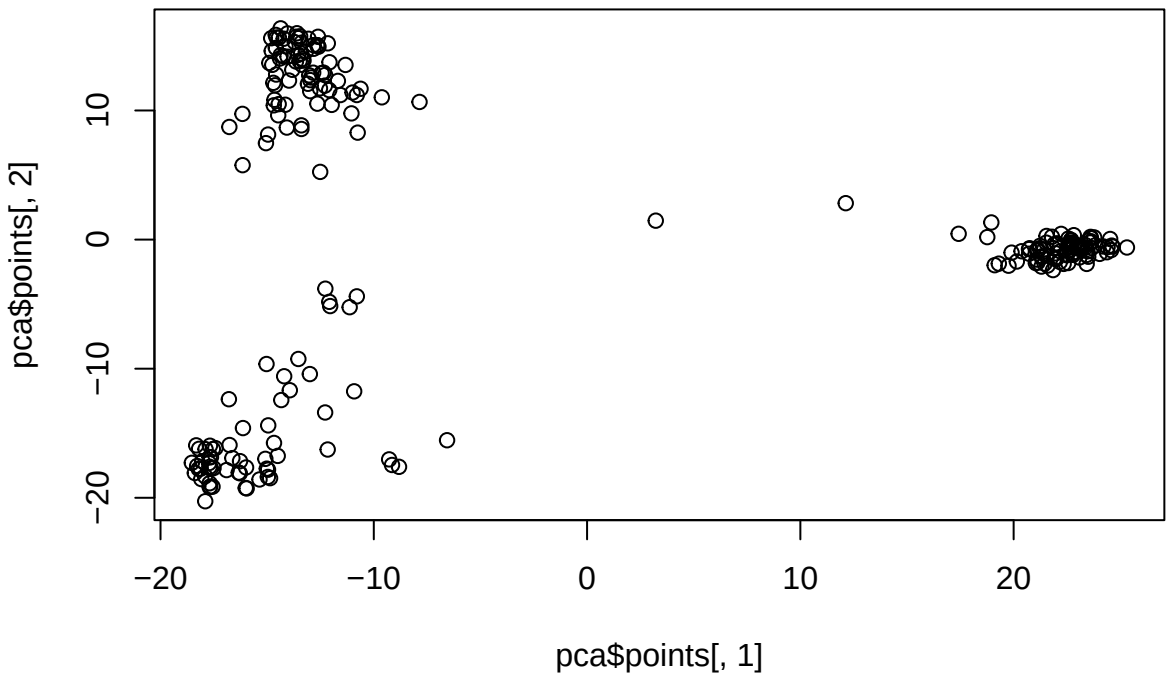
c. Plot the Manhatten and QQ-plots resulting from running it through a basic GWAS with the first 3 PCs as covariates. (Note: this may not be the same as the number you chose in part B, and that's okay; I'm specifying the number here so it's easier to check your results)

d. Note: if you have issues getting rrBLUP to do the plots right, you can tell it to skip plotting (plot=F) and manually plot them using the manhattan() and qq() functions in "exam_useful_functions.r". manhattan() takes the full results of an rrBLUP GWAS call, and qq() takes a vector of -log10 p-values. (So, just the results column from the rrBLUP output)

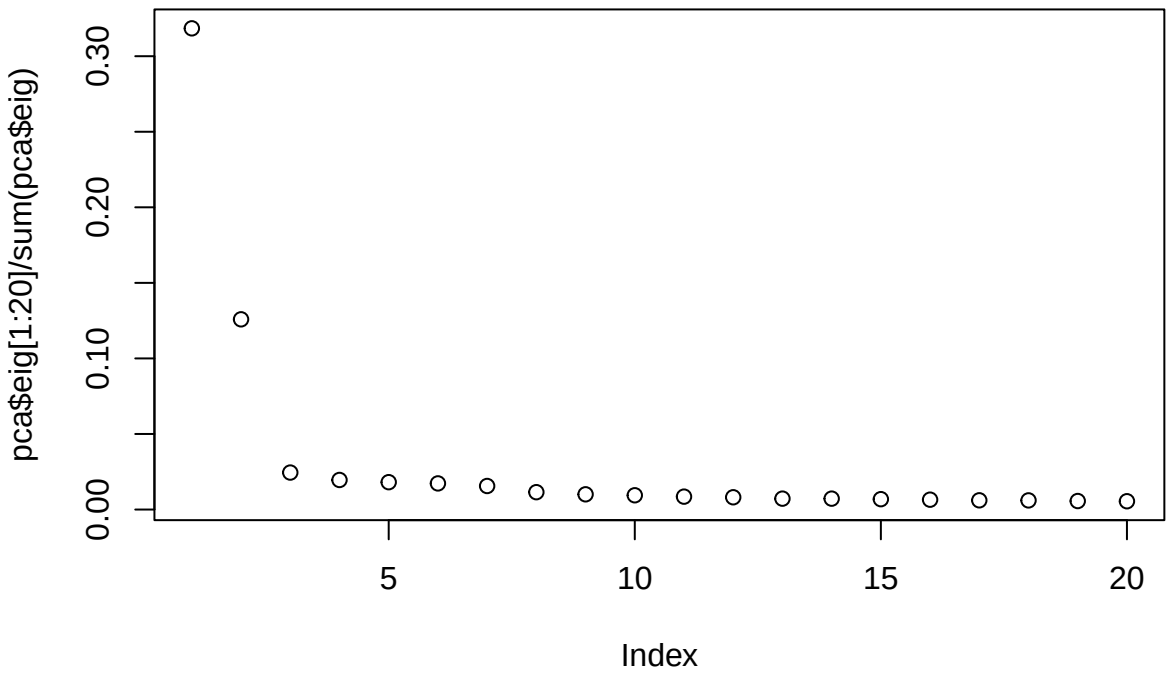
```r
source("09_useful_functions.r")
exam_phenos<- read.csv("exam_gwas_phenos.csv")
exam_genos<- read.csv("exam_gwas_genos_pca.csv")
rrblup_genos<- read.csv("exam_gwas_genos_rrblup.csv")

#plotting raw field data
hist(exam_phenos$panicle)
```




```r
#plotting the first two pcs and scree plot
mydist= dist(exam_genos)
pca= cmdscale(mydist, eig=TRUE, k=20)
plot(pca$points[,1], pca$points[,2]) #three distinct clusters of genotypes
```




```
#filtering using eigen value
plot(pca$eig[1:20] / sum(pca$eig))
```




```
percents=(pca$eig[1:20] / sum(pca$eig))
cumsum(percents)
```

```
##  [1] 0.3184276 0.4442923 0.4687640 0.4883255 0.5064343 0.5237279 0.5393075
##  [8] 0.5508004 0.5608705 0.5703172 0.5788643 0.5869794 0.5942011 0.6014116
## [15] 0.6082655 0.6147955 0.6208918 0.6269433 0.6325707 0.6380620
```

```
#plotting manhattan and qq-plots
library(rrBLUP)
kinship=A.mat(exam_genos)
kinship[1:5,1:5]
```

```
##            IRGC121155 IRGC121114 IRGC120931 IRGC120896 IRGC121060
## IRGC121155  0.7444405  0.2967191  0.2183303  0.1358498  0.2411630
## IRGC121114  0.2967191  0.9191245  0.2929621  0.1731673  0.2852233
## IRGC120931  0.2183303  0.2929621  0.9764048  0.2077105  0.4406619
## IRGC120896  0.1358498  0.1731673  0.2077105  0.8459169  0.2150429
## IRGC121060  0.2411630  0.2852233  0.4406619  0.2150429  0.8689127
```

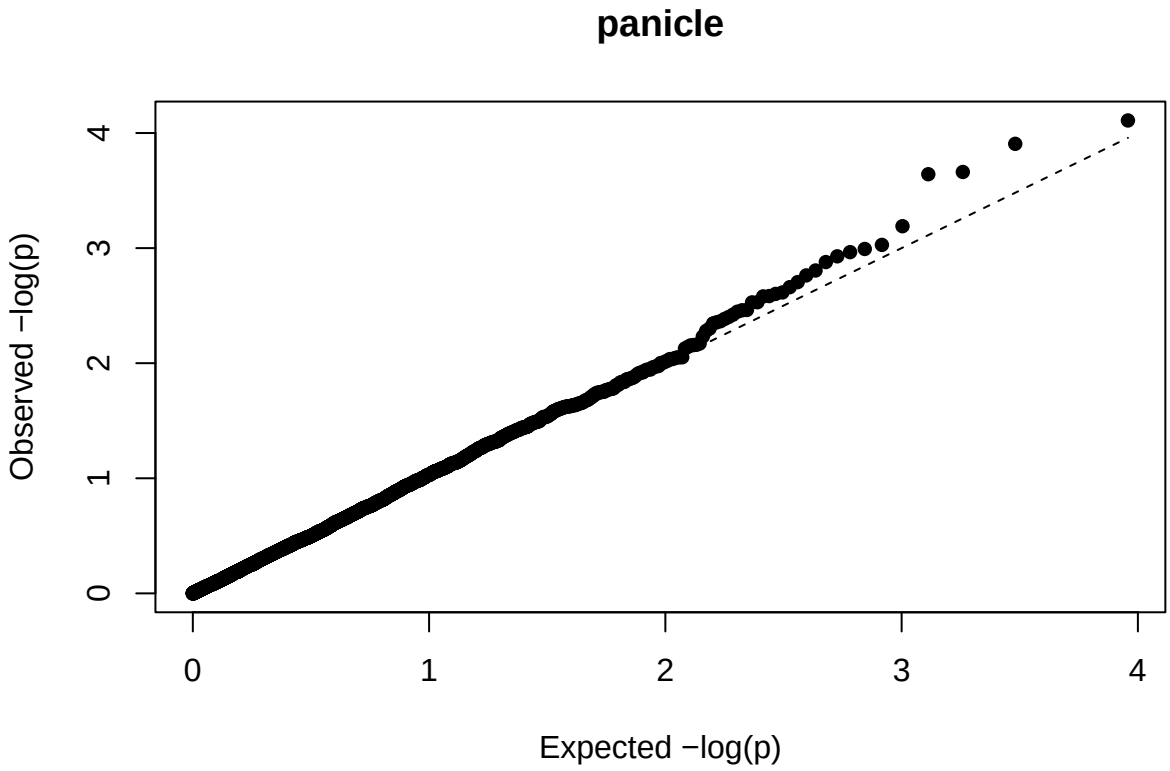
```
#only pcs as a covariates
results1= GWAS(pheno=exam_phenos, geno=rrblup_genos, n.PC=3)
```

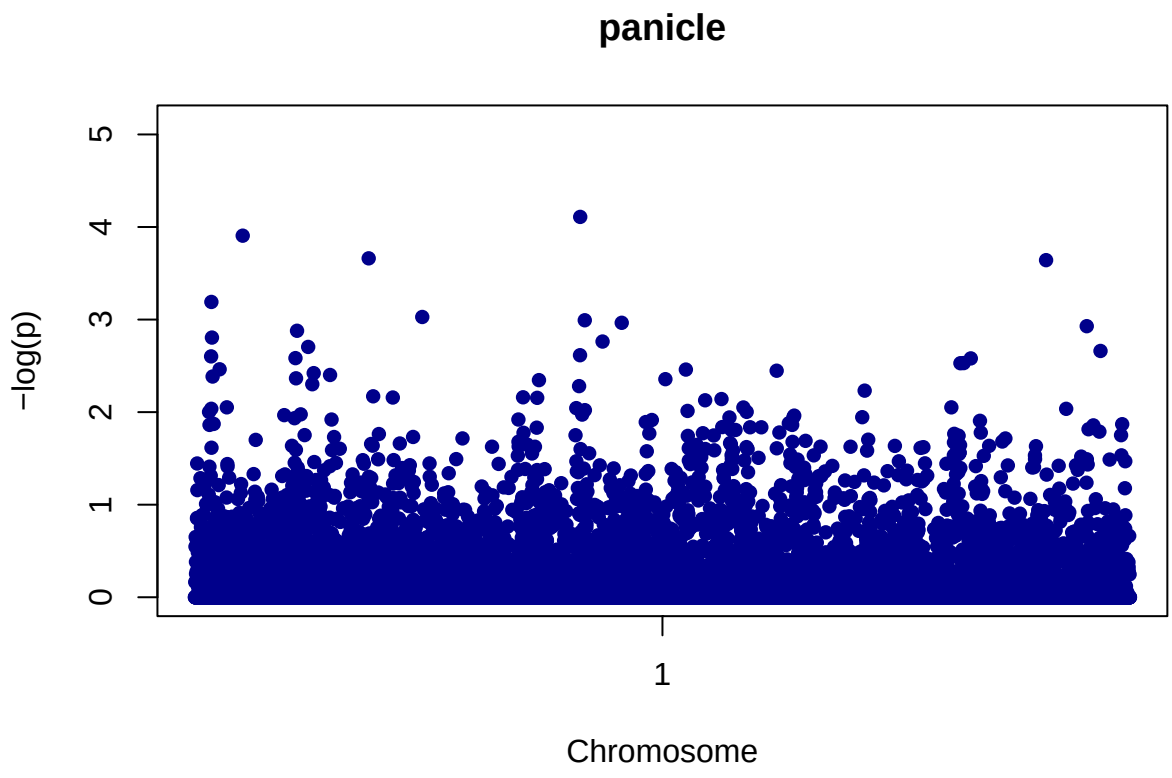
```
## [1] "GWAS for trait: panicle"
## [1] "Variance components estimated. Testing markers."
```

```
manhattan(results1)
qq(results1$panicle)
#both kinship and pcs together
```

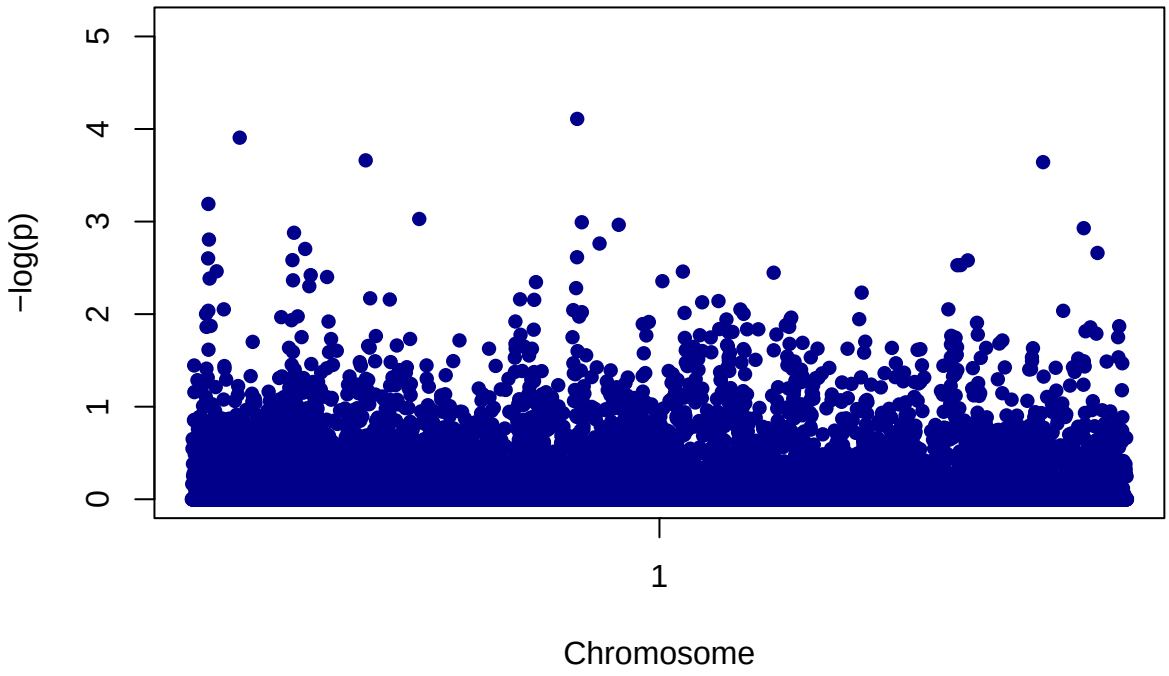
```
results= GWAS(pheno=exam_phenos, geno=rrblup_genos, K=kinship, n.PC=3)
```

```
## [1] "GWAS for trait: panicle"
## [1] "Variance components estimated. Testing markers."
```

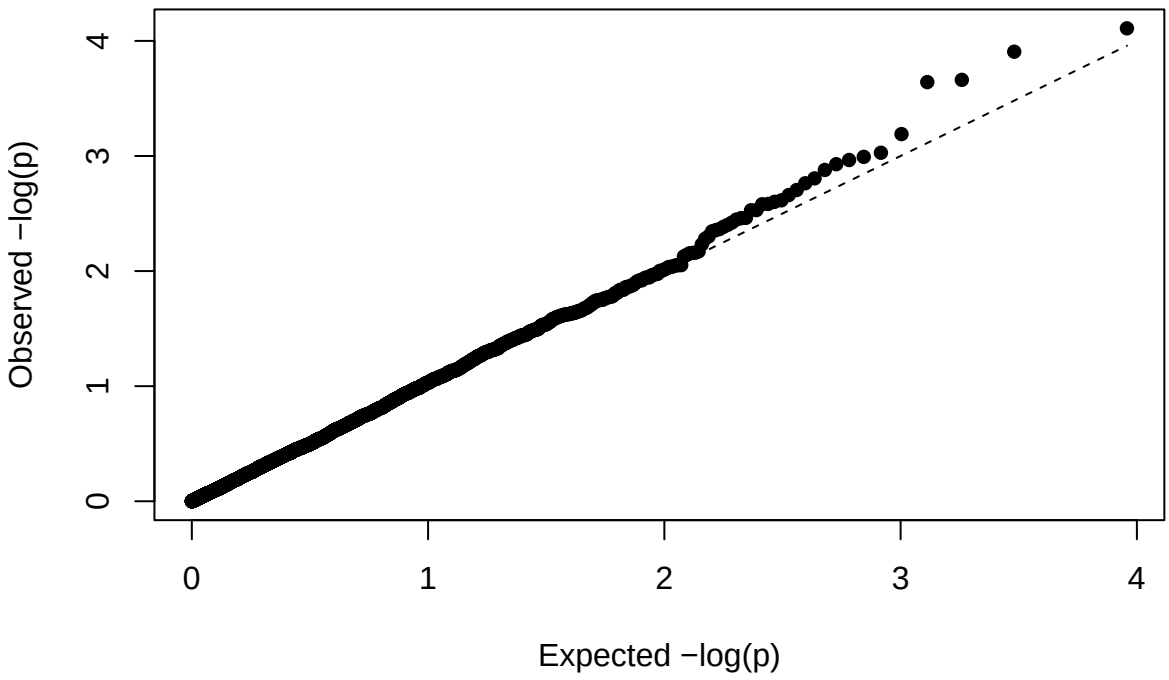





```
manhattan(results)
```




```
qq(results$panicle)
```


Observed −log(p)
Expected −log(p)
0
1
2
3
4