Practical in Quantiative Genetic Analyses using the R package qgg

Palle Duun Rohde, Izel Fourie Sørensen & Peter Sørensen

2022-10-07

Introduction

In these practicals we will be analysing quantitative traits observed in a mice population. The mouse data consist of phenotypes for traits related to growth and obesity (e.g. body weight, glucose levels in blood), pedigree information, and genetic marker data. The practicals will be a mix of theoretical and practical exercises in R that are used for illustrating/applying the theory presented in the lectures and corresponding notes.

- Practical 1: Use R for Analysing Quantitative Traits
- Practical 2: Basic Quantitative Genetics illustrated in the mouse data
- Practical 3: Estimation of Genetic Parameters for traits in the mouse data
- Practical 4: Estimation of Breeding Values for traits in the mouse data
- Practical 5: Estimation of Genomic Breeding Values for traits in the mouse data

Mouse data

The **M16 mouse** was established as an outbred animal model of early onset polygenic obesity and diabesity. This was done by selection for 3- to 6-week weight gain for 27 generations from an outbred ICR base population. Breeding criterion was within-litter selection for the male and female with the largest weight gain from 3 to 6 weeks of age. An ICR control line was maintained in parallel, with random mating from generation to generation but maintaining a similar effective population size. Mice from the M16 line ar elarger at all ages and have increased body fat percentage, fat cell size, fat cell numbers, and organ weights when compared with ICR. Mice from the M16 line are larger at all ages and have increased body fat percentage, fat cell size, fat cell numbers, and organ weights when compared with ICR. These mice also exhibit hyperphagia, accompanied by moderate obesity, and are hyperglycemic, hyperinsulinemic, and hypercholesterolemic.

The **ICR mouse** is a strain of albino mice originating in SWISS and selected by Dr. Hauschka to create a fertile mouse line. Because mice of this strain have been sent to various places from the Institute of Cancer Research in the USA, the strain was named ICR after the initial letters of the institute. Mice of this strain are relatively large albinos with a gentle nature that grow well. The ICR mouse is a general-purpose model used for studies in a wide range of fields including toxicity, pharmacology, drug efficacy, and immunology.

A large **F2 population** (n=1181) was **established by crossing the M16 and ICR lines** (for a recent description of relevant phenotypes in the parental lines, see https://onlinelibrary.wiley.com/doi/epdf/10.1038/oby.2004.176). Twelve F1 families resulted from six pair matings of M16 males x ICR females and six pair matings of the reciprocal cross. A total of 55 F1 dams were mated to 11 F1 sires in sets of five F1 full sisters mated to the same F1 sire. These same specific matings were repeated in three consecutive replicates. Thus, the F2 population consisted of 55 full-sib families of up to 24 individuals each and 11 sire families families of up to 120 individuals each. Actual numbers of mice within families varied slightly due to a small number of failed pregnancies. All litters were standardized at birth to eight pups, with approximately equal representation of males and females.

More information about the mouse data can be found in the following publications:

https://onlinelibrary.wiley.com/doi/epdf/10.1038/oby.2004.176

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1449794/

Practical 1: Use R for Analysing Quantitative Traits

Introduction:

In this practical we use R for explorative data analyses of two quantitative traits, body weight and blood glucose levels, observed in the F2 mouse population. These explorative data analyses includes computation of basic descriptive statistics such as mean, and variance used to describe each of these traits. Distribution plots (e.g., histogram) will be used to visualize whether the trait phenotypes follow a normal distribution. Boxplots will be used to spot potential effects of explanatory variables. Furthermore relationships between traits and variables will be characterized in terms of correlations and linear relationships.

Let's get started to explore our mouse data

One of the first thing to do is to explore the data used in the analysis. The goal is to understand the variables, how many records the data set contains, how many missing values, what is the variable structure, what are the variable relationships and more. Several commands/functions will be used. To read more about a specific function (e.g., `str`) write `?str`.

The mouse data set can be loaded using the following command:

```
mouse <- readRDS(url("https://github.com/psoerensen/bgcourse/raw/main/data/mouse.rds"))
```

Question 1: How many observations and which variables do we have in the data set? To get a fast overview of the data set you are working with you can use the `str` function:

Answer:

```
str(mouse)
```

```
## 'data.frame':    1177 obs. of  6 variables:
##  $ sire: Factor w/ 11 levels "25","28","34",..: 6 6 6 6 6 6 6 6 6 6 ...
##  $ dam : Factor w/ 55 levels "26","27","29",..: 8 8 8 8 8 8 8 8 8 8 ...
##  $ sex : Factor w/ 2 levels "Female","Male": 1 1 1 1 2 2 2 2 1 1 ...
##  $ reps: Factor w/ 3 levels "1","2","3": 1 1 1 1 1 1 1 1 2 2 ...
##  $ Gl  : num  187 136 115 125 112 190 169 159 111 89 ...
##  $ BW  : num  36.6 33.3 42.1 37.1 38.4 ...
```

The two quantitative traits we will be analysing are glucose levels in the blood (Gl) and body weight (BW) measured in the mice at 8 weeks of age. A more detailed view of the two quantitative traits in the `data.frame` is provided by the `summary` function:

```
summary(mouse[,5:6])
```

```
##        Gl              BW
##  Min.   : 65.0   Min.   :23.04
##  1st Qu.:121.0   1st Qu.:34.06
```

```
##  Median :139.0   Median :38.32
##  Mean   :144.2   Mean   :38.72
##  3rd Qu.:164.0   3rd Qu.:43.40
##  Max.   :292.0   Max.   :60.28
```

Question 2: What is the mean and variance of body weight and blood glucose levels? Use the `mean` and `var` functions to compute the mean and variance two traits:

Answer:

```
weight <- mouse[,"BW"]
glucose <- mouse[,"Gl"]
mean(weight)
```

```
## [1] 38.72392
```

```
mean(glucose)
```

```
## [1] 144.2234
```

```
var(weight)
```

```
## [1] 37.84458
```

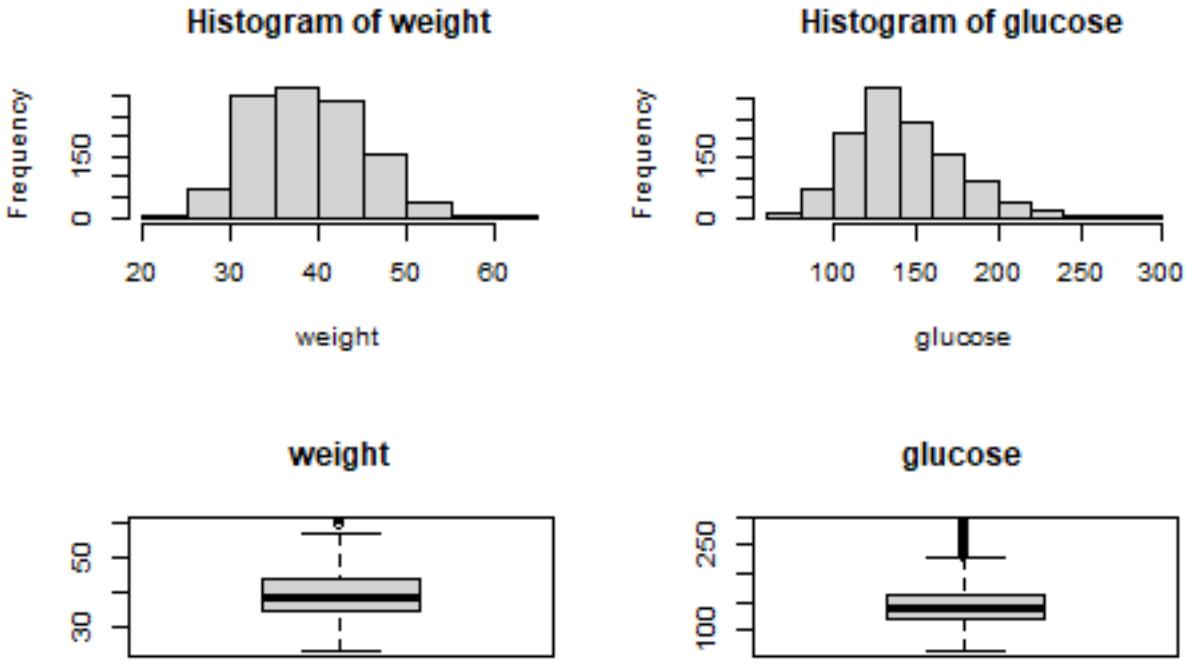
```
var(glucose)
```

```
## [1] 1150.66
```

Question 3: How are the phenotypes of weight and glucose distributed? Use the `histogram` and `boxplot` functions to visualize the distribution the two traits:

Answer:

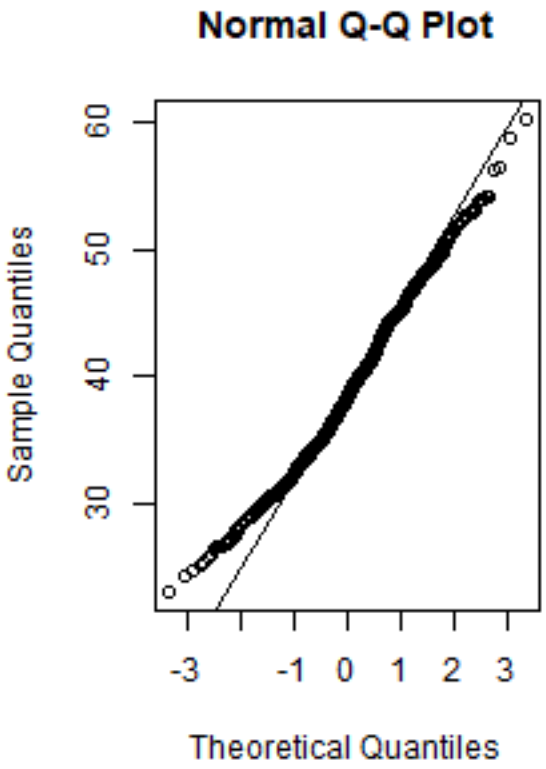
```
layout(matrix(1:4,ncol=2,byrow=TRUE))
hist(weight)
hist(glucose)
boxplot(weight, main="weight")
boxplot(glucose, main="glucose")
```

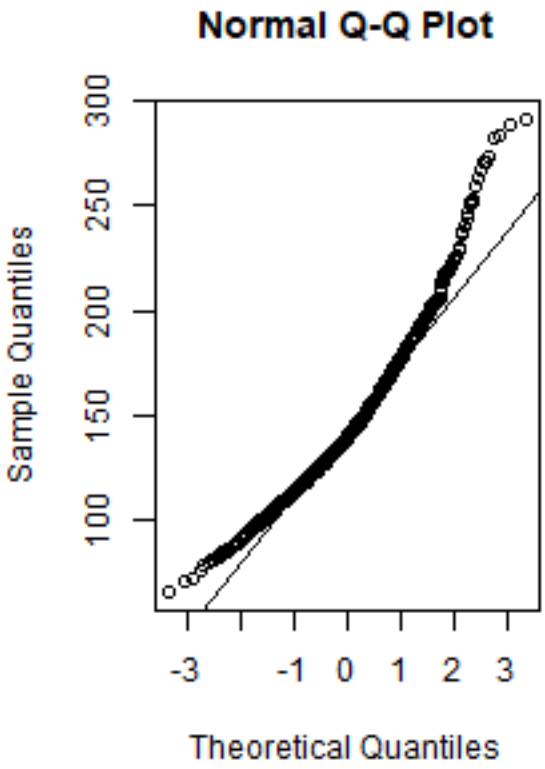


Question 4: Are the phenotypes of weight and glucose normally distributed? Use the `qqnorm` function to create a quantile-quantile (QQ) plot of the trait values. Use the qqline function to add a line to a "theoretical", by default normal, quantile-quantile plot:

Answer:

```
layout(matrix(1:2,ncol=2))
qqnorm(weight)
qqline(weight)
qqnorm(glucose)
qqline(glucose)
```

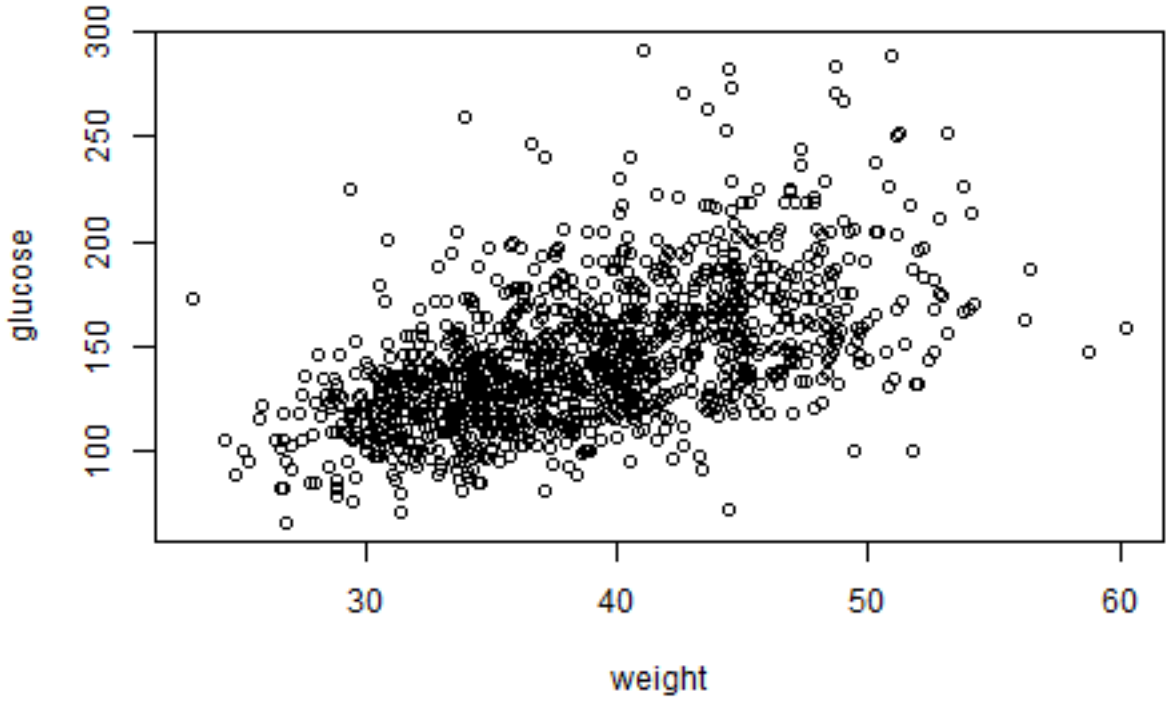




Question 5: Is there a relationship between the phenotypes of weight and glucose? Make a scatter plot of the the 2 traits using the `plot` function. Compute the correlation using the `cor` function and perform a statistical test to assess the significance of correlation between values of weight and glucose using the `cor.test` function:

Answer:

```
plot(weight,glucose)
```




```
cor(weight,glucose)
```

```
## [1] 0.5440533
```

```
cor.test(weight,glucose)
```

```
##
##  Pearson's product-moment correlation
##
## data:  weight and glucose
## t = 22.227, df = 1175, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5025357 0.5830674
## sample estimates:
##       cor
## 0.5440533
```

Let us explore the family structure. Use the `table` function to determine the family size for sires and dams:

```
table(mouse$sire)
```

```
##
##  25  28  34  40  51  63  69  72  78  79  85
## 115 114 107  95 110 103 103 119 119 118  74
```

```
table(mouse$dam)
```

```
##
## 26 27 29 30 31 32 33 35 36 37 38 39 41 42 43 44 45 46 47 48 49 50 52 53 54 55
## 24 16 23 24 24 24 24 24 23 16 16 24 23 16 24 24 16 16 24 16 23 16 16 15 16 24
## 56 57 58 59 60 61 62 64 65 66 67 68 70 71 73 74 75 76 77 80 81 82 83 84 86 87
## 16 23 24 21 24 21 24 24 24 23 22 22 15 19 23 24 23 24 24 24 23 24 24 24 16 23
## 88 89 90
## 24 24 20
```

Question 6: What are the min and max family size? Use the `table` and `min` or `max` functions to determine the min/max family size for sires and dams:

Answer:

```
min(table(mouse$sire))
```

```
## [1] 74
```

```
max(table(mouse$sire))
```

```
## [1] 119
```

```
min(table(mouse$dam))
```

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

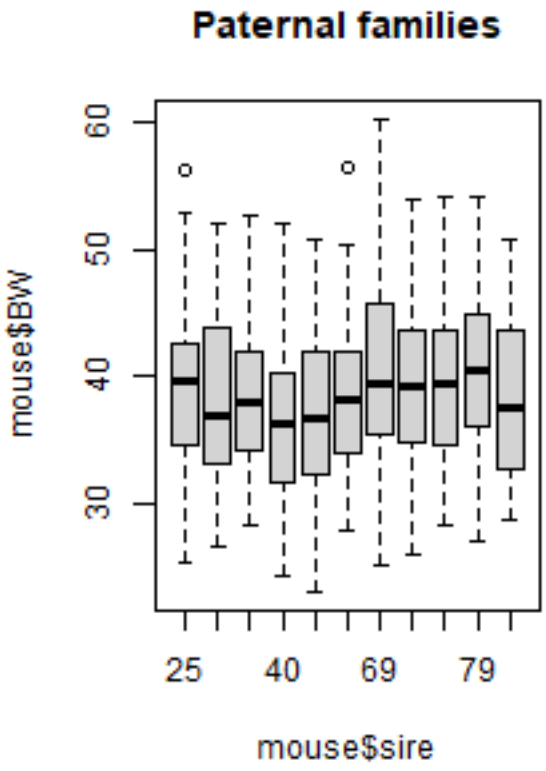
```
max(table(mouse$dam))
```

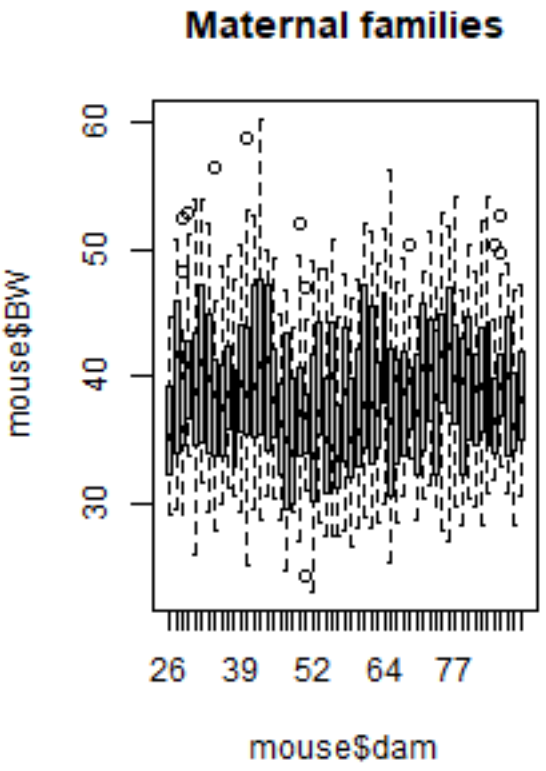
```
## [1] 24
```

Question 7: Does family influence the traits? Use the `boxplot` function to visualize the potential effect of family on the two traits:

Answer:

```
layout(matrix(1:2,ncol=2))
boxplot(mouse$BW~mouse$sire, main="Paternal families")
boxplot(mouse$BW~mouse$dam, main="Maternal families")
```





Question 8: How many males and females?

Answer:

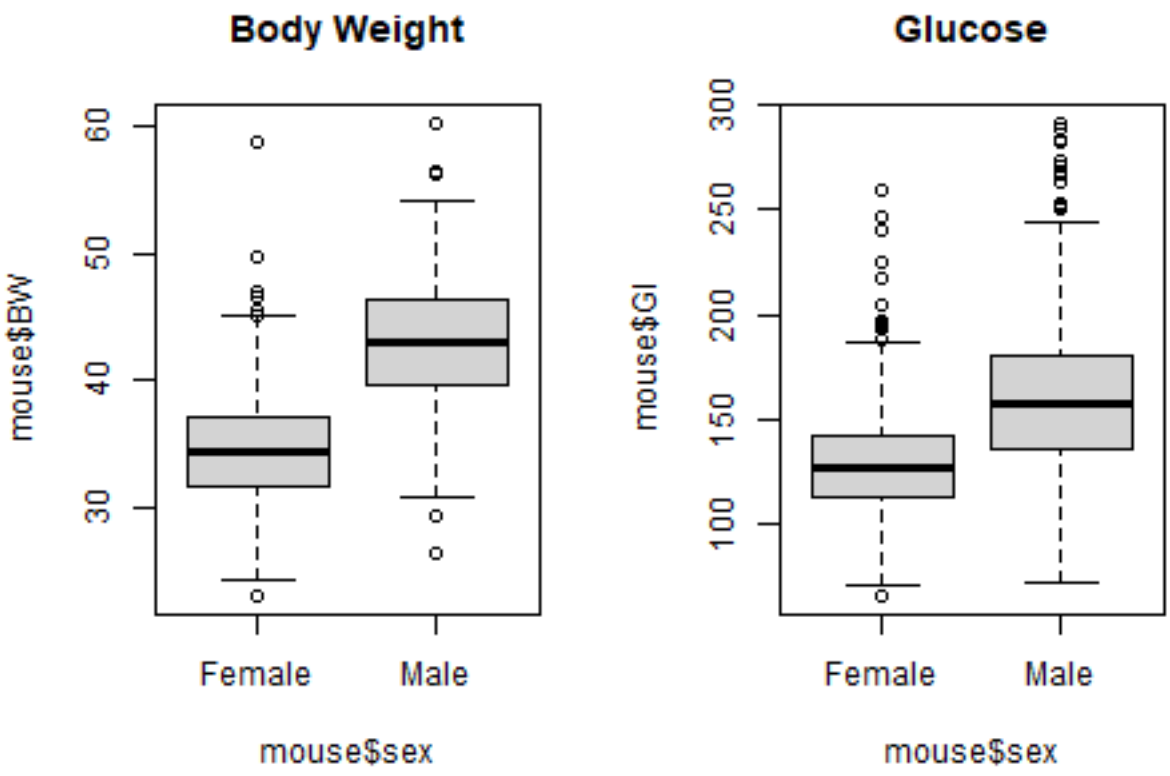
```
table(mouse$sex)
```

```
##
## Female   Male
##    589    588
```

Question 9: Does gender influence the traits? Use the `boxplot` function to visualize the potential effect of gender on the two traits:

Answer:

```
layout(matrix(1:2,ncol=2))
boxplot(mouse$BW~mouse$sex, main="Body Weight")
boxplot(mouse$Gl~mouse$sex, main="Glucose")
```



Question 10: How many observations in each replicate?

Answer:

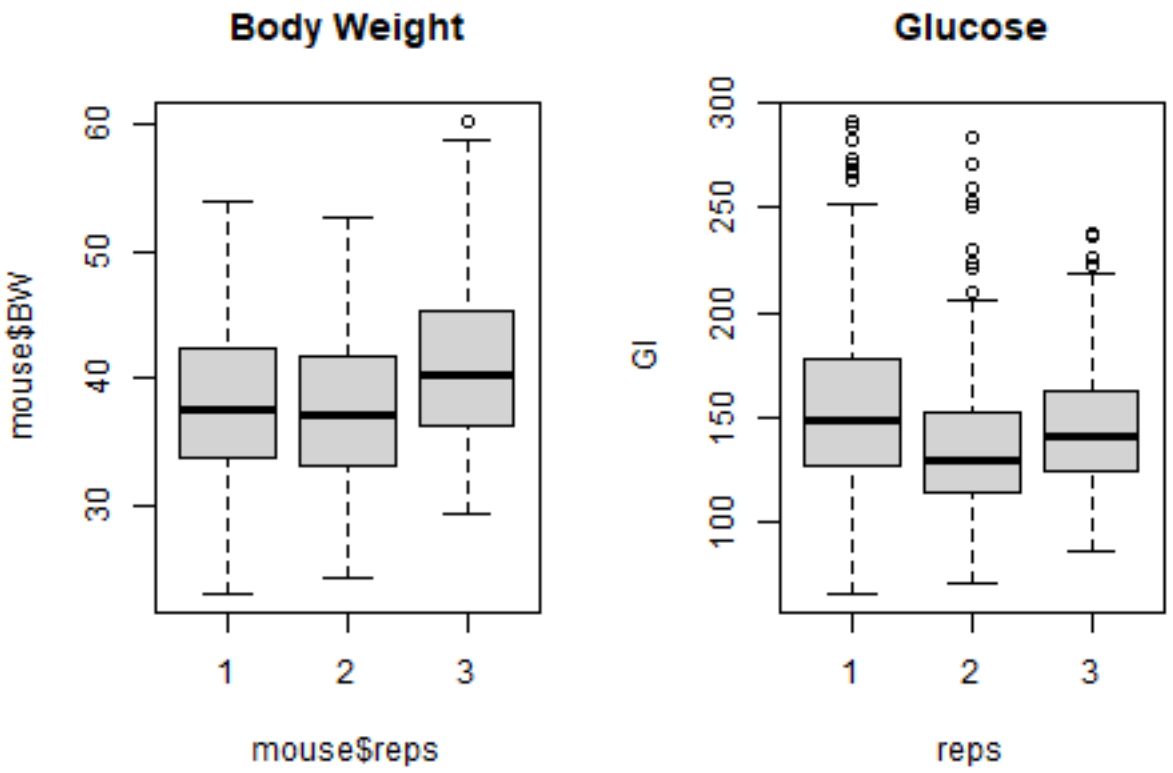
```
table(mouse$reps)
```

```
##
##   1   2   3
## 415 427 335
```

Question 11: Does replicate influence the phenotype of weight and glucose? Use the `boxplot` function to visualize the potential effect of replicate on the two traits:

Answer:

```
layout(matrix(1:2,ncol=2))
boxplot(mouse$BW~mouse$reps, main="Body Weight")
boxplot(Gl~reps, main="Glucose", data=mouse)
```



The exploratory data analysis is the process of analyzing and visualizing the data to get a better understanding of the data. It is not a formal statistical test.

Which factors should we include in the statistical model? To best answer these question we can fit a linear model that include these factors (sire, dam, sex, reps) in the model. This can be done using the `lm` function:

```
fit <- lm(BW~sire+dam+sex+reps, data=mouse)
```

To test the effect of the variables in the model use the `anova` function on the `fit` object from the `lm` function:

```
anova(fit)
```

```
## Analysis of Variance Table
##
## Response: BW
##             Df  Sum Sq Mean Sq   F value    Pr(>F)
## sire        10  1536.6   153.7    9.2514 7.705e-15 ***
```

```
## dam         44  2020.9    45.9    2.7652 1.238e-08 ***
## sex          1 20637.7 20637.7 1242.5000 < 2.2e-16 ***
## reps         2  1723.6   861.8   51.8858 < 2.2e-16 ***
## Residuals 1119 18586.4    16.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Question 12: Do genetic factors influence the traits? Look at the output of the `anova` function.

Answer:

Practical 2: Basic Quantitative Genetics

Time schedule of practical session 2:

Introduction:

In this practical we use R for explorative data analyses of two quantitative traits, body weight and blood glucose levels, observed in the F2 mouse population. We will be characterizing and investigating the potential effects of a single marker locus. This includes computation of allele and genotype frequencies, evaluating different genetic models, and estimation of the breeding values and genetic variances for the single marker locus.

Furthermore you may also want to explore these `shinyapps` that could help understand some of the basic concepts of quantitative genetics:

https://neyhartj.shinyapps.io/qgshiny/

https://shiny.cnsgenomics.com/Falconer2/

Let's continue explore our mouse data

The mouse data set can be loaded using the following command:

```
mouse <- readRDS(url("https://github.com/psoerensen/bgcourse/raw/main/data/mouseqtl.rds"))
```

Question 1: How many observations and which variables do we have in the data set? To get a fast overview of the data set you are working with you can use the `str` function:

Answer:

```
str(mouse)
```

```
## 'data.frame':    1177 obs. of  8 variables:
##  $ sire : Factor w/ 11 levels "25","28","34",..: 6 6 6 6 6 6 6 6 6 6 ...
##  $ dam  : Factor w/ 55 levels "26","27","29",..: 8 8 8 8 8 8 8 8 8 8 ...
##  $ sex  : Factor w/ 2 levels "Female","Male": 1 1 1 1 2 2 2 2 1 1 ...
##  $ reps : Factor w/ 3 levels "1","2","3": 1 1 1 1 1 1 1 1 2 2 ...
##  $ Gl   : num  187 136 115 125 112 190 169 159 111 89 ...
##  $ BW   : num  36.6 33.3 42.1 37.1 38.4 ...
##  $ M227 : Factor w/ 3 levels "AA","AB","BB": 2 1 2 2 2 1 2 1 2 2 ...
##  $ M1139: Factor w/ 3 levels "AA","AB","BB": 3 NA 1 1 1 2 3 3 2 2 ...
```

Question 2: How many observations do the two marker variables have in each genotype class? Use the `table` function to explore the two marker variables:

Answer:

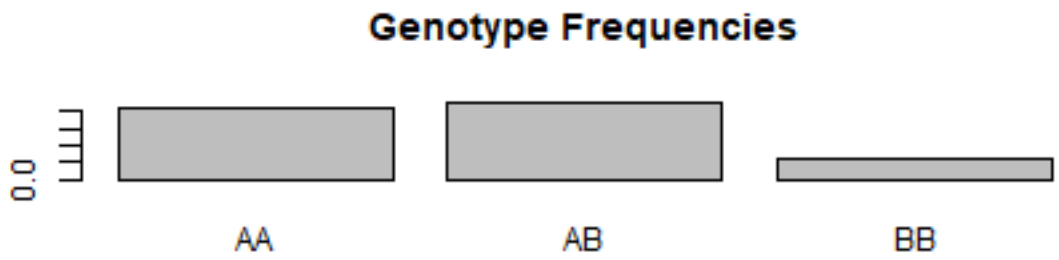
```
table(mouse$M227)
```

```
##
##  AA  AB  BB
## 493 536 145
```

Question 2: What are the genotype and allele frequencies for M227? Include the allele and genotype frequencies for M227 in the following table:

Variable	M227
f_{AA}	
f_{AB}	
f_{BB}	
f_A	
f_B	

```
freq_genotypes <- table(mouse$M227)/sum(table(mouse$M227))
fA <- sum(table(mouse$M227)*c(2,1,0))/(sum(table(mouse$M227))*2)
fB <- sum(table(mouse$M227)*c(0,1,2))/(sum(table(mouse$M227))*2)
freq_alleles <- c(fA,fB)
names(freq_alleles) <- c("A","B")
layout(matrix(1:2,nrow=2))
barplot(freq_genotypes, main="Genotype Frequencies")
barplot(freq_alleles, main="Allele Frequencies")
```




```
freq_genotypes
```

```
##
##        AA        AB        BB
## 0.4199319 0.4565588 0.1235094
```

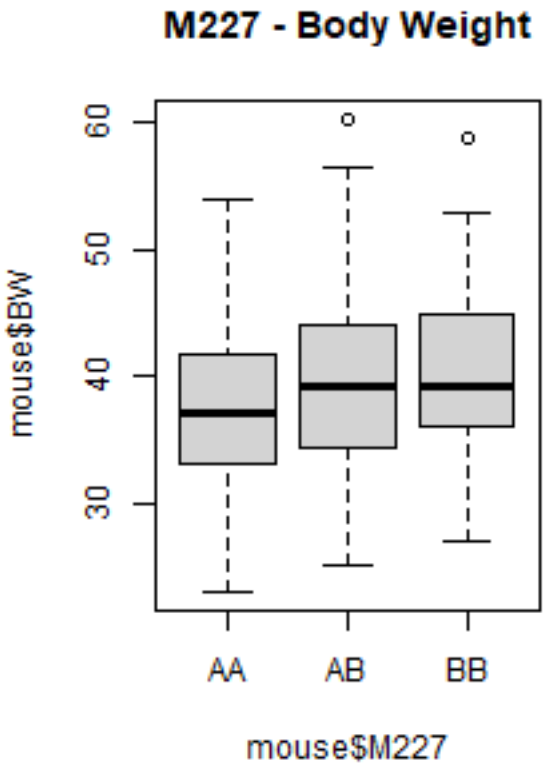
```
freq_alleles
```

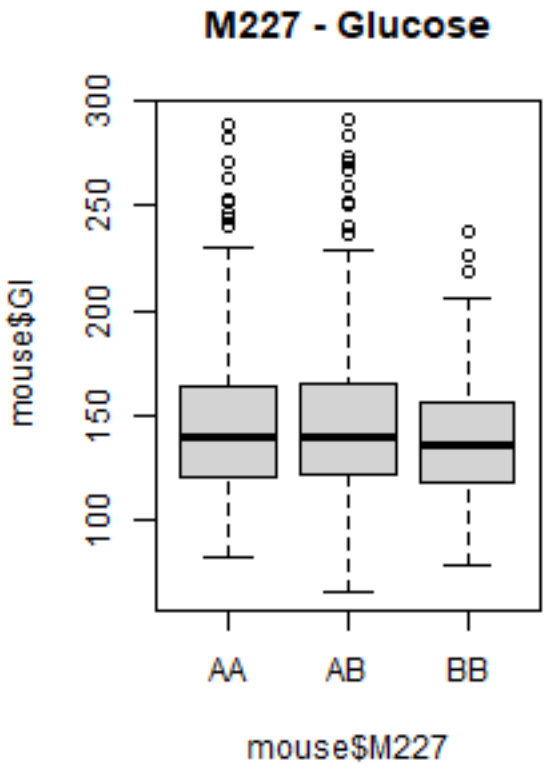
```
##         A         B
## 0.6482112 0.3517888
```

Question 3: Does the marker variable M227 potentially influence body weight and glucose? Use the `boxplot` function to visualize the potential effect of the marker variable M227 on the two traits:

Answer:

```
layout(matrix(1:2,ncol=2))
boxplot(mouse$BW~mouse$M227, main="M227 - Body Weight")
boxplot(mouse$Gl~mouse$M227, main="M227 - Glucose")
```





To best answer these question we can fit a linear model that also include the effect of the marker variable in addition to sex and reps. This can be done using the `lm` function:

```
fit <- lm(BW~ sex + reps + M227, data=mouse)
```

To test the effect of the variables in the model use the `anova` function on the `fit` object from the `lm` function:

```
anova(fit)
```

```
## Analysis of Variance Table
##
## Response: BW
##             Df  Sum Sq Mean Sq  F value    Pr(>F)
## sex          1 20542.9 20542.9 1203.352 < 2.2e-16 ***
## reps         2  2195.9  1097.9   64.315 < 2.2e-16 ***
## M227         2  1660.3   830.1   48.627 < 2.2e-16 ***
## Residuals 1168 19939.3    17.1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Question 4: Based on the linear model results do marker variable M227 influence body weight?

Answer:

The additive effect is modeled by a variable, `add`, with levels that is coded as -1, 0, and 1 (corresponding to -a, 0, a) for the genotypes AA, AB, and BB. The following lines of R code create a the `add` variable, fit the linear model and test the effects:

```
alleles <- c(-1,0,1)
names(alleles) <- c("AA","AB","BB")
mouse$add <- alleles[mouse$M227]
fit <- lm(BW~ sex + reps + add, data=mouse)
summary(fit)
```

```
##
## Call:
## lm(formula = BW ~ sex + reps + add, data = mouse)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.9743  -2.6780  -0.0483   2.5625  19.7455
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  34.3598     0.2396 143.399   <2e-16 ***
## sexMale       8.4133     0.2415  34.838   <2e-16 ***
## reps2        -0.3787     0.2852  -1.328    0.184
## reps3         2.8966     0.3043   9.518   <2e-16 ***
## add           1.7381     0.1790   9.713   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.135 on 1169 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.5492, Adjusted R-squared:  0.5477
## F-statistic: 356.1 on 4 and 1169 DF,  p-value: < 2.2e-16
```

The `summary(fit)` command produced

- parameter estimates (or Coefficients) $\widehat{\mu}$ and $\widehat{\beta}$,
- their standard errors (SE) (estimates for square root of the sampling variance of the parameter estimates),
- t-statistic (estimate/SE) and
- P-value under the null hypothesis that the parameter is 0 and errors are uncorrelated and have distribution $N(0, \sigma^2)$.

Under the assumptions of linear model, sampling distribution of t-statistic is t-distribution and hence q% confidence intervals are determined as $\widehat{\beta} \pm a \times \text{SE}$, where a is the q/2% quantile of t-distribution with $n-2$ degrees of freedom. To get a confidence interval use the `confint` function:

```
confint(fit,parm="add")
```

```
##        2.5 %   97.5 %
## add 1.387014 2.089222
```

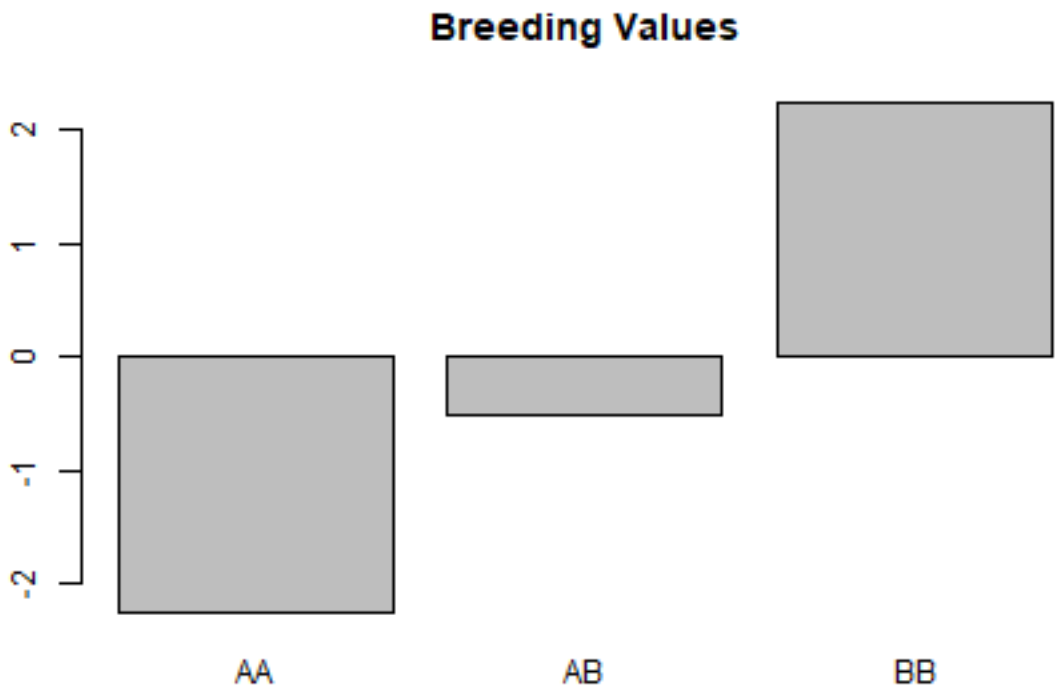
The regression coefficient for the variable `add` is 1.74. The coefficient corresponds to the allele substitution effect (α). Previously we have estimated allele and genotype frequencies for M227. The following table summarizes all genotypic values, all breeding values and the dominance deviations.

Genotyp A_iA_j	Genotypic value GV_{ij}	Breeding Value BV_{ij}	Dominance Deviation D_{ij}
A_1A_1	a	$2q\alpha$	$-2q^2d$
A_1A_2	d	$(q-p)\alpha$	$2pqd$
A_2A_2	$-a$	$-2p\alpha$	$-2p^2d$

Question 5: What are the breeding values for body weight based on the M227 locus?

Answer:

```
alpha <- -fit$coefficients["add"]
BV_AA <- 2*fA*alpha
BV_AB <- (fA-fB)*alpha
BV_BB <- -2*fA*alpha
BV <- c(BV_AA,BV_AB,BV_BB)
names(BV) <- c("AA","AB","BB")
barplot(BV, main="Breeding Values")
```



Now we want to compute the genetic variance associated with marker M227. The formula below shows that genetic variance for a single locus model σ^2_G consists of two components. The first component σ^2_A is called the **genetic additive variance** and the second component σ^2_D is termed **dominance variance**. Here σ^2_A corresponds to the variance of the breeding values. The variance of breeding values is also called the additive genetic variance, because as we have already seen the breeding values are additive in the number of favorable alleles. In populations where there is no additive genetic variance, individuals all have the same breeding value. Therefore, they will produce offspring with the same expected advantage (zero), and selection cannot generate any improvement over generations. Because σ^2_D corresponds to the variance of the dominance deviation effects it is called dominance variance.

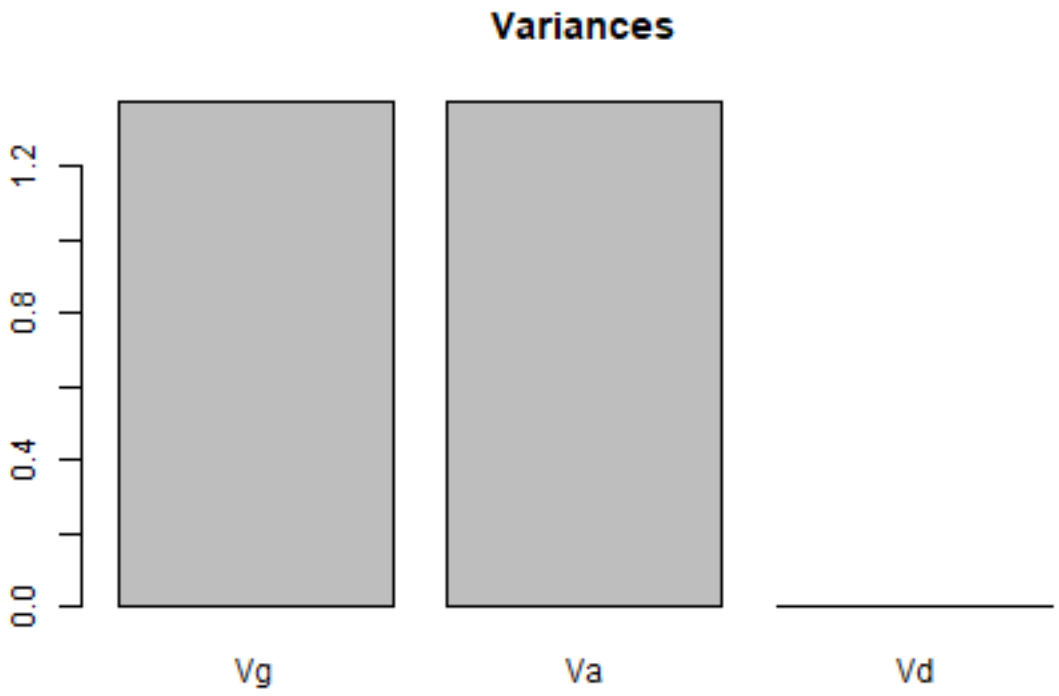
$$\begin{aligned}\sigma^2_G &= 2pq\alpha^2 + (2pqd)^2 \\ &= \sigma^2_A + \sigma^2_D\end{aligned}$$

Question 6: What is the additive genetic variance associated with M227 for body weight?

Answer:

```
alpha <- fit$coefficients["add"]
d <- 0
Va <- 2*fA*fB*alpha^2
Vd <-  (2*fA*fB*d)^2
```

```
Vg <- Va + Vd
V <- c(Vg,Va,Vd)
names(V) <- c("Vg","Va","Vd")
barplot(V, main="Variances")
```



Question 7: Should you have considered other factors in the linear model specified above?

Answer:

Now we will fit the full genetic model to locus M227 including both additive and dominance effects. The additive effect is modeled as previously shown by a variable `add` that is coded as -1, 0, and 1 (corresponding to -a, 0, a) for the genotypes AA, AB, and BB. The dominance effect is modeled by a variable `dom` that is coded as 0, 1, and 0 (corresponding to 0,d,0) for the genotypes AA, AB, and BB. The corresponding R code is shown below:

```
alleles <- c(-1,0,1)
names(alleles) <- c("AA","AB","BB")
mouse$add <- alleles[mouse$M227]
mouse$dom <- as.numeric(mouse$add==1)
fit <- lm(BW~sex + reps + add+dom, data=mouse)
summary(fit)
```

```
##
## Call:
## lm(formula = BW ~ sex + reps + add + dom, data = mouse)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.1773  -2.7642  -0.0437   2.5549  20.1121
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  34.5549     0.2666 129.635  < 2e-16 ***
## sexMale       8.4130     0.2413  34.863  < 2e-16 ***
## reps2        -0.3706     0.2850  -1.300   0.1937
## reps3         2.9062     0.3041   9.555  < 2e-16 ***
## add           2.0479     0.2580   7.937 4.82e-15 ***
## dom          -0.8811     0.5290  -1.665   0.0961 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.132 on 1168 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.5503, Adjusted R-squared:  0.5484
## F-statistic: 285.8 on 5 and 1168 DF,  p-value: < 2.2e-16
```

```
confint(fit,parm="add")
```

```
##        2.5 %   97.5 %
## add 1.541656 2.554078
```

```
confint(fit,parm="dom")
```

```
##         2.5 %    97.5 %
## dom -1.919038 0.1569128
```

The results from the linear model analysis suggest that only the additive genetic effect, `add`, is significantly different from 0. However in the following exercise we will be using the both the additive effect (`add`) and dominance effect (`dom`) estimated for locus M227, and the frequency of the positive allele (p) to explore the effect of changes in allele frequency.

Use the following shinyapp, https://shiny.cnsgenomics.com/Falconer2/, to understand the relationship between allelic substitution effect (α) and additive gene action (a), dominance gene action (d), and allele frequency (p).

Question 8: Use the estimated gene actions (Question 7) and the estimated allele frequency (Question 2) to obtain the predicted allelic substitution effect? Use rounded values if necessary

Answer:

Question 9: Does the value of α match the estimate of the (marginal) additive effect from Question 5?

Answer:

Question 10: How does α depend on a larger dominance gene action d (e.g., maximum value, 10)?

Answer:

Question 11: How does α depend on a different allele frequency p (e.g., 0.95)?

Answer:

Question 12: Under that new value of p, how does α depend on d (e.g., from the initial value of d to the maximum value)?

Answer:

Practical 3: Estimation of Genetic Parameters

Introduction:

In this practical we will estimate genetic parameters (heritability) for quantitative traits observed in the F2 mouse population. We will be using the REML method. This method allow for estimation of genetic parameters using phenotypic information for individuals from a general pedigree. REML is based on linear mixed model methodology and uses a likelihood approach to estimate genetic parameters. The REML method also require us to calculate an genetic relationship matrix using a recursive algorithm. These methods and algorithms are implemented in the R package `qgg`.

This package provides an infrastructure for efficient processing of large-scale genetic and phenotypic data including core functions for:

- fitting linear mixed models
- constructing genetic relationship matrices
- estimating genetic parameters (heritability and correlation)
- performing genomic prediction and genetic risk profiling
- single or multi-marker association analyses

We will also be using the qgg package for the remaining practicals.

Installation of the R package qgg:

You can install qgg from CRAN with:

```
install.packages("qgg")
```

You can install the latest version of qgg from github with:

```
#install.packages("devtools") # needed if devtools is not allready installed
library(devtools)
options(devtools.install.args=" --no-multiargs")
devtools::install_github("psoerensen/qgg")
```

Load R packages that will be used in this practical

```
library(qgg) # R package used for REML analysis
#install.packages("corrplot")
library(corrplot)
```

Explore mouse pedigree data

The mouse data set can be loaded using the following command:

```
mouse <- readRDS(url("https://github.com/psoerensen/bgcourse/raw/main/data/mouseqtl.rds"))
```

The mouse pedigree is loaded in a similar way using the following command:

```
pedigree <- readRDS(url("https://github.com/psoerensen/bgcourse/raw/main/data/pedigree.rds"))
```

Question 1: Which variables do we have in the pedigree? Use the `str` function to get a fast overview of the pedigree you are working.

Answer:

```
str(pedigree)
```

```
## 'data.frame':    1267 obs. of  6 variables:
##  $ id        : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ sire      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ dam       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ family    : Factor w/ 68 levels "0/0","1/2","11/12",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ sex       : chr  "Male" "Female" "Male" "Female" ...
##  $ generation: chr  "M6" "IC" "IC" "M6" ...
```

Question 2: How many individuals do we have in the pedigree?

Answer:

```
nrow(pedigree)
```

```
## [1] 1267
```

```
dim(pedigree)
```

```
## [1] 1267    6
```

Question 3: How many generations and number of mice in each generation do we have in the pedigree? Use the `table` function on the generation variable.

Answer:

```
table(pedigree$generation)
```

```
##
##   F1   F2   IC   M6
##   66 1177   12   12
```

Computing genetic relationship matrix for the mouse pedigree:

The REML analysis require us to calculate the genetic relationship matrix A. This is done using information about the id, mother, and father which is avaliable in our pedigree data file.

To illustrate this step we will first calculate it for a small part of the mouse pedigree. We are given the following pedigree and we want to compute the matrix A.

```
family <- c(13,14,84,1244,1248)
pedigree[family,]
```

```
##        id sire dam family    sex generation
## 13     13    0   0    0/0   Male         IC
## 14     14    0   0    0/0 Female         M6
## 84     84   13  14  13/14 Female         F1
## 1244 1244   78  84  78/84 Female         F2
## 1248 1248   78  84  78/84   Male         F2
```

The additive genetic relationship (A_{ij}) between the various sources (j) and the individual itself, i.e. the candidate to be evaluated (i), can be seen in the table below.

Relative	A_{ij}
Self	1.0
Unrelated	0
Mother	0.5
Father	0.5
Grandparent	0.25
Half-sib	0.25
Full-sib	0.5
Progeny	0.5

Answer:

Next we will compute the genetic relationship matrix for the entire mouse pedigree. The matrix A can be computed using a recursive algorithm implemented in the function `grm` from the qgg package. Use the command below to compute the genetic relationship matrix for the mouse pedigree:

```
A <- grm(pedigree=pedigree)
```

Question 4: What is the dimension of the genetic relationship matrix?

Answer:

```
dim(A)
```

```
## [1] 1267 1267
```

The number of rows and columns should be equal to the number of individuals in the pedigree. Check the first 5 individuals in the matrix using the following command:

```
A[1:5,1:5]
```

```
##   1 2 3 4 5
## 1 1 0 0 0 0
## 2 0 1 0 0 0
## 3 0 0 1 0 0
## 4 0 0 0 1 0
## 5 0 0 0 0 1
```

Question 5: Are these individuals related?

Answer:

To further explore the genetic relationship we compute the mean of diagonal elements of A using the following command:

```
mean(diag(A))
```

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

Question 6: How should we interpret this value?

Answer:

Previously we have determined the genetic relationship matrix for a small part of the mouse pedigree. We can extract the corresponding elements from the A matrix for the entire mouse pedigree using the following command:

```
ids <- c(13,14,84,1244,1248)
A[ids,ids]
```

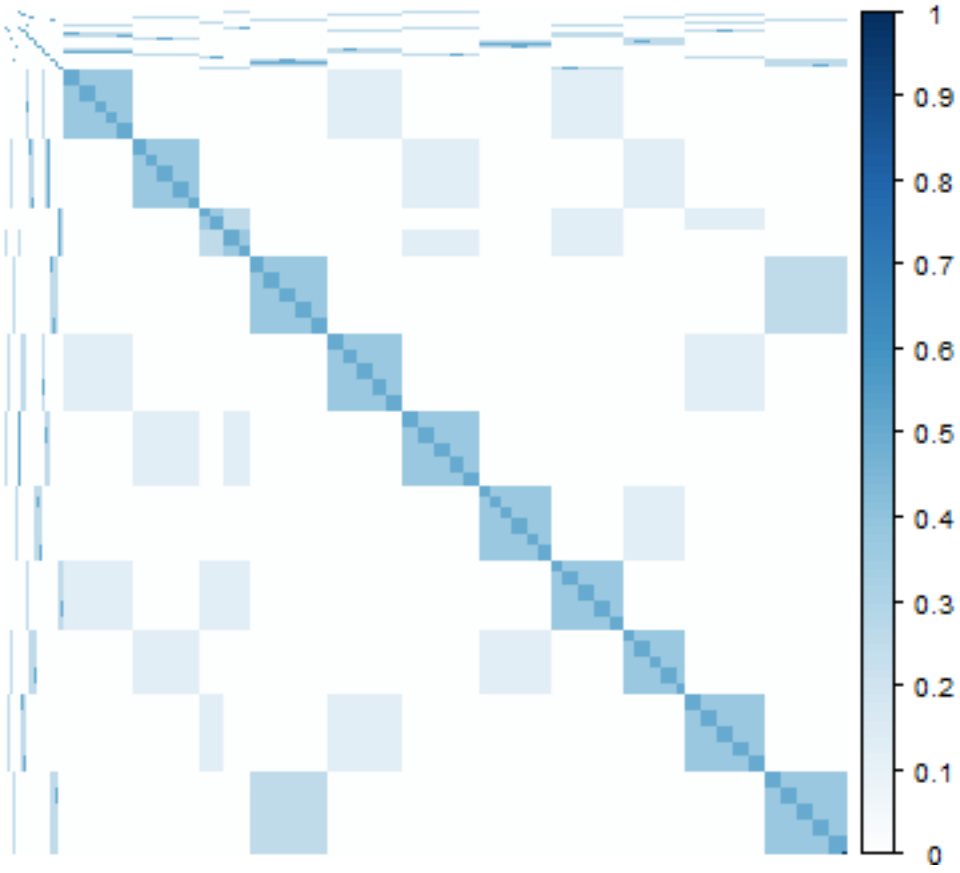
```
##        13   14  84 1244 1248
## 13   1.00 0.00 0.5 0.25 0.25
## 14   0.00 1.00 0.5 0.25 0.25
## 84   0.50 0.50 1.0 0.50 0.50
## 1244 0.25 0.25 0.5 1.00 0.50
## 1248 0.25 0.25 0.5 0.50 1.00
```

Question 6: Are the values in this part of the genetic relationship matrix the same as you have found using the "manual" approach?

Answer:

Make a plot of the genetic relationship matrix using the `corrplot` function from the corrplot R package:

```
corrplot(A, method="color", bg="white",
         outline=FALSE, col=NULL, tl.pos="n",is.corr = FALSE, xlab=FALSE, ylab=FALSE)
```



Question 7: Describe the plot you just made of the genetic relationship?

Answer:

Specifying the linear mixed model for the mouse data:

The next step is to prepare the linear mixed model for the mouse data. Recall that the linear mixed model contains the observation vector for the trait(s) of interest (y), the `fixed effects` that explain systematic differences in y, and the `random genetic effects` a and random residual effects e.

A matrix formulation of a general model equation is:

$$y = Xb + a + e$$

where

y : is the vector of observed values of the trait,
b : is a vector of fixed effects,
a : is a vector of random genetic effects,
e : is a vector of random residual effects,
X : is a known design matrix that relates the elements of b to their corresponding element in y.

In the statistical model (specified above) the random effects (a and e) and the phenotypes (y) are considered to be random variables which follow a multivariate normal distribution: In general terms the expectations of these random variables are:

$$\begin{aligned} E(y) &= Xb \\ E(a) &= 0 \\ E(e) &= 0 \end{aligned} \tag{1}$$

and the variance-covariance matrices are:

$$\begin{aligned} Var(a) &= A\sigma_a^2 \\ Var(e) &= I\sigma_e^2 \\ Var(y) &= A\sigma_a^2 + I\sigma_e^2 \end{aligned}$$

where $A\sigma_a^2$, and $I\sigma_e^2$ are square matrices of genetic and residual (co)variances among the individuals, respectively. In the previous section we have allready constructed the genetic relationship matrix A.

In order to perform the REML analysis we need to construct y and X from the mouse data. Let us just have a quick look at the mouse data again:

```
str(mouse)
```

```
## 'data.frame':    1177 obs. of  8 variables:
##  $ sire : Factor w/ 11 levels "25","28","34",..: 6 6 6 6 6 6 6 6 6 6 ...
##  $ dam  : Factor w/ 55 levels "26","27","29",..: 8 8 8 8 8 8 8 8 8 8 ...
##  $ sex  : Factor w/ 2 levels "Female","Male": 1 1 1 1 2 2 2 2 1 1 ...
##  $ reps : Factor w/ 3 levels "1","2","3": 1 1 1 1 1 1 1 1 2 2 ...
##  $ Gl   : num  187 136 115 125 112 190 169 159 111 89 ...
##  $ BW   : num  36.6 33.3 42.1 37.1 38.4 ...
##  $ M227 : Factor w/ 3 levels "AA","AB","BB": 2 1 2 2 2 1 2 1 2 2 ...
##  $ M1139: Factor w/ 3 levels "AA","AB","BB": 3 NA 1 1 1 2 3 3 2 2 ...
```

Here we will estimate the heritability for body weight. The vector of observed trait values for body weight can be extracted from the mouse data as follows:

```
y <- mouse[,"BW"]
```

Let us explore the trait values using the `head`, `tail` and `summary` functions:

```
head(y)
```

```
## [1] 36.65 33.29 42.07 37.15 38.39 39.82
```

```
tail(y)
```

```
## [1] 39.67 39.35 44.80 52.23 47.63 54.10
```

```
summary(y)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   23.04   34.06   38.32   38.72   43.40   60.28
```

To make the X matrix we need to decide which variables we should include as fixed effects in the model. We have sex, reps, sire, dam, M227 and M1139 in the mouse data frame.

Question 8: Which variables should we include as fixed effects in the model?

Answer:

The `model.matrix` function can be used to construct the X matrix in the linear mixed model specified above:

```
X <- model.matrix(BW ~ sex + reps, data=mouse)
```

We can use the `head` and `tail` functions to look at the X matrix:

```
head(X)
```

```
##    (Intercept) sexMale reps2 reps3
## 91           1       0     0     0
## 92           1       0     0     0
## 93           1       0     0     0
## 94           1       0     0     0
## 95           1       1     0     0
## 96           1       1     0     0
```

```
tail(X)
```

```
##      (Intercept) sexMale reps2 reps3
## 1262           1       0     0     1
## 1263           1       0     0     1
## 1264           1       1     0     1
## 1265           1       1     0     1
## 1266           1       1     0     1
## 1267           1       1     0     1
```

Estimating genetic parameters on the mouse data using REML:

The goal of the REML analysis to estimate the parameters (i.e. variance components σ_a^2 and σ_e^2) in the linear mixed model specified above. In this analysis we find the set of parameters which maximizes the **likelihood** of the data, i.e., the probability of observations given the model and its parameter estimates: $p(y|\hat{b}, \hat{\sigma}_a^2, \hat{\sigma}_e^2)$.

The input required the vector of observed values of the trait (y), the deisgn matrix for the fixed effects (X), and the genetic relationship matrix (A). The A matrix calculated previously include genetic relationships for all individuals in the pedigree. However only a subset of the inviduals have phenotypes recorded for body weight. Therefore we need to subset the A matrix as shown in the R code below:

```
ids <- rownames(X)
A <- A[ids,ids]
```

The REML method is implemented in the `greml` function from the "qgg" package. The REML analysis is done using the following command:

```
fit <- greml(y=y,X=X, GRM=list(A=A))
```

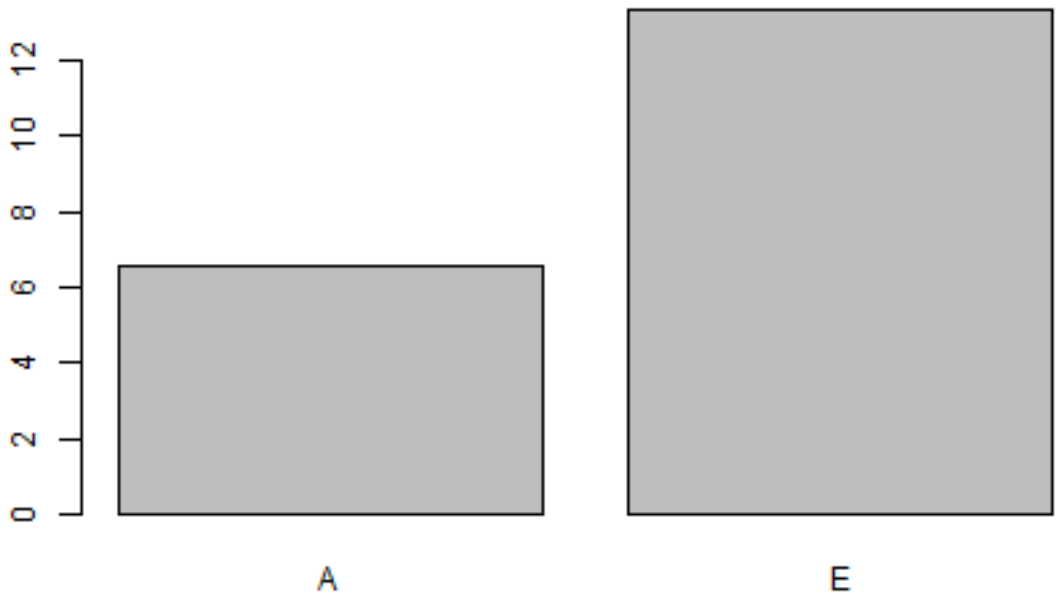
The fit object (i.e., output from the `greml` function) contains estimates of variance components, fixed and random effects, first and second derivatives of log-likelihood, and the asymptotic standard deviation of parameter estimates.

Our main interest is the variance components σ_a^2 and σ_e^2 which are in the `fit$theta` slot of the fit. The following commands extract and makes a barplot of the estimates of the variance components:

```
fit$theta
```

```
##         A         E
##  6.569611 13.384147
```

```
barplot(fit$theta)
```



The first element in the `theta` vector is the estimate of the additive genetic variance ($\hat{\sigma}_a^2$) and the second element is the estimate of the residual variance ($\hat{\sigma}_e^2$).

```
Va <- fit$theta[1]
Ve <- fit$theta[2]
Va
```

```
##        A
## 6.569611
```

```
Ve
```

```
##        E
## 13.38415
```

From the REML estimate of the variance components, the heritability can easily be computed by:

$$\hat{h}^2 = \hat{\sigma}_a^2/(\hat{\sigma}_a^2 + \hat{\sigma}_e^2) \tag{2}$$

where the hat ($\hat{}$) refers to estimators.

Question 9: What is the heritability for body weight?

Answer:

```
Va/(Va+Ve)
```

```
##         A
## 0.3292418
```

In the experiment the mice were feed ad libitum. Now we want to perform a simlar experiment where mice are reared under restricted feed intake, We will record phenotypes for body weight and blood glucose levels and use mice from the same F2 population.

Question 10: Should we re-estimate the heritability?

Answer:

Question 11: What is the heritability for glucose levels in the blood?

Answer:

```
y <- mouse[,"Gl"]
X <- model.matrix(Gl ~ sex + reps, data=mouse)
ids <- rownames(X)
A <- grm(pedigree=pedigree)
A <- A[ids,ids]
fit <- greml(y=y,X=X, GRM=list(A=A))
Va <- fit$theta[1]
Ve <- fit$theta[2]
Va/(Va+Ve)
```

```
##         A
## 0.4058474
```

Practical 4: Estimation of Breeding Values

Introduction:

In this practical we will estimate breeding values for quantitative traits in the mouse population. We will be using the BLUP method. This method allow for estimation of breeding values using phenotypic information for individuals from a general pedigree. BLUP is based on linear mixed model methodology and estimates of breeding values can be obtained by solving the mixed model equations. The BLUP method also require a genetic relationship matrix and estimates of variance components (e.g., σ^2_a and σ^2_e). Furthermore, we will compute reliabilities to determine how well we have estimated the breeding value in relation to the true breeding value. These methods and algorithms are implemented in the R package `qgg` introduced previously.

Load R packages that will be used in this practical

Use the following code to load the qgg package:

```
library(qgg) # R package used for REML/BLUP analysis
```

Explore mouse pedigree data

The mouse data and pedigre set can be loaded using the following commands:

```
mouse <- readRDS(url("https://github.com/psoerensen/bgcourse/raw/main/data/mouseqtl.rds"))
pedigree <- readRDS(url("https://github.com/psoerensen/bgcourse/raw/main/data/pedigree.rds"))
```

First let us have a quick look at the mouse data again. Use the `str` function to get a fast overview of the pedigree you are working.

```
str(pedigree)
```

```
## 'data.frame':    1267 obs. of  6 variables:
##  $ id        : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ sire      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ dam       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ family    : Factor w/ 68 levels "0/0","1/2","11/12",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ sex       : chr  "Male" "Female" "Male" "Female" ...
##  $ generation: chr  "M6" "IC" "IC" "M6" ...
```

The number of individuals and generations in the pedigree can be found using the following commands:

```
nrow(pedigree)
```

```
## [1] 1267
```

```
dim(pedigree)
```

```
## [1] 1267    6
```

```
table(pedigree$generation)
```

```
##
##   F1   F2   IC   M6
##   66 1177   12   12
```

Computing genetic relationship matrix for the mouse pedigree:

The genetic relationship matrix A is used for estimating breeding values. The matrix A can be computed using the recursive algorithm implemented in the function `grm` from the qgg package. Use the command below to compute the genetic relationship matrix for the mouse pedigree:

```
A <- grm(pedigree=pedigree)
```

The dimension of the genetic relationship matrix can be determined using the following command:

```
dim(A)
```

```
## [1] 1267 1267
```

The number of rows and columns should be equal to the number of individuals in the pedigree.

Specifying the linear mixed model for the mouse data:

The next step is to prepare the linear mixed model for the mouse data. Recall that the linear mixed model contains the observation vector for the trait(s) of interest (y), the `fixed effects` that explain systematic differences in y, and the `random genetic effects` a and random residual effects e.

A matrix formulation of a general model equation is:

$$y = Xb + a + e$$

where

y : is the vector of observed values of the trait,
b : is a vector of fixed effects,
a : is a vector of random genetic effects,
e : is a vector of random residual effects,
X : is a known design matrix that relates the elements of b to their corresponding element in y.

In the statistical model (specified above) the random effects (a and e) and the phenotypes (y) are considered to be random variables which follow a multivariate normal distribution. In general terms the expectations of these random variables are:

$$\begin{aligned} a &\sim MVN(0, A\sigma_a^2) \\ e &\sim MVN(0, I\sigma_e^2) \\ y &\sim MVN(Xb, V) \end{aligned} \tag{3}$$

where $A\sigma_a^2$, and $I\sigma_e^2$ are square matrices of genetic and residual (co)variances among the individuals, respectively, and $V = A\sigma_a^2 + I\sigma_e^2$ is the overall phenotypic covariance matrix. In the previous section we have allready constructed the genetic relationship matrix A.

In order to specify the linear mixed model we need to construct y and X from the mouse data. Let us just have a quick look at the mouse data again:

```
str(mouse)
```

```
## 'data.frame':    1177 obs. of  8 variables:
##  $ sire : Factor w/ 11 levels "25","28","34",..: 6 6 6 6 6 6 6 6 6 6 ...
##  $ dam  : Factor w/ 55 levels "26","27","29",..: 8 8 8 8 8 8 8 8 8 8 ...
##  $ sex  : Factor w/ 2 levels "Female","Male": 1 1 1 1 2 2 2 2 1 1 ...
##  $ reps : Factor w/ 3 levels "1","2","3": 1 1 1 1 1 1 1 1 2 2 ...
##  $ Gl   : num  187 136 115 125 112 190 169 159 111 89 ...
##  $ BW   : num  36.6 33.3 42.1 37.1 38.4 ...
##  $ M227 : Factor w/ 3 levels "AA","AB","BB": 2 1 2 2 2 1 2 1 2 2 ...
##  $ M1139: Factor w/ 3 levels "AA","AB","BB": 3 NA 1 1 1 2 3 3 2 2 ...
```

Here we will estimate breeding values for body weight. The vector of observed trait values for body weight can be extracted from the mouse data as follows:

```
y <- mouse[,"BW"]
```

Let us explore the trait values using the `head`, `tail` and `summary` functions:

```
head(y)
```

```
## [1] 36.65 33.29 42.07 37.15 38.39 39.82
```

```
tail(y)
```

```
## [1] 39.67 39.35 44.80 52.23 47.63 54.10
```

```
summary(y)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   23.04   34.06   38.32   38.72   43.40   60.28
```

To make the X matrix we need to decide which variables we should include as fixed effects in the model. Here we use the variables sex and reps. The `model.matrix` function can be used to construct the X matrix in the linear mixed model specified above:

```
X <- model.matrix(BW ~ sex + reps, data=mouse)
```

We can use the `head` and `tail` functions to look at the X matrix:

```
head(X)
```

```
##    (Intercept) sexMale reps2 reps3
## 91           1       0     0     0
## 92           1       0     0     0
## 93           1       0     0     0
## 94           1       0     0     0
## 95           1       1     0     0
## 96           1       1     0     0
```

```
tail(X)
```

```
##      (Intercept) sexMale reps2 reps3
## 1262           1       0     0     1
## 1263           1       0     0     1
## 1264           1       1     0     1
## 1265           1       1     0     1
## 1266           1       1     0     1
## 1267           1       1     0     1
```

Question 1: Why do we not include the effect of sire and dam in the model?

Answer:

Estimating genetic parameters on the mouse data using REML:

The BLUP analysis is based on estimates of the variance components (i.e. σ_a^2 and σ_e^2). The variance components are estimated using REML method. The input required the vector of observed values of the trait (y), the design matrix for the fixed effects (X), and the genetic relationship matrix (A).

The genetic relationship matrix A include relationships for all individuals in the pedigree. However only a subset of the individuals have phenotypes recorded for body weight and glucose levels in blood. Therefore we need to subset the A matrix:

```
ids <- rownames(X)
A <- A[ids,ids]
```

The REML analysis is done using the following command:

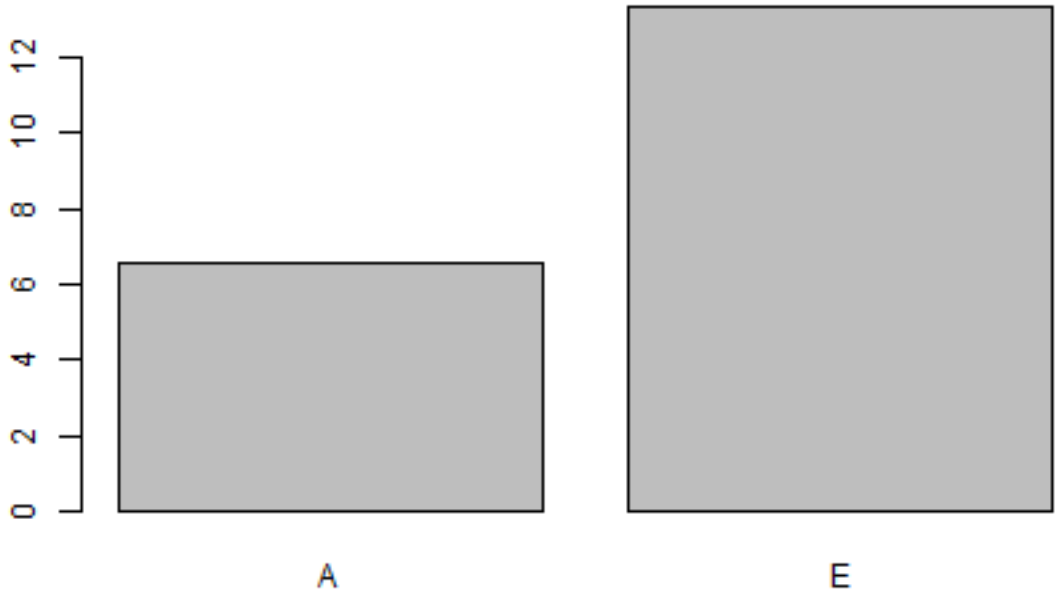
```
fit <- greml(y=y,X=X, GRM=list(A=A))
```

The fit object contains estimates of variance components, fixed and random effects, first and second derivatives of log-likelihood, and the asymptotic standard deviation of parameter estimates. Our main interest is the variance components σ_a^2 and σ_e^2 which are in the `fit$theta` slot of the fit. The following commands extract and makes a barplot of the estimates of the variance components:

```
fit$theta
```

```
##         A         E
##  6.569611 13.384147
```

```r
Va <- fit$theta[1] # First element in theta is the additive genetic variance
Ve <- fit$theta[2] # Second element in theta is the residual variance
barplot(fit$theta)
```



Estimating breeding values for traits in the mouse data using BLUP:

The goal of the BLUP analysis is the estimate the fixed ,b, and random genetic effects, a, in the linear mixed model specified above. This can be done using the `BLUE` and 'BLUP´ equations shown below:

The best linear unbiased prediction (BLUP) of $\hat{a}$ is:

$$\hat{a} = A\sigma_a^2 V^{-1}(y - X\hat{b}) \tag{4}$$

The best linear unbiased estimator (BLUE) of $\hat{b}$ is:

$$\hat{b} = (X'V^{-1}X)^{-1}X'V^{-1}y \tag{5}$$

The matrix $(X'V^{-1}X)^{-1}$ denotes the inverse of the matrix $(X'V^{-1}X)$.

We have allready determined y and X and therefore just need to construct the phenotypic covariance matrix V (and it's inverse). This can be done using the following lines of R code:

```
n <- nrow(X)        # Number of individuals in the data set
I <- diag(1,n)      # Identity matrix for residual effects
V <-  A*Va + I*Ve   # Phenotypic variance covariance matrix
Vi <- solve(V)      # Inverse of phenotypic covariance matrix
```

The solution to the fixed effects, b, can be found using the following R command:

```
bhat <- solve(t(X) %*% Vi %*% X)%*%t(X) %*% Vi %*% y
bhat
```

```
##                   [,1]
## (Intercept) 33.8873546
## sexMale      8.3453194
## reps2       -0.3684327
## reps3        2.6411388
```

The solution to the random genetic effects, a, can be found using the following R command:

```
ahat <- (A*Va)%*% Vi %*% (y-X%*%bhat)
head(ahat)
```

```
##            [,1]
## 91 -0.001564943
## 92 -0.663690910
## 93  1.066507303
## 94  0.096965707
## 95 -1.303217779
## 96 -1.021420120
```

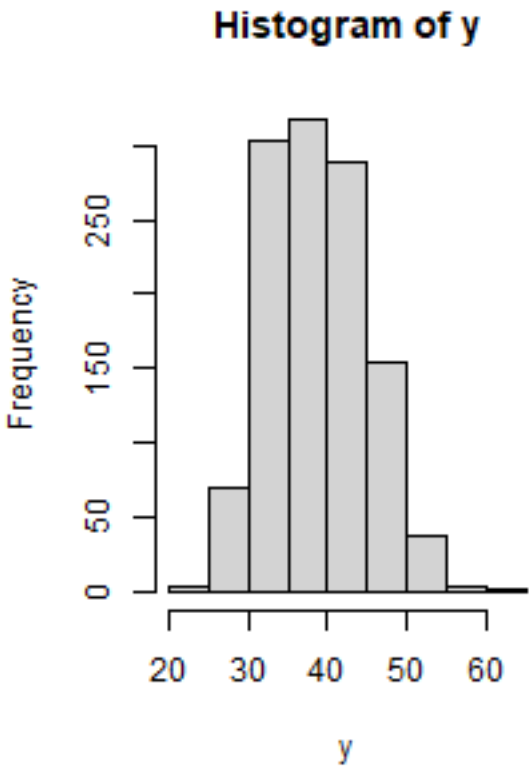
```
tail(ahat)
```

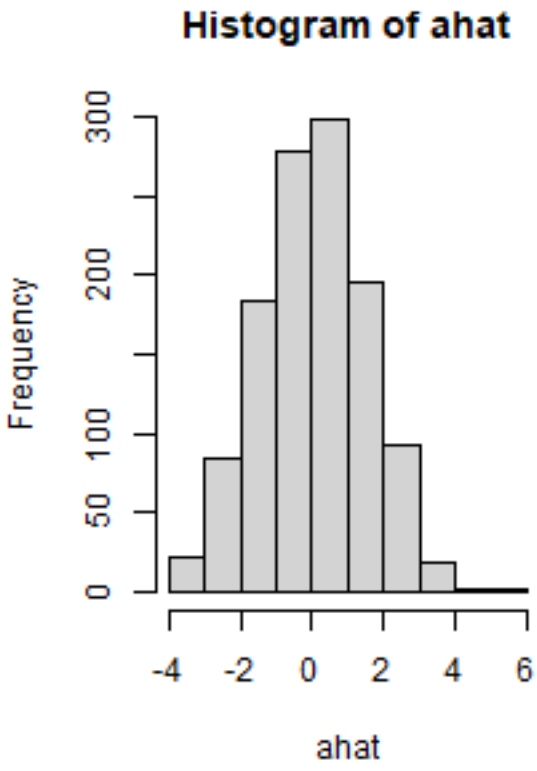
```
##          [,1]
## 1262 1.111041
## 1263 1.047981
## 1264 0.477426
## 1265 1.941591
## 1266 1.035109
## 1267 2.310096
```

Question 2: Make histogram for y and the estimated breeding values. What do you think about their distribution?

Answer:

```
layout(matrix(1:2,ncol=2))
hist(y)
hist(ahat)
```

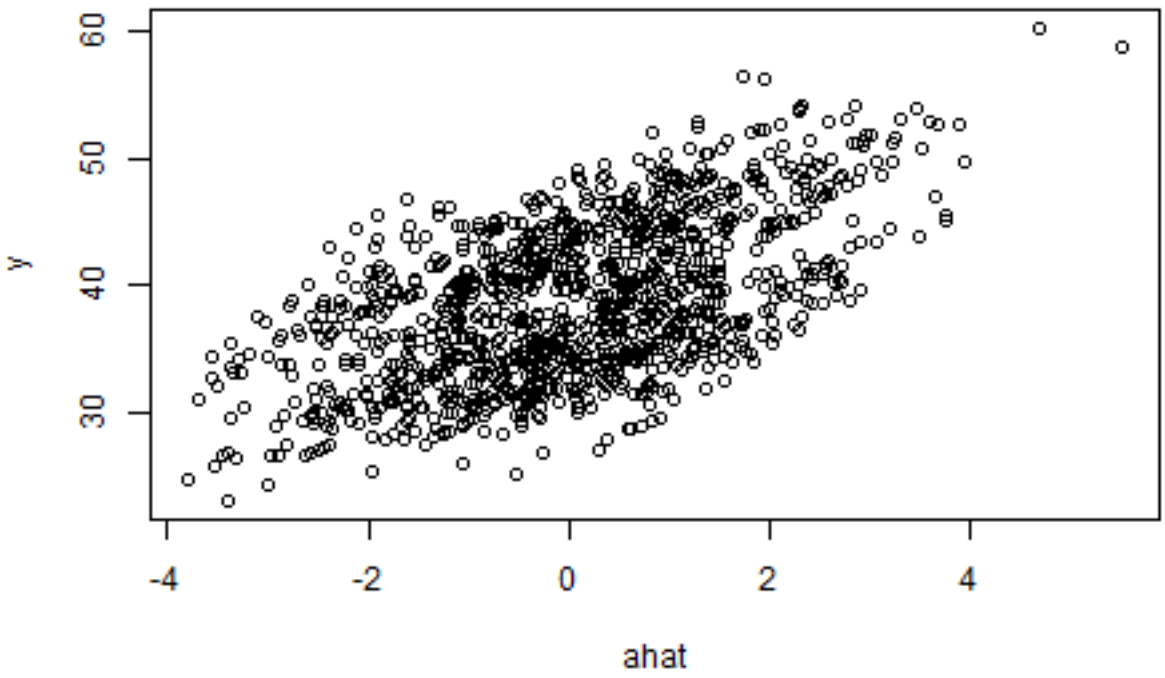




Question 3: Make a scatter plot of y and the estimated breeding values. What do you think about their relationship?

Answer:

```
plot(ahat,y)
```




```
cor(ahat,y)
```

```
##           [,1]
## [1,] 0.5708277
```

Question 4: Which of the sires has the highest breeding value for body weight? Which dam has the highest breeding value for body weight?

```
ids <- rownames(X)
A <- grm(pedigree=pedigree)
ahat <- (A[,ids]*Va)%*% Vi %*% (y-X%*%bhat)
head(ahat)
```

```
##         [,1]
## 1  1.5347891
## 2  1.5347891
## 3 -0.8589335
## 4 -0.8589335
## 5 -0.5407432
## 6 -0.5407432
```

```
tail(ahat)
```

```
##          [,1]
## 1262 1.111041
## 1263 1.047981
## 1264 0.477426
## 1265 1.941591
## 1266 1.035109
## 1267 2.310096
```

```
ids_sire <- unique(pedigree$sire)
ids_dam <- unique(pedigree$dam)
sort(ahat[rownames(ahat)%in%ids_sire,])
```

```
##          51           9          40          25           3          85
## -1.84832353 -1.73579652 -1.70022244 -0.87046098 -0.85893355 -0.75837120
##          28           5          63          17          21          34
## -0.63937383 -0.54074318 -0.52743859 -0.40905151 -0.11252701 -0.10615906
##           7          11          23          78          13          15
## -0.08532606 -0.07695294 -0.04642724  0.24057510  0.68208036  0.68208036
##          19          72           1          79          69
##  0.96680814  1.43552985  1.53478914  1.80566599  3.08110571
```

```
sort(ahat[rownames(ahat)%in%ids_dam,])
```

```
##          26          47          58          48          56          29
## -3.365322998 -3.329528267 -3.224397087 -2.703590554 -1.980714112 -1.915216260
##          46          10          86          89          59          74
## -1.839637115 -1.735796519 -1.600304121 -1.510410842 -1.340775327 -1.297453682
##          52          50          45          49          54          90
## -1.285914822 -1.277075596 -1.231350628 -1.152420570 -1.147065162 -1.012595106
##           4          35          80           6          62          36
## -0.858933547 -0.832504416 -0.618370882 -0.540743183 -0.527436542 -0.483533806
##          38          18          33          82          41          31
## -0.461424689 -0.409051511 -0.392776477 -0.340256073 -0.247210944 -0.139915988
##          22           8          12          24          70          57
## -0.112527012 -0.085326059 -0.076952936 -0.046427237 -0.007581342  0.011503061
##          61          65          37          44          88          66
##  0.132637169  0.166400547  0.249953318  0.338571044  0.453309089  0.515442959
##          14          16          77          27          83          20
##  0.682080364  0.682080364  0.718427198  0.800049793  0.867918000  0.966808139
##          84          43          73          67          30          39
##  0.984852636  0.992532759  0.997447238  1.079766729  1.081110987  1.308677997
##          42          32          55           2          87          68
##  1.309009661  1.408889155  1.417475259  1.534789135  1.565011598  1.570600737
##          60          75          81          76          53          71
##  1.702208101  1.971620505  2.074752880  2.143946188  2.197026694  2.439284048
##          64
##  4.330207270
```

Answer:

Computing reliabilities for the estimated breeding values for traits in the mouse data:

The last step is to compute reliabilities to determine how well we have estimated the breeding value in relation to the true breeding value. The reliability (i.e., variances of prediction error are often expressed as a number going from 0 to 1. The general formula is:

REL = (Var(TBV) - Var(TBV-EBV))/(Var(TBV)

TBV=True Breeding Values EBV=Estimated Breeding Values

The standard error of prediction, or SEP, is the square root of the variance of prediction error.

Estimation of breeding values and reliabilities can also be done by solving the mixed model equation. Procedures for solving the mixed model equations are implemented in the `gsolve` function from the R package `qgg` introduced previously. The input to this function is y, X, A or G and estimates of the variance components (e.g., σ^2_a and σ^2_e).

```
A <- grm(pedigree=pedigree)
fit <- gsolve(y=y,X=X, GRM=list(A=A), Ve=Ve, Va=Va)
```

We can use the `str`, `head` and `tail` functions to look at the output from the `gsolve` function:

```
str(fit)
head(fit)
tail(fit)
```

Question 5: Which mouse has the hight reliability and what could be the reason for this?

```
o <- order(fit$rel)
tail(fit[o,])
```

Answer:

To further explore the reliabilities we can make a plot of them using the following command:

```
plot(fit$rel)
F0 <- pedigree$generation=="IC" | pedigree$generation=="M6"
F1 <- pedigree$generation=="F1"
F2 <- pedigree$generation=="F2"
n <- length(fit$rel)
points(x=(1:n)[F0],y=fit$rel[F0],col="blue", pch=4, cex=2, lwd=2 )
points(x=(1:n)[F1],y=fit$rel[F1],col="red", pch=4, cex=2, lwd=2 )
points(x=(1:n)[F2],y=fit$rel[F2],col="green", pch=4, cex=2, lwd=2 )
```

The reliabilities for generation F0 is blue, F1 is red and F2 is green. What we can observed is that mice from the F0 generation (i.e., IC and M6) has the lowest reliability.

Question 6: Could you explain this?

Answer:

To further explore the value of using phenotypic information from different relatives consider the general formula for reliability of estimated breeding value using different sources of information: {-}

$$r^2_{a,\hat{a}} = \frac{(a')^2 n h^2}{1 + (n-1)r} \tag{6}$$

where a' is the genetic relationship between the breeding individual and individuals with phenotypes, n is the number of phenotypic records, h^2 is the trait heritability, and r is correlation between individuals with observations ($r = a^{''} h2 + c2$, where $a^{''}$ = genetic relationship between individuals with records and target, $c2$ = common environmental component).

We want to compare the reliabity of the estimated breeding value for an individual computed based on phenotypic observation on different types of relatives. Assume that the trait narrow sense trait heritability $h^2 = 0.35$ and that the common environmental component $c2 = 0$.

What is the reliability if we compute the breeding values based on:

1) Own
2) Mother
3) 50 paternal halfsibs (same father)
4) 20 offspring that halfsibs (different mothers)

Question 8: Which phenotypic information source give the highest reliability?

Answer:

Question 8: Which of the sires has the highest breeding value for blood glucose levels?

Answer:

Practical 5: Estimation of Genomic Breeding Values

Introduction:

In this practical we will estimate genomic breeding values for quantitative traits in the mouse population. We will be using the GBLUP method. This method allow for estimation of genomic breeding values using phenotypic and genotypic information for individuals from a general pedigree. GBLUP is based on linear mixed model methodology and estimates of genomic breeding values can be obtained by solving the mixed model equations. The GBLUP method also require a genomic relationship matrix estimated from genetic marker data and estimates of variance components (e.g., σ_a^2 and σ_e^2). These methods are implemented in the R package `qgg` introduced previously.

Load R packages that will be used in this practical

Use the following code to load the qgg package:

```
library(qgg) # R package used for REML/BLUP analysis
```

Explore mouse pedigree data

The mouse phenotype data, pedigree and genotype data can be loaded using the following commands:

```
mouse <- readRDS(url("https://github.com/psoerensen/bgcourse/raw/main/data/mouseqtl.rds"))
pedigree <- readRDS(url("https://github.com/psoerensen/bgcourse/raw/main/data/pedigree.rds"))
genotypes <- readRDS(url("https://github.com/psoerensen/bgcourse/raw/main/data/genotypes_imputed.rds"))
```

First let us have a quick look at the mouse genotype data. Use the `str` function to get a fast overview of the genotypes you are working with.

The genotypes for each marker are coded as 0,1 or 2 corresponding to the number of copies of the minor allele. The number of individuals and number of genetic markers in the data can be found using the following commands:

```
nrow(genotypes)
```

```
## [1] 1267
```

```
dim(genotypes)
```

```
## [1] 1267 1813
```

Computing genomic relationship matrix using marker data:

The genomic relationship matrix G is used for estimating genomic breeding values. The matrix G can be computed using genetic marker data. This is implemented in the function `grm` from the `qgg` package. Use the command below to compute the genomic relationship matrix for the mouse pedigree:

```
W <- scale(genotypes) # here we center and scale columns in genotypes (i.e., mean=0, sd=1)
G <- grm(W=W)
```

The dimension of the genomic relationship matrix can be determined using the following command:

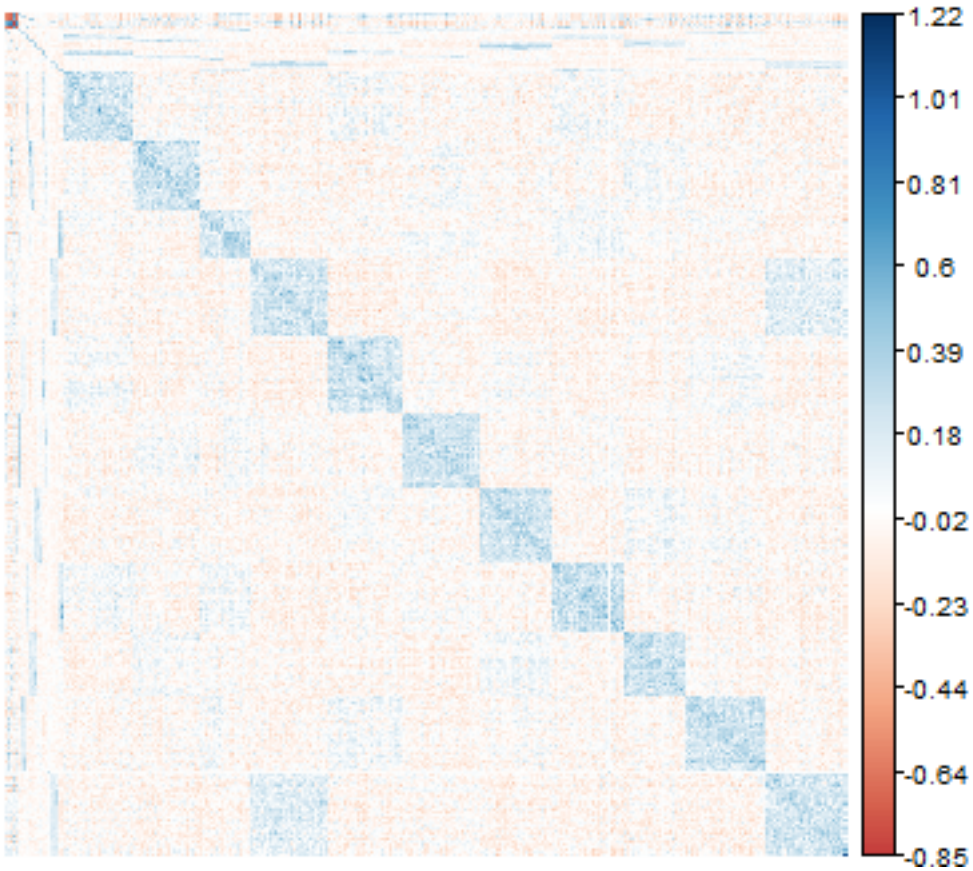
```
dim(G)
```

```
## [1] 1267 1267
```

The number of rows and columns should be equal to the number of individuals in the genotype matrix.

Make a plot of the genomic relationship matrix using the `corrplot` function from the corrplot R package:

```
corrplot(G, method="color", bg="white",
         outline=FALSE, col=NULL, tl.pos="n",is.corr = FALSE, xlab=FALSE, ylab=FALSE)
```



Question 1: Describe the plot you just made of the genomic relationship?

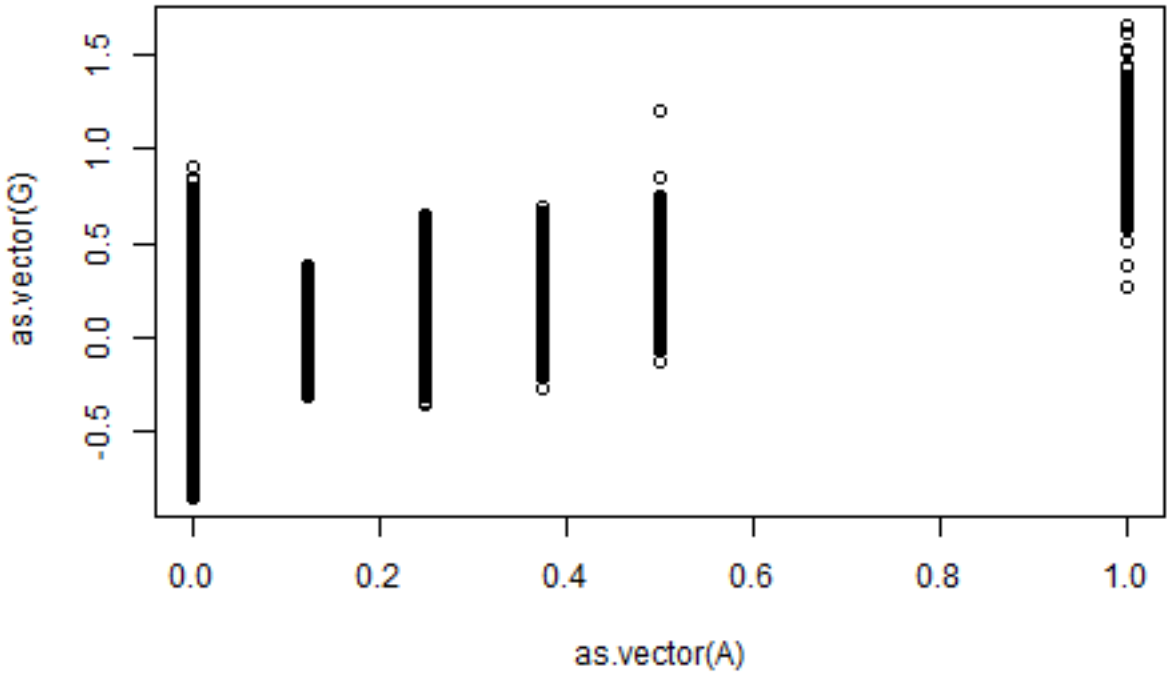
Answer:

To better compare the pedigree based genetic relationship matrix A and the genomic relationship matrix G we can make a scatter plot of the values from the two matrices:

```
A <- grm(pedigree=pedigree)
dim(A)
```

```
## [1] 1267 1267
```

```
plot(as.vector(A),as.vector(G))
```



Question 2: Are the two relationship matrices similar?

Answer:

Specifying the linear mixed model for the mouse data:

The next step is to prepare the linear mixed model for the mouse data. The linear mixed model contains the observation vector for the trait(s) of interest (y), the `fixed effects` that explain systematic differences in y, and the `random genetic effects` a and random residual effects e.

A matrix formulation of a general model equation is:

$$y = Xb + a + e$$

where

y : is the vector of observed values of the trait,

b : is a vector of fixed effects,

a : is a vector of random genetic effects,

e : is a vector of random residual effects,

X : is a known design matrix that relates the elements of b to their corresponding element in y.

In the statistical model (specified above) the random effects (a and e) and the phenotypes (y) are considered to be random variables which follow a multivariate normal distribution. In general terms the expectations of these random variables are:

$$\begin{aligned} a &\sim MVN(0, G\sigma_a^2) \\ e &\sim MVN(0, I\sigma_e^2) \\ y &\sim MVN(Xb, V) \end{aligned} \tag{7}$$

where $G\sigma_a^2$, and $I\sigma_e^2$ are square matrices of genetic and residual (co)variances among the individuals, respectively, and $V = G\sigma_a^2 + I\sigma_e^2$ is the overall phenotypic covariance matrix.

The main difference is that we use the genomic relationship matrix G instead of the pedigree based genetic relationship matrix A.

In order to specify the linear mixed model we need to construct y and X from the mouse data. Let us just have a quick look at the mouse data again:

Here we will estimate genomic breeding values for body weight. The vector of observed trait values for body weight can be extracted from the mouse data as follows:

```
y <- mouse[,"BW"]
```

To make the X matrix we need to decide which variables we should include as fixed effects in the model. Here we use the variables sex and reps. The `model.matrix` function can be used to construct the X matrix in the linear mixed model specified above:

```
X <- model.matrix(BW ~ sex + reps, data=mouse)
```

Estimating genetic parameters on the mouse data using REML:

The GBLUP analysis is based on estimates of the variance components (i.e. σ_a^2 and σ_e^2). The variance components are estimated using REML method. The input required the vector of observed values of the trait (y), the design matrix for the fixed effects (X), and the genetic relationship matrix (A).

The genetic relationship matrix A include relationships for all individuals in the pedigree. However only a subset of the individuals have phenotypes recorded for body weight. Therefore we need to subset the A matrix. The REML analysis is done using the following command:

```
ids <- rownames(X)
fit <- greml(y=y,X=X, GRM=list(A=A[ids,ids]))
```

The fit object contains estimates of variance components σ_a^2 and σ_e^2 which are in the `fit$theta` slot of the fit:

```
fit$theta
```

```
##         A         E
##  6.569611 13.384147
```

```
Va <- fit$theta[1] # First element in theta is the additive genetic variance
Ve <- fit$theta[2] # Second element in theta is the residual variance
```

Estimating genomic breeding values for traits in the mouse data using GBLUP:

The goal of the GBLUP analysis is the estimate the fixed ,b, and random genetic effects, a, in the linear mixed model specified above. This can be done using the `BLUE` and 'BLUP´ equations as shown previously. The best linear unbiased prediction (BLUP) of $\hat{a}$ is:

$$\hat{a} = G\sigma_a^2 V^{-1}(y - X\hat{b}) \tag{8}$$

The best linear unbiased estimator (BLUE) of $\hat{b}$ is:

$$\hat{b} = (X'V^{-1}X)^{-1}X'V^{-1}y \tag{9}$$

.

We have allready determined y and X and therefore just need to construct the phenotypic covariance matrix V (and it's inverse). This can be done using the following lines of R code:

```
ids <- rownames(X) # Individuals with phenotypes
n <- nrow(X)       # Number of individuals in the data set
I <- diag(1,n)     # Identity matrix for residual effects
V <-  G[ids,ids]*Va + I*Ve  # Phenotypic variance covariance matrix
Vi <- solve(V)     # Inverse of phenotypic covariance matrix
```

The solution to the fixed effects, b, can be found using the following R command:

```
bhat <- solve(t(X) %*% Vi %*% X)%*%t(X) %*% Vi %*% y
bhat
```

```
##                   [,1]
## (Intercept) 33.9244213
## sexMale      8.4114162
## reps2       -0.5009359
## reps3        2.6993539
```

The solution to the random genetic effects, a, can be found using the following R command:

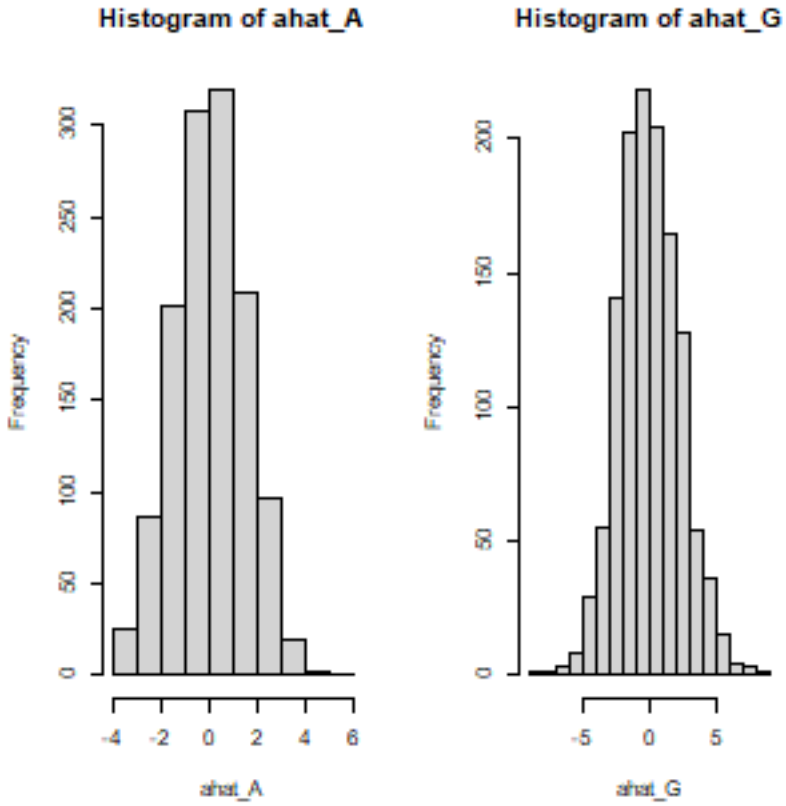
```
ahat_G <- (G[,ids]*Va)%*% Vi %*% (y-X%*%bhat)  # Genomic based BLUP
```

```
V <-  A[ids,ids]*Va + I*Ve  # Phenotypic variance covariance matrix
Vi <- solve(V)     # Inverse of phenotypic covariance matrix
bhat <- solve(t(X) %*% Vi %*% X)%*%t(X) %*% Vi %*% y
ahat_A <- (A[,ids]*Va)%*% Vi %*% (y-X%*%bhat)  # Genomic based BLUP
```

Question 3: Make histogram for the estimated (genomic) breeding values. What do you think about their distribution?

Answer:

```
layout(matrix(1:3,ncol=3))
hist(ahat_A)
hist(ahat_G)
```



Question 4: Which of the sires has the highest genomic breeding value for body weight? Which of the sires has the highest breeding value for body weight?

Answer:

```
ids_sire <- as.character(unique(pedigree$sire))[-1]
sort(ahat_G[ids_sire,])
```

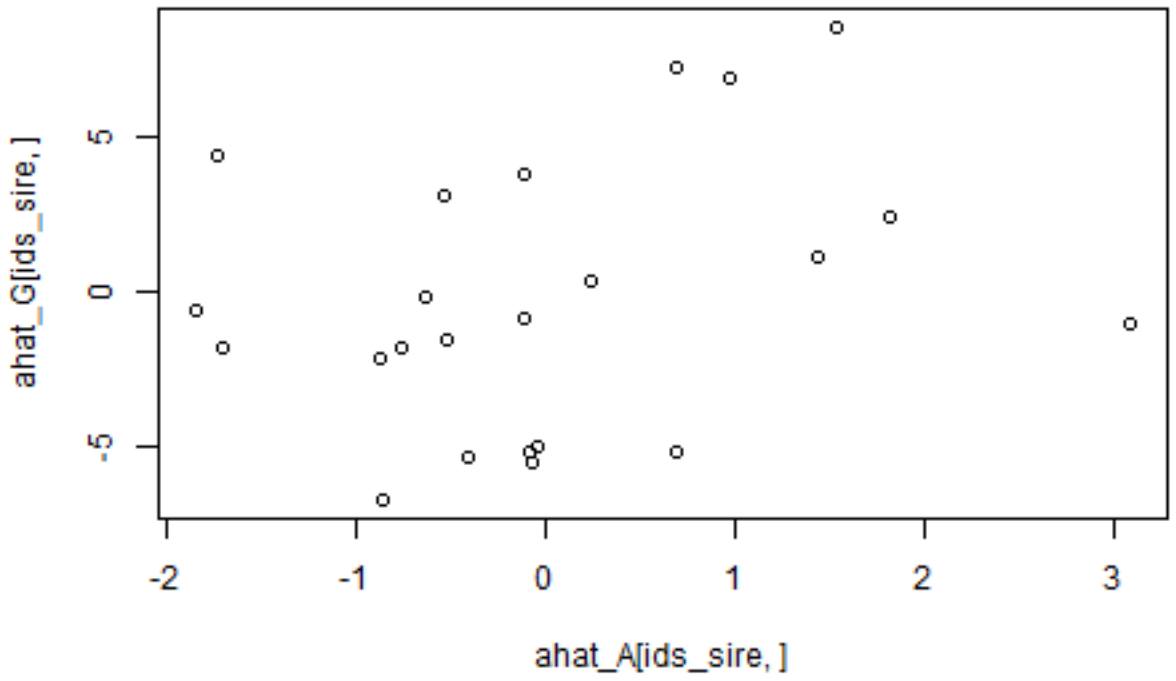
```
##          3         11         17          7         13         23         25
## -6.7678497 -5.5863038 -5.4031113 -5.2552701 -5.2480131 -5.0091092 -2.1866988
##         85         40         63         69         34         51         28
## -1.8641167 -1.8557966 -1.5441644 -1.0810038 -0.9225157 -0.6494719 -0.2121856
```

```
##         78         72         79          5         21          9         19
##  0.3515734  1.1039104  2.3753030  3.0664318  3.8026316  4.4157806  6.8656149
##         15          1
##  7.2855319  8.5783126
```

```
sort(ahat_A[ids_sire,])
```

```
##          51           9          40          25           3          85
## -1.84832353 -1.73579652 -1.70022244 -0.87046098 -0.85893355 -0.75837120
##          28           5          63          17          21          34
## -0.63937383 -0.54074318 -0.52743859 -0.40905151 -0.11252701 -0.10615906
##           7          11          23          78          15          13
## -0.08532606 -0.07695294 -0.04642724  0.24057510  0.68208036  0.68208036
##          19          72           1          79          69
##  0.96680814  1.43552985  1.53478914  1.80566599  3.08110571
```

```
plot(ahat_A[ids_sire,],ahat_G[ids_sire,])
```

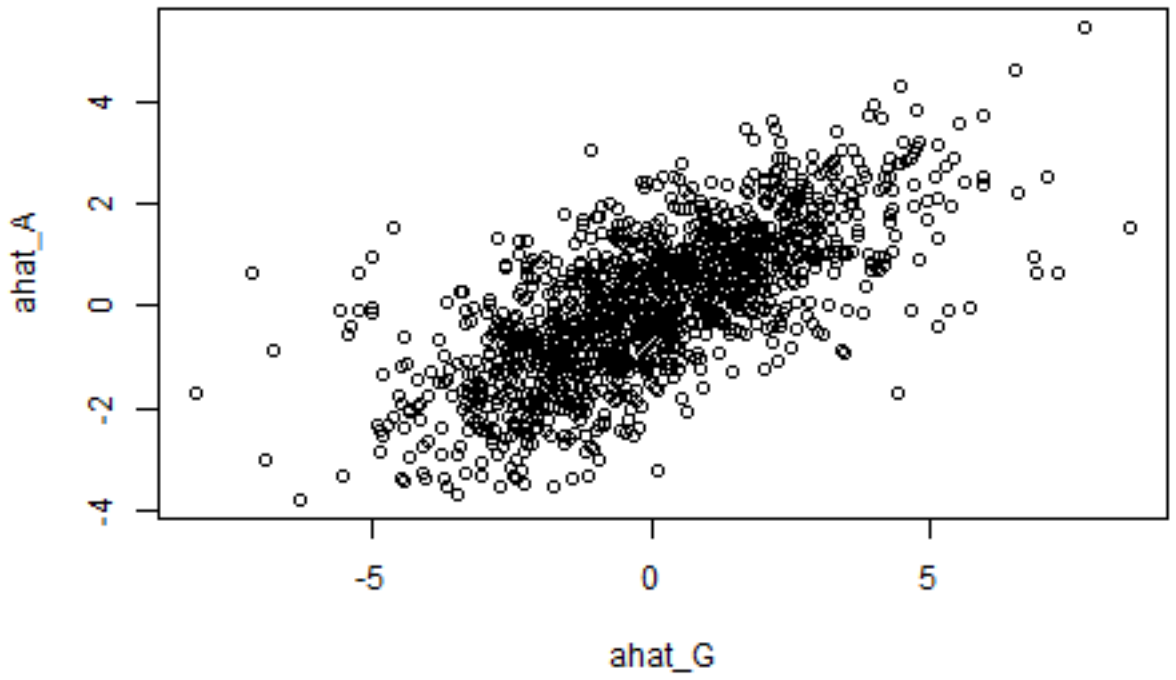


We want to compare the breeding values estimated using pedigree and genomic information. This can be done by computing the correlation or by making a scatter plot for the breeding values obtained using pedigree or genetic marker information.

Question 5: Make a scatter plot and compute correlation for the estimated breeding values and genomic breeding values. What do you think about their relationship?

Answer:

```
plot(ahat_G,ahat_A)
```




```
cor(ahat_G,ahat_A)
```

```
##           [,1]
## [1,] 0.6996325
```