ETH Zurich
D-USYS
Institute of Agricultural Sciences

Solutions To Exam
Livestock Breeding and Genomics
FS 2021

Peter von Rohr

DATE *17. DECEMBER 2021*
BEGIN *09:15*
END *11:15*

Name:

Legi-Nr:

Problem	Maximum Number of Points	Number of Points Reached
1	67	
2	14	
3	17	
4	22	
5	24	
Total	144	

Problem 1 Numerator Relationship Matrix and Inbreeding

Given is the following list of animals.

Gegeben ist die folgende Tierliste.

Animal	Birthdate	Sire	Dam
GINA	18.01.2020	HARRY	CH120.1208.5899.1
CH 120.1208.5899.1	22.11.2015	NA	Gitta
Gitta	31.05.2001	HARRY	Gibsy
Gibsy	09.12.1990	Ginger Hill Angus 133	Bianca
HARRY	22.02.1997	HIBISCUS	WALBURGA

a) Set up the numerator relationship matrix for the animals shown above.

Stellen Sie die genetisch-additive Verwandtschaftsmatrix auf für die oben gezeigten Tiere.

45

Solution:

Start by ordering the list by birthdates

```
tbl_ped_p1
```

```
## # A tibble: 5 x 4
##   Animal             Birthdate  Sire                  Dam
##   <chr>              <date>     <chr>                 <chr>
## 1 Gibsy              1990-12-09 Ginger Hill Angus 133 Bianca
## 2 HARRY              1997-02-22 HIBISCUS              WALBURGA
## 3 Gitta              2001-05-31 HARRY                 Gibsy
## 4 CH 120.1208.5899.1 2015-11-22 <NA>                  Gitta
## 5 GINA               2020-01-18 HARRY                 CH120.1208.5899.1
```

Extend the pedigree with parents not as animals

```
tbl_ped_p1_ext
```

```
## # A tibble: 9 x 4
##   Animal                Birthdate  Sire                  Dam
##   <chr>                 <date>     <chr>                 <chr>
## 1 Bianca                NA         <NA>                  <NA>
## 2 Ginger Hill Angus 133 NA         <NA>                  <NA>
## 3 HIBISCUS              NA         <NA>                  <NA>
## 4 WALBURGA              NA         <NA>                  <NA>
## 5 Gibsy                 1990-12-09 Ginger Hill Angus 133 Bianca
```

```
## 6 HARRY                 1997-02-22 HIBISCUS             WALBURGA
## 7 Gitta                 2001-05-31 HARRY                Gibsy
## 8 CH 120.1208.5899.1    2015-11-22 <NA>                 Gitta
## 9 GINA                  2020-01-18 HARRY                CH120.1208.5899.1
```

Convert to numeric IDs

```
tbl_ped_p1_ext$ID <- c(1:nrow(tbl_ped_p1_ext))
tbl_ped_p1_ext <- tbl_ped_p1_ext[,c("ID", "Animal", "Sire", "Dam", "Birthdate")]
```

Adding IDs for sire and dam

```
library(dplyr)
tbl_ped_sire_id <- tbl_ped_p1_ext %>%
  left_join(tbl_ped_p1_ext, by = c("Sire" = "Animal")) %>%
  select(ID.y)
colnames(tbl_ped_sire_id) <- "SireID"
tbl_ped_dam_id <- tbl_ped_p1_ext %>%
  left_join(tbl_ped_p1_ext, by = c("Dam" = "Animal")) %>%
  select(ID.y)
colnames(tbl_ped_dam_id) <- "DamID"
tbl_ped_p1_ext <- bind_cols(tbl_ped_p1_ext, tbl_ped_sire_id, tbl_ped_dam_id)
tbl_ped_p1_ext[9,"DamID"] <- 8
tbl_ped_p1_ext
```

```
## # A tibble: 9 x 7
##      ID Animal                Sire           Dam         Birthdate  SireID DamID
##   <int> <chr>                 <chr>          <chr>       <date>      <int> <int>
## 1     1 Bianca                <NA>           <NA>        NA             NA    NA
## 2     2 Ginger Hill Angus 133 <NA>           <NA>        NA             NA    NA
## 3     3 HIBISCUS              <NA>           <NA>        NA             NA    NA
## 4     4 WALBURGA              <NA>           <NA>        NA             NA    NA
## 5     5 Gibsy                 Ginger Hill A~ Bianca      1990-12-09      2     1
## 6     6 HARRY                 HIBISCUS       WALBURGA    1997-02-22      3     4
## 7     7 Gitta                 HARRY          Gibsy       2001-05-31      6     5
## 8     8 CH 120.1208.5899.1    <NA>           Gitta       2015-11-22     NA     7
## 9     9 GINA                  HARRY          CH120.1208~ 2020-01-18      6     8
```

Setting up the pedigree with IDs and computing the nrm

```
ped <- pedigreemm::pedigree(sire = tbl_ped_p1_ext$SireID,
                            dam  = tbl_ped_p1_ext$DamID,
                            label = as.character(1:nrow(tbl_ped_p1_ext)))
mat_A <- as.matrix(pedigreemm::getA(ped = ped))
colnames(mat_A) <- tbl_ped_p1_ext$Animal
mat_A
```

```
##   Bianca Ginger Hill Angus 133 HIBISCUS WALBURGA Gibsy HARRY Gitta
## 1 1.0000                0.0000   0.0000   0.0000 0.500 0.000  0.25
```

```
## 2 0.0000                1.0000   0.0000   0.0000 0.500 0.000  0.25
## 3 0.0000                0.0000   1.0000   0.0000 0.000 0.500  0.25
## 4 0.0000                0.0000   0.0000   1.0000 0.000 0.500  0.25
## 5 0.5000                0.5000   0.0000   0.0000 1.000 0.000  0.50
## 6 0.0000                0.0000   0.5000   0.5000 0.000 1.000  0.50
## 7 0.2500                0.2500   0.2500   0.2500 0.500 0.500  1.00
## 8 0.1250                0.1250   0.1250   0.1250 0.250 0.250  0.50
## 9 0.0625                0.0625   0.3125   0.3125 0.125 0.625  0.50
##   CH 120.1208.5899.1   GINA
## 1              0.125 0.0625
## 2              0.125 0.0625
## 3              0.125 0.3125
## 4              0.125 0.3125
## 5              0.250 0.1250
## 6              0.250 0.6250
## 7              0.500 0.5000
## 8              1.000 0.6250
## 9              0.625 1.1250
```

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0.5 & 0 & 0.25 & 0.125 & 0.0625 \\ 0 & 1 & 0 & 0 & 0.5 & 0 & 0.25 & 0.125 & 0.0625 \\ 0 & 0 & 1 & 0 & 0 & 0.5 & 0.25 & 0.125 & 0.3125 \\ 0 & 0 & 0 & 1 & 0 & 0.5 & 0.25 & 0.125 & 0.3125 \\ 0.5 & 0.5 & 0 & 0 & 1 & 0 & 0.5 & 0.25 & 0.125 \\ 0 & 0 & 0.5 & 0.5 & 0 & 1 & 0.5 & 0.25 & 0.625 \\ 0.25 & 0.25 & 0.25 & 0.25 & 0.5 & 0.5 & 1 & 0.5 & 0.5 \\ 0.125 & 0.125 & 0.125 & 0.125 & 0.25 & 0.25 & 0.5 & 1 & 0.625 \\ 0.0625 & 0.0625 & 0.3125 & 0.3125 & 0.125 & 0.625 & 0.5 & 0.625 & 1.125 \end{bmatrix}$$

■

b) Compute the inbreeding coefficients F_i of the following animals and indicate whether the animals are inbred

Berechnen Sie den Inzuchtkoeffizienten F_i der folgenden Tiere und geben Sie an, ob die jeweiligen Tiere ingezüchtet sind.

18

Animal	Inbreeding Coefficient	Animal Inbred (yes/no)
Bianca		
Ginger Hill Angus 133		
HIBISCUS		
WALBURGA		
Gibsy		
HARRY		
Gitta		
CH 120.1208.5899.1		
GINA		

Solution:

Animal	Inbreeding Coefficient	Animal Inbred (yes/no)
Bianca	0.000	no
Ginger Hill Angus 133	0.000	no
HIBISCUS	0.000	no
WALBURGA	0.000	no
Gibsy	0.000	no
HARRY	0.000	no
Gitta	0.000	no
CH 120.1208.5899.1	0.000	no
GINA	0.125	yes

■

c) The owner of GINA wants to find a mate to have an offspring. Compute the inbreeding coefficients of all possible offspring of GINA with all available sires. Select the mate for GINA, among the available sires, such that the offpsring has the lowest inbreeding coefficient.

Der/die Besitzer/In von GINA möchte einen Paarungspartner für GINA finden. Berechnen Sie die Inzuchtkoeffizienten aller möglichen Nachkommen von GINA mit allen möglichen Vätern. Wählen Sie den Paarungspartner von GINA unter allen verfügbaren Stieren, so dass das Nachkommentier einen minimalen Inzuchtkoeffizienten hat.

4

Sire	Offspring Inbreeding Coefficient
Ginger Hill Angus 133	
HIBISCUS	
HARRY	

Solution:

```
vec_inb_offspring <- 0.5 * mat_A[c(vec_sire_id), s_cow_id];vec_inb_offspring
```

```
##       2       3       6
## 0.03125 0.15625 0.31250
```

```
tbl_mate_gina <- tibble::tibble(Sire = c(vec_sire),
                                `Offspring Inbreeding Coefficient` = vec_inb_offspring)

knitr::kable(tbl_mate_gina,
             booktabs = TRUE,
             escape = FALSE,
             format = 'latex')
```

Sire	Offspring Inbreeding Coefficient
Ginger Hill Angus 133	0.03125
HIBISCUS	0.15625
HARRY	0.31250

The mate which results in the offspring with minimal offspring is

```
vec_sire[which(vec_inb_offspring == min(vec_inb_offspring))]
```

```
## [1] "Ginger Hill Angus 133"
```

■

Problem 2 Variance and Inbreeding

The cattle breed "Rätisches Grauvieh" is a robust alpine cattle breed. In a recent survey about 550 calvings per year were counted. For reasons of simplicity, we can set in the following computations, the variable N to the number of calvings per year.

Die Rindviehrasse "Rätisches Grauvieh" ist eine robuste Rasse im Alpenraum. In einer kürzlich gemachten Umfrage wurden 550 Abkalbungen pro Jahr von Rätischen Grauviehkühen gezählt. Zur Vereinfachung können wir in den folgenden Berechnungen die Variable N der Anzahl Abkalbungen pro Jahr gleichsetzen.

a) What is the expected inbreeding coefficients F_t after 50 years assuming traditional selection with a generation interval of 5 years.

Wie gross ist der erwartete Inzuchtkoeffizient F_t nach 50 Jahren? Dabei nehmen wir ein traditionelles Zuchtprogramm an mit einem Generationenintervall von 5 Jahren.

4

Solution:

```
delta_f <- 1/(2*nr_tgv_cow)
nr_gen_trad <- nr_year / gen_int_trad
inb_coef <- 1 - (1 - delta_f)^nr_gen_trad
```

The inbreeding coefficient is computed as

$$F_t = 1 - (1 - \Delta F)^t$$

where $\Delta F = \frac{1}{2N} = \frac{1}{2*550} = 9 \times 10^{-4}$ and t corresponds to the number of generations which is computed as the ratio of the number of years (50) and the generation interval (5), $t = 50/5 = 10$

Hence

$$F_t = 1 - (1 - 9 \times 10^{-4})^{10} = 0.009$$

■

b) What is the expected inbreeding coefficient F_t after 50 years, if the generation interval is reduced to 2 years due to introduction of genomic selection?

Wie gross ist der erwartete Inzuchtkoeffizient F_t nach 50 Jahren, falls das Generationenintervall durch die Einführung der genomischen Selektion auf 2 Jahre reduziert wird?

4

Solution:

```
delta_f <- 1/(2*nr_tgv_cow)
nr_gen_gsel <- nr_year / gen_int_gsel
inb_coef_gsel <- 1 - (1 - delta_f)^nr_gen_gsel
```

The same solution as under a) but with different numbers.

The inbreeding coefficient is computed as

$$F_t = 1 - (1 - \Delta F)^t$$

where $\Delta F = \frac{1}{2N} = \frac{1}{2*550} = 9 \times 10^{-4}$ and t corresponds to the number of generations which is computed as the ratio of the number of years (50) and the generation interval (2), $t = 50/2 = 25$

Hence

$$F_t = 1 - (1 - 9 \times 10^{-4})^{25} = 0.022$$

■

c) After how many years is the expected inbreeding depression at a single bi-allelic locus (minor allele frequency $p = 0.25$) bigger than 0.5 in the population of "Rätisches Grauvieh" with $N = 550$, assuming traditional selection and genomic selection? The domiance deviation d is assumed to be 50.

Nach wie vielen Jahren ist die erwartete Inzuchtdepression an einem Genlocus mit zwei Allelen (Minorallelfrequenz $p = 0.25$) grösser als 0.5 in der Population des "Rätischen Grauviehs" mit $N = 550$, einmal unter der Annahme eines traditionellen Zuchtprogramms und einmal unter Genomischer Selektion? Die Dominanzabweichung d beträgt 50.

6

Solution:

```
limit_inb_coef <- inbr_dep / (2 * dom_dev * maf * (1-maf))
delta_f <- 1/(2*nr_tgv_cow)
limit_nr_gen <- log(1 - limit_inb_coef) / log(1 - delta_f)
limit_year_trad <- ceiling(limit_nr_gen * gen_int_trad)
limit_year_gsel <- ceiling(limit_nr_gen * gen_int_gsel)
```

Inbreeding depression ΔM is computed as

$$\Delta M = M_0 - M_F = 2dp(1-p)F$$

Solving für F and inserting the numbers given in the problem leads to

$$F = \frac{\Delta M}{2dp(1-p)} = \frac{0.5}{2 * 50 * 0.25 * (1 - 0.25)} = 0.0266667$$

The number of generations t is computed from $F_t = 1 - (1 - \Delta F)^t$ with $\Delta F = \frac{1}{2N} = \frac{1}{2*550} = 9 \times 10^{-4}$ which leads to

$$t = \frac{log(1-F)}{log(1-\Delta F)} = \frac{log(1 - 0.0266667)}{log(1 - 9.0909091 \times 10^{-4})} = 29.7180232$$

- Traditional Selection with generation interval 5 years: The limit for the inbreeding depression is reached after 149 years
- Genomic selection with generation interval 2 years: The limit for the inbreeding depression is reached after 60 years

■

Problem 3 Quantitative Genetics

Given is the following dataset with genotypes of a single bi-allelic locus and with observations of a quantitative trait. The minor allele frequency of the positive allele is assumed to be $p = 0.15$.

The dataset is available from: https://charlotte-ngs.github.io/lbgfs2021/data/exam_lbgfs2021_problem3.csv.

Gegeben ist der folgende Datensatz mit Genotypen eines Genortes mit zwei Allelen und mit Beobachtungen eines quantitativen Merkmals. Die Frequenz des Allels mit positiver Wirkung sei $p = 0.15$.

Der Datensatz ist auch verfügbar unter: https://charlotte-ngs.github.io/lbgfs2021/data/exam_lbgfs2021_problem3.csv.

Animal	Genotype	Observation
1	G_1G_2	31.3
2	G_1G_2	27.4
3	G_1G_2	17.3
4	G_1G_1	32.8
5	G_2G_2	20.4
6	G_1G_2	31.9
7	G_2G_2	4.5
8	G_1G_2	26.6
9	G_2G_2	18.8
10	G_1G_2	38.2
11	G_2G_2	7.2
12	G_1G_2	26.3
13	G_2G_2	22.3
14	G_2G_2	10.9
15	G_1G_2	27.5
16	G_1G_2	32.7
17	G_2G_2	17.3
18	G_2G_2	15.8
19	G_1G_2	31.1
20	G_1G_2	24.3
21	G_2G_2	16.9
22	G_1G_1	37.0
23	G_2G_2	18.7

a) Estimate the genotypic values for the three genotypes G_1G_1, G_1G_2 and G_2G_2 using a linear fixed effects model.

Schätzen Sie die genotypischen Werte für die drei Genotypen G_1G_1, G_1G_2 and G_2G_2 unter Verwendung eines linearen fixen Modells

9

Solution:

For a linear fixed effects model, the column with the genotypes must be converted into factors

```
tbl_data_p3$Genotype <- as.factor(tbl_data_p3$Genotype)
```

The linear fixed effects model is fitted

```
lm_single_locus <- lm(formula = Observation ~ Genotype, data = tbl_data_p3)
summary(lm_single_locus)
```

```
##
## Call:
## lm(formula = Observation ~ Genotype, data = tbl_data_p3)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -11.30  -2.20   1.62   3.36   9.60
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)         34.900      3.897   8.955 1.96e-08 ***
## Genotype$G_1G_2$   -6.300      4.237  -1.487 0.152625
## Genotype$G_2G_2$  -19.620      4.269  -4.596 0.000175 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.512 on 20 degrees of freedom
## Multiple R-squared:  0.6678, Adjusted R-squared:  0.6346
## F-statistic:  20.1 on 2 and 20 DF,  p-value: 1.638e-05
```

Transforming the solutions means

```
(coef_single_locus <- coefficients(lm_single_locus))
```

```
##      (Intercept) Genotype$G_1G_2$ Genotype$G_2G_2$
##            34.90            -6.30           -19.62
```

The solutions show that the effect of G_1G_1 is set to 0. The parameter a corresponding to the genotypic value of G_1G_1 is estimated via the difference between the homozygous genotypes. This means that

```
(parameter_a <- (0 - coef_single_locus[["Genotype$G_2G_2$"]])/2)
```

```
## [1] 9.81
```

$$a = \frac{0 - (-19.62)}{2} = 9.81$$

The genotypic value of G_1G_2 is obtained by adding a to the effect obtained for genotype G_1G_2.

```
(parameter_d <- parameter_a + coef_single_locus[["Genotype$G_1G_2$"]])
```

```
## [1] 3.51
```

The genotypic values are

Genotype	Value
G_2G_2	-9.81
G_1G_2	3.51
G_1G_1	9.81

■

b) Compute the breeding values and the dominance deviations as defined in the section of "Quantitative Genetics" for the data shown above and using the results under 3a. If you were not able to solve 3a, you can use the values $a = 10$ and $d = 2$.

Berechnen Sie die Zuchtwerte und die Dominanzabweichungen, wie sie im Kapitel "Quantitative Genetik" definiert wurden für die oben gezeigten Daten. Falls Sie Aufgabe 3a nicht lösen konnten, können Sie die Werte $a = 10$ *und* $d = 2$ *verwenden.*

6

Solution:

```
alpha = parameter_a + (1-2*maf_p3) * parameter_d
bv11 <- 2 * (1-maf_p3) * alpha
bv12 <- (1-2*maf_p3) * alpha
bv22 <- -2*maf_p3 * alpha
d11 <- -2*(1-maf_p3)^2 * parameter_d
d12 <- 2*(1-maf_p3)*maf_p3 * parameter_d
d22 <- -2*maf_p3 * parameter_d
```

Using $\alpha = a + (q - p)d = 9.81 + (0.85 - 0.15) * 3.51 = 12.267$

Genotype	Breeding Value	Dominance Deviation
G_1G_1	$2q\alpha = 2 * 0.85 * 12.267 = 20.8539$	$-2q^2d = -5.07195$
G_1G_2	$(q - p)\alpha = (0.85 - 0.15) * 12.267 = 8.5869$	$2pqd = 0.89505$
G_2G_2	$-2p\alpha = -2 * 0.15 * 12.267 = -3.6801$	$-2p^2d = -1.053$

```
parameter_a_not_solved <- 10
parameter_d_not_solved <- 2
alpha_not_solved = parameter_a_not_solved + (1-2*maf_p3) * parameter_d_not_solved
bv11 <- 2 * (1-maf_p3) * alpha_not_solved
bv12 <- (1-2*maf_p3) * alpha_not_solved
bv22 <- -2*maf_p3 * alpha_not_solved
d11 <- -2*(1-maf_p3)^2 * parameter_d_not_solved
d12 <- 2*(1-maf_p3)*maf_p3 * parameter_d_not_solved
d22 <- -2*maf_p3 * parameter_d_not_solved
```

Genotype	Breeding Value	Dominance Deviation
G_1G_1	$2q\alpha = 19.38$	$-2q^2d = -2.89$
G_1G_2	$(q - p)\alpha = 7.98$	$2pqd = 0.51$
G_2G_2	$-2p\alpha = -3.42$	$-2p^2d = -0.6$

■

c) Compute the additive genetic variance and the dominance variance for the data shown above. If you were not able to solve 3a, you can use the values $a = 10$ and $d = 2$.

Berechnen Sie die additive genetische Varianz und die Dominanzvarianz für die oben gezeigten Daten. Falls Sie Aufgabe 3a nicht lösen konnten, können Sie die Werte $a = 10$ *und* $d = 2$ *verwenden.*

2

Solution:

```
sigma_a2 <- 2 * maf_p3 * (1-maf_p3) * alpha^2
sigma_d2 <- (2 * maf_p3 * (1-maf_p3) * parameter_d)^2
```

- The genetic additive variance $\sigma_A^2 = 2pq\alpha^2 = 2 * 0.15 * 0.85 * 12.267^2 = 38.3722187$
- The dominance variance $\sigma_D^2 = (2pqd)^2 = (2*0.15*0.85*3.51)^2 = 0.8011145$

Not solved

```
sigma_a2 <- 2 * maf_p3 * (1-maf_p3) * alpha_not_solved^2
sigma_d2 <- (2 * maf_p3 * (1-maf_p3) * parameter_d_not_solved)^2
```

- The genetic additive variance $\sigma_A^2 = 2pq\alpha^2 = 2 * 0.15 * 0.85 * 12.267^2 = 33.1398$
- The dominance variance $\sigma_D^2 = (2pqd)^2 = (2 * 0.15 * 0.85 * 3.51)^2 = 0.2601$

■

Problem 4 Prediction of Breeding Values

Use the following dataset to predict breeding values. The phenotypic variance of the data is assumed to be $\sigma_p^2 = 1$. The heritability of the trait shown in the column 'Phen' of the following table is $h^2 = 0.2$.

The dataset is available from https://charlotte-ngs.github.io/lbgfs2021/data/exam_lbgfs2021_problem4.csv.

Verwenden Sie den folgenden Datensatz für die Schätzung von Zuchtwerten. Die phänotypische Varianz der Daten betrage $\sigma_p^2 = 1$. *Die Heritabilität des Merkmals in der Kolonnen 'Phen' in der nachfolgenden Tabelle betrage* $h^2 = 0.2$

Der Datensatz ist verfügbar unter: https://charlotte-ngs.github.io/lbgfs2021/data/exam_lbgfs2021_problem4.csv.

Progeny	Sire	Dam	Sex	Phen
7519	6662	6108	F	-1.669972
7399	6561	6687	F	1.030195
7151	6258	6127	M	0.085925
8418	7151	7399	F	-0.476189
8419	7151	7519	F	-0.071148
8420	7151	7519	M	0.578070

a) Use the own performance records of the animals shown above to predict breeding values. The mean of all observations above can be used as population mean.

   Verwenden Sie die Eigenleistungen der Tiere in der oben gezeigten Tabelle um deren Zuchtwerte zu schäten. Verwenden Sie den Mittelwert der Beobachtungen als Populationsmittel.

6

Solution:

The population mean is computed as

```
pop_mean <- mean(tbl_data_p4$Phen)
```

The predicted breeding values are computed as

```
pred_bv <- h2_p4 * (tbl_data_p4$Phen - pop_mean)
```

Listing the results

Progeny	Breeding Value
7519	-0.3165571
7399	0.2234763
7151	0.0346223
8418	-0.0778005
8419	0.0032077
8420	0.1330513

■

b) Use a BLUP animal model to predict breeding values for all animals given in the above shown dataset.

Verwenden Sie das BLUP Tiermodell zur Schätzung der Zuchtwerte aller Tiere, welche im obigen Datensatz gegeben sind.

16

Solution:

The BLUP animal model corresponds to the following linear mixed effects model

$$y = Xb + Zu + e$$

with y the vector of observations; b the vector of fixed effects corresponding to the 'Sex' of each animal; u the vector of random breeding values for all animals in the pedigree; e the vector of random residuals. The matrices X and Z are known design matrices.

Expected values and variance-covariance matrices are given by

$$E \begin{bmatrix} y \\ u \\ e \end{bmatrix} = \begin{bmatrix} Xb \\ 0 \\ 0 \end{bmatrix}$$

$$var \begin{bmatrix} y \\ u \\ e \end{bmatrix} = \begin{bmatrix} ZGZ^T + R & ZG & R \\ GZ^T & G & 0 \\ R & 0 & R \end{bmatrix}$$

The first step is the extension of the pedigree.

```
vec_sire_extend <- tbl_data_p4$Sire[sapply(tbl_data_p4$Sire,
                                           function(x)
                                             !(is.element(x, tbl_data_p4$Progeny)),
                                           USE.NAMES = FALSE)]
nr_sire_extend <- length(vec_sire_extend)
tbl_sire_extend <- tibble::tibble(Progeny = vec_sire_extend,
                                  Sire = rep(NA, nr_sire_extend),
                                  Dam  = rep(NA, nr_sire_extend))

vec_dam_extend <- tbl_data_p4$Dam[sapply(tbl_data_p4$Dam,
                                         function(x)
                                           !(is.element(x, tbl_data_p4$Progeny)),
                                         USE.NAMES = FALSE)]
nr_dam_extend <- length(vec_dam_extend)
tbl_dam_extend <- tibble::tibble(Progeny = vec_dam_extend,
                                 Sire = rep(NA, nr_dam_extend),
```

```
                                    Dam  = rep(NA, nr_dam_extend))

tbl_ped_ext_p4 <- dplyr::bind_rows(tbl_sire_extend,
                                    tbl_dam_extend,
                                    tbl_data_p4[,c("Progeny", "Sire", "Dam")])
tbl_ped_ext_p4
```

```
## # A tibble: 12 x 3
##    Progeny  Sire   Dam
##      <dbl> <dbl> <dbl>
##  1    6662    NA    NA
##  2    6561    NA    NA
##  3    6258    NA    NA
##  4    6108    NA    NA
##  5    6687    NA    NA
##  6    6127    NA    NA
##  7    7519  6662  6108
##  8    7399  6561  6687
##  9    7151  6258  6127
## 10    8418  7151  7399
## 11    8419  7151  7519
## 12    8420  7151  7519
```

The numerator relationship matrix

```
ped_p4 <- pedigreemm::pedigree(sire = tbl_ped_ext_p4$Sire,
                               dam = tbl_ped_ext_p4$Dam,
                               label = as.character(tbl_ped_ext_p4$Progeny))
ped_p4
```

```
##      sire  dam
## 6662 <NA> <NA>
## 6561 <NA> <NA>
## 6258 <NA> <NA>
## 6108 <NA> <NA>
## 6687 <NA> <NA>
## 6127 <NA> <NA>
## 7519 6662 6108
## 7399 6561 6687
## 7151 6258 6127
## 8418 7151 7399
## 8419 7151 7519
## 8420 7151 7519
```

```
mat_Ainv_p4 <- as.matrix(pedigreemm::getAInv(ped = ped_p4))
mat_Ainv_p4
```

```
##      6662 6561 6258 6108 6687 6127 7519 7399 7151 8418 8419 8420
## 6662  1.5  0.0  0.0  0.5  0.0  0.0   -1  0.0  0.0    0    0    0
## 6561  0.0  1.5  0.0  0.0  0.5  0.0    0 -1.0  0.0    0    0    0
## 6258  0.0  0.0  1.5  0.0  0.0  0.5    0  0.0 -1.0    0    0    0
## 6108  0.5  0.0  0.0  1.5  0.0  0.0   -1  0.0  0.0    0    0    0
## 6687  0.0  0.5  0.0  0.0  1.5  0.0    0 -1.0  0.0    0    0    0
## 6127  0.0  0.0  0.5  0.0  0.0  1.5    0  0.0 -1.0    0    0    0
## 7519 -1.0  0.0  0.0 -1.0  0.0  0.0    3  0.0  1.0    0   -1   -1
## 7399  0.0 -1.0  0.0  0.0 -1.0  0.0    0  2.5  0.5   -1    0    0
## 7151  0.0  0.0 -1.0  0.0  0.0 -1.0    1  0.5  3.5   -1   -1   -1
## 8418  0.0  0.0  0.0  0.0  0.0  0.0    0 -1.0 -1.0    2    0    0
## 8419  0.0  0.0  0.0  0.0  0.0  0.0   -1  0.0 -1.0    0    2    0
## 8420  0.0  0.0  0.0  0.0  0.0  0.0   -1  0.0 -1.0    0    0    2
```

The design matrices are given by the following chunks. First the matrix X

```
nr_records <- nrow(tbl_data_p4)
(mat_X <- matrix(c(0, 1,
                   0, 1,
                   1, 0,
                   0, 1,
                   0, 1,
                   1, 0), ncol = 2, byrow = TRUE))
```

```
##      [,1] [,2]
## [1,]    0    1
## [2,]    0    1
## [3,]    1    0
## [4,]    0    1
## [5,]    0    1
## [6,]    1    0
```

The matrix Z is defined as

```
vec_base <- tbl_ped_ext_p4$Progeny[sapply(tbl_ped_ext_p4$Progeny,
                                          function(x)
                                            !is.element(x, tbl_data_p4$Progeny),
                                          USE.NAMES = FALSE)]
(mat_Z <- cbind(matrix(0, nrow = nr_records, ncol = length(vec_base)),
                diag(1, nrow = nr_records)))
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]
## [1,]    0    0    0    0    0    0    1    0    0     0     0     0
## [2,]    0    0    0    0    0    0    0    1    0     0     0     0
## [3,]    0    0    0    0    0    0    0    0    1     0     0     0
## [4,]    0    0    0    0    0    0    0    0    0     1     0     0
## [5,]    0    0    0    0    0    0    0    0    0     0     1     0
## [6,]    0    0    0    0    0    0    0    0    0     0     0     1
```

The mixed model equations

```
lambda <- (1-h2_p4)/ h2_p4
mat_xtx <- crossprod(mat_X)
mat_xtz <- crossprod(mat_X, mat_Z)
mat_ztx <- crossprod(mat_Z, mat_X)
mat_ztzlAinv <- crossprod(mat_Z) + lambda * mat_Ainv_p4
mat_coef <- rbind(cbind(mat_xtx, mat_xtz), cbind(mat_ztx, mat_ztzlAinv))
mat_rhs <- rbind(crossprod(mat_X, tbl_data_p4$Phen),
                 crossprod(mat_Z, tbl_data_p4$Phen))
(mat_sol <- solve(mat_coef, mat_rhs))
```

```
##             [,1]
##       0.38845865
##      -0.30206265
## 6662 -0.10737069
## 6561  0.12071478
## 6258 -0.01334410
## 6108 -0.10737069
## 6687  0.12071478
## 6127 -0.01334410
## 7519 -0.21474138
## 7399  0.24142957
## 7151 -0.02668819
## 8418  0.07609324
## 8419 -0.08164485
## 8420 -0.08623410
```

■

Problem 5 Genomics

Use the following dataset to predict genomic breeding values. The minor allele frequencies of the three loci are given as

Verwenden Sie den folgenden Datensatz zur Schätzung von genomischen Zuchtwerten. Die minor Allelfrequenzen der drei Loci sind gegeben als

- $p_G = 0.45$
- $p_H = 0.35$
- $p_I = 0.4$

The dataset can be obtained from https://charlotte-ngs.github.io/lbgfs2021/data/exam_lbgfs2021_problem5.csv.

Der Datensatz ist verfügbar unter: https://charlotte-ngs.github.io/lbgfs2021/data/exam_lbgfs2021_problem5.csv.

Animal	Locus G	Locus H	Locus I	Observation
1	1	0	-1	25.3
2	0	-1	0	20.7
3	-1	1	0	33.2
4	0	-1	0	8.4
5	0	-1	0	18.8
6	1	0	0	35.9
7	0	0	-1	1.4
8	-1	0	-1	-6.4
9	0	-1	-1	6.3
10	0	0	-1	13.6
11	1	0	0	34.0
12	0	1	1	54.1
13	1	-1	0	25.8
14	-1	1	0	29.0
15	0	-1	0	14.8
16	0	-1	-1	1.6
17	0	-1	1	23.8

a) Use a marker effect model to predict genomic breeding values from the above data. Use a value of $\lambda = 10$ for solving the mixed model equations.

Verwenden Sie ein Markereffektmodell zur Schätzung von genomischen Zuchtwerten. Verwenden Sie $\lambda = 10$ *für die Mischmodellgleichungen*

12

Solution:

The marker effect model corresponds to the following linear mixed effect model

$$y = 1\mu + Wq + e$$

with y the vector of observations; μ the fixed general intercept; q the vector of random marker effects; e the vector of random residuals.

First the marker effects are predicted using the following MME

```
nr_records <- nrow(tbl_data_p5)
mat_X <- matrix(1, nrow = nr_records, ncol = 1)
mat_W <- as.matrix(tbl_data_p5[,c("Locus G", "Locus H", "Locus I")])
n_nr_snp <- ncol(mat_W)
mat_xtx <- crossprod(mat_X)
mat_xtw <- crossprod(mat_X, mat_W)
mat_wtx <- crossprod(mat_W, mat_X)
mat_wtwlI <- crossprod(mat_W) + lambda_p5a + diag(1,nrow = n_nr_snp)
mat_coef <- rbind(cbind(mat_xtx, mat_xtw), cbind(mat_wtx, mat_wtwlI))
mat_rhs <- rbind(crossprod(mat_X, tbl_data_p5$Observation),
                 crossprod(mat_W, tbl_data_p5$Observation))
(mat_sol <- solve(mat_coef, mat_rhs))
```

```
##              [,1]
##         22.209503
## Locus G -1.897469
## Locus H  2.048225
## Locus I  6.280737
```

The genomic breeding values are obtained by a multiplication of the matrix W with the marker effects.

```
nr_fix_eff <- ncol(mat_X)
mat_mrk_eff <- mat_sol[(nr_fix_eff+1):nrow(mat_sol),]
mat_gen_bv <- crossprod(t(mat_W), mat_mrk_eff)
mat_gen_bv
```

```
##            [,1]
##  [1,] -8.178206
##  [2,] -2.048225
```

```
##  [3,]  3.945695
##  [4,] -2.048225
##  [5,] -2.048225
##  [6,] -1.897469
##  [7,] -6.280737
##  [8,] -4.383268
##  [9,] -8.328962
## [10,] -6.280737
## [11,] -1.897469
## [12,]  8.328962
## [13,] -3.945695
## [14,]  3.945695
## [15,] -2.048225
## [16,] -8.328962
## [17,]  4.232512
```

The numeric values of the genomic breeding values are not that important, but the ranking of the animals according to the predicted breeding values

```
order(mat_gen_bv[,1], decreasing = TRUE)
```

```
##  [1] 12 17  3 14  6 11  2  4  5 15 13  8  7 10  1  9 16
```

■

b) Use a breeding-value-based model to predict genomic breeding values from the above data. Use a value of $\lambda = 5$ for solving the mixed model equations.

Verwenden Sie ein Zuchtwert-basiertes Modell zur Schätzung von genomischen Zuchtwerten. Verwenden Sie $\lambda = 5$ für die Mischmodellgleichungen

12

Solution:

The breeding-value based model corresponds to the following linear mixed effect model

$$y = 1\mu + Zg + e$$

We first have to compute the genomic relationship matrix G and its inverse. The following function is used to setup the matrix G

```
computeMatGrm <- function(pmatData) {
  matData <- pmatData
  # check the coding, if matData is -1, 0, 1 coded, then add 1 to get to 0, 1, 2 coding
  if (min(matData) < 0) matData <- matData + 1
  # Allele frequencies, column vector of P and sum of frequency products
  freq <- apply(matData, 2, mean) / 2
  P <- 2 * (freq - 0.5)
  sumpq <- sum(freq*(1-freq))
  # Changing the coding from (0,1,2) to (-1,0,1) and subtract matrix P
  Z <- matData - 1 - matrix(P, nrow = nrow(matData),
                            ncol = ncol(matData),
                            byrow = TRUE)
  # Z%*%Zt is replaced by tcrossprod(Z)
  return(tcrossprod(Z)/(2*sumpq))
}
nr_records <- nrow(tbl_data_p5)
mat_W <- as.matrix(tbl_data_p5[,c("Locus G", "Locus H", "Locus I")])
mat_G <- computeMatGrm(pmatData = mat_W)
mat_Ginv <- solve(mat_G + 0.1 * diag(1, nrow = nr_records))
```

The solution is via the following mixed model equation.

```
mat_X <- matrix(1, nrow = nr_records, ncol = 1)
mat_Z <- diag(1, nrow = nr_records)
mat_xtx <- crossprod(mat_X)
mat_xtz <- crossprod(mat_X, mat_Z)
mat_ztx <- crossprod(mat_Z, mat_X)
mat_ztzlGinv <- crossprod(mat_Z) + lambda_p5b * mat_Ginv
mat_coef <- rbind(cbind(mat_xtx, mat_xtz), cbind(mat_ztx, mat_ztzlGinv))
```

```
mat_rhs <- rbind(crossprod(mat_X, tbl_data_p5$Observation),
                 crossprod(mat_Z, tbl_data_p5$Observation))
(mat_sol <- solve(mat_coef, mat_rhs))
```

```
##               [,1]
##  [1,]  20.01764706
##  [2,]  -0.05202814
##  [3,]  -2.62222084
##  [4,]   5.31580514
##  [5,]  -2.86339731
##  [6,]  -2.65947574
##  [7,]   7.69254279
##  [8,]  -4.63416255
##  [9,]  -8.90061068
## [10,] -10.44130157
## [11,]  -4.39494686
## [12,]   7.65528788
## [13,]  17.37584381
## [14,]   1.59128612
## [15,]   5.23345220
## [16,]  -2.73790711
## [17,] -10.53345843
## [18,]   4.97529127
```

The first element is the estimate of μ and all other elements in the solution vector are the genomic breeding values.

```
nr_fix_eff <- ncol(mat_X)
(mat_bv <- mat_sol[(nr_fix_eff+1):nrow(mat_sol),])
```

```
##  [1]  -0.05202814  -2.62222084   5.31580514  -2.86339731  -2.65947574
##  [6]   7.69254279  -4.63416255  -8.90061068 -10.44130157  -4.39494686
## [11]   7.65528788  17.37584381   1.59128612   5.23345220  -2.73790711
## [16] -10.53345843   4.97529127
```

The numeric values of the genomic breeding values are not that important, but the ranking of the animals according to the predicted breeding values

```
order(mat_bv, decreasing = TRUE)
```

```
##  [1] 12  6 11  3 14 17 13  1  2  5 15  4 10  7  8  9 16
```

■