Applied Statistical Methods - Solution 11

Peter von Rohr

2022-05-22

Problem 1: Marker Effects Model

Predict genomic breeding values using a marker effects model. The dataset is available from

```
## https://charlotte-ngs.github.io/asmss2022/data/asm_geno_sim_data.csv
```

Hints

- The variance σ_q^2 of the marker effect is 3.
- The residual variance σ_e^2 is 36
- The sex of each animal can be modelled as a fixed effect

Solution

- Read the data

```
tbl_ex11_p01 <- readr::read_csv(s_ex11_p01_data_path)
```

```
## Rows: 8 Columns: 105

## -- Column specification ----------------------------------------------------------------------------
## Delimiter: ","
## chr   (1): SEX
## dbl (104): ID, SIRE, DAM, P, SNP1, SNP2, SNP3, SNP4, SNP5, SNP6, SNP7, SNP8, SNP9, SNP10, SNP11, SNP

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
tbl_ex11_p01
```

```
## # A tibble: 8 x 105
##      ID  SIRE   DAM SEX       P  SNP1  SNP2  SNP3  SNP4  SNP5  SNP6  SNP7  SNP8  SNP9 SNP10 SNP11 SN
##   <dbl> <dbl> <dbl> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <d
## 1     5     1     3 m      37.5     2     1     1     1     0     1     2     0     1     1     0
## 2     6     2     3 f      18       2     2     0     1     1     1     2     0     1     2     0
## 3     7     1     4 m      22.4     1     0     0     1     1     2     2     0     1     0     0
## 4     8     2     4 f      36.7     1     2     1     1     2     2     2     0     2     1     0
## 5     9     1     8 f      33       0     2     0     2     1     1     2     0     1     0     1
## 6    10     2     6 f      33.1     2     2     0     1     1     1     2     0     2     2     0
## 7    11     1     8 m      32.4     2     1     0     2     1     1     2     0     1     0     1
## 8    12     2     6 m      18.8     2     2     1     1     1     1     2     0     0     1     0
## # ... with 85 more variables: SNP16 <dbl>, SNP17 <dbl>, SNP18 <dbl>, SNP19 <dbl>, SNP20 <dbl>, SNP21
## #   SNP23 <dbl>, SNP24 <dbl>, SNP25 <dbl>, SNP26 <dbl>, SNP27 <dbl>, SNP28 <dbl>, SNP29 <dbl>, SNP30
## #   SNP32 <dbl>, SNP33 <dbl>, SNP34 <dbl>, SNP35 <dbl>, SNP36 <dbl>, SNP37 <dbl>, SNP38 <dbl>, SNP39
## #   SNP41 <dbl>, SNP42 <dbl>, SNP43 <dbl>, SNP44 <dbl>, SNP45 <dbl>, SNP46 <dbl>, SNP47 <dbl>, SNP48
## #   SNP50 <dbl>, SNP51 <dbl>, SNP52 <dbl>, SNP53 <dbl>, SNP54 <dbl>, SNP55 <dbl>, SNP56 <dbl>, SNP57
## #   SNP59 <dbl>, SNP60 <dbl>, SNP61 <dbl>, SNP62 <dbl>, SNP63 <dbl>, SNP64 <dbl>, SNP65 <dbl>, SNP66
## #   SNP68 <dbl>, SNP69 <dbl>, SNP70 <dbl>, SNP71 <dbl>, SNP72 <dbl>, SNP73 <dbl>, SNP74 <dbl>, SNP75
```

- Setup mixed model equations to predict marker effects for all the SNP-loci. The model is given as

$$y = Xb + Wq + e$$

where y is the vector of observations, b is the vector of fixed effects and q is the vector of random marker effects for each SNP. The matrices X and W are design matrices. The matrix W is special because it contains the genotype encodings.

From that model the mixed model equations can be specified as

$$\left[\begin{array}{cc} X^TX & X^TW \\ W^TX & W^TW + \lambda_q * I \end{array}\right] \left[\begin{array}{c} \hat{b} \\ \hat{q} \end{array}\right] = \left[\begin{array}{c} X^Ty \\ W^Ty \end{array}\right]$$

with $\lambda_q = \sigma_e^2/\sigma_q^2$.

The matrix X

```
mat_X <- model.matrix(lm(P ~ 0 + SEX, data = tbl_ex11_p01))
attr(mat_X, "assign") <- NULL
attr(mat_X, "contrasts") <- NULL
mat_X
```

```
##   SEXf SEXm
## 1    0    1
## 2    1    0
## 3    0    1
## 4    1    0
## 5    1    0
## 6    1    0
## 7    0    1
## 8    0    1
```

The matrix W

```
library(dplyr)
tbl_geno_ex11_p01 <- tbl_ex11_p01 %>%
  select(SNP1:SNP100)
mat_W <- as.matrix(tbl_geno_ex11_p01)
mat_W[,1:10]
```

```
##      SNP1 SNP2 SNP3 SNP4 SNP5 SNP6 SNP7 SNP8 SNP9 SNP10
## [1,]    2    1    1    1    0    1    2    0    1     1
## [2,]    2    2    0    1    1    1    2    0    1     2
## [3,]    1    0    0    1    1    2    2    0    1     0
## [4,]    1    2    1    1    2    2    2    0    2     1
## [5,]    0    2    0    2    1    1    2    0    1     0
## [6,]    2    2    0    1    1    1    2    0    2     2
## [7,]    2    1    0    2    1    1    2    0    1     0
## [8,]    2    2    1    1    1    1    2    0    0     1
```

The vector y

```
vec_y <- tbl_ex11_p01$P
vec_y
```

```
## [1] 37.5 18.0 22.4 36.7 33.0 33.1 32.4 18.8
```

The mixed model equations

```
# coefficient matrix
mat_xtx <- crossprod(mat_X)
mat_xtw <- crossprod(mat_X, mat_W)
mat_wtx <- t(mat_xtw)
lambda_q <- sigma_e2 / sigma_q2
mat_ztz_lambda_I <- crossprod(mat_W) + lambda_q * diag(1, nrow = ncol(mat_W))
mat_coef <- rbind(cbind(mat_xtx, mat_xtw),
                  cbind(mat_wtx, mat_ztz_lambda_I))
# right hand side
mat_xty <- crossprod(mat_X, vec_y)
mat_wty <- crossprod(mat_W, vec_y)
mat_rhs <- rbind(mat_xty, mat_wty)
# solution
mat_sol <- solve(mat_coef, mat_rhs)
mat_sol[1:10,]
```

```
##          SEXf          SEXm          SNP1          SNP2          SNP3          SNP4          SNP5
##  3.002412e+01  2.831841e+01  8.637400e-02  1.423242e-01  3.568333e-01  8.887511e-02 -7.332053e-02 -5
##          SNP7          SNP8
##  4.935867e-15  0.000000e+00
```

- Compute predicted genomic breeding values based on the estimated marker effects

Problem 2: Breeding Value Based Model

Use the same dataset as in Problem 1 to predict genomic breeding values based on a breeding-value model. The dataset is available from

```
## https://charlotte-ngs.github.io/asmss2022/data/asm_geno_sim_data.csv
```

Hints

- The genomic variance σ_g^2 of the marker effect is 9.
- The residual variance σ_e^2 is 36
- The sex of each animal can be modelled as a fixed effect
- Use the following function to compute the genomic relationship matrix G based on the matrix of genotypes

```
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))
}
```

- If the genomic relationship matrix G which is computed by the function above cannot be inverted, add $0.05 * I$ to G which results in G^* and use G^* as genomic relationship matrix.

Solution