Written examination, date: 9th of December 2019 *Page 1 of 39 pages* *Enclosure: XX pages*
Course name: Multivariate Statistics
Course number: 02409
Aids allowed: All
Exam duration: 4 hours
Weighting: The questions are given equal weight

This exam is answered by:

_____ _____ _____
(name) (signature) (study no.)

There is a total of 30 questions for the 6 problems. The answers to the 30 questions must be written into the table below.

Problem	1	1	1	1	1	1	1	1	2	2
Question	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	2.1	2.2
Answer	2	1	5	3	3	2	1	4	5	2

Problem	2	3	3	3	3	3	3	3	3	4
Question	2.3	3.1	3.2	3.3	3.4	3.5	3.6	3.7	3.8	4.1
Answer	4	1	1	2	4	4	3	2	4	4

Problem	4	4	5	5	5	6	6	6	6	6
Question	4.2	4.3	5.1	5.2	5.3	6.1	6.2	6.3	6.4	6.5
Answer	2	(3)	4	2	2	2	2	5	1	5

The possible answers for each question are numbered from 1 to 6. If you enter a wrong number, you may correct it by crossing the wrong number in the table and writing the correct answer immediately below. If there is any doubt about the meaning of a correction then the question will be considered not answered.

Only the front page must be returned. The front page must be returned even if you do not answer any of the questions or if you leave the exam prematurely. Drafts and/or comments are not considered, only the numbers entered above are registered.

A correct answer gives 5 points, a wrong answer gives – 1 point. Unanswered questions or a 6 (corresponding to "don't know") give 0 points. The total number of points needed for a satisfactorily answered exam is determined at the final evaluation of the exam. Especially note that the grade 10 may be given even if only one answer is wrong or unanswered.
Remember to write your name, signature, and study number on the front page.

Please note, that there is one and only one correct answer to each question. Furthermore, some of the possible alternative answers may not make sense. When the text refers to SAS-output, the values may be rounded to fewer decimal places than in the output itself. The enclosures do not necessarily contain all the output generated by the given SAS programs. Please check that all pages of the exam paper and the enclosures are present.

Problem 1.

You are encouraged to use statistical software in this problem.

We consider the following model

$$\begin{bmatrix} Y_1 & Z_1 & V_1 & W_1 \\ Y_2 & Z_2 & V_2 & W_2 \\ Y_3 & Z_3 & V_3 & W_3 \\ Y_4 & Z_4 & V_4 & W_4 \\ Y_5 & Z_5 & V_5 & W_5 \end{bmatrix} = \begin{bmatrix} 1 & -2 & -4 \\ 1 & -1 & -1 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 4 \end{bmatrix} \begin{bmatrix} \alpha_y & \alpha_z & \alpha_v & \alpha_w \\ \beta_y & \beta_z & \beta_v & \beta_w \\ \gamma_y & \gamma_z & \gamma_v & \gamma_w \end{bmatrix} + \begin{bmatrix} \delta_1 & \varepsilon_1 & \epsilon_1 & \vartheta_1 \\ \delta_2 & \varepsilon_2 & \epsilon_2 & \vartheta_2 \\ \delta_3 & \varepsilon_3 & \epsilon_3 & \vartheta_3 \\ \delta_4 & \varepsilon_4 & \epsilon_4 & \vartheta_4 \\ \delta_5 & \varepsilon_5 & \epsilon_5 & \vartheta_5 \end{bmatrix}$$

Where the error terms $[\delta_i \quad \varepsilon_i \quad \epsilon_i \quad \vartheta_i], for\ i = 1, \dots, 5$ are independent and normally distributed $N_4(\mathbf{0}, \mathbf{\Sigma})$, and where $\mathbf{\Sigma}$ is the unknown dispersion matrix.

We have obtained the following observations

$$\begin{bmatrix} 1 & 8 & 2 & 9 \\ 0 & 9 & 4 & 6 \\ 2 & 4 & 4 & 2 \\ 1 & 5 & 9 & 5 \\ 1 & 2 & 8 & 7 \end{bmatrix}$$

We can further calculate $(\boldsymbol{x}^T\boldsymbol{x})^{-1} = \begin{bmatrix} 0.2 & 0 & 0 \\ 0 & 2.125 & -1.125 \\ 0 & -1.125 & 0.625 \end{bmatrix}$

Question 1.1.

The maximum likelihood estimate for the parameters $[\alpha_y \quad \beta_y \quad \gamma_y]$ *are*

This can be done by hand – or easier – using R.

```
# Load the dataset in R
# Create a data frame with the provided data
data_MGLM <- data.frame(
  y = c(1, 0, 2, 1, 1),
  z = c(8, 9, 4, 5, 2),
  v = c(2, 4, 4, 9, 8),
  w = c(9, 6, 2, 5, 7),
  X1 = c(1, 1, 1, 1, 1),
  X2 = c(-2, -1, 0, 1, 2),
  X3 = c(-4, -1, 0, 1, 4)
)
```

In R:

```
# Fit the GLM without intercept
# In the first question we want to find the maximum likelihood estimation for the parameters [a_y b_y g_y],
# Therefore we need only the y variable as dependent variable
# In SAS the results for this question are executed automatically because SAS gives you the results by each dependent variable
model <- lm(y ~ X1 + X2 + X3 -1, data = data_MGLM)
model_summary  <-  summary(model)
model_summary
# View the estimated coefficients
model_summary$coefficients
```

Result in R:

```
> model_summary$coefficients
   Estimate Std. Error    t value  Pr(>|t|)
X1      1.0  0.3872983  2.5819889 0.1229420
X2      1.0  1.2624381  0.7921180 0.5113222
X3     -0.5  0.6846532 -0.7302967 0.5411685
```

with yields:
ANSWER 2: $[\mathbf{1} \quad \mathbf{1} \quad -\mathbf{0.5}]$

Question 1.2.

The covariance between the maximum likelihood estimates for α_x and β_x is

We can do immediately see from the given $(x^T x)^{-1} = \begin{bmatrix} 0.2 & 0 & 0 \\ 0 & 2.125 & -1.125 \\ 0 & -1.125 & 0.625 \end{bmatrix}$, that it is zero. We can also use:

In R:

```
# Get the covariance matrix of parameter estimates
cov_matrix <- vcov(model)

# View the covariance matrix
cov_matrix
```

Result in R:

```
> cov_matrix
      X1       X2       X3
X1 0.15  0.00000  0.00000
X2 0.00  1.59375 -0.84375
X3 0.00 -0.84375  0.46875
```

which confirms the results
ANSWER 1: 0

Question 1.3.

The covariance between the maximum likelihood estimates for β_x and γ_x is

We already have $(x^T x)^{-1}$ given, we need to estimate σ_x^2, which can be done by theorem

Theorem 4.18

We consider the situation from theorem 4.14. Then the maximum likelihood estimate for $\boldsymbol{\Sigma}$ equals

$$\begin{aligned}\hat{\boldsymbol{\Sigma}}^* &= \frac{1}{n}\sum_{i=1}^{n}(\mathbf{Y}_i - \hat{\boldsymbol{\theta}}^T x_i)(\mathbf{Y}_i - \hat{\boldsymbol{\theta}}^T x_i)^T \\ &= \frac{1}{n}(\mathbf{Y} - \mathbf{x}\hat{\boldsymbol{\theta}})^T(\mathbf{Y} - \mathbf{x}\hat{\boldsymbol{\theta}}) \\ &= \frac{1}{n}[\mathbf{Y}^T\mathbf{Y} - (\mathbf{x}\hat{\boldsymbol{\theta}})^T(\mathbf{x}\hat{\boldsymbol{\theta}})].\end{aligned}$$

The (i, j)'th element can also be written

$$\hat{\sigma}_{ij}^* = \frac{1}{n}(\mathbf{Y}_{i|} - \mathbf{x}\hat{\boldsymbol{\theta}}_{i|})^T(\mathbf{Y}_{j|} - \mathbf{x}\hat{\boldsymbol{\theta}}_{j|}).$$

Alternatively, in R we use the matrix which we have found in the previous question:

```
> cov_matrix
     X1       X2       X3
X1 0.15  0.00000  0.00000
X2 0.00  1.59375 -0.84375
X3 0.00 -0.84375  0.46875
```

ANSWER 5: -0.84375

Question 1.4.

The variance of the maximum likelihood estimates for α_x is

We again need to estimate σ_x^2.
We use in R the previous matrix:

```
> cov_matrix
     X1       X2       X3
X1 0.15  0.00000  0.00000
X2 0.00  1.59375 -0.84375
X3 0.00 -0.84375  0.46875
```

Other way in R:
model_summary$coefficients

Result in R:

```
> model_summary$coefficients
   Estimate Std. Error    t value  Pr(>|t|)
X1      1.0  0.3872983  2.5819889 0.1229420
X2      1.0  1.2624381  0.7921180 0.5113222
X3     -0.5  0.6846532 -0.7302967 0.5411685
```

Calculating the squared of the standard error of X1 coefficient we receive the variance of the maximum likelihood estimates for α_x.

$$0.38730^2 = 0.15$$

ANSWER 3: 0.15

Question 1.5.

The observation with the lowest leverage is:

In R:

```
DFFITS <- round(dffits(model), 4)
R_student <- round(rstudent(model),4)
HatDiagH <- round(hatvalues(model),4)
Residual <- round(residuals(model),4)
Covratio <- round(covratio(model), 4)
Predicted_value <- round(predict(model),4)

Stats  <- data.frame(Obs,Predicted_value,Residual,R_student,HatDiagH,Covratio,DFFITS)
print(Stats)

# Find out the lowest HatDiagH value
obs_with_min_hat_diag_H <- which.min(Stats$HatDiagH)

# Print the observation number with the minimum HatDiagH
cat("Observation with Min HatDiagH:", obs_with_min_hat_diag_H, "\n")
```

Result in R:

We need to find the observation with the lowest leverage, so we look at the Hat Diag H column taking the lowest value:

```
> print(Stats)
  Obs Predicted_value Residual R_student HatDiagH Covratio  DFFITS
1   1             1.0      0.0    0.0000      0.7  26.6667  0.0000
2   2             0.5     -0.5   -1.1180      0.7   2.3411 -1.7078
3   3             1.0      1.0    2.2361      0.2   0.0463  1.1180
4   4             1.5     -0.5   -1.1180      0.7   2.3411 -1.7078
5   5             1.0      0.0    0.0000      0.7  26.6667  0.0000

> cat("Observation with Min HatDiagH:", obs_with_min_hat_diag_H, "\n")
Observation with Min HatDiagH: 3
```

ANSWER 3: 3

Question 1.6.

The dependent variable with the lowest MSE is:

In R:

```
model_y <- lm(y ~ X1 + X2 + X3 -1, data = data_MGLM)
# Calculate the residual sums of squares (RSS)
RSS_y <- sum(residuals(model_y)^2)
# Calculate the degrees of freedom for error
df_error_y <- nobs(model_y) - length(coefficients(model_y))
# Calculate the Sum of squares and products for error (SSPE)
SSPE_y <- RSS_y / df_error_y
# View the SSPE
SSPE_y

model_z <- lm(z ~ X1 + X2 + X3 -1, data = data_MGLM)
# Calculate the residual sums of squares (RSS)
RSS_z <- sum(residuals(model_z)^2)
# Calculate the degrees of freedom for error
df_error_z <- nobs(model_z) - length(coefficients(model_z))
# Calculate the Sum of squares and products for error (SSPE)
SSPE_z <- RSS_z / df_error_z
# View the SSPE
SSPE_z

model_v <- lm(v ~ X1 + X2 + X3 -1, data = data_MGLM)
# Calculate the residual sums of squares (RSS)
RSS_v <- sum(residuals(model_v)^2)
# Calculate the degrees of freedom for error
df_error_v <- nobs(model_v) - length(coefficients(model_v))
# Calculate the Sum of squares and products for error (SSPE)
SSPE_v <- RSS_v / df_error_v
# View the SSPE
SSPE_v

model_w <- lm(w ~ X1 + X2 + X3 -1, data = data_MGLM)
# Calculate the residual sums of squares (RSS)
RSS_w <- sum(residuals(model_w)^2)
# Calculate the degrees of freedom for error
df_error_w <- nobs(model_w) - length(coefficients(model_w))
# Calculate the Sum of squares and products for error (SSPE)
SSPE_w <- RSS_w / df_error_w
# View the SSPE
SSPE_w
```

Result in R:

```
> SSPE_y
[1] 0.75
> SSPE_z
[1] 3.6
> SSPE_v
[1] 2.35
> SSPE_w
[1] 12.15
```

ANSWER 2: Y

We now test whether $[\beta_x \quad \beta_y \quad \beta_z \quad \beta_w]$ *are all equal to 0 with the following model*

$$H_0: \boldsymbol{A}\begin{bmatrix}\alpha_y & \alpha_z & \alpha_v & \alpha_w \\ \beta_y & \beta_z & \beta_v & \beta_w \\ \gamma_y & \gamma_z & \gamma_v & \gamma_w\end{bmatrix}\boldsymbol{B}^T = \boldsymbol{C} \quad vs. \quad H_1: \boldsymbol{A}\begin{bmatrix}\alpha_y & \alpha_z & \alpha_v & \alpha_w \\ \beta_y & \beta_z & \beta_v & \beta_w \\ \gamma_y & \gamma_z & \gamma_v & \gamma_w\end{bmatrix}\boldsymbol{B}^T \neq \boldsymbol{C}$$

Question 1.7.

In the above model A is equal to:

A selects the rows, and we need to select the second row.

ANSWER 1: $[0 \quad 1 \quad 0]$

Question 1.8.

The usual test-statistic for the above model has – under the null-hypothesis – the following distribution:

We find the relevant theorem

Theorem 4.21

We consider the above mentioned situation including the assumption of the normality of the observations. Furthermore we consider the hypothesis

$$H_0 : \mathbf{A}\,\boldsymbol{\theta}\,\mathbf{B}^T = \mathbf{C} \quad \text{against} \quad H_1 : \mathbf{A}\,\boldsymbol{\theta}\,\mathbf{B}^T \neq \mathbf{C},$$

where $\mathbf{A}(r \times k)$, $\mathbf{B}(s \times p)$ and $\mathbf{C}(r \times s)$ are given (known) matrices. We introduce

$$\begin{aligned} \boldsymbol{\Delta} &= \mathbf{A}\,\hat{\boldsymbol{\theta}}\,\mathbf{B}^T - \mathbf{C} \\ \mathbf{R} &= n\hat{\boldsymbol{\Sigma}}^* = (\mathbf{Y} - \mathbf{x}\,\hat{\boldsymbol{\theta}})^T(\mathbf{Y} - \mathbf{x}\,\hat{\boldsymbol{\theta}}) = \mathbf{Y}^T\mathbf{Y} - \hat{\boldsymbol{\theta}}^T(\mathbf{x}^T\mathbf{x})\hat{\boldsymbol{\theta}} \end{aligned}$$

and

$$\begin{aligned} \mathbf{E} &= \mathbf{B}\,\mathbf{R}\,\mathbf{B}^T \\ \mathbf{H} &= \boldsymbol{\Delta}^T[\mathbf{A}(\mathbf{x}^T\mathbf{x})^{-1}\mathbf{A}^T]^{-1}\boldsymbol{\Delta}. \end{aligned}$$

The likelihood ratio test for testing H_0 against H_1 is equivalent to the test given by the critical region

$$\{\mathbf{y} | \frac{\det(\mathbf{e})}{\det(\mathbf{e}+\mathbf{h})} \leq \mathrm{U}(s, r, n-k)_\alpha\},$$

where $\mathrm{U}(s, r, n-k)_\alpha$ is the α quantile in the null-hypothesis distribution of the test statistic (see below).

U(s, r, n-k)

We know C(1 x 4)
As we need the entire row B must be the identity matrix B(4 x 4)
And A is A(1 x 3)
Finally we have 5 observations

ANSWER 4: $U(4, 1, 5-3) = U(4, 1, 2)$

Problem 2.

Enclosure A with SAS program and SAS output belongs to this problem. We consider data for the 98 municipalities (kommuner) in Denmark. We have the rates (pr. 1000 capita) of different types of library use, e.g. book loan, music loan, etc. Further, we consider the educational levels as the fraction of the population with that educational level, e.g. a H1 at 0.25 means that 25 % of the population in a given municipality has primary school has the highest education. (Source http://www.statistikbanken.dk)

We shall now investigate the relations between the use of libraries and the educational level by means of a Canonical Correlation Analysis.

We consider the following variables for library use

SAS-name	Meaning
U1	Books
U2	Serial publications
U3	Audio books
U4	Music
U5	Live images (movies)
U6	Multi media material
U7	Other material

And for educational level

SAS-name	Meaning
H1	Primary school
H2	High school
H3	Vocational school
H4	Short further educations
H5	Medium further education
H6	Bachelor level
H7	Master level
H8	Ph.D. level

We shall now investigate the relations between the use of libraries and the educational level by means of a Canonical Correlation Analysis.

In R:

```
# Firstly, in R we need to find the correlation for the variables which we will use in the problem
# SAS do it automatically
# Read the data from the csv file
data_P2=read.csv("Uddannelse2019.csv")
# View the first rows of the data
head(data_P2)
# find the correlation of variables using in this problem
df = data_P2[, c("U1", "U2", "U3", "U4", "U5", "U6", "U7","H1", "H2", "H3", "H4", "H5", "H6", "H7", "H8")]
df_corr = cor(df)
```

Question 2.1.

The first canonical correlation describes which fraction of the variation between V1 and W1

In R:

```
## Computing Canonical correlations and Eigenvalues.
## var U1-U7 in SAS are the variables U1-U7 for y in R in E..
## with H1-H8 in SAS are the variables H1-H8 for x in R in E..
Exx = as.matrix(df_corr[8:15,8:15])
Eyx = as.matrix(df_corr[1:7,8:15])
Exy = as.matrix(df_corr[8:15,1:7])
Eyy = as.matrix(df_corr[1:7,1:7])
invExx = solve(Exx)
invEyy = solve(Eyy)

# Add diagonal loading (regularization) to make the matrices positive definite
epsilon <- 1e-5  # A small positive value
Eyx <- Eyx + epsilon * diag(dim(Eyx)[1])
Eyy <- Eyy + epsilon * diag(dim(Eyy)[1])
#install.packages("geigen")
library("geigen")

#canonical correlation:
Cancorr = geigen(Eyx%*%invExx%*%Exy,Eyy,symmetric = TRUE)
values = sort(Cancorr$values,decreasing = TRUE)


#E is the residual variation after having predicted Y by means of X
H = Eyx%*%invExx%*%Exy
E = Eyy - Eyx%*%invExx%*%Exy
invE = solve(E)
Ev <- eigen(invE%*%H)
var = Ev$values
# Eigenvalues, Proportion and Cumulative proportion of Variance:
varPC <- var/sum(var)
cumu = c(1:7)
for (i in 1:7){
  cumu[i] = sum(varPC[1:i])
}
results <- data.frame("CanCor" = sqrt(values),"Squared CanCor" =
values,"eigenvalues"=var,"proportion"=varPC,
                      "cumulative" = cumu)
print(results)
```

Result in R:

```
> print(results)
      CanCor Squared.CanCor  eigenvalues   proportion cumulative
1 0.76477768   0.5848849069 1.4089704678 0.4822514189  0.4822514
2 0.65086130   0.4236204326 0.7349678173 0.2515590503  0.7338105
3 0.56807817   0.3227128108 0.4764785396 0.1630853570  0.8968958
4 0.39189426   0.1535811135 0.1814481174 0.0621046460  0.9590005
5 0.27825383   0.0774251919 0.0839229418 0.0287244898  0.9877250
6 0.18452075   0.0340479078 0.0352480295 0.0120644205  0.9997894
7 0.02479864   0.0006149726 0.0006153511 0.0002106176  1.0000000
```

We look at the squared canonical correlation column in the table and we receive the requested value.

ANSWER 5: 0.5849

Question 2.2.

How much of the variance in U1 is explained by V1

In R:

```
# Correlations between the Var variables (U) and their Canonical variables
Cor_U = Eyy%*%Cancorr$vectors

#Corresponding standardized canonical coefficients / correlations:
Coefficients = data.frame("Variables" = colnames(df_corr)[1:7],
                          "CorrV1" = round(Cor_U[,7],4),
                          "CorrV2" = -round(Cor_U[,6],4),
                          "CorrV3" = round(Cor_U[,5],4),
                          "CorrV4" = -round(Cor_U[,4],4),
                          "CorrV5" = round(Cor_U[,3],4),
                          "CorrV6" = -round(Cor_U[,2],4),
                          "CorrV7" = -round(Cor_U[,1],4))

print("Correlations between the Var variables (U) and their Canonical variables:")
Coefficients
```

Result in R:

```
[1] "Correlations between the Var variables (U) and their Canonical variables:"
> Coefficients
   Variables  CorrV1  CorrV2  CorrV3  CorrV4 CorrV5  CorrV6  CorrV7
U1        U1 -0.7532 -0.4078  0.0131  0.4316 0.1035  0.2455 -0.0950
U2        U2  0.2246 -0.5347  0.0256  0.3848 0.6136  0.3431 -0.1436
U3        U3 -0.4422  0.2017  0.5650  0.3946 0.4631  0.1152 -0.2472
U4        U4 -0.5616  0.4018 -0.3959  0.3622 0.4362 -0.0245 -0.2108
U5        U5 -0.3860  0.1324  0.2111  0.2110 0.3682  0.6565 -0.4216
U6        U6  0.0958  0.4143  0.1493  0.6380 0.0214  0.6237 -0.0204
U7        U7 -0.4215  0.2987 -0.0629 -0.0146 0.3268  0.2678  0.7420
```

We take the correlation between U1 and V1 and then we square it.

We square it $(-0.7532)^2 = 0.5673$

ANSWER 2: 0.5673

Question 2.3.

The first canonical variate W1 can be interpreted as

In R:

```
#for H variables:
Cancorr2 = geigen(Exy%*%invEyy%*%Eyx,Exx,symmetric = TRUE)

# Correlations between the Var variables (H) and their Canonical variables
Cor_H = Exx%*%Cancorr2$vectors
```

```
#Corresponding standardized canonical coefficients / correlations:
Coefficients = data.frame("Variables" = colnames(df_corr)[8:15],
                          "CorrW1" = -round(Cor_H[,8],4),
                          "CorrW2" = round(Cor_H[,7],4),
                          "CorrW3" = round(Cor_H[,6],4),
                          "CorrW4" = -round(Cor_H[,5],4),
                          "CorrW5" = round(Cor_H[,4],4),
                          "CorrW6" = round(Cor_H[,3],4),
                          "CorrW7" = round(Cor_H[,2],4))

print("Correlations between the Var variables (H) and their Canonical variables:")
Coefficients
```

Result in R:

```
[1] "Correlations between the Var variables (H) and their Canonical variables:"
> Coefficients
   Variables  CorrW1  CorrW2  CorrW3  CorrW4  CorrW5  CorrW6  CorrW7
H1        H1  0.8011  0.2581  0.1093  0.1871  0.2245 -0.0680  0.1485
H2        H2 -0.5932  0.4694 -0.3295 -0.0183 -0.3781  0.1419 -0.1080
H3        H3  0.8678 -0.0786  0.2586  0.0179  0.3341 -0.2469  0.0214
H4        H4 -0.4394  0.1912  0.3919 -0.1344 -0.3453 -0.4744 -0.4129
H5        H5 -0.7879  0.0105 -0.3570 -0.1090  0.1755 -0.4051 -0.1169
H6        H6 -0.5541  0.2595 -0.4377 -0.3858 -0.3719  0.2241  0.0428
H7        H7 -0.8623 -0.0016 -0.1228 -0.2225 -0.2708  0.3133  0.0311
H8        H8 -0.8735 -0.0707 -0.0324 -0.0798 -0.2877  0.2479  0.2460
```

We see that it is a contrast between primary and vocational school against all other educations.

ANSWER 4: A contrast between primary and vocational school against all other educations.

Problem 3.

Enclosure B with SAS program and SAS output belongs to this problem. As in Problem 2, we consider library data, but now for the Capital region only. We want to predict the music loans (pr. 1000 capita) pr. municipality (the U4 variable in Problem 2) based on 5 financial variables (in 1.000 DKK pr. 1.000 capita) for the libraries in each municipality

SAS-name	Meaning
F1	Salary costs
F2	Material costs
F3	Other costs
F4	Income (e.g. late fees)
F5	Netto costs

We consider two models – all with an intercept:

- M1: All variables
- M2: which is the resulting model, after we have reduced the number of explanatory variables by stepwise model selection.

Load the data in R:

```
# Read the data from the csv file
data_P3=read.csv("Bibliotek2019.csv")
# View the first rows of the data
head(data_P3)
```

Question 3.1.

The reduction in variance explained when going from model M1 to model M2 is

In R:

```
########## Model M1 #######
# Fit a linear regression model
Model_M1 <- lm(U4 ~  F1+F2+F3+F4+F5,data=data_P3)

# Summary of the Model_M1 model
summary(Model_M1)

# Calculate the R-squared value
summary_model_M1 <- summary(Model_M1)
r_squared_M1 <- summary_model_M1$r.squared

# Print the R-squared value
cat("R-squared value:", r_squared_M1, "\n")

########## Model M2 #######
# Initialize an empty model
best_model <- lm(U4 ~ 1, data = data_P3)
```

```
# Create a list of predictor variables
predictors <- c("F1", "F2", "F3", "F4","F5")

# Initialize a list to keep track of selected predictors
selected_predictors <- character(0)

# Initialize an empty best AIC (you can use BIC or other criteria)
best_aic <- Inf

# Loop through predictors
for (predictor in predictors) {
  # Add the predictor to the model
  temp_model <- update(best_model, formula = as.formula(paste("U4 ~", paste(selected_predictors, collapse
= " + "), " + ", predictor)), data = data_P3)

  # Calculate AIC (or other criterion) for the new model
  temp_aic <- AIC(temp_model)

  # Check if AIC is improved
  if (temp_aic < best_aic) {
    best_aic <- temp_aic
    best_model <- temp_model
    selected_predictors <- c(selected_predictors, predictor)
  }
}

# Final selected model
final_model_M2 <- best_model

# Display the selected predictors
cat("Selected Predictors:", selected_predictors, "\n")

# Summary of the final model
summary(final_model_M2)

# Calculate the R-squared value
summary_model_M2 <- summary(final_model_M2)
r_squared_M2 <- summary_model_M2$r.squared

# Print the R-squared value
cat("R-squared value:", r_squared_M2, "\n")

# Reduction
cat("The reduction is:", r_squared_M1-r_squared_M2, "\n")
```

Result in R:

```
> # Print the R-squared value
> cat("R-squared value:", r_squared_M1, "\n")
R-squared value: 0.6151237
```

```
> # Print the R-squared value
> cat("R-squared value:", r_squared_M2, "\n")
R-squared value: 0.5749666
```

The reduction is R_squared_M1 - R_squared_M2 = 0.6151-0.575 = 0.04015705

```
> # Reduction
> cat("The reduction is:", r_squared_M1-r_squared_M2, "\n")
The reduction is: 0.04015705
```

ANSWER 1: 0.0401

Question 3.2.

If we performed backwards elimination from M1, the first variable to be excluded is

In R:

```
# Summary of the Model_M1 model
summary(Model_M1)
```

Result in R:

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   -41.75      72.36  -0.577    0.570
F1            -22.42      25.46  -0.881    0.388
F2            -24.17      25.64  -0.943    0.356
F3            -23.23      25.60  -0.907    0.374
F4             23.85      25.64   0.930    0.362
F5             23.15      25.54   0.906    0.374

Residual standard error: 73.54 on 23 degrees of freedom
Multiple R-squared:  0.6151,    Adjusted R-squared:  0.5315
F-statistic: 7.352 on 5 and 23 DF,  p-value: 0.0003028
```

We look at the p-value column in the table to find the variable with the highest p-value.

ANSWER 1: F1

Question 3.3.

What is the usual test statstic for M1 vs M2:

In R:

```
anova(Model  M1)
anova_M1 = anova(Model_M1)

anova(final_model_M2)
anova_M2 = anova(final_model_M2)

cat("SS res M2:",anova_M2$`Sum Sq`[3], "\n")
cat("SS res M1:",anova_M1$`Sum Sq`[6], "\n")
cat("DF res M2:",anova_M2$Df[3], "\n")
cat("SS res M1:",anova_M1$Df[6], "\n")
```

Result in R:

```
> cat("SS res M2:",anova_M2$`Sum Sq`[3], "\n")
SS res M2: 137360.2
> cat("SS res M1:",anova_M1$`Sum Sq`[6], "\n")
SS res M1: 124382.4
> cat("DF res M2:",anova_M2$Df[3], "\n")
DF res M2: 26
> cat("SS res M1:",anova_M1$Df[6], "\n")
SS res M1: 23
```

The relevant test is given in

Test statistic for $H_0: E(\boldsymbol{Y}) \in H_{i+1}$ against $H_1: E(\boldsymbol{Y}) \in H_i \backslash H_{i+1}$:

$$\frac{\left\|p_{H_i}(Y)-p_{H_{i+1}}(Y)\right\|^2/(r_i-r_{i+1})}{\left\|Y-p_{H_i}(Y)\right\|^2/(n-r_i)} = \frac{[SS_{res}(H_{i+1})-SS_{res}(H_i)]/[DF_{res}(H_{i+1})-DF_{res}(H_i)]}{SS_{res}(H_i)/DF_{res}(H_i)}$$

We insert:

$$\frac{(137360-124382)/(26-23)}{\frac{124382}{23}}$$

ANSWER 2

Question 3.4.

The distribution of the above statistic under the null hypothesis is

We find in the notes:

$$\left\{\boldsymbol{Y} \,\middle|\, F > F(DF_{res}(H_{i+1}) - DF_{res}(H_i), DF_{res}(H_i))_{1-p}\right\}$$

F(26-23,23)

ANSWER 4: F(3,23)

We now only consider M2

Question 3.5.

The observation with the highest leverage is

In R:

```
Obs <- 1:length(data_P3$U4)
DFFITS <- round(dffits(final_model_M2), 4)
R_student <- round(rstudent(final_model_M2),4)
HatDiagH <- round(hatvalues(final_model_M2),4)
Residual <- round(residuals(final_model_M2),4)
Covratio <- round(covratio(final_model_M2), 4)
Predicted_value <- round(predict(final_model_M2),4)

Stats  <- data.frame(Obs,Predicted_value,Residual,R_student,HatDiagH,Covratio,DFFITS)
```

```
print(Stats)

# Find out the maximum HatDiagH value
obs_with_max_hat_diag_H <- which.max(Stats$HatDiagH)

# Print the observation number with the maximum HatDiagH
cat("Observation with Max HatDiagH:", obs_with_max_hat_diag_H, "\n")
```

Result in R:

```
> print(Stats)
   Obs Predicted_value  Residual R_student HatDiagH Covratio  DFFITS
1    1         66.9892   36.0108    0.5196   0.1163   1.2326  0.1885
2    2        173.7829  123.2171    1.7979   0.0346   0.8090  0.3403
3    3        245.8493 -190.8493   -3.2262   0.0979   0.4389 -1.0629
4    4        262.8382   -8.8382   -0.1276   0.1257   1.2840 -0.0484
5    5        329.3542  117.6458    1.8943   0.1972   0.9370  0.9389
6    6        236.8447 -100.8447   -1.4649   0.0635   0.9382 -0.3813
7    7        108.1888  -55.1888   -0.7757   0.0565   1.1101 -0.1898
8    8        352.0847   -2.0847   -0.1112   0.9360  17.5583 -0.4255
9    9        273.4345   98.5655    1.4890   0.1318   1.0041  0.5801
10  10        167.2975   17.7025    0.2439   0.0389   1.1621  0.0491
11  11        182.8171   83.1829    1.1784   0.0427   0.9992  0.2490
12  12        205.0262   37.9738    0.5312   0.0594   1.1563  0.1335
13  13        129.0066  -30.0066   -0.4160   0.0464   1.1554 -0.0917
14  14        172.5641 -124.5641   -1.8210   0.0355   0.8026 -0.3492
15  15        211.4022   43.5978    0.6074   0.0484   1.1312  0.1370
16  16        148.6520   -4.6520   -0.0640   0.0386   1.1694 -0.0128
17  17         33.4659   19.5341    0.2850   0.1423   1.2987  0.1161
18  18        134.6479  -24.6479   -0.3404   0.0414   1.1573 -0.0708
19  19         84.3695   -7.3695   -0.1035   0.0765   1.2164 -0.0298
20  20        164.5000  -63.5000   -0.8882   0.0404   1.0679 -0.1824
21  21        108.9383    7.0617    0.0980   0.0555   1.1896  0.0238
22  22        100.4049  -18.4049   -0.2567   0.0617   1.1894 -0.0658
23  23        140.7586  -88.7586   -1.2609   0.0408   0.9747 -0.2602
24  24         85.8387   19.1613    0.2690   0.0740   1.2042  0.0760
25  25        313.0220  -30.0220   -0.4464   0.1702   1.3236 -0.2021
26  26        122.5045   31.4955    0.4368   0.0464   1.1531  0.0964
27  27        116.8929   48.1071    0.6743   0.0568   1.1298  0.1654
28  28        169.9855   18.0145    0.2480   0.0371   1.1596  0.0487
29  29         76.5392   48.4608    0.6909   0.0876   1.1648  0.2141

> # Print the observation number with the maximum HatDiagH
> cat("Observation with Max HatDiagH:", obs_with_max_hat_diag_H, "\n")
Observation with Max HatDiagH: 8
```

ANSWER 4: 8

Question 3.6.

The observation with the highest impact on the intercept is

In R:

```
dfbetas_values <- round(dfbetas(final_model_M2), 4)
dfbetas_values
# Find out the maximum DFBETAS value for the intercept
obs_with_max_DFBETAS_intercept <- which.max(abs(dfbetas_values[, "(Intercept)"]))
```

# Print the observation number with the maximum DFBETAS value for the intercept
cat("Observation with Max DFBETAS value for the intercept:", obs_with_max_DFBETAS_intercept, "\n")

Result in R:

```
> dfbetas_values
   (Intercept)      F1      F4
1       0.1776 -0.1581  0.0469
2       0.0783  0.0182 -0.0009
3       0.5743 -0.8387  0.4138
4       0.0294 -0.0405  0.0192
5      -0.7064  0.8379 -0.1010
6       0.0933 -0.1382 -0.1657
7      -0.1312  0.0757  0.0641
8       0.0080  0.0546 -0.4112
9      -0.3719  0.4962 -0.1946
10      0.0084  0.0081 -0.0162
11     -0.0065  0.0882 -0.0883
12     -0.0377  0.0786 -0.0581
13     -0.0497  0.0202  0.0338
14     -0.0658 -0.0424  0.0508
15     -0.0359  0.0734 -0.0230
16     -0.0053  0.0011  0.0035
17      0.1125 -0.0960 -0.0010
18     -0.0457  0.0271  0.0017
19     -0.0267  0.0209  0.0005
20     -0.0324 -0.0298  0.0694
21      0.0192 -0.0137 -0.0006
22     -0.0554  0.0410  0.0023
23     -0.1644  0.1026 -0.0292
24      0.0672 -0.0516 -0.0042
25      0.1470 -0.1785  0.0281
26      0.0661 -0.0391 -0.0163
27      0.1334 -0.1036  0.0330
28      0.0085  0.0074 -0.0123
29      0.1979 -0.1621  0.0118
```

```
> # Print the observation number with the maximum DFBETAS value for the intercept
> cat("Observation with Max DFBETAS value for the intercept:", obs_with_max_DFBETAS_intercept, "\n")
Observation with Max DFBETAS value for the intercept: 5
```

We use the table above and look at the DFBETAS

ANSWER 3: 5

Question 3.7.

What is the 95% confidence interval for observation no. 3

We use

Theorem 2.15

Let the situation be as above. Then the $(1-\alpha)$-*confidence interval* for the expected value of a new observation Y will be

$$[u - \mathrm{t}(n-k)_{1-\frac{\alpha}{2}} s\sqrt{c},\quad u + \mathrm{t}(n-k)_{1-\frac{\alpha}{2}} s\sqrt{c}].$$

In R:

```
# Calculate the sum of squares for the error
SS_error <- anova_M2$`Mean Sq`[3]

# Print SS error
cat("Sum of Squares (SS) Error:", SS_error, "\n")

# Calculate rmse
Rmse <- sqrt(SS_error)

# Print SS error
cat("Sum of Squares (SS) Error:", Rmse, "\n")

# Extract the predicted value and HatDiagH value for observation 3
obs_3 <- 3
predicted_value_3 <- Predicted_value[obs_3]
HatDiagH_3 <- HatDiagH[obs_3]

cat("Predicted Value for Observation 3:", predicted_value_3, "\n")
cat("HatDiagH Value for Observation 3:", HatDiagH_3, "\n")

# Number of observations in the library dataset
num_observations <- nrow(data_P3)

# Number of independent variables in the model
num_variables <- length(coefficients(final_model_M2))

# Print the results
cat("Number of Observations in Dataset:", num_observations, "\n")
cat("Number of Variables in the Model:", num_variables, "\n")
```

Result in R:

```
> # Print SS error
> cat("Sum of Squares (SS) Error:", SS_error, "\n")
Sum of Squares (SS) Error: 5283.084
> # Print SS error
> cat("Sum of Squares (SS) Error:", Rmse, "\n")
Sum of Squares (SS) Error: 72.68482
> cat("Predicted Value for Observation 3:", predicted_value_3, "\n")
Predicted Value for Observation 3: 245.8493
> cat("HatDiagH Value for Observation 3:", HatDiagH_3, "\n")
HatDiagH Value for Observation 3: 0.0979
> # Print the results
> cat("Number of Observations in Dataset:", num_observations, "\n")
Number of Observations in Dataset: 29
> cat("Number of Variables in the Model:", num_variables, "\n")
Number of Variables in the Model: 3
```

We have the predicted value from the results above, 245.8493, further we need the number of observation n=29, and the number of parameters in the model k=3. As $c = h_{ii} = 0.0979$ from the results above in R,

We have the the MSE=5283.08, leading to s=RMSE=72.6848

$$s\sqrt{c} = 72.6848\sqrt{0.0979} = 22.7423$$

ANSWER 2 : $\mathbf{245.8493 \pm t(26)_{0.975} \times 22.7423}$

Question 3.8.

What is the variance of the model M2, if observation 8 is deleted?

In R:

```
# Extract residual, R_student, and HatDiagH for observation 8
obs_8 <- 8  # Replace with the desired observation number
residual_8 <- Stats$Residual[obs_8]
R_student_8 <- Stats$R_student[obs_8]
HatDiagH_8 <- Stats$HatDiagH[obs_8]

cat("Residual for Observation 8:", residual_8, "\n")
cat("R_student for Observation 8:", R_student_8, "\n")
cat("HatDiagH for Observation 8:", HatDiagH_8, "\n")

new_variance = residual_8/(R_student_8*sqrt(1-HatDiagH_8))
cat("The new variance of the model M2, if observation 8 is deleted:", new_variance^2, "\n")
```

Result in R:

```
> cat("Residual for Observation 8:", residual_8, "\n")
Residual for Observation 8: -2.0847
> cat("R_student for Observation 8:", R_student_8, "\n")
R_student for Observation 8: -0.1112
> cat("HatDiagH for Observation 8:", HatDiagH_8, "\n")
HatDiagH for Observation 8: 0.936
>
> new_variance = residual_8/(R_student_8*sqrt(1-HatDiagH_8))
> cat("The new variance of the model M2, if observation 8 is deleted:", new_variance^2, "\n")
The new variance of the model M2, if observation 8 is deleted: 5491.583
```

We use

RSTUDENT is a so-called "studentised" residual, i.e.

$$\text{RSTUDENT}_i = \frac{r_i}{\hat{\sigma}(i)\sqrt{1-h_{ii}}},$$

where $\hat{\sigma}(i)^2$ is the estimate of variance corresponding to deletion of the i'th observation.

We isolate and insert

$$\hat{\sigma}(8) = \frac{r_8}{RSTUDENT_8\sqrt{1-h_{88}}} = \frac{-2.0847}{-0.1112\sqrt{1-0.9360}} = 74.1052$$

And the variance is thus 5491.58

ANSWER 4 : 5491.58

Problem 4.

Enclosure C with SAS program and SAS output belongs to this problem. We now consider a subset of municipalities, that are either high or low crime. There might be a link between crime levels and library use. We will test if we can classify these municipalities as a high or low crime municipality, based on the 7 use of library variables (U1 – U7) described in problem 2. (Source http://www.statistikbanken.dk)

Load the data in R

```
# Read the data from the csv file
data_P4=read.csv("KrimBiblio2019.csv")
# View the first rows of the data
head(data_P4)
```

Question 4.1.

The number of misclassifications by resubstitution when going from Linear Discriminant Analysis with all variables to Quadratic Discriminant Analysis with all variables is reduced with:

In R:

```
library(MASS)

# Prior probabilities for the classes
# SAS by default has equal prior probabilities, so we need equal prior in R to receive the same results
different_types <- unique(data_P4$type)
num_types  <- length(different_types)
prior  <-  rep(1/num_types, num_types)

#linear discriminant analysis
# Define the classes (type)
data_P4$class <- as.factor(data_P4$type)
# Define the variables for the analysis
variables <- c("U1", "U2", "U3", "U4", "U5" ,"U6" ,"U7","U8")
# Perform Linear Discriminant Analysis
z <- lda(class ~ ., data = data_P4[, c("class", variables)],prior=prior)


Class_Level_Information_LDA = data.frame("Frequency" =
z$counts,"Proportion"=z$counts/z$N,"Prior"=z$prior)
print("Class Level Information LDA:")
Class_Level_Information_LDA


n <- nrow(data_P4)
Classes <- nlevels(data_P4$type)
```

```
paste0("DF Within Classes = ",n-Classes)
paste0("DF Between Classes = ",Classes-1)

zpred <- predict(z)

#Confusion Matrix:
print("Confusion Matrix Linear Discriminant Analysis:")
xtabs(~data_P4$type+zpred$class)

#########################################################
library(MASS)

# Prior probabilities for the classes
# SAS by default has equal prior probabilities, so we need equal prior in R to receive the same results
different_types <- unique(data_P4$type)
num_types  <- length(different_types)
prior  <-  rep(1/num_types, num_types)

# Define the classes (type)
data_P4$class <- as.factor(data_P4$type)
# Define the variables for the analysis
variables <- c("U1", "U2", "U3", "U4", "U5" ,"U6" ,"U7","U8")

# Perform Quadratic Discriminant Analysis
zqda <- qda(class ~ ., data = data_P4[, c("class", variables)], prior = prior)

Class_Level_Information_QDA = data.frame("Frequency" =
zqda$counts,"Proportion"=zqda$counts/zqda$N,"Prior"=zqda$prior)
print("Class Level Information QDA:")
Class_Level_Information_QDA


n <- nrow(data_P4)
Classes <- nlevels(data_P4$type)

paste0("DF Within Classes = ",n-Classes)
paste0("DF Between Classes = ",Classes-1)

zpred_qda <- predict(zqda)

#Confusion Matrix:
print("Confusion Matrix Quadratic discriminant analysis:")
xtabs(~data_P4$type+zpred_qda$class)
```

Result in R:

```
[1] "Confusion Matrix Linear Discriminant Analysis:"
> xtabs(~data_P4$type+zpred$class)
             zpred$class
data_P4$type HighCrim LowCrime
    HighCrim       17        4
    LowCrime        1       11


[1] "Confusion Matrix Quadratic discriminant analysis:"
> xtabs(~data_P4$type+zpred_qda$class)
             zpred_qda$class
data_P4$type HighCrim LowCrime
    HighCrim       20        1
    LowCrime        0       12
```

We go from 5 misclassification to 1, so
ANSWER 4 : 4

Question 4.2.

The Hotelling T^2 for the hypothesis of same mean in the two groups is

Here we have to be a bit careful, as we have both the table from the QDA and the LDA. We consider

Definition 5.15

Assuming that the hypothesis $H_0 : \Sigma_1 = \cdots = \Sigma_k$ is true, we define *the squared generalized distance from* $\widehat{\mu}_j$ *to population* π_i as

$$D_i^2(\widehat{\mu}_j) = \begin{cases} \left(\widehat{\mu}_i - \widehat{\mu}_j\right)^T \widehat{\Sigma}^{-1} \left(\widehat{\mu}_i - \widehat{\mu}_j\right) & \text{if the priors are equal} \\ \left(\widehat{\mu}_i - \widehat{\mu}_j\right)^T \widehat{\Sigma}^{-1} \left(\widehat{\mu}_i - \widehat{\mu}_j\right) - 2\log p_i & \text{if the priors are not all equal} \end{cases}$$

If the hypothesis is *not* true, we define *the squared generalized distance from* $\widehat{\mu}_j$ *to population* π_i as

$$D_i^2(\widehat{\mu}_j) = \begin{cases} \left(\widehat{\mu}_i - \widehat{\mu}_j\right)^T \widehat{\Sigma}_i^{-1} \left(\widehat{\mu}_i - \widehat{\mu}_j\right) & \text{if the priors are equal} \\ \left(\widehat{\mu}_i - \widehat{\mu}_j\right)^T \widehat{\Sigma}_i^{-1} \left(\widehat{\mu}_i - \widehat{\mu}_j\right) + \log\det\widehat{\Sigma}_i - 2\log p_i & \text{if the priors are not all equal} \end{cases}$$

And see that we should use the one from LDA. We use

Theorem 4.9

We use the same notation as given above. Now, let

$$T^2 = \frac{nm}{n+m}(\bar{X} - \bar{Y})^T \mathbf{S}^{-1}(\bar{X} - \bar{Y}).$$

Then the critical region for a test of H_0 against H_1 at level α is equal to

$$C = \{x_1, \ldots, x_n, y_1, \ldots, y_m | \frac{n+m-p-1}{(n+m-2)p} t^2 > \mathrm{F}(p, n+m-p-1)_{1-\alpha}\}$$

Here t^2 is the observed value of T^2.

In R:

```
print("Class Level Information LDA:")
Class_Level_Information_LDA

data_P4$type = as.factor(data_P4$type)
library(Rfast)
pcov  <- pooled.cov(as.matrix(data_P4[,variables]),data_P4$type)
Means <- as.matrix(z$means)
invCov <- solve(pcov)
# Extract unique levels from data_P4$type
unique_levels <- levels(data_P4$type)
num_col <- length(unique(data_P4$type))

# Create an empty matrix to store the Mahalanobis distances with the equal priors ################
maha <- matrix(c(rep(0, num_col^2)), ncol = num_col)

# Define the names for rows and columns
rownames(maha) <- unique_levels
colnames(maha) <- unique_levels

for (i in 1:num_col) {
  for (j in 1:num_col) {
    mu <- Means[i, ] - Means[j, ]
    maha[i, j] <- mu %*% invCov %*% mu
  }
}

# squared Mahalanobis distances between the 2 crime types.
# maha    : Assuming equal priors

print("Generalized Squared Distance to crimes (equal priors):")
```

maha

Result in R:

```
[1] "Generalized Squared Distance to crimes (equal priors):"
> maha
          HighCrim LowCrime
HighCrim 0.000000 3.620014
LowCrime 3.620014 0.000000
```

```
[1] "Class Level Information LDA:"
> Class_Level_Information_LDA
          Frequency Proportion Prior
HighCrim         21  0.6363636   0.5
LowCrime         12  0.3636364   0.5
```

Thus n=21 and m=12

And the answer is $T^2 = \frac{n \cdot m}{n+m} D^2 = \frac{21 \cdot 12}{21+12} 3.62001 = 27.6437$

ANSWER 2 : 27.6437

Question 4.3.

We now test if U1, U2, U3, U5, and U8 contribute to the discrimination between the groups using Linear Discriminant Analysis. The usual test statistic is given by

We use

Theorem 5.21

The critical region for testing the hypothesis that the last $p - q$ variables do not contribute to the discrimination between the populations π_1 and π_2, i.e. the hypothesis that $\Delta^2_{(2|1)} = 0$ against all alternatives is

$$\left\{ x_{11}, \cdots, x_{2n_2} \;\middle|\; \frac{n_1+n_2-p-1}{p-q} \frac{d^2-d_1^2}{(n_1+n_2)(n_1+n_2-2)/(n_1 n_2)+d_1^2} > F(p-q, n_1+n_2-p-1)_{1-\alpha} \right\}$$

Here d^2 and d_1^2 are the observed values of D^2 and D_1^2.

In R:

```
############ Reduced model ##############
variables_reduced <- c("U4", "U6", "U7")
# Perform Linear Discriminant Analysis
z_reduced <- lda(class ~ ., data = data_P4[, c("class", variables_reduced)],prior=prior)

library(Rfast)
pcov_reduced  <- pooled.cov(as.matrix(data_P4[,variables_reduced]),data_P4$type)
```

```
Means_reduced <- as.matrix(z_reduced$means)
invCov <- solve(pcov_reduced)

# Extract unique levels from data_P4$type
unique_levels <- levels(data_P4$type)
num_col <- length(unique(data_P4$type))

# Create an empty matrix to store the Mahalanobis distances with the equal priors ################
maha_reduced <- matrix(c(rep(0, num_col^2)), ncol = num_col)


# Define the names for rows and columns
rownames(maha_reduced) <- unique_levels
colnames(maha_reduced) <- unique_levels

for (i in 1:num_col) {
  for (j in 1:num_col) {
    mu <- Means_reduced[i, ] - Means_reduced[j, ]
    maha_reduced[i, j] <- mu %*% invCov %*% mu
  }
}

# squared Mahalanobis distances between the 2 crime types.
# maha    : Assuming equal priors

print("Generalized Squared Distance to crimes (equal priors):")
maha_reduced

#### Class Level Information from the previous question
print("Class Level Information LDA:")
Class_Level_Information_LDA
```

Result in R:

```
[1] "Generalized Squared Distance to crimes (equal priors):"
> maha_reduced
          HighCrim LowCrime
HighCrim 0.000000 1.524151
LowCrime 1.524151 0.000000
```

```
[1] "Class Level Information LDA:"
> Class_Level_Information_LDA
          Frequency Proportion Prior
HighCrim         21  0.6363636   0.5
LowCrime         12  0.3636364   0.5
```

We can then insert:

ANSWER 3 : $\frac{21+12-8-1}{8-3} \cdot \frac{3.62001-1.52415}{(21+12)(21+12-2)/(21\cdot 12)+1.52415}$

Problem 5.

Enclosure D with SAS program and SAS output belongs to this problem. We now consider the library contents among the 98 municipalities in Denmark given in items pr. 1000 capita. We will analyse if there are any patterns or trends by means of a factor analysis. (Source http://www.statistikbanken.dk)

We consider the following variables for library contents

SAS-name	Meaning
B1	Books
B2	Audio books
B3	Music
B4	Live images (movies)
B5	Multi media material
B6	Other material
B7	Electronic resources

Load the data in R

```
data_5 = read.csv("BiblioFactor2019.csv")
# Define the variables for the analysis
variables <- c("B1", "B2", "B3", "B4", "B5" ,"B6" ,"B7")
data_P5 = data_5[, variables]
```

Question 5.1.

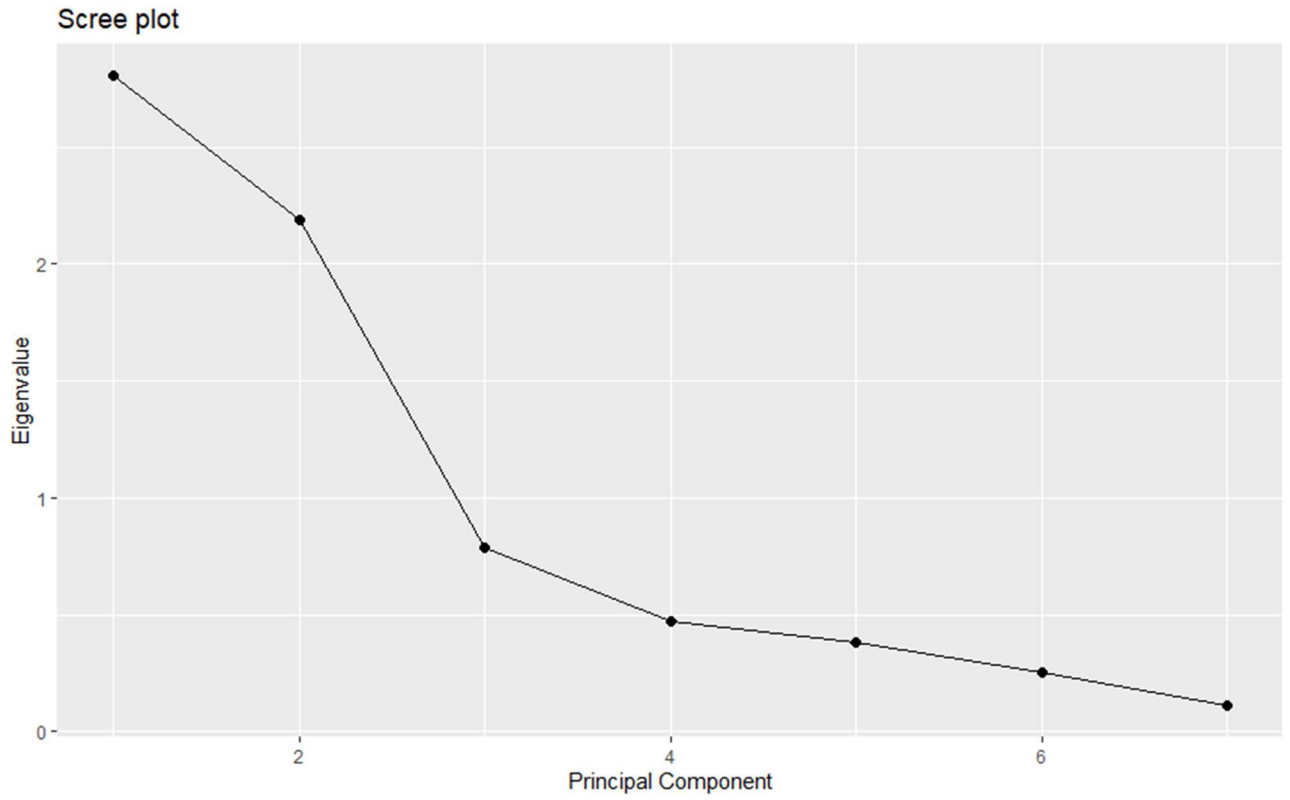
Considering the screeplot we should retain, how many factors in the model

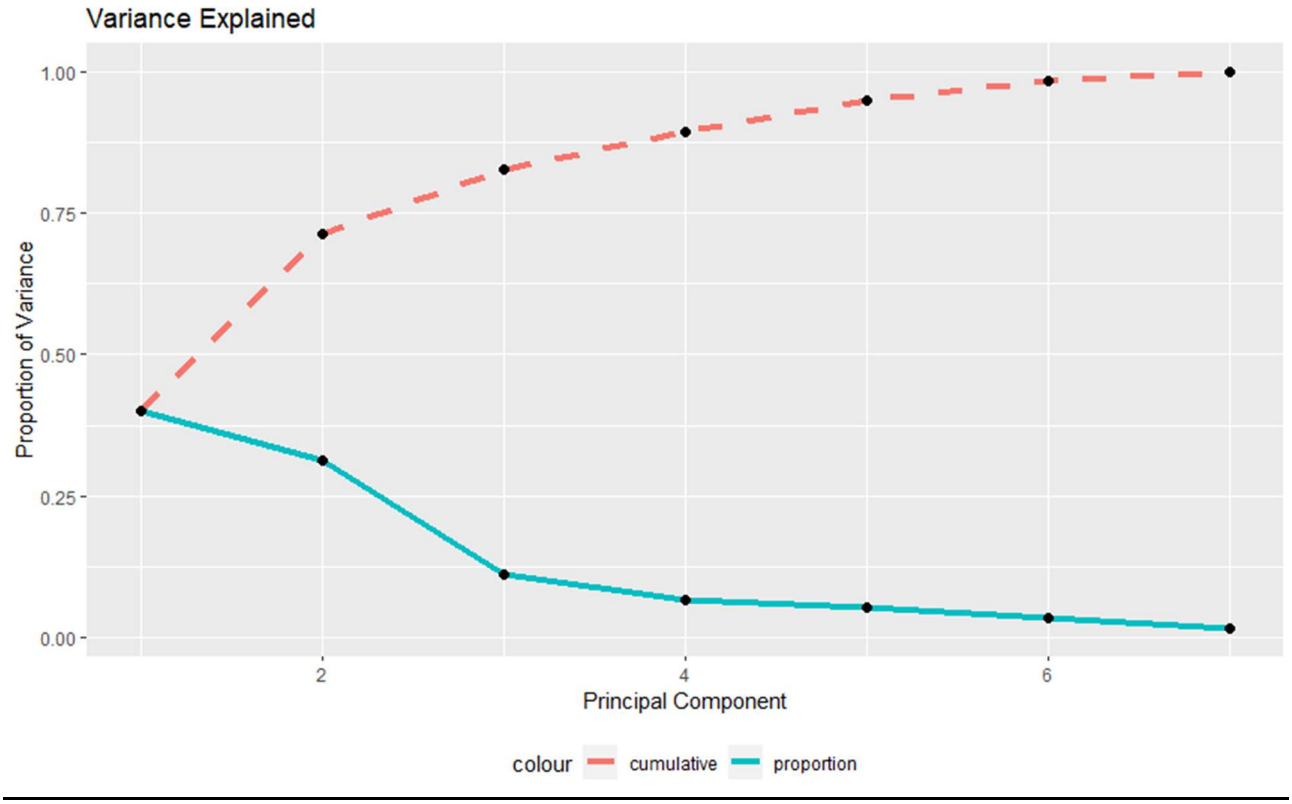
In R:

```
# Principal Component Analysis on the correlation matrix.
pca <- princomp(data_P5,cor=TRUE,scores=TRUE)

# variance for each Principal Component:
var <- pca$sdev^2

# proportion of variance:
varPC <- var/sum(var)

# cumulative variance:
cumu = c(1:7)
for (i in 1:7){
  cumu[i] = sum(varPC[1:i])
}
results_PCA <- data.frame("eigenvalues"=var,"proportion"=varPC,
```

```
                    "cumulative" = cumu)

print("Eigenvalues of the Correlation matrix:")
results_PCA

library(ggplot2)

# Scree plot
Scree = ggplot(results_PCA,aes(x=c(1:7),y=eigenvalues,group=1))+
  geom_line()+
  geom_point(size=2)+
  labs(x='Principal Component',y='Eigenvalue')+
  ggtitle('Scree plot')

# Print the scree plot
print(Scree)

# Variance Explained plot
VarExp = ggplot(results_PCA,aes(x=c(1:7),group=1))+
  geom_line(aes(y=cumulative,col = 'cumulative'),linetype = 'dashed',lwd=1.5)+
  geom_line(aes(y=proportion,col = 'proportion'),lwd=1.5)+
  geom_point(aes(y=cumulative),size=2)+
  geom_point(aes(y=proportion),size=2)+
  labs(x='Principal Component',y='Proportion of Variance')+
  theme(legend.position='bottom')+
  ggtitle('Variance Explained')

# Print the Variance Explained plot
print(VarExp)
```

Result in R:





There is a clear elbow after the first 2. We should thus include 2 factors

ANSWER 4: 2

Question 5.2.

Which of the original variables has the least amount of its variance described by the Factor Analysis?

In R:

```
#### Factor analysis 3 Factors ####
#install.packages("psych")
# Load the psych package
library(psych)
fa3 <- principal(cor(data_P5),nfactors = 3,rotate = "none")
fa3l <- fa3$loadings[,1:3]

#Communality:
fa3com <- fa3$communality
print("Final Communality Estimates unrotated:")
fa3com
```

Result in R:

```
> print("Final Communality Estimates unrotated:")
[1] "Final Communality Estimates unrotated:"
> fa3com
       B1        B2        B3        B4        B5        B6        B7
0.9115998 0.6892683 0.9910335 0.8287701 0.7718174 0.7563000 0.8361301
```

From the table with communalities in R we take the lowest value representing the variable with the least amount of its variance described by Factor Analysis.

ANSWER 2 : B2

Question 5.3.

Looking at the score plots, we can conclude

First we will investigate the meaning of the different rotated factors.
We inspect the factor patterns.

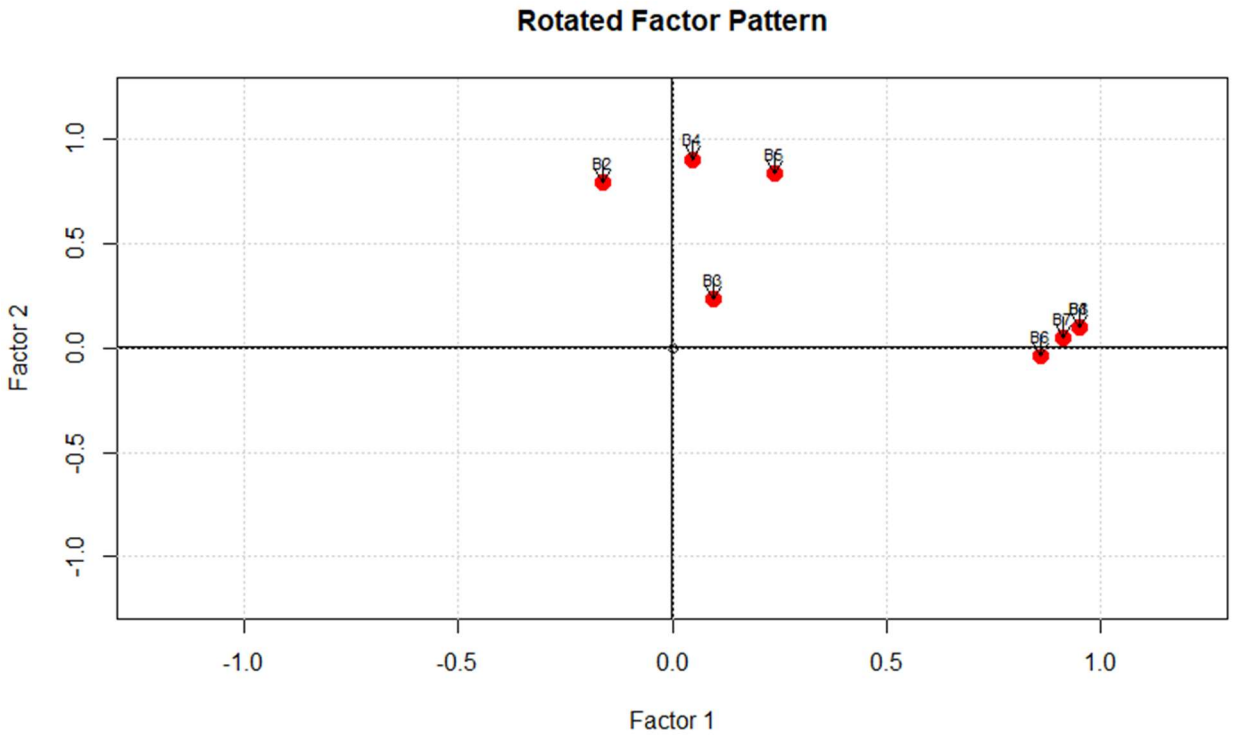
In R:

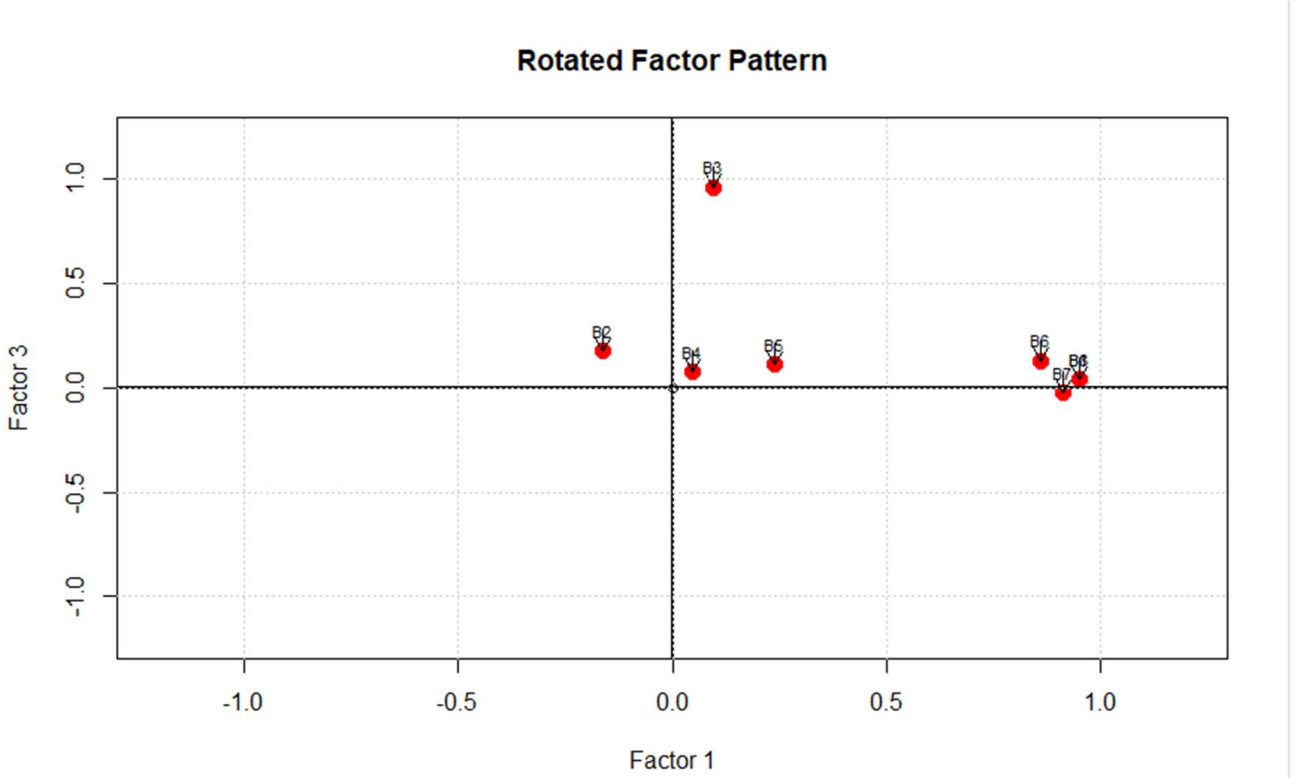
```
#rotated:
rfa3 <- principal(cor(data_P5),nfactors = 3,rotate = "varimax")
rfa3l <- rfa3$loadings[,1:3]
print(rfa3l)

# Plots for factor analysis with 3 factors:
par(mfrow = c(1, 1))
circle <- seq(-3.2, 3.2, by = 0.1)

# Different combinations of plots
ij <- matrix(c(1, 1, 2, 2, 3, 3), ncol = 2)
```

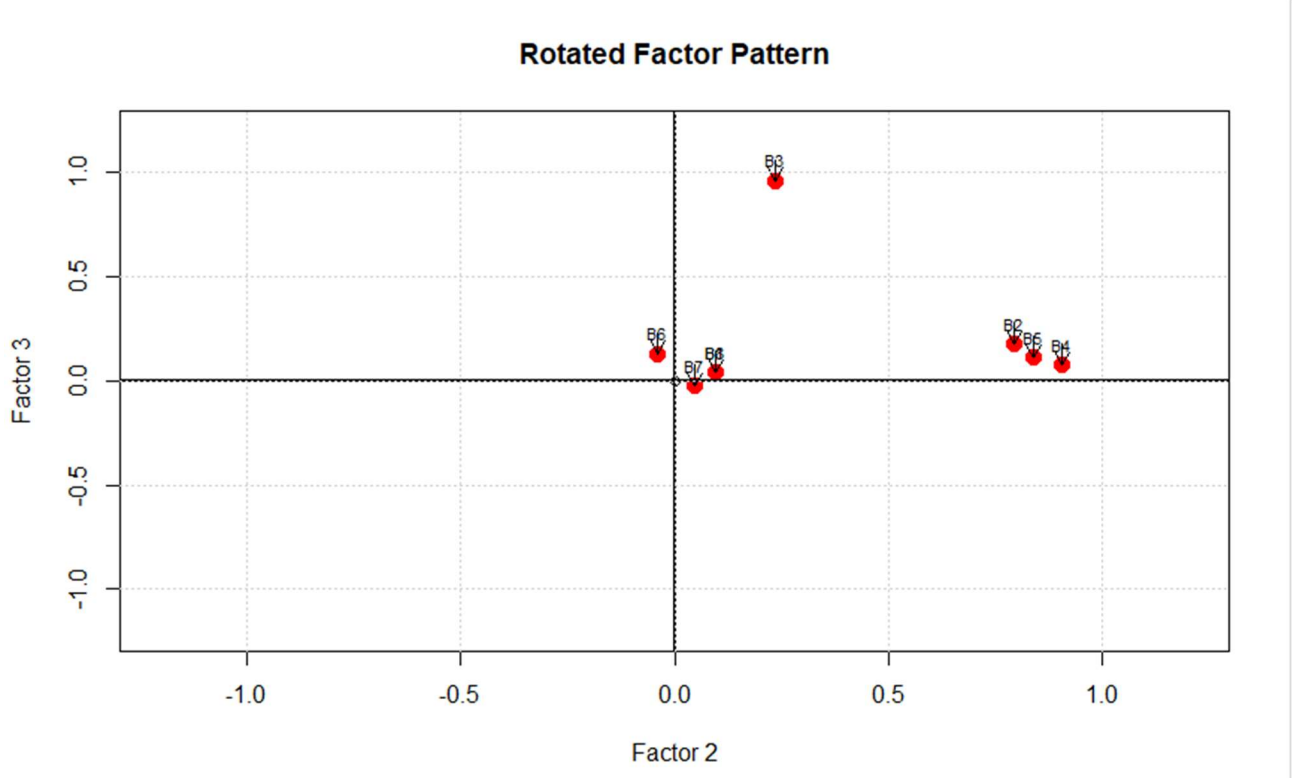
```
Names <- c("B1", "B2", "B3", "B4", "B5" ,"B6" ,"B7","B8")
for (i in 1:3) {
  l <- ij[i, 1]
  k <- ij[i, 2]


  # Plot for rotated Factors
  plot(0, 0, xlim = c(-1.2, 1.2), ylim = c(-1.2, 1.2), xlab = paste0("Factor ", l),
       ylab = paste0("Factor ", k), main = "Rotated Factor Pattern")
  points(rfa3l[, l], rfa3l[, k], col = 'red', pch = 19, cex = 1.5)
  text(rfa3l[, l], rfa3l[, k] + 0.1, Names, cex = 0.7)
  arrows(rfa3l[, l], rfa3l[, k], rfa3l[, l], rfa3l[, k] + 0.1, length = 0.1, code = 1, angle = 30)
  abline(h = 0, v = 0, col = 'black', lwd = 2)  # Add thick black zero lines
  grid()
}

################ Rotated Factors Score Plots ###############
#Factors
Factors3 <- data.frame("FA" = fa3l,"Rot FA" = rfa3l)
print("Non rotated and rotated factors:")
Factors3


municipalities = data_5$LA

#Scores plot for 3 Factors
F3 <- matrix(c(Factors3[,4],Factors3[,5],Factors3[,6]),ncol=3)

library(psych)

#Thurstone: regression based weights
ScoringF3 <- factor.scores(data_P5,F3,method = "Thurstone")
Scoring_PointsF3 <- ScoringF3$scores

scores2 = data.frame(Scoring_PointsF3)

# Correct the encoding of municipality names
municipalities <- iconv(municipalities, from = "UTF-8", to = "ASCII//TRANSLIT")

#install.packages("ggrepel")
library(ggrepel)

######## Plot Rotated Factor 1 vs Factor 2###############
ggplot(scores2,aes(x = scores2[,1],y = scores2[,2]))+
  geom_point(col = 'blue',size = 2)+
  geom_text_repel(aes(label = municipalities),box.padding = 0.5,cex=3,max.overlaps=100)+
  labs(x = 'Factor 1',y = 'Factor 2')+
  ggtitle('3 rotated factors')

######## Plot Rotated Factor 1 vs Factor 3###############
ggplot(scores2,aes(x = scores2[,1],y = scores2[,3]))+
  geom  point(col = 'blue',size = 2)+
  geom_text_repel(aes(label = municipalities),box.padding = 0.5,cex=3,max.overlaps=100)+
```

```
  labs(x = 'Factor 1',y = 'Factor 3')+
  ggtitle('3 rotated factors')

######## Plot Rotated Factor 2 vs Factor 3###############
ggplot(scores2,aes(x = scores2[,2],y = scores2[,3]))+
  geom_point(col = 'blue',size = 2)+
  geom_text_repel(aes(label = municipalities),box.padding = 0.5,cex=3,max.overlaps=100)+
  labs(x = 'Factor 2',y = 'Factor 3')+
  ggtitle('3 rotated factors')
```

Result in R:

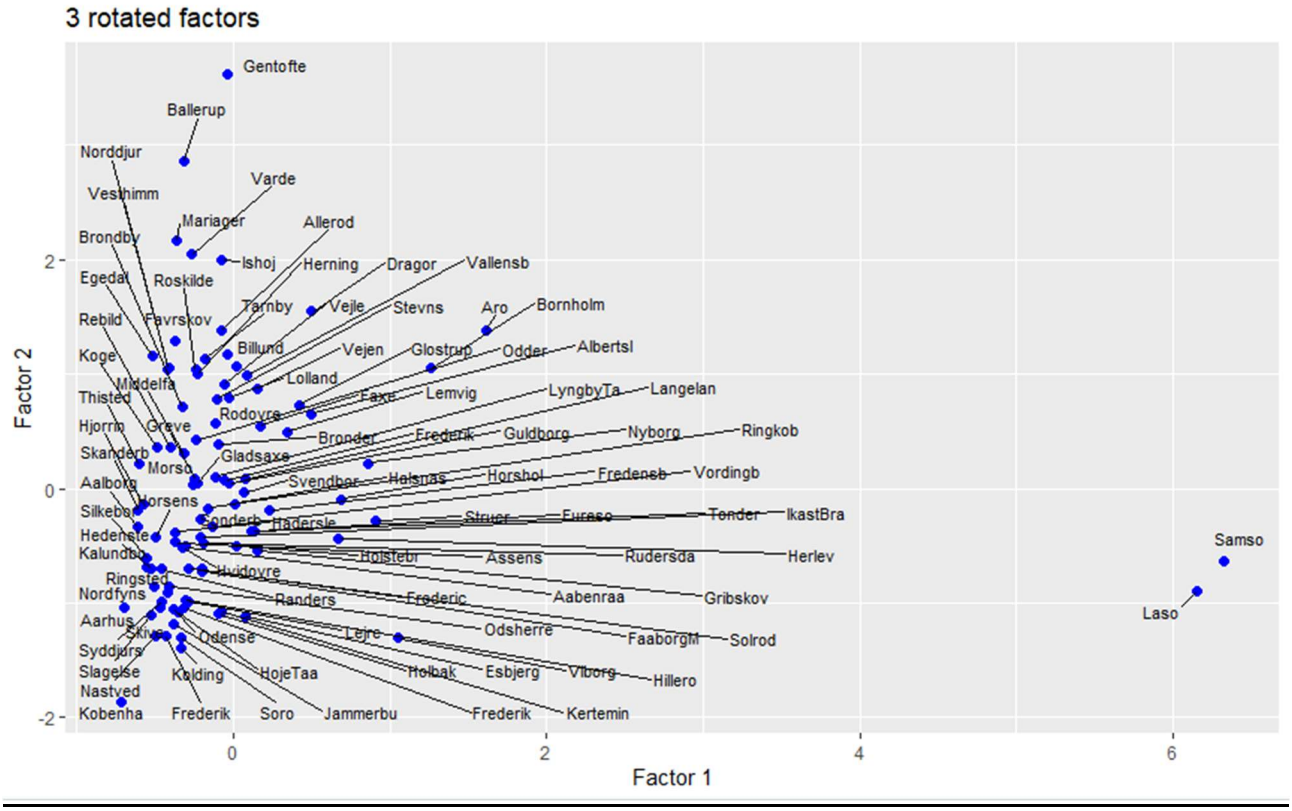


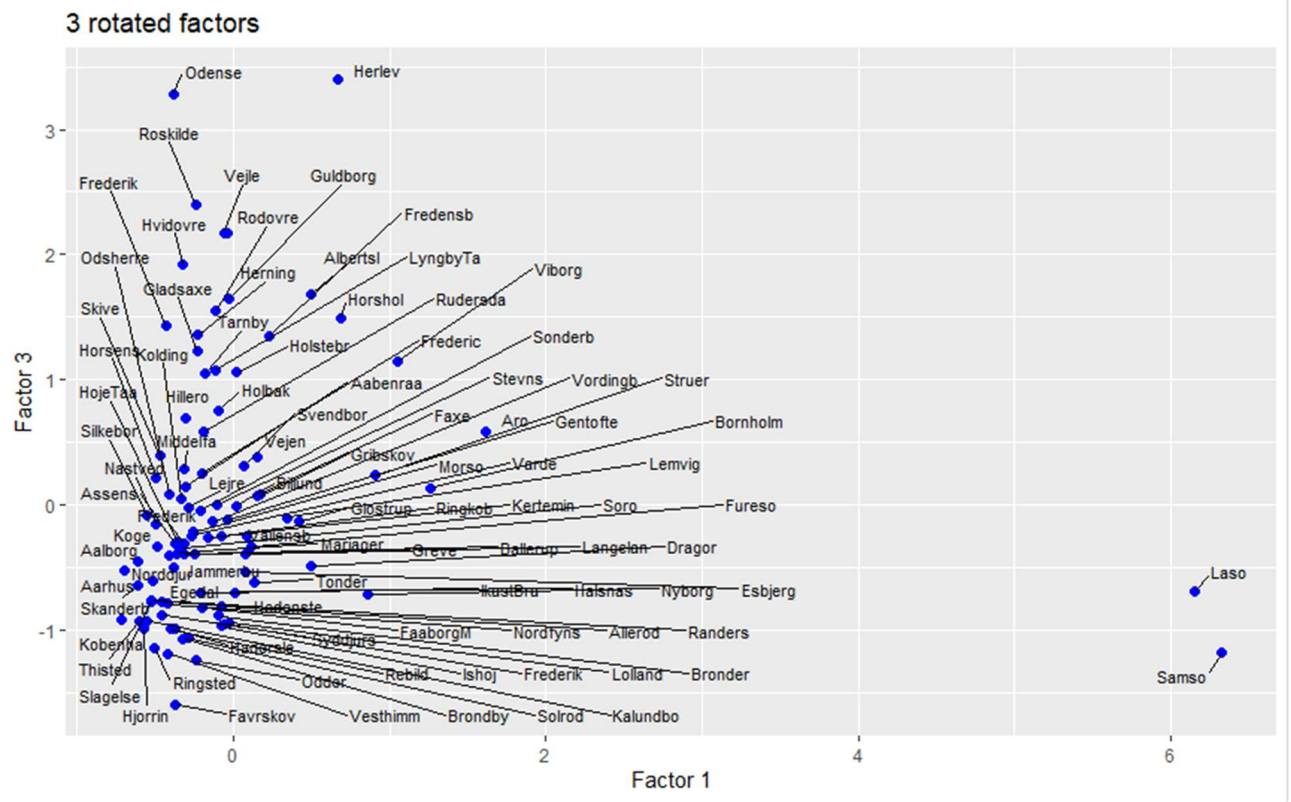


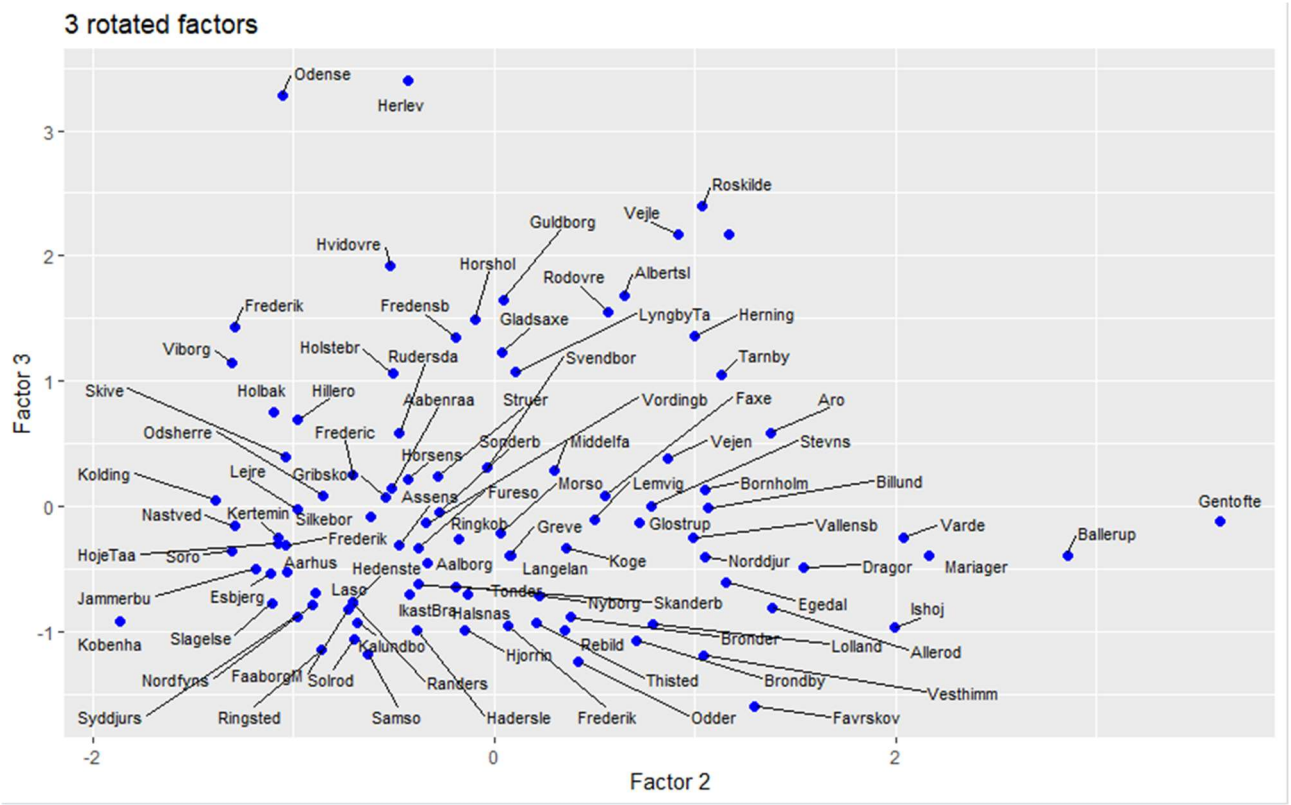


ROTATED FACTOR 1: B1, B6, B7 – Books, Other Material, Electronic Resources
ROTATED FACTOR 2: B2, B4, B5 – Audio Books, Live Images (Movies), Multi Media Material
ROTATED FACTOR 3: B3 - Music

Result in R:


3 rotated factors
Factor 2
Factor 1
Gentofte
Ballerup
Norddjur
Varde
Vesthimm
Mariager
Allerod
Brondby
Ishoj
Herning
Dragor
Vallensb
Egedal
Roskilde
Tarnby
Vejle
Stevns
Aro
Bornholm
Rebild
Favrskov
Billund
Vejen
Glostrup
Odder
Albertsl
Koge
Middelfa
Lolland
Thisted
Faxe
Lemvig
LyngbyTa
Langelan
Hjorrm
Greve
Rodovre
Bronder
Frederik
Guldborg
Nyborg
Ringkob
Skanderb
Gladsaxe
Morso
Aalborg
Svendbor
Halsnas
Horshol
Fredensb
Vordingb
Horsens
Silkebor
Sonderb
Hadersle
Struer
Fureso
Tonder
IkastBra
Hedensle
Kalundbo
Holstebr
Assens
Rudersda
Herlev
Ringsted
Hvidovre
Nordfyns
Randers
Frederic
Aabenraa
Gribskov
Aarhus
Skive
Odense
Lejre
Odsherre
FaaborgM
Solrod
Syddjurs
Slagelse
Kolding
HojeTaa
Holbak
Esbjerg
Viborg
Hillero
Nastved
Kobenha
Frederik
Soro
Jammerbu
Frederik
Kertemin
Samso
Laso


3 rotated factors
Factor 3
Factor 1
Odense
Herlev
Roskilde
Frederik
Vejle
Guldborg
Hvidovre
Rodovre
Fredensb
Odsherre
Albertsl
LyngbyTa
Viborg
Herning
Gladsaxe
Horshol
Ruderada
Skive
Tarnby
Holstebr
Frederic
Sonderb
Horsens
Kolding
Aabenraa
Stevns
Vordingb
Struer
HojeTaa
Hillero
Holbak
Svendbor
Faxe
Aro
Gentofte
Bornholm
Silkebor
Middelfa
Vejen
Gribskov
Morso
Varde
Lemvig
Nastved
Billund
Lejre
Assens
Glostrup
Ringkob
Kertemin
Soro
Fureso
Frederik
Koge
Aalborg
Mariager
Greve
Ballerup
Langelan
Dragor
Norddjur
Jammerbu
Tonder
Aarhus
Egedal
IkastBru
Halsnas
Nyborg
Esbjerg
Skanderb
Hedensle
Kobenha
Hadersle
Syddjurs
FaaborgM
Nordfyns
Allerod
Randers
Thisted
Rebild
Ishoj
Frederik
Lolland
Bronder
Slagelse
Ringsted
Odder
Hjorrin
Favrskov
Vesthimm
Brondby
Solrod
Kalundbo
Laso
Samso



Læsø and Samsø has a larger collection of books, Other Material and Electronic Resources than average. Københa(vn) has a smaller and Gentofte a larger Audio Books, Live Images (Movies), Multi Media Material collection. Odense and Herlev have a larger Music collection than average, while Favrskov has a smaller

ANSWER 2

Problem 6.

We consider a random variable

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

with mean value and dispersion matrix respectively equal to

$$\mu = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \text{ and } \boldsymbol{\Sigma} = \begin{bmatrix} 1 & \rho & \rho^2 \\ \rho & 1 & \rho \\ \rho^2 & \rho & 1 \end{bmatrix}$$

Furthermore we consider the random variables

$$S = X - Y$$
$$T = Y - Z$$

Question 6.1.

The mean value of the two-dimensional random variable $\begin{bmatrix} S \\ T \end{bmatrix}$ *is*

We use

Remark 1.10 Rules for computing moments of simple functions

$$\mathrm{E}(a+bX) = a + b\,\mathrm{E}(X)$$
$$\mathrm{E}(X+Y) = \mathrm{E}(X) + \mathrm{E}(Y)$$

$$\mathrm{V}(a+bX) = b^2\,\mathrm{V}(X)$$
$$\mathrm{V}(X+Y) = \mathrm{V}(X) + \mathrm{V}(Y) + 2\mathrm{Cov}(X,Y)$$
$$= \mathrm{V}(X) + \mathrm{V}(Y) \quad \text{iff } X, Y \text{ independent}$$

$$\mathrm{Cov}(X,X) = \mathrm{V}(X)$$
$$\mathrm{Cov}(aX,bY) = ab\mathrm{Cov}(X,Y)$$
$$\mathrm{Cov}(X+U,Y) = \mathrm{Cov}(X,Y) + \mathrm{Cov}(U,Y)$$
$$\mathrm{Cov}(X,Y+V) = \mathrm{Cov}(X,Y) + \mathrm{Cov}(X,V)$$

$$\mathrm{E}(\mathbf{A}+\mathbf{X}) = \mathbf{A} + \mathrm{E}(\mathbf{X})$$
$$\mathrm{E}(\mathbf{AX}) = \mathbf{A}\,\mathrm{E}(\mathbf{X})$$
$$\mathrm{E}(\mathbf{XB}) = \mathrm{E}(\mathbf{X})\mathbf{B}$$
$$\mathrm{E}(\mathbf{X}+\mathbf{Y}) = \mathrm{E}(\mathbf{X}) + \mathrm{E}(\mathbf{Y})$$
$$\mathrm{D}(\mathbf{b}+\mathbf{X}) = \mathrm{D}(\mathbf{X})$$
$$\mathrm{D}(\mathbf{AX}) = \mathbf{A}\,\mathrm{D}(\mathbf{X})\mathbf{A}^T$$
$$\mathrm{D}(\mathbf{X}+\mathbf{Y}) = \mathrm{D}(\mathbf{X}) + \mathrm{D}(\mathbf{Y}) + \mathrm{C}(\mathbf{X},\mathbf{Y}) + \mathrm{C}(\mathbf{Y},\mathbf{X})$$
$$= \mathrm{D}(\mathbf{X}) + \mathrm{D}(\mathbf{Y}) \quad \text{iff } X, Y \text{ independent}$$
$$\mathrm{C}(\mathbf{X},\mathbf{X}) = \mathrm{D}(\mathbf{X})$$
$$\mathrm{C}(\mathbf{X},\mathbf{Y}) = \mathrm{C}(\mathbf{Y},\mathbf{X})^T$$
$$\mathrm{C}(\mathbf{AX},\mathbf{BY}) = \mathbf{A}\,\mathrm{C}(\mathbf{X},\mathbf{Y})\mathbf{B}^T$$
$$\mathrm{C}(\mathbf{X}+\mathbf{U},\mathbf{Y}) = \mathrm{C}(\mathbf{X},\mathbf{Y}) + \mathrm{C}(\mathbf{U},\mathbf{Y})$$
$$\mathrm{C}(\mathbf{X},\mathbf{Y}+\mathbf{V}) = \mathrm{C}(\mathbf{X},\mathbf{Y}) + \mathrm{C}(\mathbf{X},\mathbf{V})$$

$$\begin{bmatrix} S \\ T \end{bmatrix} = \begin{bmatrix} X - Y \\ Y - Z \end{bmatrix} = \begin{bmatrix} 1 - 2 \\ 2 - 3 \end{bmatrix} = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$$

ANSWER 2 : $\begin{bmatrix} \mathbf{-1} \\ \mathbf{-1} \end{bmatrix}$

Question 6.2.

The dispersion matrix for the two-dimensional random variable $\begin{bmatrix} S \\ T \end{bmatrix}$ *is*

We again use Remark 1.10

$$V(S) = \begin{bmatrix} 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} 1 & \rho & \rho^2 \\ \rho & 1 & \rho \\ \rho^2 & \rho & 1 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix} = (1 - \rho) - (\rho - 1) = 2 - 2\rho$$

$$V(T) = \begin{bmatrix} 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & \rho & \rho^2 \\ \rho & 1 & \rho \\ \rho^2 & \rho & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ -1 \end{bmatrix} = (1 - \rho) - (\rho - 1) = 2 - 2\rho$$

$$C(S,T) = \begin{bmatrix} 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} 1 & \rho & \rho^2 \\ \rho & 1 & \rho \\ \rho^2 & \rho & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ -1 \end{bmatrix} = (\rho - \rho^2) - (1 - \rho) = 2\rho - \rho^2 - 1$$

Collecting

$$D\left(\begin{bmatrix} S \\ T \end{bmatrix}\right) = \begin{bmatrix} 2 - 2\rho & 2\rho - \rho^2 - 1 \\ 2\rho - \rho^2 - 1 & 2 - 2\rho \end{bmatrix} = \begin{bmatrix} 2(1-\rho) & -(1-\rho)^2 \\ -(1-\rho)^2 & 2(1-\rho) \end{bmatrix} = (1-\rho)\begin{bmatrix} 2 & \rho - 1 \\ \rho - 1 & 2 \end{bmatrix}$$

ANSWER 2 : $(1-\rho)\begin{bmatrix} 2 & \rho - 1 \\ \rho - 1 & 2 \end{bmatrix}$

Question 6.3.

The covariance between X and S is:

$$C(X,S) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & \rho & \rho^2 \\ \rho & 1 & \rho \\ \rho^2 & \rho & 1 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix} = 1 - \rho$$

ANSWER 5: $\mathbf{1 - \rho}$

Question 6.4.

The conditional mean $\mathrm{E}(X|Y)$ is

We use

Theorem 1.27

If X_2 is regularly distributed, i.e. if Σ_{22} has full rank, then the distribution of X_1 conditioned on $X_2 = x_2$ is again a normal distribution, and the following holds

$$\begin{aligned} \mathrm{E}(X_1|X_2 = x_2) &= \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(x_2 - \mu_2) \\ \mathrm{D}(X_1|X_2 = x_2) &= \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}. \end{aligned}$$

If Σ_{22} does not have full rank then the conditional distribution is still normal and Σ_{22}^{-1} in the above equations should be substituted by a generalised inverse Σ_{22}^{-}.

$$\mathrm{E}(X|Y) = 1 + \rho \cdot 1 \cdot (y - 2)$$

ANSWER 1: $\boldsymbol{\rho(Y - 2) + 1}$

Question 6.5.

The conditional dispersion matrix $\mathrm{D}\left(\begin{bmatrix} X \\ Z \end{bmatrix} \middle| Y\right)$ *is*

We again use Theorem 1.27

$$\mathrm{D}\left(\begin{bmatrix} X \\ Z \end{bmatrix} \middle| Y\right) = \begin{bmatrix} 1 & \rho^2 \\ \rho^2 & 1 \end{bmatrix} - \begin{bmatrix} \rho \\ \rho \end{bmatrix} \cdot 1 \cdot [\rho \quad \rho] = \begin{bmatrix} 1 & \rho^2 \\ \rho^2 & 1 \end{bmatrix} - \begin{bmatrix} \rho^2 & \rho^2 \\ \rho^2 & \rho^2 \end{bmatrix} = \begin{bmatrix} 1-\rho^2 & 0 \\ 0 & 1-\rho^2 \end{bmatrix}$$

ANSWER 5 : $\begin{bmatrix} \boldsymbol{1-\rho^2} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{1-\rho^2} \end{bmatrix}$

LAST PAGE:
END OF THE EXAM SET