STAT 5703 Homework 2 Exercise 2

Shijie He(sh3975), Yunjun Xia(yx2569), Shuyu Huang(sh3967)

3/1/2020

Part 1

```r
scores = read.delim("scores.txt", sep = " ")
```

(a)

```r
(cov_a <- cov(scores, use="complete.obs"))
```

```
##        x1     x2     x3     x4     x5
## x1 216.30  -7.50  45.05  77.65  94.50
## x2  -7.50 221.50 117.50  77.00 226.75
## x3  45.05 117.50 157.30  85.90 242.00
## x4  77.65  77.00  85.90  75.20 132.25
## x5  94.50 226.75 242.00 132.25 422.00
```

(b)

```r
(cov_b <- cov(scores, use="pairwise.complete.obs"))
```

```
##            x1         x2        x3        x4       x5
## x1 121.363636   4.563636  35.79091  42.12727  94.5000
## x2   4.563636 179.134199 112.26840 114.60173 172.5000
## x3  35.790909 112.268398 151.48918 125.96537 182.3727
## x4  42.127273 114.601732 125.96537 153.56061 142.8636
## x5  94.500000 172.500000 182.37273 142.86364 294.5636
```

(c)

```r
scores_imp <- scores
for (i in 1:dim(scores)[2]){
  ind <- which(is.na(scores[,i]))
  scores_imp[ind, i] <- mean(na.omit(scores[,i]))
}
(cov_c <- cov(scores_imp))
```

```
##          x1        x2        x3        x4        x5
## x1 57.79221   2.17316  17.04329  20.06061  21.50138
## x2  2.17316 179.13420 112.26840 114.60173  82.14286
## x3 17.04329 112.26840 151.48918 125.96537  86.84416
## x4 20.06061 114.60173 125.96537 153.56061  68.03030
## x5 21.50138  82.14286  86.84416  68.03030 140.26840
```

(d)

```r
c <- matrix(0, nrow = dim(scores)[2], ncol = dim(scores)[2])
for (i in 1:1000){
  n <- dim(scores)[1]
  new_ind <- sample(1:n, size = n, replace = TRUE)
  scores_boot <- scores[new_ind,]
  for (j in 1:dim(scores)[2]){
    ind <- which(is.na(scores_boot[,j]))
    scores_boot[ind, j] <- mean(na.omit(scores_boot[,j]))
  }
  c <- c + cov(scores_boot)
}
(cov_d <- c/1000)
```

```
##           x1         x2        x3        x4        x5
## x1 53.003527   1.648386  15.40612  17.67104  19.44182
## x2  1.648386 172.153955 108.10460 110.14272  75.60857
## x3 15.406122 108.104604 145.89841 121.40361  80.07345
## x4 17.671038 110.142716 121.40361 147.64234  62.81708
## x5 19.441815  75.608567  80.07345  62.81708 129.85824
```

(e)

```r
scores_em <- EMimpute(scores, max.score = 1000)
```

```r
(cov_e <- cov(scores_em))
```

```
##           x1        x2        x3       x4       x5
## x1 233.56061  18.12554  87.96537 100.4177 151.1082
## x2  18.12554 179.13420 112.26840 114.6017 135.8874
## x3  87.96537 112.26840 151.48918 125.9654 182.5368
## x4 100.41775 114.60173 125.96537 153.5606 107.2035
## x5 151.10823 135.88745 182.53680 107.2035 321.2987
```

Using imputation with and without bootstrap gives smaller covariance value compared to the rest.

Part 2

The asymptotic distribution for $\hat{\lambda}_1$ is:

$$\sqrt{n}(\log \hat{\lambda}_1 - \log \lambda_1) \xrightarrow[n\to\infty]{\mathcal{D}} \mathcal{N}(0, 2)$$

Use delta method, we get:

$$\sqrt{n}(\hat{\lambda}_1 - \lambda_1) \xrightarrow[n\to\infty]{\mathcal{D}} \mathcal{N}(0, 2\lambda_1^2)$$

Thus,

$$\frac{\sqrt{n}(\hat{\lambda}_1 - \lambda_1)}{\sqrt{2}\lambda_1} \xrightarrow[n\to\infty]{\mathcal{D}} \mathcal{N}(0, 1)$$

The confidence interval for λ_1 is:

$$\mathbb{P}[-z_{1-\alpha/2} < \frac{\sqrt{n}(\hat{\lambda}_1 - \lambda_1)}{\sqrt{2}\lambda_1} < z_{1-\alpha/2}] = 1-\alpha$$
$$\mathbb{P}[-z_{1-\alpha/2}\sqrt{\frac{2}{n}}\lambda_1 < \hat{\lambda}_1 - \lambda_1 < z_{1-\alpha/2}\sqrt{\frac{2}{n}}\lambda_1] = 1-\alpha$$
$$\mathbb{P}[(-z_{1-\alpha/2}\sqrt{\frac{2}{n}}+1)\lambda_1 < \hat{\lambda}_1 < (z_{1-\alpha/2}\sqrt{\frac{2}{n}}+1)\lambda_1] = 1-\alpha$$
$$\mathbb{P}[\frac{\hat{\lambda}_1}{z_{1-\alpha/2}\sqrt{\frac{2}{n}}+1} < \lambda_1 < \frac{\hat{\lambda}_1}{-z_{1-\alpha/2}\sqrt{\frac{2}{n}}+1}] = 1-\alpha$$

Thus,

$$\lambda_1 \in (\frac{\hat{\lambda}_1}{z_{1-\alpha/2}\sqrt{\frac{2}{n}}+1}, \frac{\hat{\lambda}_1}{-z_{1-\alpha/2}\sqrt{\frac{2}{n}}+1})$$

(a)

Here, we set the significance level for confidence interval to be 0.05.

```r
z <- qnorm(0.975)
n <- dim(scores)[1]
lambda_a <- max(eigen(cov_a)$values)
lambda_a_low <- lambda_a/(z*sqrt(2/n)+1)
lambda_a_high <- lambda_a/(-z*sqrt(2/n)+1)
cat(paste0("The confidence interval for eigenvalue of complete case analysis is: (",
           round(lambda_a_low, 3), ", ", round(lambda_a_high, 3), ")."))
```

```
## The confidence interval for eigenvalue of complete case analysis is: (482.301, 1875.86).
```

(b)

```r
lambda_b <- max(eigen(cov_b)$values)
lambda_b_low <- lambda_b/(z*sqrt(2/n)+1)
lambda_b_high <- lambda_b/(-z*sqrt(2/n)+1)
cat(paste0("The confidence interval for eigenvalue of available case analysis is: (",
           round(lambda_b_low, 3), ", ", round(lambda_b_high, 3), ")."))
```

```
## The confidence interval for eigenvalue of available case analysis is: (412.135, 1602.954).
```

(c)

```r
lambda_c <- max(eigen(cov_c)$values)
lambda_c_low <- lambda_c/(z*sqrt(2/n)+1)
lambda_c_high <- lambda_c/(-z*sqrt(2/n)+1)
cat(paste0("The confidence interval for eigenvalue of mean imputation is: (",
           round(lambda_c_low, 3), ", ", round(lambda_c_high, 3), ")."))
```

```
## The confidence interval for eigenvalue of mean imputation is: (288.024, 1120.239).
```

(d)

```r
lambda_d <- max(eigen(cov_d)$values)
lambda_d_low <- lambda_d/(z*sqrt(2/n)+1)
lambda_d_high <- lambda_d/(-z*sqrt(2/n)+1)
cat(paste0("The confidence interval for eigenvalue of mean imputation with bootstrap is: (",
           round(lambda_d_low, 3), ", ", round(lambda_d_high, 3), ")."))
```

```
## The confidence interval for eigenvalue of mean imputation with bootstrap is: (273.974, 1065.593).
```

(e)

```r
lambda_e <- max(eigen(cov_e)$values)
lambda_e_low <- lambda_e/(z*sqrt(2/n)+1)
lambda_e_high <- lambda_e/(-z*sqrt(2/n)+1)
cat(paste0("The confidence interval for eigenvalue of EM-algorithm is: (",
           round(lambda_e_low, 3), ", ", round(lambda_e_high, 3), ")."))
```

```
## The confidence interval for eigenvalue of EM-algorithm is: (432.987, 1684.059).
```

Confidence interval are wider and larger for complete case, available case and EM-algorithm. This might be because the covariance matrix and different for these cases.

Part 3

```r
library(SMPracticals)
cov_comp <- cov(mathmarks)

lambda_comp <- max(eigen(cov_comp)$values)
lambda_comp_low <- lambda_comp/(z*sqrt(2/n)+1)
lambda_comp_high <- lambda_comp/(-z*sqrt(2/n)+1)
cat(paste0("The confidence interval for eigenvalue of the complete data is: (",
           round(lambda_comp_low, 3), ", ", round(lambda_comp_high, 3), ")."))
```

```
## The confidence interval for eigenvalue of the complete data is: (431.811, 1679.482).
```

EM-algorithm gives the most accurate estimation of the eigenvalue. Imputation gives relatively worst estimation. The reason might be because the data has few rows so that the mean imputation is not quite general in our case.

Part 4

The log-likelihood function for this model is:

$$\ell_i(\boldsymbol{\mu}, \boldsymbol{\Sigma} \mid \mathbf{x_i}) = -\frac{d}{2}\log(2\pi) - \frac{1}{2}\log|\boldsymbol{\Sigma}| - \frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}_i - \boldsymbol{\mu})$$

We need to first get the expected value of this log-likelihood function and then figure out the optimal value for the parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ that makes the expected value maximum. We use MLE to get the maximum value.

We first take a look at the partial derivative respect to $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$:

$$\frac{\partial}{\partial \boldsymbol{\mu}}\ell_i(\boldsymbol{\mu}, \boldsymbol{\Sigma} \mid \mathbf{x}_i) = -\boldsymbol{\Sigma}^{-1}(\mathbf{x}_i - \boldsymbol{\mu})$$

For the derivative of $\boldsymbol{\Sigma}$, we will take the derivative with respect to $\boldsymbol{\Sigma}^{-1}$ instead.

$$\frac{\partial}{\partial \mathbf{\Sigma}^{-1}} \ell_i(\boldsymbol{\mu}, \mathbf{\Sigma} \mid \mathbf{x}_i) = \frac{1}{2}\mathbf{\Sigma} - \frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^T$$

Thus, we need the conditional expectation $\mathbb{E}[\mathbf{X_i} \mid \mathbf{X}_{im}]$ and $\mathbb{E}[(\mathbf{X_i} - \boldsymbol{\mu})^T(\mathbf{X_i} - \boldsymbol{\mu}) \mid \mathbf{X}_{im}]$ for E-step.

E-step:

$$\mathbb{E}[\mathbf{X_i} \mid \mathbf{X}_{im}, \boldsymbol{\mu}^{(k)}, \mathbf{\Sigma}^{(k)}] = ((\hat{\mathbf{X}}_{im})^T, (\hat{\mathbf{X}}_{io}^{(k)})^T)^T$$

, where $\hat{\mathbf{X}}_{io} = \mathbf{X}_{io}$ and

$$\hat{\mathbf{X}}_{im}^{(k)} = E[\mathbf{X}_{im} \mid \mathbf{X}_{io}, \boldsymbol{\mu}^k, \mathbf{\Sigma}^k] = \boldsymbol{\mu}_{im}^{(k)} + \mathbf{\Sigma}_{imo}^{(k)}(\mathbf{\Sigma}_{ioo}^{(k)})^{-1}(\mathbf{X}_{io} - \boldsymbol{\mu}_{io}^k)$$

$$\begin{aligned}
\mathbb{E}[(\mathbf{X_i} - \boldsymbol{\mu})(\mathbf{X_i} - \boldsymbol{\mu})^T \mid \mathbf{X}_{im}, \boldsymbol{\mu}^{(k)}, \mathbf{\Sigma}^{(k)}] &= \mathbb{E}\begin{bmatrix} (\mathbf{X}_{io} - \boldsymbol{\mu}_{io}^{(k)})(\mathbf{X}_{io} - \boldsymbol{\mu}_{io}^{(k)})^T & (\mathbf{X}_{io} - \boldsymbol{\mu}_{io}^{(k)})(\mathbf{X}_{im} - \boldsymbol{\mu}_{im}^{(k)})^T \\ (\mathbf{X}_{im} - \boldsymbol{\mu}_{im}^{(k)})(\mathbf{X}_{io} - \boldsymbol{\mu}_{io}^{(k)})^T & (\mathbf{X}_{im} - \boldsymbol{\mu}_{im}^{(k)})(\mathbf{X}_{im} - \boldsymbol{\mu}_{im}^{(k)})^T \end{bmatrix} \\
&= \begin{bmatrix} (\hat{\mathbf{X}}_{io} - \boldsymbol{\mu}_{io}^{(k)})(\hat{\mathbf{X}}_{io} - \boldsymbol{\mu}_{io}^{(k)})^T & (\hat{\mathbf{X}}_{io} - \boldsymbol{\mu}_{io}^{(k)})(\hat{\mathbf{X}}_{im} - \boldsymbol{\mu}_{im}^{(k)})^T \\ (\hat{\mathbf{X}}_{im} - \boldsymbol{\mu}_{im}^{(k)})(\hat{\mathbf{X}}_{io} - \boldsymbol{\mu}_{io}^{(k)})^T & (\hat{\mathbf{X}}_{im} - \boldsymbol{\mu}_{im}^{(k)})(\hat{\mathbf{X}}_{im} - \boldsymbol{\mu}_{im}^{(k)})^T + \mathbf{C}_{imm} \end{bmatrix}
\end{aligned}$$

where $\mathbf{C}_{imm}^{(k)} = \mathbf{\Sigma}_{imm}^{(k)} - \mathbf{\Sigma}_{imo}^{(k)}(\mathbf{\Sigma}_{ioo}^{(k)})^{-1}\mathbf{\Sigma}_{iom}^{(k)}$

M-step:

For $\boldsymbol{\mu}^{(k+1)}$:

$$\frac{\partial}{\partial \boldsymbol{\mu}} \ell_i(\boldsymbol{\mu}, \mathbf{\Sigma} \mid \mathbf{X}_i) = -\mathbf{\Sigma}^{-1}(\mathbf{X}_i - \boldsymbol{\mu})$$

Thus,

$$\frac{\partial}{\partial \boldsymbol{\mu}} \ell(\boldsymbol{\mu}, \mathbf{\Sigma} \mid \mathbf{X}) = \sum_{i=1}^{n} -\mathbf{\Sigma}^{-1}(\mathbf{X}_i - \boldsymbol{\mu}) = 0$$

This gives that $\boldsymbol{\mu}^{(k+1)}$: $\sum_{i=1}^{n}(\hat{\mathbf{X}}_i - \boldsymbol{\mu}) = 0$.

For $\mathbf{\Sigma}^{(k+1)}$:

$$\frac{\partial}{\partial \mathbf{\Sigma}^{-1}} \ell_i(\boldsymbol{\mu}, \mathbf{\Sigma} \mid \mathbf{X}_i) = \frac{1}{2}\mathbf{\Sigma} - \frac{1}{2}(\mathbf{X}_i - \boldsymbol{\mu})(\mathbf{X}_i - \boldsymbol{\mu})^T$$

Thus,

$$\frac{\partial}{\partial \mathbf{\Sigma}^{-1}} \ell(\boldsymbol{\mu}, \mathbf{\Sigma} \mid \mathbf{X}) = \sum_{i=1}^{n} \frac{1}{2}\mathbf{\Sigma} - \frac{1}{2}(\mathbf{X}_i - \boldsymbol{\mu})(\mathbf{X}_i - \boldsymbol{\mu})^T = 0$$

This gives $\mathbf{\Sigma}^{(k+1)}$: $\sum_{i=1}^{n}(\mathbf{\Sigma} - (\hat{\mathbf{X}}_i - \boldsymbol{\mu})(\hat{\mathbf{X}}_i - \boldsymbol{\mu})^T - \mathbf{C}_i) = 0$, where $\mathbf{C}_{imm} = \mathbf{\Sigma}_{imm}^{(k)} - \mathbf{\Sigma}_{imo}^{(k)}(\mathbf{\Sigma}_{ioo}^{(k)})^{-1}\mathbf{\Sigma}_{iom}^{(k)}$ and all other entries to be 0.