Compulsory HWA4, Multivariate Analysis, 2023HT

Shaobo Jin

1. (1pt) Suppose that X follows an exponential distribution with mean μ_1 in π_1, and an exponential distribution with mean μ_2 in π_2. It is known that $p_1 = 0.6$ and $p_2 = 0.4$. Let $c(2 \mid 1) = 5$ and $c(1 \mid 2) = 10$. Find the classifier that minimizes the ECM. How should you classifier the subject $x_0 = 2$ if $\mu_1 = 1$ and $\mu_2 = 3$?
   Solution: The classifier that minimizes ECM is

$$\begin{aligned} R_1 &= \left\{ \boldsymbol{x}; \quad \frac{f_1(x)}{f_2(x)} \geq \frac{c(1 \mid 2)}{c(2 \mid 1)} \frac{p_2}{p_1} \right\}, \\ R_2 &= \left\{ \boldsymbol{x}; \quad \frac{f_1(x)}{f_2(x)} < \frac{c(1 \mid 2)}{c(2 \mid 1)} \frac{p_2}{p_1} \right\}, \end{aligned}$$

   where

$$\begin{aligned} \frac{f_1(x)}{f_2(x)} &= \frac{\mu_1^{-1} \exp\{-x/\mu_1\}}{\mu_2^{-1} \exp\{-x/\mu_2\}} = \frac{\mu_2}{\mu_1} \exp\left\{ -\left(\frac{1}{\mu_1} - \frac{1}{\mu_2} \right) x \right\}, \\ \frac{c(1 \mid 2)}{c(2 \mid 1)} \frac{p_2}{p_1} &= \frac{10}{5} \frac{0.4}{0.6} = \frac{4}{3}. \end{aligned}$$

   If a new subject with $x_0 = 2$ satisfies

$$\frac{\mu_2}{\mu_1} \exp\left\{ -\left(\frac{1}{\mu_1} - \frac{1}{\mu_2} \right) x_0 \right\} \geq \frac{4}{3},$$

   then it will be classified to π_1. Otherwise, it is classified to π_2. Note that

$$\frac{\mu_2}{\mu_1} \exp\left\{ -\left(\frac{1}{\mu_1} - \frac{1}{\mu_2} \right) x_0 \right\} = 3 \exp\left\{ -\frac{4}{3} \right\} \approx 0.79.$$

   Hence, we will classify it to π_2.

2. (2pt) During the lecture, we have derived the Gaussian discriminant classifier for two classes when $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2$ but unknown. Derive the Gaussian discriminant classifier when $\boldsymbol{\Sigma}_1 \neq \boldsymbol{\Sigma}_2$ and unknown. Is the decision boundary quadratic? Hint: follow our derivation of the Gaussian discriminant classifier for two classes when $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2$.
   Solution: Let $Z \sim \text{Bernoulli}(\phi)$ be a random variable that indicates the class, where $Z = 1$ or 0. Let

$$\boldsymbol{X} \mid Z = 1 \sim N(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1), \quad \boldsymbol{X} \mid Z = 0 \sim N(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_2).$$

The log-likelihood function is

$$\begin{aligned}
\ell(\boldsymbol{\mu}_1, \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2, \phi) &= \sum_{j=1}^{n} \left\{ z_j \log\left[f_1\left(\boldsymbol{x}_j\right) \phi \right] + \left(1 - z_j\right) \log\left[f_0\left(\boldsymbol{x}_j\right) (1-\phi) \right] \right\} \\
&= \sum_{j=1}^{n} \left\{ z_j \left[\log\phi - \frac{p}{2}\log(2\pi) - \frac{1}{2}\log\det\left(\boldsymbol{\Sigma}_1\right) - \frac{1}{2}\left(\boldsymbol{x}_j - \boldsymbol{\mu}_1\right)^T \boldsymbol{\Sigma}_1^{-1}\left(\boldsymbol{x}_j - \boldsymbol{\mu}_1\right) \right] \right\} \\
&\quad + \sum_{j=1}^{n} \left\{ \left(1 - z_j\right) \left[\log(1-\phi) - \frac{p}{2}\log(2\pi) - \frac{1}{2}\log\det\left(\boldsymbol{\Sigma}_2\right) - \frac{1}{2}\left(\boldsymbol{x}_j - \boldsymbol{\mu}_2\right)^T \boldsymbol{\Sigma}_2^{-1}\left(\boldsymbol{x}_j \right. \right. \right. \\
&= \sum_{j=1}^{n} \left[z_j \log\phi + \left(1 - z_j\right) \log(1-\phi) \right] \\
&\quad + \sum_{j=1}^{n} \left\{ z_j \left[-\frac{p}{2}\log(2\pi) - \frac{1}{2}\log\det\left(\boldsymbol{\Sigma}_1\right) - \frac{1}{2}\left(\boldsymbol{x}_j - \boldsymbol{\mu}_1\right)^T \boldsymbol{\Sigma}_1^{-1}\left(\boldsymbol{x}_j - \boldsymbol{\mu}_1\right) \right] \right\} \\
&\quad + \sum_{j=1}^{n} \left\{ \left(1 - z_j\right) \left[-\frac{p}{2}\log(2\pi) - \frac{1}{2}\log\det\left(\boldsymbol{\Sigma}_2\right) - \frac{1}{2}\left(\boldsymbol{x}_j - \boldsymbol{\mu}_2\right)^T \boldsymbol{\Sigma}_2^{-1}\left(\boldsymbol{x}_j - \boldsymbol{\mu}_2\right) \right] \right\}
\end{aligned}$$

Since

$$\frac{\partial \ell}{\partial \phi} = \frac{\sum_{j=1}^{n} z_j}{\phi} - \frac{n - \sum_{j=1}^{n} z_j}{1-\phi},$$

the MLE of ϕ is $\hat{\phi} = n^{-1}\sum_{j=1}^{n} z_j$. $\boldsymbol{\mu}_1$ is only involved in

$$\sum_{j=1}^{n} \left\{ z_j \left[-\frac{1}{2}\left(\boldsymbol{x}_j - \boldsymbol{\mu}_1\right)^T \boldsymbol{\Sigma}_1^{-1}\left(\boldsymbol{x}_j - \boldsymbol{\mu}_1\right) \right] \right\} = -\frac{1}{2} \sum_{j:z_j=1} \left(\boldsymbol{x}_j - \boldsymbol{\mu}_1\right)^T \boldsymbol{\Sigma}_1^{-1}\left(\boldsymbol{x}_j - \boldsymbol{\mu}_1\right). \quad (1)$$

By the lemma in Chapter 4, $\boldsymbol{\mu}_1$ that maximizes (1) is

$$\hat{\boldsymbol{\mu}}_1 = \frac{\sum_{j:z_j=1} \boldsymbol{x}_j}{\sum_{j=1}^{n} I\left(z_j = 1\right)} = \frac{\sum_{j=1}^{n} z_j \boldsymbol{x}_j}{\sum_{j=1}^{n} z_j}.$$

Likewise

$$\hat{\boldsymbol{\mu}}_2 = \frac{\sum_{j=1}^{n} \left(1 - z_j\right) \boldsymbol{x}_j}{\sum_{j=1}^{n} \left(1 - z_j\right)}.$$

To find the MLE of $\boldsymbol{\Sigma}_1$, we consider

$$\begin{aligned}
\sum_{j=1}^{n} \left\{ z_j \left[-\frac{1}{2}\log\det\left(\boldsymbol{\Sigma}_1\right) - \frac{1}{2}\left(\boldsymbol{x}_j - \boldsymbol{\mu}_1\right)^T \boldsymbol{\Sigma}_1^{-1}\left(\boldsymbol{x}_j - \boldsymbol{\mu}_1\right) \right] \right\} &= \frac{1}{2} \sum_{j:z_j=1} \left[-\log\det\left(\boldsymbol{\Sigma}_1\right) - \left(\boldsymbol{x}_j - \boldsymbol{\mu}_1\right)^T \boldsymbol{\Sigma}_1^{-1} \left(\right. \right. \\
&= \frac{1}{2} \left[-\left(\sum_{j=1}^{n} z_j \right) \log\det\left(\boldsymbol{\Sigma}_1\right) - \mathrm{tr} \left\{ \boldsymbol{\Sigma}_1^{-1} \sum_{j:z_j=1} \right.
\end{aligned}$$

By the lemma in Chapter 4, the maximizer is

$$\hat{\boldsymbol{\Sigma}}_1 = \frac{\sum_{j:z_j=1}\left(\boldsymbol{x}_j-\hat{\boldsymbol{\mu}}_1\right)\left(\boldsymbol{x}_j-\hat{\boldsymbol{\mu}}_1\right)^T}{\sum_{j=1}^n z_j} = \frac{\sum_{j=1}^n z_j\left(\boldsymbol{x}_j-\hat{\boldsymbol{\mu}}_1\right)\left(\boldsymbol{x}_j-\hat{\boldsymbol{\mu}}_1\right)^T}{\sum_{j=1}^n z_j}.$$

Likewise,

$$\hat{\boldsymbol{\Sigma}}_2 = \frac{\sum_{j=1}^n \left(1-z_j\right)\left(\boldsymbol{x}_j-\hat{\boldsymbol{\mu}}_2\right)\left(\boldsymbol{x}_j-\hat{\boldsymbol{\mu}}_2\right)^T}{\sum_{j=1}^n z_j}.$$

To classify a new subject $\boldsymbol{x}_0$, we need to compute the posterior probabilities. The posterior probabilities are given by

$$\begin{aligned} P\left(Z=1 \mid \boldsymbol{x}_0\right) &= \frac{\frac{\hat{\phi}}{(2\pi)^{p/2}\sqrt{\det\left(\hat{\boldsymbol{\Sigma}}_1\right)}}\exp\left\{-\frac{1}{2}\left(\boldsymbol{x}_0-\hat{\boldsymbol{\mu}}_1\right)^T\hat{\boldsymbol{\Sigma}}_1^{-1}\left(\boldsymbol{x}_0-\hat{\boldsymbol{\mu}}_1\right)\right\}}{P\left(\boldsymbol{x}_0\right)}, \\ P\left(Z=0 \mid \boldsymbol{x}_0\right) &= \frac{\frac{1-\hat{\phi}}{(2\pi)^{p/2}\sqrt{\det\left(\hat{\boldsymbol{\Sigma}}_2\right)}}\exp\left\{-\frac{1}{2}\left(\boldsymbol{x}_0-\hat{\boldsymbol{\mu}}_2\right)^T\hat{\boldsymbol{\Sigma}}_2^{-1}\left(\boldsymbol{x}_0-\hat{\boldsymbol{\mu}}_2\right)\right\}}{P\left(\boldsymbol{x}_0\right)}. \end{aligned}$$

Hence, we classify $\boldsymbol{x}_0$ to the first class if

$$\frac{\frac{\hat{\phi}}{\sqrt{\det\left(\hat{\boldsymbol{\Sigma}}_1\right)}}\exp\left\{-\frac{1}{2}\left(\boldsymbol{x}_0-\hat{\boldsymbol{\mu}}_1\right)^T\hat{\boldsymbol{\Sigma}}_1^{-1}\left(\boldsymbol{x}_0-\hat{\boldsymbol{\mu}}_1\right)\right\}}{\frac{1-\hat{\phi}}{\sqrt{\det\left(\hat{\boldsymbol{\Sigma}}_2\right)}}\exp\left\{-\frac{1}{2}\left(\boldsymbol{x}_0-\hat{\boldsymbol{\mu}}_2\right)^T\hat{\boldsymbol{\Sigma}}_2^{-1}\left(\boldsymbol{x}_0-\hat{\boldsymbol{\mu}}_2\right)\right\}} \geq 1.$$

Taking the logarithm on both sides, this decision rule can be simplified to

$$\begin{aligned} \log\left(\frac{\hat{\phi}}{1-\hat{\phi}}\right)-\frac{1}{2}\log\left(\frac{\det\left(\hat{\boldsymbol{\Sigma}}_1\right)}{\det\left(\hat{\boldsymbol{\Sigma}}_2\right)}\right) &\geq \frac{1}{2}\left(\boldsymbol{x}_0-\hat{\boldsymbol{\mu}}_1\right)^T\hat{\boldsymbol{\Sigma}}_1^{-1}\left(\boldsymbol{x}_0-\hat{\boldsymbol{\mu}}_1\right)-\frac{1}{2}\left(\boldsymbol{x}_0-\hat{\boldsymbol{\mu}}_2\right)^T\hat{\boldsymbol{\Sigma}}_2^{-1}\left(\boldsymbol{x}_0-\hat{\boldsymbol{\mu}}_2\right) \\ &= \frac{1}{2}\left[\boldsymbol{x}_0^T\left(\hat{\boldsymbol{\Sigma}}_1^{-1}-\hat{\boldsymbol{\Sigma}}_2^{-1}\right)\boldsymbol{x}_0-2\left(\hat{\boldsymbol{\mu}}_1^T\hat{\boldsymbol{\Sigma}}_1^{-1}-\hat{\boldsymbol{\mu}}_2^T\hat{\boldsymbol{\Sigma}}_2^{-1}\right)\boldsymbol{x}_0\right] \\ &\quad +\frac{1}{2}\left(\hat{\boldsymbol{\mu}}_1^T\hat{\boldsymbol{\Sigma}}_1^{-1}\hat{\boldsymbol{\mu}}_1-\hat{\boldsymbol{\mu}}_2^T\hat{\boldsymbol{\Sigma}}_2^{-1}\hat{\boldsymbol{\mu}}_2\right). \end{aligned}$$

Hence the decision boundary is quadratic in $\boldsymbol{x}_0$.

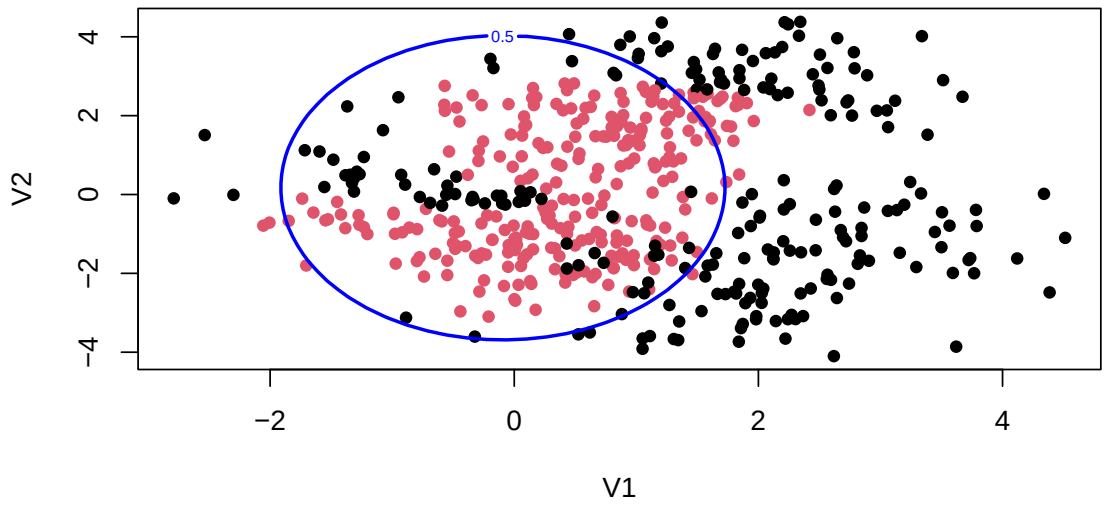
3. (1pt) Consider the data set TwoClass.RData that we used for illustration of Chapter 11 Classification. We have hand-coded the Gaussian discriminant analysis (naive Bayes) during the lectures. The R package e1071 has a function naiveBayes(x, y) that performs Gaussian discriminant analysis under the assumption that the covariance matrices from different groups are different but are diagonal. To extract the predicted posterior probability, the function is predict(object, newdata, type = "raw"). Apply such function to perform Gaussian discriminant analysis to the data set. Plot the decision boundary. How does the decision boundary look like?

Solution: We run the function naiveBayes

```
library(e1071)
NB <- naiveBayes(x = Data[, c("V1", "V2")], y = factor(Data[, "Label"]))
```

To plot the decision boundary, we will use the code during the lecture.

```
## Decision boundary
V1grid <- seq(-3, 5, length.out = 50)
V2grid <- seq(-6, 5, length.out = 50)
Vgrid <- expand.grid(V1grid, V2grid)
names(Vgrid) <- c("V1", "V2")
Decision <- predict(NB, newdata = Vgrid, type = "raw")[, "Y"]
plot(Data[, "V1"], Data[, "V2"], col = (Data$Label == "Y") + 1, pch = 16,
     xlab = "V1", ylab = "V2")
contour(x = V1grid, y = V2grid, z = matrix(Decision, 50, 50), level = 0.5,
    add = TRUE, col = "blue", lwd = 2)
```



It is clearly seen that the decision boundary is not linear in V1. It is only linear in V1, V2 and V1sq.

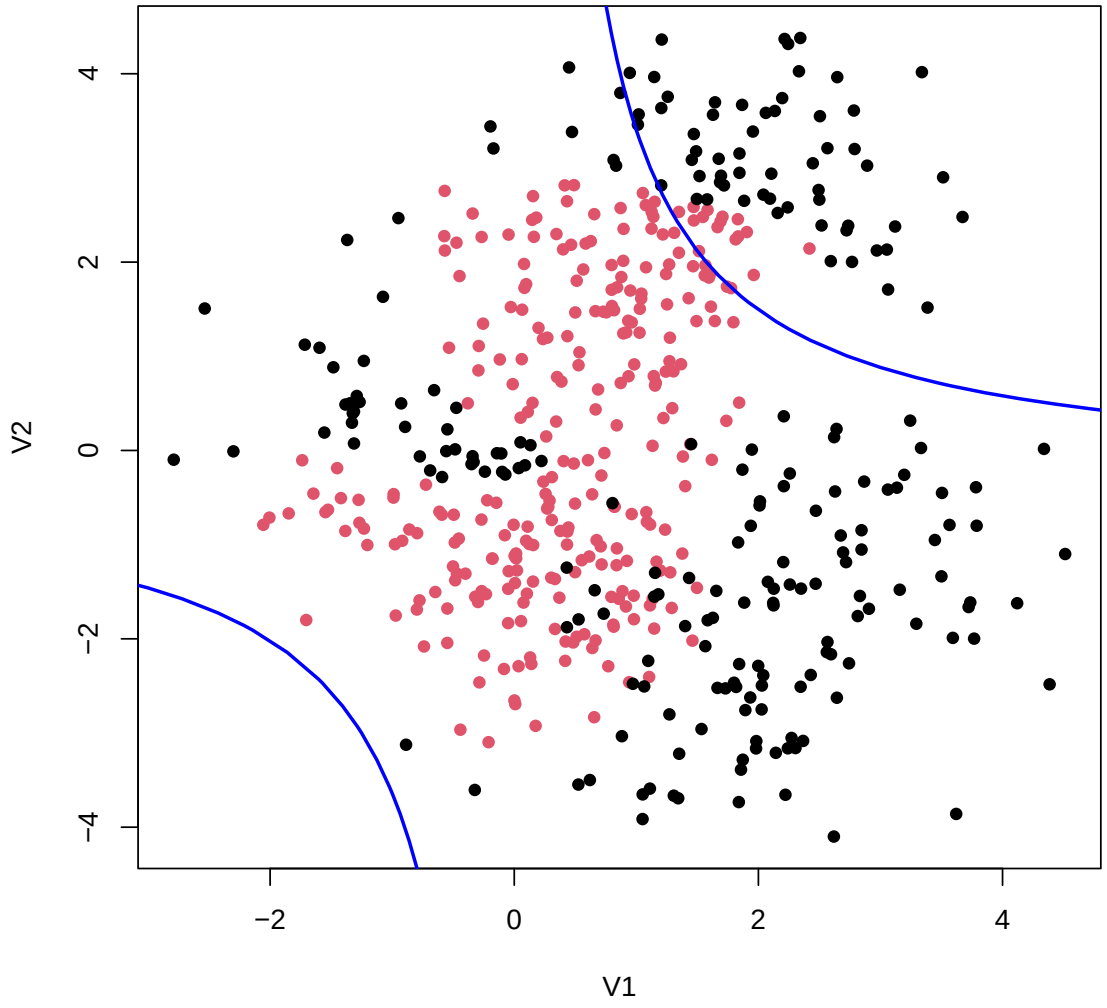
4. (1pt) Consider again the data set TwoClass.RData that we used for illustration of Chapter 11 Classification. You can download the data set from the Studium page Modules/code. Assume that both variables V1 and V2 in the data set are normally distributed. If we create a new variable V1sq as the square of V1, i.e., Data$V1sq <- Data$V1 * Data$V1. Can we apply Gaussian discriminant analysis (naive Bayes) to the data set with variables V1, V2, and V1sq? What about Fisher's LDA? If so, will the decision boundary be linear in V1? Otherwise, state the reason.
Solution: Gaussian discriminant analysis is derived under the assumption that data follow

a normal distribution in each class. When V1sq is included in the data, it is certainly not normally distributed. Hence, the Gaussian discriminant analysis is questionable. However, Fisher's LDA can be viewed from a distribution-free perspective. Hence, we will apply Fisher's LDA here.

```
library(MASS)
Data$V1sq <- Data$V1 * Data$V1
LDA <- lda(x = Data[, c("V1", "V2", "V1sq")], grouping = Data$Label)
```

To plot the decision boundary, we will use the code during the lecture.

```
## Decision boundary
V1grid <- seq(-7, 7, length.out = 50)
V2grid <- seq(-7, 7, length.out = 50)
Vgrid <- expand.grid(V1grid, V2grid)
Vgrid <- cbind(Vgrid, Vgrid[, 1] * Vgrid[, 2])
names(Vgrid) <- c("V1", "V2", "V1sq")
Decision <- predict(LDA, newdata = Vgrid)$posterior[, "Y"]
plot(Data[, "V1"], Data[, "V2"], col = (Data$Label == "Y") + 1, pch = 16,
     xlab = "V1", ylab = "V2")
contour(x = V1grid, y = V2grid, z = matrix(Decision, 50, 50), level = 0.5,
    add = TRUE, col = "blue", lwd = 2)
```



It is clearly seen that the decision boundary is not linear in V1.

5. (1pt) Consider the data set in HWA4.RData. The data set contains 718 observations from a study of heart disease. Our goal is to classify the patients into two classes: with heart disease (1) or without heart disease (0) using the other variables in the data set. Use SVM, and Random Forest to build the classifiers. Even though the tuning parameters need to be chosen by cross validation, you can arbitrarily choose values for tuning parameters for this task. Test your classifiers in the new data set TestData. Which classifier do you prefer?
Solution: We fit these classifiers in R and also produce the corresponding confusion matrix.

```
library(caret)
## Load data
load("C:/Users/shaji948/Box/Teaching/Math Department/Multivariate analysis 1MS003/2
```

```r
## SVM
svmfit <- svm(HeartDisease ~ Age + Sex + RestingBP + Cholesterol + FastingBS + MaxH
data = Data,
kernel = "radial", #Gaussian kernel
type = "C-classification",
cost = 1, gamma = 1, # For illustration. Cross validation in practice
scale = TRUE) # Standardize my data
SVMpred <- predict(svmfit, newdata = TestData)
confusionMatrix(data = SVMpred,
reference = factor(TestData$HeartDisease, levels = levels(SVMpred)),
mode = "everything",
positive = "1")

## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 76 25
##          1 29 70
##
##                Accuracy : 0.73
##                  95% CI : (0.6628, 0.7902)
##     No Information Rate : 0.525
##     P-Value [Acc > NIR] : 2.343e-09
##
##                   Kappa : 0.4597
##
##  Mcnemar's Test P-Value : 0.6831
##
##             Sensitivity : 0.7368
##             Specificity : 0.7238
##          Pos Pred Value : 0.7071
##          Neg Pred Value : 0.7525
##               Precision : 0.7071
##                  Recall : 0.7368
##                      F1 : 0.7216
##              Prevalence : 0.4750
##          Detection Rate : 0.3500
##    Detection Prevalence : 0.4950
##       Balanced Accuracy : 0.7303
##
##        'Positive' Class : 1
##

## Random forest
```

```r
library(randomForest)
RF <- randomForest(x = Data[, -9],
y = factor(Data[, "HeartDisease"]),
ntree = 1000)
RFpred <- predict(RF, newdata = TestData)
confusionMatrix(data = RFpred,
reference = factor(TestData$HeartDisease, levels = levels(SVMpred)),
mode = "everything",
positive = "1")

## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 91 33
##          1 14 62
##
##                Accuracy : 0.765
##                  95% CI : (0.7, 0.8219)
##     No Information Rate : 0.525
##     P-Value [Acc > NIR] : 2.221e-12
##
##                   Kappa : 0.5243
##
##  Mcnemar's Test P-Value : 0.00865
##
##             Sensitivity : 0.6526
##             Specificity : 0.8667
##          Pos Pred Value : 0.8158
##          Neg Pred Value : 0.7339
##               Precision : 0.8158
##                  Recall : 0.6526
##                      F1 : 0.7251
##              Prevalence : 0.4750
##          Detection Rate : 0.3100
##    Detection Prevalence : 0.3800
##       Balanced Accuracy : 0.7596
##
##        'Positive' Class : 1
##
```

The confusion matrices are similar. To compare two classifiers, we use the McNemar test.

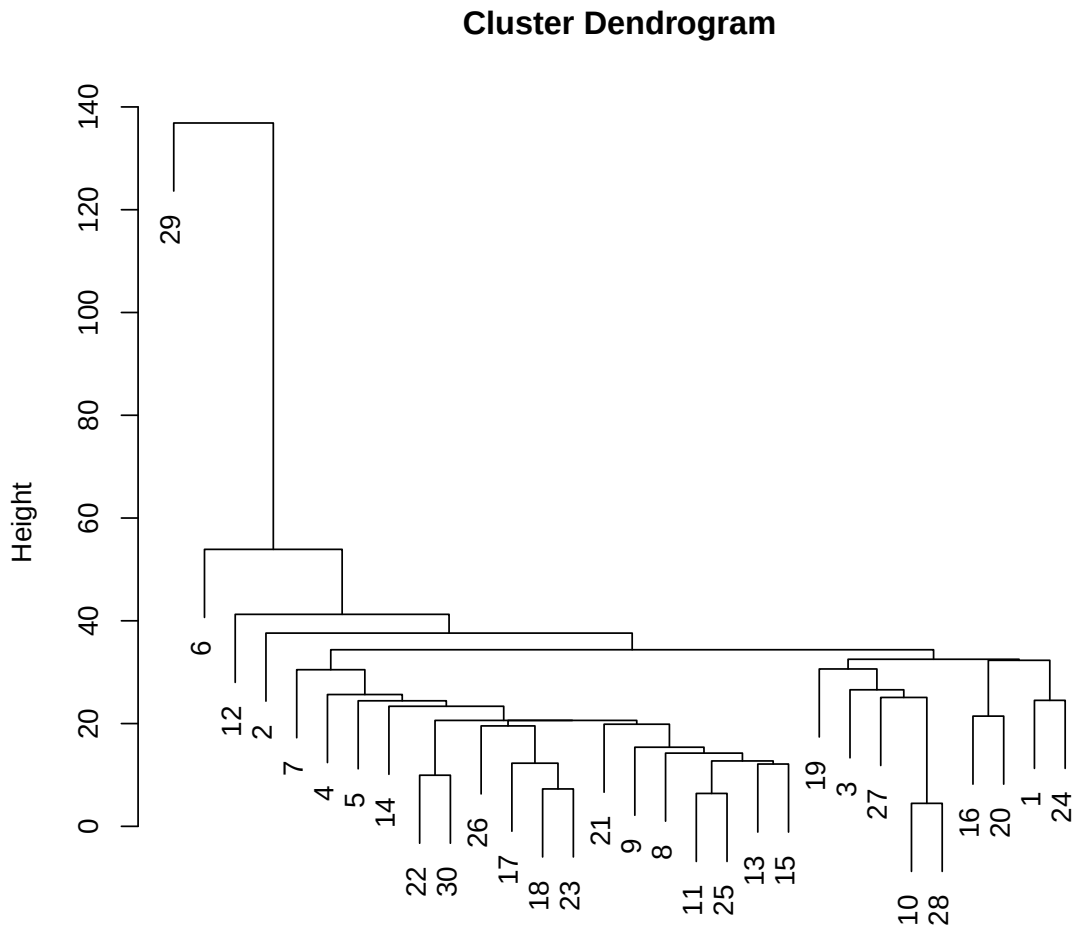
```
CW.SVMpred <- (SVMpred == TestData$HeartD)
CW.RF <- (RFpred == TestData$HeartD)
mcnemar.test(x = CW.SVMpred, y = CW.RF)

##
##  McNemar's Chi-squared test with continuity correction
##
## data:  CW.SVMpred and CW.RF
## McNemar's chi-squared = 1.1613, df = 1, p-value = 0.2812
```

The McNemar test suggests that these classifers are not statistically indifferent.

6. (1pt) Consider again the heart disease data. Apply hierarchical clustering to cluster the first 30 observations in the data set Data using the variables Age, RestingBP, Cholesterol, and MaxHR. Also determine the number of clusters using the CH index. Report the dendrodiagram.Based on the dendrodiagram, how would you group the observations if you decided to have 3 clusters?
Solution: For illustration, we choose the single linkage.

```
#### Clustering ####
hclust.single <- hclust(dist(Data[1 : 30, c("Age", "RestingBP", "Cholesterol", "Max
plot(hclust.single)
```

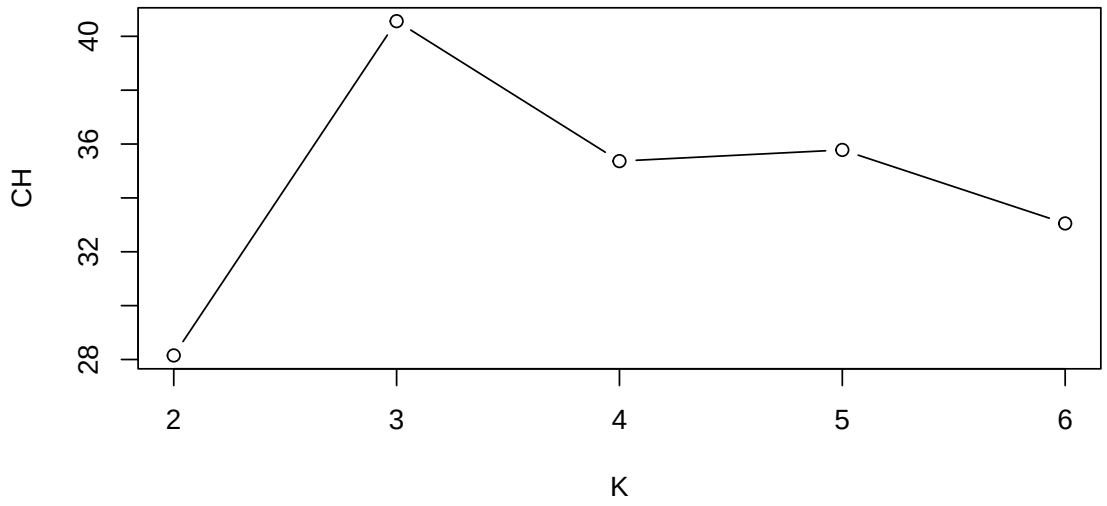


If we want to have 3 clusters, cluster 1 only consists of 29th observation, cluster 2 consists of the 6th observations, and cluster 3 consists of the remaining ones.

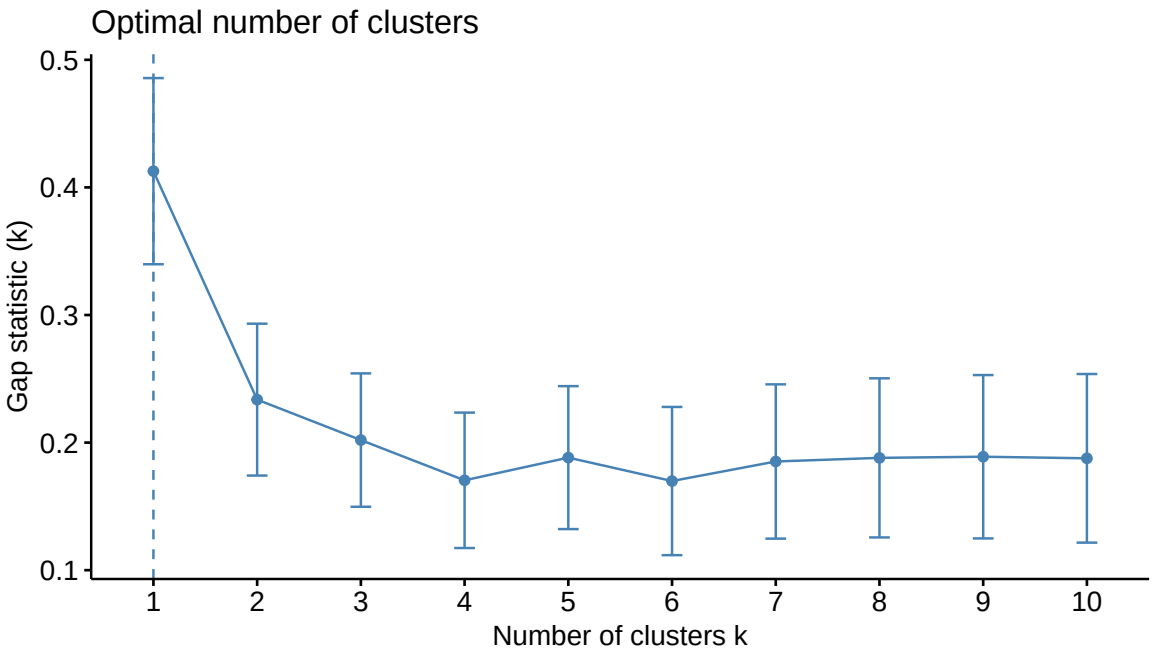
7. (1pt) Consider again the heart disease data. Apply k-means clustering to cluster the data set Data using the variables Age, RestingBP, Cholesterol, and MaxHR. Also determine the number of clusters. Ignore the warning messages, if any.
Solution:

```
## Scale data
SData <- cbind(Data[1 : 30, ])
## CH index and elbow method
WithinSS <- rep(0, 5)
BetwSS <- rep(0, 5)
for(k in 2 : 6){
KM <- kmeans(SData[1 : 30, c("Age", "RestingBP", "Cholesterol", "MaxHR")], centers
```

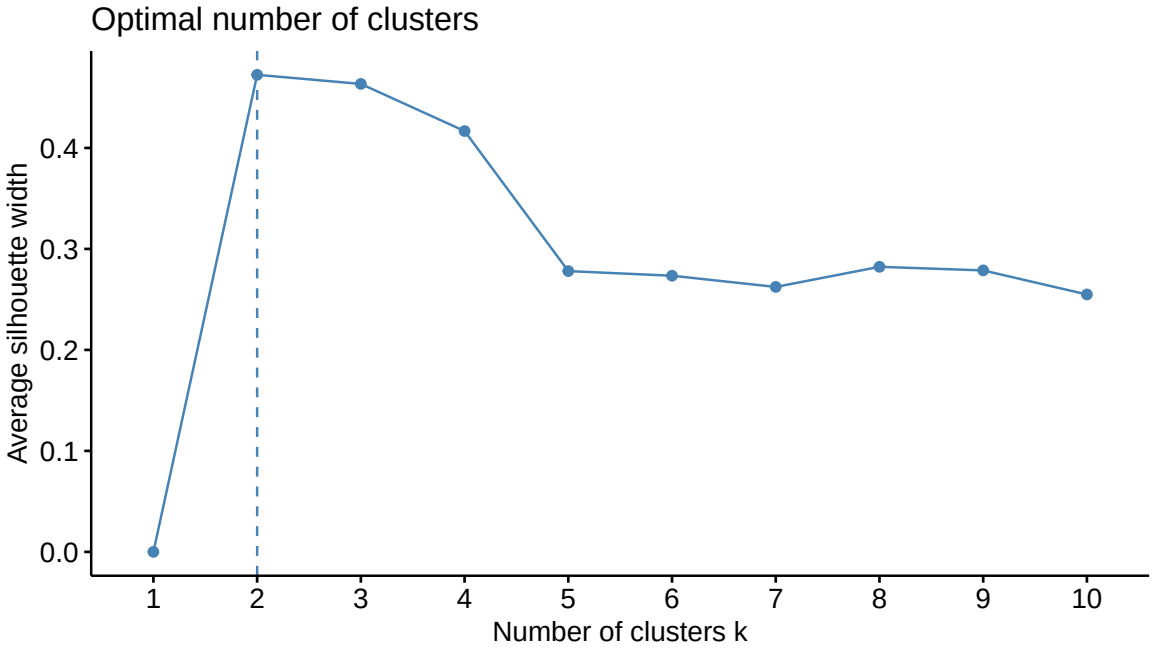
```
WithinSS[k - 1] <- KM$tot.withinss
BetwSS[k - 1] <- KM$betweenss
}
K <- 2 : 6
N <- nrow(SData)
CH <- (BetwSS / (K - 1)) / (WithinSS / (N - K))
plot(K, CH, type = "b") # Maximum CH ind
```




```
## Gap statistic:
library(cluster)
library(factoextra)
gap_stat <- cluster::clusGap(SData, FUN = kmeans, nstart = 100, K.max = 10, B = 100
fviz_gap_stat(gap_stat)
```




```
## Silhouette method
fviz_nbclust(SData, FUNcluster = kmeans, method = "silhouette", nstart = 100)
```



We can then investigate either 2 or 3 clusters.

8. (2pt) In this task, we will complete the Gaussian mixture analysis using the EM algorithm. Suppose that we have observed a random sample $\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_N$ from K populations, but we do not observe which population each observation comes from. We assume that

$$\boldsymbol{X}|Z = k \quad \sim \quad N_q(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k),$$

$$P(Z=k) \quad = \quad p_k.$$

A recap of what we have done first. The E step computes the conditional expectation

$$Q := \mathbb{E}\left(\log f(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta}) \mid \boldsymbol{X};\hat{\boldsymbol{\theta}}^{(t)},\hat{\boldsymbol{p}}^{(t)}\right) \quad = \quad \sum_{j=1}^{N}\sum_{k=1}^{K}\left[\log p_k + \log f\left(\boldsymbol{x}_j \mid Z_j=k;\boldsymbol{\theta}\right)\right] P\left(Z_j=k \mid \boldsymbol{x}_j;\boldsymbol{\theta}^{(t)}\right).$$

With the normal assumption, it becomes

$$Q \quad = \quad \sum_{j=1}^{N}\sum_{k=1}^{K}\left[\log p_k - \frac{q}{2}\log(2\pi) - \frac{1}{2}\log\det(\boldsymbol{\Sigma}_k) - \frac{1}{2}\left(\boldsymbol{x}_j-\boldsymbol{\mu}_k\right)^T\boldsymbol{\Sigma}_k^{-1}\left(\boldsymbol{x}_j-\boldsymbol{\mu}_k\right)\right] P\left(Z_j=k \mid \boldsymbol{x}_j;\boldsymbol{\theta}^{(t)}\right).$$

We have shown during the lecture that

$$p_k \quad = \quad \frac{\sum_{j=1}^{N} P\left(Z_j=k \mid \boldsymbol{x}_j;\boldsymbol{\theta}^{(t)}\right)}{\sum_{j=1}^{N} P\left(Z_j=K \mid \boldsymbol{x}_j;\boldsymbol{\theta}^{(t)}\right)} p_K,$$

where

$$P\left(Z_j=k \mid \boldsymbol{x}_j;\boldsymbol{\theta}^{(t)}\right) \quad = \quad \frac{p_k^{(t)} P\left(\boldsymbol{x}_j \mid Z_j=k;\boldsymbol{\theta}^{(t)}\right)}{\sum_{m=1}^{K} p_m^{(t)} P\left(\boldsymbol{x}_j \mid Z_j=m;\boldsymbol{\theta}^{(t)}\right)}.$$

Hence,

$$\begin{aligned}\hat{p}_k \quad &= \quad \frac{\sum_{j=1}^{N} P\left(Z_j=k \mid \boldsymbol{x}_j;\boldsymbol{\theta}^{(t)}\right)}{\sum_{k=1}^{K}\sum_{j=1}^{N} P\left(Z_j=k \mid \boldsymbol{x}_j;\boldsymbol{\theta}^{(t)}\right)} \\ &= \quad \frac{1}{N}\sum_{j=1}^{N}\frac{p_k^{(t)} P\left(\boldsymbol{x}_j \mid Z_j=k;\boldsymbol{\theta}^{(t)}\right)}{\sum_{m=1}^{K} p_m^{(t)} P\left(\boldsymbol{x}_j \mid Z_j=m;\boldsymbol{\theta}^{(t)}\right)}.\end{aligned}$$

In this task, we will complete the M step. For simplicity, we consider the univariate Gaussian mixture where $q=1$. Derive the expression of $\hat{\mu}_k^{(t+1)}$ and $\hat{\Sigma}_k^{(t+1)}$. Also speculate (no need to derive) the expression of $\hat{\boldsymbol{\mu}}_k^{(t+1)}$ and $\hat{\boldsymbol{\Sigma}}_k^{(t+1)}$ if $q>1$.

$$-\frac{1}{2}\sum_{j=1}^{N}\left[\left(\boldsymbol{x}_j-\boldsymbol{\mu}_k\right)^T\boldsymbol{\Sigma}_k^{-1}\left(\boldsymbol{x}_j-\boldsymbol{\mu}_k\right)\right] P\left(Z_j=k \mid \boldsymbol{x}_j;\boldsymbol{\theta}^{(t)}\right),$$

which is equivalent to

$$Q_{\mu,k} \quad := \quad -\frac{1}{2}\sum_{j=1}^{N}\left[\left(\boldsymbol{x}_j-\boldsymbol{\mu}_k\right)^T\boldsymbol{\Sigma}_k^{-1}\left(\boldsymbol{x}_j-\boldsymbol{\mu}_k\right)\right] P\left(Z_j=k \mid \boldsymbol{x}_j;\boldsymbol{\theta}^{(t)}\right).$$

The part of Q that involves $\boldsymbol{\Sigma}_k$ is

$$Q_{\Sigma,k} = \sum_{j=1}^{N}\left[-\frac{1}{2}\log\det\left(\boldsymbol{\Sigma}_k\right)-\frac{1}{2}\left(\boldsymbol{x}_j-\boldsymbol{\mu}_k\right)^T\boldsymbol{\Sigma}_k^{-1}\left(\boldsymbol{x}_j-\boldsymbol{\mu}_k\right)\right]P\left(Z_j=k\mid\boldsymbol{x}_j;\boldsymbol{\theta}^{(t)}\right).$$

Solution: With univariate normal mixture, we have

$$Q = \sum_{j=1}^{N}\sum_{k=1}^{K}\left[\log p_k-\frac{1}{2}\log\left(2\pi\right)-\frac{1}{2}\log\left(\Sigma_k\right)-\frac{\left(x_j-\mu_k\right)^2}{2\Sigma_k}\right]P\left(Z_j=k\mid x_j;\boldsymbol{\theta}^{(t)}\right).$$

Note that

$$\frac{\partial Q}{\partial\mu_k} = -\sum_{j=1}^{N}\frac{x_j-\mu_k}{\Sigma_k}P\left(Z_j=k\mid x_j;\boldsymbol{\theta}^{(t)}\right)=-\frac{\sum_{j=1}^{N}\left(x_j-\mu_k\right)P\left(Z_j=k\mid x_j;\boldsymbol{\theta}^{(t)}\right).}{\Sigma_k}$$

Hence,

$$\hat{\mu}_k^{(t+1)} = \frac{\sum_{j=1}^{N}P\left(Z_j=k\mid x_j;\boldsymbol{\theta}^{(t)}\right)x_j}{\sum_{j=1}^{N}P\left(Z_j=k\mid x_j;\boldsymbol{\theta}^{(t)}\right)}.$$

Note that

$$\frac{\partial Q}{\partial\Sigma_k} = \sum_{j=1}^{N}\left[-\frac{1}{2\Sigma_k}+\frac{\left(x_j-\mu_k\right)^2}{2\Sigma_k^2}\right]P\left(Z_j=k\mid x_j;\boldsymbol{\theta}^{(t)}\right).$$

Hence,

$$\hat{\Sigma}_k^{(t+1)} = \frac{\sum_{j=1}^{N}P\left(Z_j=k\mid x_j;\boldsymbol{\theta}^{(t)}\right)\left(x_j-\hat{\mu}_k^{(t)}\right)^2}{\sum_{j=1}^{N}P\left(Z_j=k\mid x_j;\boldsymbol{\theta}^{(t)}\right)}.$$

If we have $q>1$, then we will have

$$\begin{aligned}\hat{\boldsymbol{\mu}}_k^{(t+1)} &= \frac{\sum_{j=1}^{N}P\left(Z_j=k\mid x_j;\boldsymbol{\theta}^{(t)}\right)\boldsymbol{x}_j}{\sum_{j=1}^{N}P\left(Z_j=k\mid x_j;\boldsymbol{\theta}^{(t)}\right)},\\ \hat{\boldsymbol{\Sigma}}_k^{(t+1)} &= \frac{\sum_{j=1}^{N}P\left(Z_j=k\mid x_j;\boldsymbol{\theta}^{(t)}\right)\left(\boldsymbol{x}_j-\hat{\boldsymbol{\mu}}_k^{(t)}\right)\left(\boldsymbol{x}_j-\hat{\boldsymbol{\mu}}_k^{(t)}\right)^T}{\sum_{j=1}^{N}P\left(Z_j=k\mid x_j;\boldsymbol{\theta}^{(t)}\right)}.\end{aligned}$$