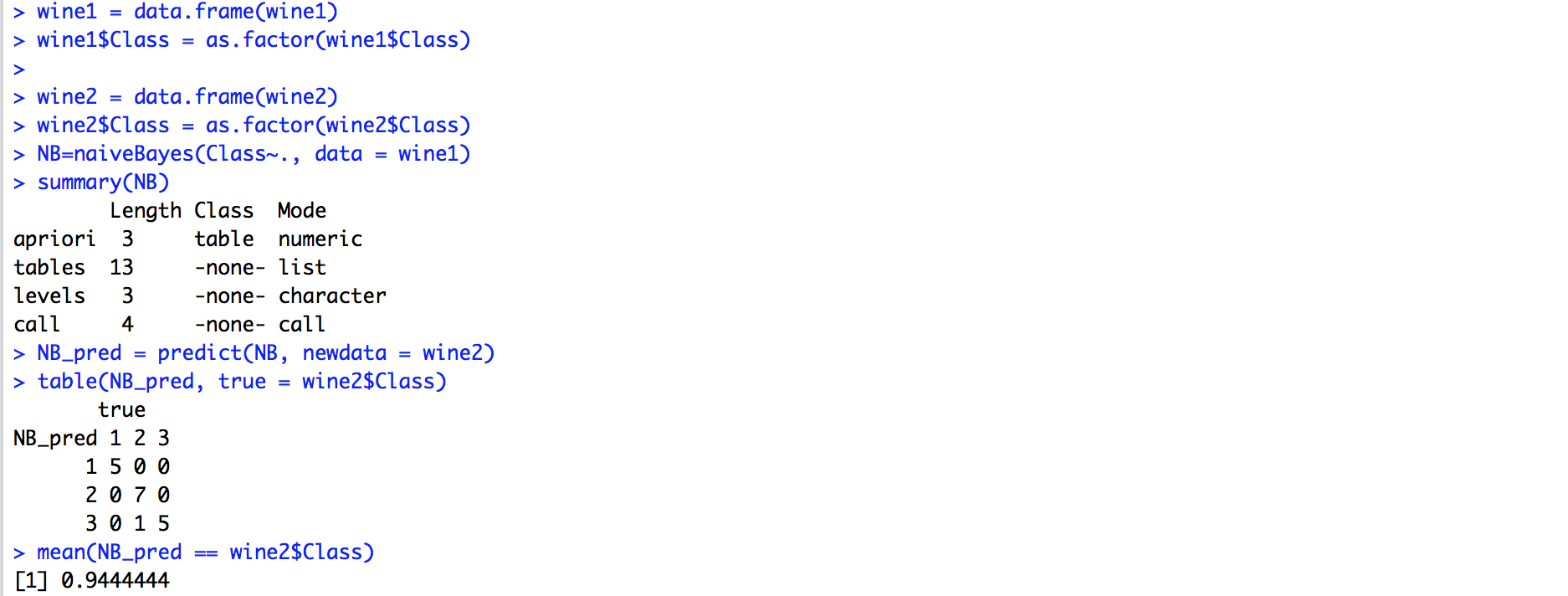
Statistical Data Mining Assignment 4

Saurav Chakraborty

1. From the slides, see the code in slide 26. You will have to modify this code to do the following:

The Wine.dat contains 178 rows describing bottles of wine. The first 160 can be used to train a Naïve Bayes classifier and the last 18 can be used to test it. How accurate do you classify the last 18 wine options? Give the confusion matrix (table) as part of the word document you will submit.

Solution:



So the confusion matrix can be seen above and the accuracy of the predictions is around 94.4%.

True

NB\_pred 1 2 3

1 5 0 0

2 0 7 0

3 0 1 5

2. The OptDigits.csv file consists of 5620 rows. This dataset contains 64 columns of data to predict the digit {0-9}. Think about how an input box on a form might read handwriting. In this case, each attribute is the number of pixels (0-15) that are shaded in a very small 4x4 square, and these squares are numbered 0-63. The top row is 0-7, row 2 is 8-15, row 3 is 16-23, and so on.

Some of these columns are all 0’s – meaning that no one marked that small square on the form. This happens near the edges of the input box, so numbers that represent boundaries are very sparse (0, 7, 8, 15, 16, ...). These will have to be removed or the math fails.

Break this into 80% training and 20% training. Use Naïve Bayes to predict the digits based on the non-zero attribute columns. How accurate are you at predicting? Give the table as part of the word document you will submit.

Solution:



The accuracy of prediction comes to be around 84 percent here.

The two columns with only 0 values were the first and 40th column.

3. On the recording, we performed Principal Components in Excel on the Olympic Decathletes data. It was a bit of work. In R of course, it is one line of code to run the algorithm. The book describes PCA in detail. Using R, calculate the principal components. For each athlete, which component best describes them? Give your R code for this.

Submit your R code and a brief summary of how the components were used to describe the athletes. Upload this entire word doc to Canvas.

Solution:

olympic = read.csv(file.choose())

head(olympic)

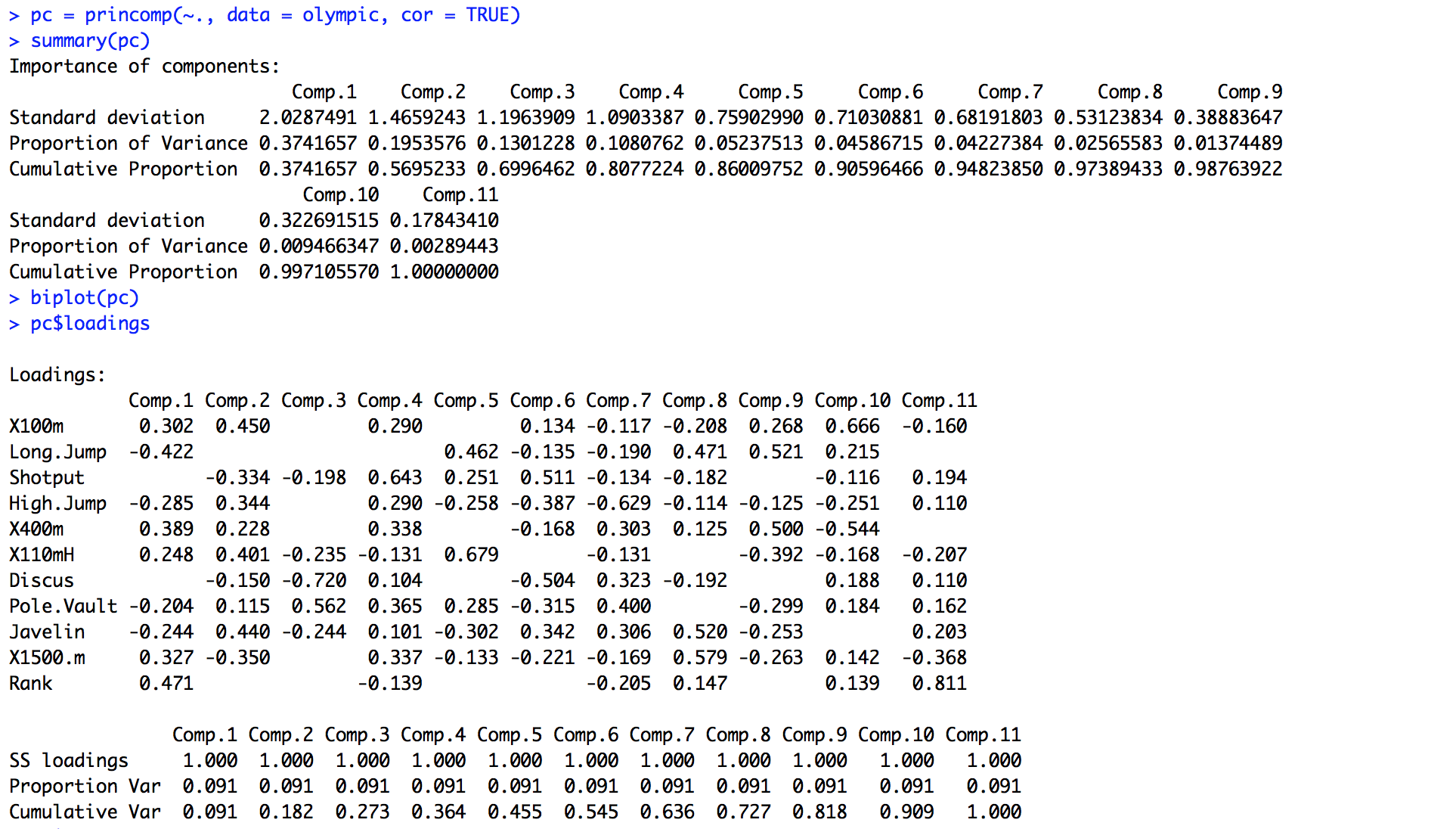
pc = princomp(~., data = olympic, cor = TRUE)

summary(pc)

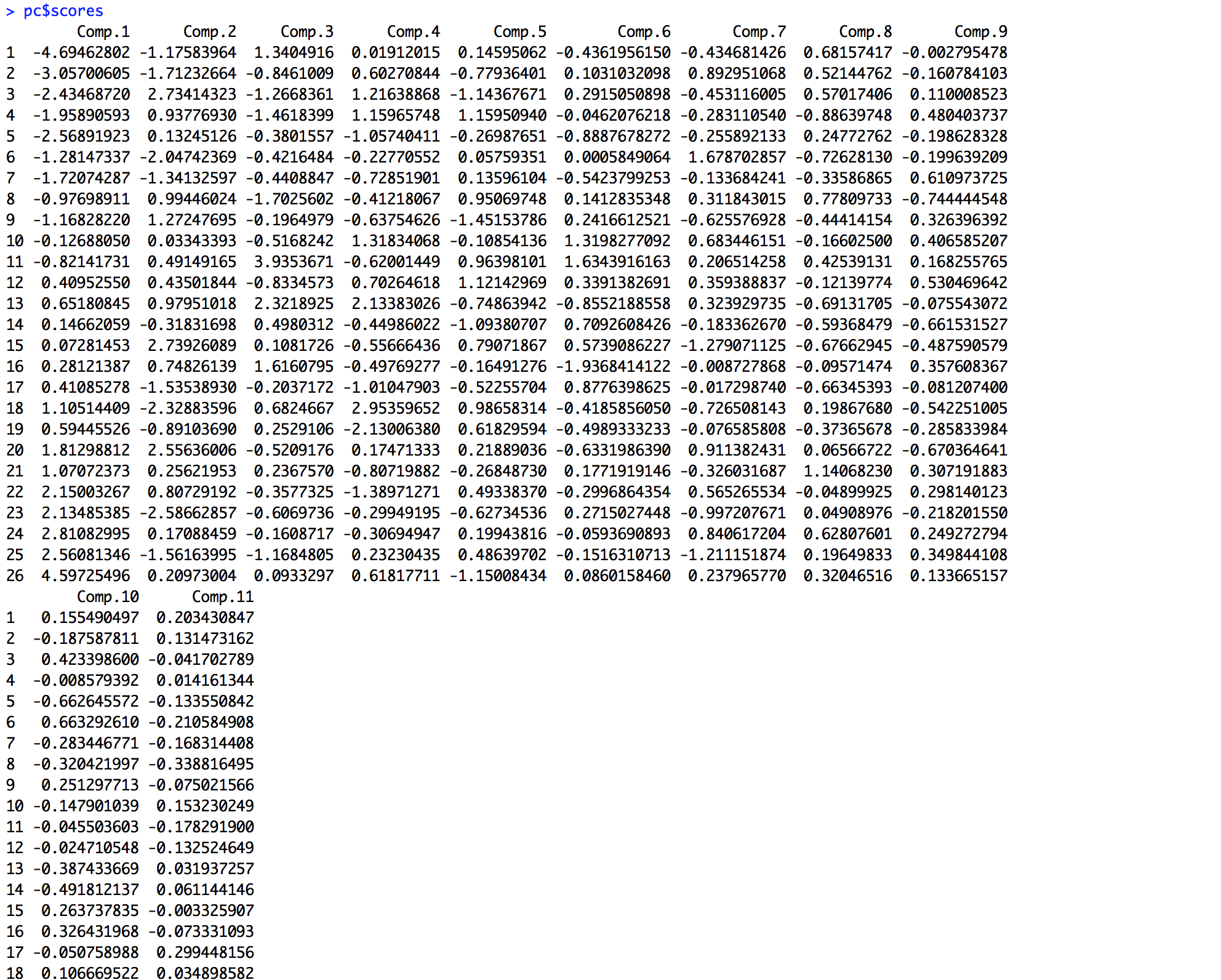
pc$loadings

pc$scores

The loadings are as follows:



The scores are as follows:



So for each player the most important component is as follows:

|  |  |  |
| --- | --- | --- |
| **Rank** | **Athlete** | **Best Described by** |
| 1 | Ashton Eaton | Comp.1 (-4.48134602) |
| 2 | Trey Hardee | Comp.1 (-2.94580574 ) |
| 3 | Leonel Suarez | Comp. 2(2.542500844) |
| 4 | Hans Van Alphen | Comp. 3 (2.1894492) |
| 5 | Damian Warner | Comp. 1 (-2.44745314) |
| 6 | Rico Freimuth | Comp. 2 (-1.899953933) |
| 7 | Oleksiy Kasyanov | Comp. 1(-1.64633687) |
| 8 | Sergey Sviridov | Comp. 3 (1.4264670) |
| 9 | Willem Coertzen | Comp. 7 (1.49898908) |
| 10 | Pascal Behrenbruch | Comp. 6 (1.53814522) |
| 11 | Eelco Sintnicolaas | Comp. 3 (-3.0370769) |
| 12 | Brent Newdick | Comp. 7 (-1.16552775), Comp.6 (1.16314136) |
| 13 | Gonzalo Barroilhet | Comp. 5 (2.2009187) |
| 14 | Yordanis Garcia | Comp. 7 (1.08505430) |
| 15 | Kevin Mayer | Comp. 2 (2.682120625) |
| 16 | Ilya Shkurenyov | Comp. 3 (-1.7986211) |
| 17 | Eduard Mikhan | Comp. 5 (-1.7690239) |
| 18 | Dmitriy Karpov | Comp. 2 (-2.964271880) |
| 19 | Luiz Alberto de Araujo | Comp. 5 (-2.3959121) |
| 20 | Keisuke Ushiro | Comp. 2 (2.551114082) |
| 21 | Ingmar Vos | Comp. 9 (-1.35740417) |
| 22 | Edgars Erins | Comp. 1 (1.48852218) |
| 23 | Jangy Addy | Comp. 2(-2.800870612) |
| 24 | Attila Szabo | Comp. 4(-2.359752236) |
| 25 | Darius Draudvila | Comp. 1(2.58994697) |
| 26 | Rifat Artikov | Comp. 1(4.50216644) |

R CODE:

#Question 01

Wine = read.csv(file.choose())

wine1 <- Wine[1:160,]

wine2 <- Wine[161:178,]

wine1 = data.frame(wine1)

wine1$Class = as.factor(wine1$Class)

wine2 = data.frame(wine2)

wine2$Class = as.factor(wine2$Class)

NB=naiveBayes(Class~., data = wine1)

summary(NB)

NB\_pred = predict(NB, newdata = wine2)

table(NB\_pred, true = wine2$Class)

mean(NB\_pred == wine2$Class)

#Question 02

OptDigit = read.csv(file.choose())

head(OptDigit)

j=1

out <- NULL

temp <- NULL

#taking out the columns with 0 values

for(i in 1:ncol(OptDigit))

{

temp <- unique(OptDigit[,i])

if(length(temp)==1 & (temp)==0)

{

out[j] <- i;

j = j+1

}

}

temp2 <- OptDigit[,-out]

#Building training and testing data

smp\_size = floor(0.80\*nrow(temp2))

set.seed(123)

train\_temp = sample(seq\_len(nrow(temp2)), size = smp\_size)

train = temp2[train\_temp, ]

test = temp2[-train\_temp, ]

nrow(train)

nrow(test)

train = data.frame(train)

train$OptDigit = as.factor(train$OptDigit)

test = data.frame(test)

test$OptDigit = as.factor(test$OptDigit)

NB=naiveBayes(OptDigit~., data = train)

summary(NB)

NB\_pred = predict(NB, newdata = test)

table(NB\_pred, true = test$OptDigit)

mean(NB\_pred == test$OptDigit)NB\_pred = predict(NB, newdata = test)

table(NB\_pred, true = test$OptDigit)

mean(NB\_pred == test$OptDigit)

#Question 3:

olympic = read.csv(file.choose())

head(olympic)

pc = princomp(~., data = olympic, cor = TRUE)

summary(pc)

pc$loadings

pc$scores