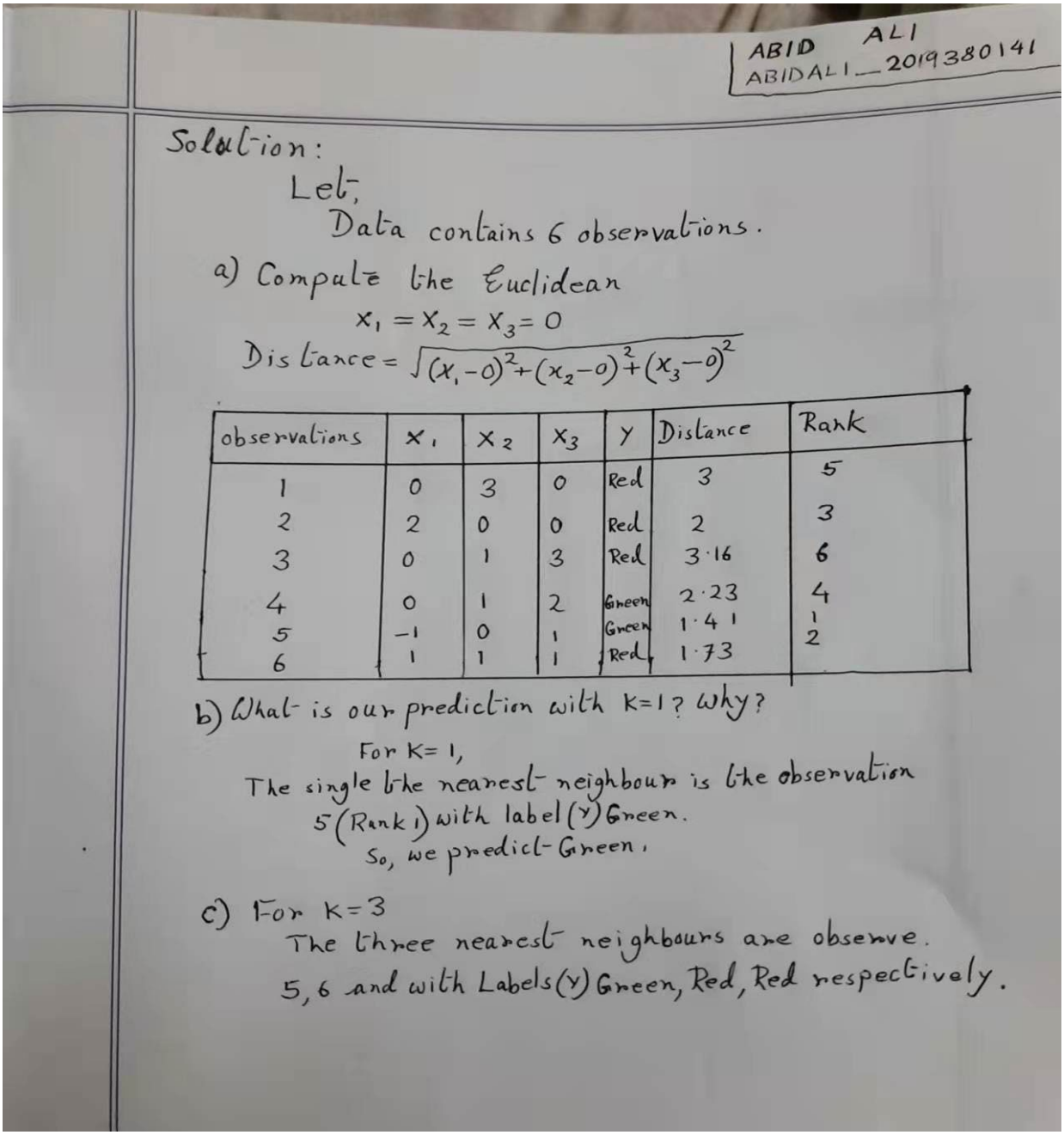
Student Name : ABID ALI
Student Roll : 2019380141
Teacher's Email : dayong.tian@nwpu.edu.cn

Homework -1

Page 53, Question 7

7. The table below provides a training data set containing six observations,three predictors, and one qualitative response variable.

Solution:



ABID ALI
ABIDALI_2019380141

Solution:

Let,

Data contains 6 observations.

a) Compute the Euclidean

$$x_1 = x_2 = x_3 = 0$$

$$\text{Distance} = \sqrt{(x_1-0)^2+(x_2-0)^2+(x_3-0)^2}$$

observations	x_1	x_2	x_3	Y	Distance	Rank
1	0	3	0	Red	3	5
2	2	0	0	Red	2	3
3	0	1	3	Red	3.16	6
4	0	1	2	Green	2.23	4
5	-1	0	1	Green	1.41	1
6	1	1	1	Red	1.73	2

b) What is our prediction with K=1? why?

For K= 1,

The single the nearest neighbour is the observation 5 (Rank 1) with label (Y) Green.

So, we predict Green.

c) For K=3

The three nearest neighbours are observe.

5, 6 and with Labels (Y) Green, Red, Red respectively.

Majority is red
∴ We predict Red.

d) ⇒ As K becomes larger,
The boundary becomes inflexible (linear).

⇒ In this case,
We would expect the best value
for K to be small.

Page 59, Question 9

This exercise involves the Auto data set studied in the lab. Make sure that the missing values have been removed from the data.

Solution:

First read "Auto.csv" file using read.csv() auto=read.csv("Auto.csv",head=T)

dimensions

dim(auto)

create data frame with missing values removed(using na)

auto=na.omit(auto)

now see dimentions after remove missing values

dim(auto)

(a) Which of the predictors are quantitative, and which are qualitative?

Solution:

convert origin to factor

auto$origin=as.factor(auto$origin)

create factor version of cylinder and merge

```
cylinders=as.factor(auto$cylinder)

auto=data.frame(auto,cylinders)

rm(cylinders) # remove cylinder factor

# Rename integer version of cylinder

auto$cylinders.int=auto$cylinders

# Drop old version of cylinder

auto=subset(auto, select = -cylinders)

# Convert horsepower to numeric

auto$horsepower=as.numeric(as.character(auto$horsepower)) # convert to character
```

(b) What is the range of each quantitative predictor? You can answer this using the range() function.

Solution:

```
# Assign temp(data.frame())

temp=auto[, !sapply(auto, is.factor)] # variables are not factors

# Use apply

temp=t(apply(temp, 2, function(x) range(x))) # 2x7 transpose matrix
```

```
# Add column names Min and Max

colnames(temp)=c("Min", "Max")
# Round to two digits

round(temp, digits = 2)

rm(temp) # remove temp
```

(c) What is the mean and standard deviation of each quantitative predictor?

Solution:

```
# Assign temp(data.frame())

temp=auto[, !sapply(auto, is.factor)] # no factors variables

# Use apply

temp=t(apply(temp, 2, function(x) c(mean(x), sd(x)))) # 2x7 transpose matrix

# Add column names Mean and Std. Deviation

colnames(temp)=c("Mean", "Std. Deviation")

# Round to two digits

round(temp, digits = 2)

rm(temp) # remove temp
```

(d) Now remove the 10th through 85th observations. What is the range, mean, and standard deviation of each predictor in the subset of the data that remains?

Solution:

```
# sorted list

auto=auto[order(as.numeric(row.names(auto))), ]

# Remove 10th through 85th observation (using rm())

auto.rm=auto[-c(10:85), ]

# Assign temp(data.frame())

temp=auto.rm[, !sapply(auto.rm, is.factor)] # no factor variables

# Use apply()

temp=t(apply(temp, 2, function(x) c(range(x), mean(x), sd(x)))) # 4x7 transpose matrix

# Add column names Min, Max, Mean and Std. Deviation

colnames(temp)=c("Min", "Max", "Mean", "Std. Deviation")

# Round to two digits

round(temp, digits = 2)
rm(temp) # remove temp
```

(e) Using the full data set, investigate the predictors graphically, using scatterplots or other tools of your choice. Create some plots highlighting the relationships among the predictors. Comment on your findings.

Solution:

```
# Assign temp(data.frame())


temp=auto[, !sapply(auto, is.factor)] # no factor variables


# Scatterplot matrix of non-factor variables


pairs(temp, main = "Scatterplot Matrix: Non-factor Variables of 'Auto.csv'")


par(mfcol = c(2, 2))


# Create histograms


for (i in 1:ncol(temp)) {


hist(temp[, i], col = "beige",


main = paste("Histogram of auto$", names(temp)[i], sep = ""),


xlab = paste("auto$", names(temp)[i], sep = ""))


}
par(mfcol = c(1, 1))


par(mfcol = c(2, 2))
```

```
# Create boxplots

for (i in 1:ncol(temp)) {

boxplot(temp[, i], col = "beige",

main = paste("Boxplot of auto$", names(temp)[i], sep = ""),

ylab = paste("auto$", names(temp)[i], sep = ""))

}

par(mfcol = c(1, 1))

# Remove temp

rm(temp)
```

(f) Suppose that we wish to predict gas mileage (mpg) on the basis of the other variables. Do your plots suggest that any of the other variables might be useful in predicting mpg? Justify your answer.

Solution:

```
# Examine correlation between scatterplot variables

sapply(auto[, !sapply(auto, is.factor)], function(x) cor(auto$mpg, x))

auto.rm <- auto[-c(10:85), ]

temp <- NULL
```

```
for (i in 1:ncol(auto)) {

if(is.factor(auto[, i]) == F) {

temp=rbind(temp, data.frame(colnames(auto.rm[i]),

round(min(auto.rm[, i]), digits = 2),

round(max(auto.rm[, i]), digits = 2),

round(mean(auto.rm[, i]), digits = 2),

round(sd(auto.rm[, i]), digits = 2)))

}

}

colnames(temp)=c("Variable", "Min", "Max", "Mean", "Standard Deviation") temp

rm(temp)
```