---

title: "ASSIGNMENT 8"

author: "Rajasekhar Reddy Karna"

date: '2020-10-24'

output:

word\_document: default

html\_document: default

pdf\_document: default

bibliography: bibliography.bib

---

##Include all of your answers in a R Markdown report. Here is an example R Markdown report that you can use as a guide.

##Fit a logistic regression model to the binary-classifier-data.csv dataset from the previous assignment.

##a. What is the accuracy of the logistic regression classifier?

##Output Analysis: Accuracy of 58.3%.

***Code:***

```{r include=FALSE}

options(warn=-1)

library(tidyr)

library(readr)

library(foreign)

library(caTools)

library(class)

library(caret)

setwd("C:/Users/vahin/Documents/GitHub/dsc520/")

classifier\_df <- read.csv("data/binary-classifier-data.csv")

head(classifier\_df)

str(classifier\_df)

summary(classifier\_df)

split<-sample.split(classifier\_df, SplitRatio=0.8)

str(split)

train <- subset(classifier\_df, split="TRUE")

str(train)

test <- subset(classifier\_df, split="FALSE")

str(test)

logistic\_model<-glm(label ~ x + y, data = train, family = "binomial")

summary(logistic\_model)

result <- predict(logistic\_model, test, type="response")

result <- predict(logistic\_model, train, type="response")

confusion\_matrix <- table(Actual\_Value=train$label, Predicted\_Value= result >0.5)

confusion\_matrix

#Accuracy calculation based on confusion matrix

(confusion\_matrix[[1,1]] + confusion\_matrix[[2,2]])/sum(confusion\_matrix)

#KNN implementation

#The value of K is decided as square root of number of observations

sqrt(nrow(train))

# Based on above value we get K = 38 or K = 39 (if we round to nearest integer)

knn.38 <- knn(train=train, test=test, cl=train$label, k=38)

knn.39 <- knn(train=train, test=test, cl=train$label, k=39)

accuracy.38 <- 100 \* sum(test$label == knn.38)/nrow(test)

accuracy.38

accuracy.39 <- 100 \* sum(test$label == knn.39)/nrow(test)

accuracy.39

table(knn.38, test$label)

table(knn.39, test$label)

confusionMatrix(table(knn.39, test$label))

```

***Coding Output:***

> options(warn=-1)

> library(tidyr)

> library(readr)

> library(foreign)

> library(caTools)

> library(class)

> library(caret)

> setwd("C:/Users/vahin/Documents/GitHub/dsc520/")

The working directory was changed to C:/Users/vahin/Documents/GitHub/dsc520 inside a notebook chunk. The working directory will be reset when the chunk is finished running. Use the knitr root.dir option in the setup chunk to change the working directory for notebook chunks.> classifier\_df <- read.csv("data/binary-classifier-data.csv")

> head(classifier\_df)

> str(classifier\_df)

'data.frame': 1498 obs. of 3 variables:

$ label: int 0 0 0 0 0 0 0 0 0 0 ...

$ x : num 70.9 75 73.8 66.4 69.1 ...

$ y : num 83.2 87.9 92.2 81.1 84.5 ...

> summary(classifier\_df)

label x y

Min. :0.000 Min. : -5.20 Min. : -4.019

1st Qu.:0.000 1st Qu.: 19.77 1st Qu.: 21.207

Median :0.000 Median : 41.76 Median : 44.632

Mean :0.488 Mean : 45.07 Mean : 45.011

3rd Qu.:1.000 3rd Qu.: 66.39 3rd Qu.: 68.698

Max. :1.000 Max. :104.58 Max. :106.896

> split<-sample.split(classifier\_df, SplitRatio=0.8)

> str(split)

logi [1:3] FALSE TRUE TRUE

> train <- subset(classifier\_df, split="TRUE")

> str(train)

'data.frame': 1498 obs. of 3 variables:

$ label: int 0 0 0 0 0 0 0 0 0 0 ...

$ x : num 70.9 75 73.8 66.4 69.1 ...

$ y : num 83.2 87.9 92.2 81.1 84.5 ...

> test <- subset(classifier\_df, split="FALSE")

> str(test)

'data.frame': 1498 obs. of 3 variables:

$ label: int 0 0 0 0 0 0 0 0 0 0 ...

$ x : num 70.9 75 73.8 66.4 69.1 ...

$ y : num 83.2 87.9 92.2 81.1 84.5 ...

> logistic\_model<-glm(label ~ x + y, data = train, family = "binomial")

> summary(logistic\_model)

Call:

glm(formula = label ~ x + y, family = "binomial", data = train)

Deviance Residuals:

Min 1Q Median 3Q Max

-1.3728 -1.1697 -0.9575 1.1646 1.3989

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) 0.424809 0.117224 3.624 0.00029 \*\*\*

x -0.002571 0.001823 -1.411 0.15836

y -0.007956 0.001869 -4.257 2.07e-05 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 2075.8 on 1497 degrees of freedom

Residual deviance: 2052.1 on 1495 degrees of freedom

AIC: 2058.1

Number of Fisher Scoring iterations: 4

> result <- predict(logistic\_model, test, type="response")

> result <- predict(logistic\_model, train, type="response")

> confusion\_matrix <- table(Actual\_Value=train$label, Predicted\_Value= result >0.5)

> confusion\_matrix

Predicted\_Value

Actual\_Value FALSE TRUE

0 429 338

1 286 445

> #Accuracy calculation based on confusion matrix

> (confusion\_matrix[[1,1]] + confusion\_matrix[[2,2]])/sum(confusion\_matrix)

[1] 0.5834446

> #KNN implementation

> #The value of K is decided as square root of number of observations

> sqrt(nrow(train))

[1] 38.704

> # Based on above value we get K = 38 or K = 39 (if we round to nearest integer)

> knn.38 <- knn(train=train, test=test, cl=train$label, k=38)

> knn.39 <- knn(train=train, test=test, cl=train$label, k=39)

> accuracy.38 <- 100 \* sum(test$label == knn.38)/nrow(test)

> accuracy.38

[1] 97.46328

> accuracy.39 <- 100 \* sum(test$label == knn.39)/nrow(test)

> accuracy.39

[1] 97.39653

> table(knn.38, test$label)

knn.38 0 1

0 747 18

1 20 713

> table(knn.39, test$label)

knn.39 0 1

0 746 18

1 21 713

> confusionMatrix(table(knn.39, test$label))

Confusion Matrix and Statistics

knn.39 0 1

0 746 18

1 21 713

Accuracy : 0.974

95% CI : (0.9646, 0.9814)

No Information Rate : 0.512

P-Value [Acc > NIR] : <2e-16

Kappa : 0.9479

Mcnemar's Test P-Value : 0.7488

Sensitivity : 0.9726

Specificity : 0.9754

Pos Pred Value : 0.9764

Neg Pred Value : 0.9714

Prevalence : 0.5120

Detection Rate : 0.4980

Detection Prevalence : 0.5100

Balanced Accuracy : 0.9740

'Positive' Class : 0

##b. How does the accuracy of the logistic regression classifier compare to the nearest neighbors algorithm?

##Output Analysis: KNN is showing accuracy of 97.4%.

##c. Why is the accuracy of the logistic regression classifier different from that of the nearest neighbors?

##Output Analysis: Theoritically KNN is non parametric model and logistic regression is parametric model. KNN model showing higher accuracy than logistic regression. KNN tries to predict binary result by indicating outcome as 0 or 1 and LR tries to find the probability of outcome to have values retain between 0 and 1.

## References

Discovering Statistics Using R

Open source like Github and Google searches.