Exercise 14: Fit a Logistic Regression Model to Previous Dataset

Patkhedkar Ninad

2020-10-27

Exercise 14: Fit a Logistic Regression model to Previous Dataset

```
library(knitr)
library(caTools)
library(ggplot2)
library(class)
library(pander)
setwd("/cloud/project")

bcd_df <- read.csv("data/binary-classifier-data.csv")
bcd_df$label <- as.factor(bcd_df$label)
ggplot(data = bcd_df, aes(x = x, y = y, color = label)) + geom_point() +
ggtitle("Binary Classification Data")
```

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

```
sample <- sample.split(bcd_df$label, SplitRatio = 0.80)
training_data = subset(bcd_df, sample == TRUE)
test_data = subset(bcd_df, sample == FALSE)
glm.model = glm(label ~ . , family = binomial(logit), data = training_data)
summary(glm.model)
```

```
##
## Call:
## glm(formula = label ~ ., family = binomial(logit), data = training_data)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.370  -1.170  -0.960   1.166   1.392
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.419027   0.131112   3.196 0.001394 **
## x           -0.002632   0.002031  -1.296 0.194941
## y           -0.007732   0.002082  -3.714 0.000204 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1661.5  on 1198  degrees of freedom
## Residual deviance: 1643.1  on 1196  degrees of freedom
## AIC: 1649.1
##
## Number of Fisher Scoring iterations: 4
```

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

```
test_data$predicted = predict(glm.model, newdata=test_data, type="response")
pander(table(test_data$label, test_data$predicted> 0.5))
```

	FALSE	TRUE
0	84	69
1	50	96

Accuracy is poor.More attributes would have helped to make better predictions.

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

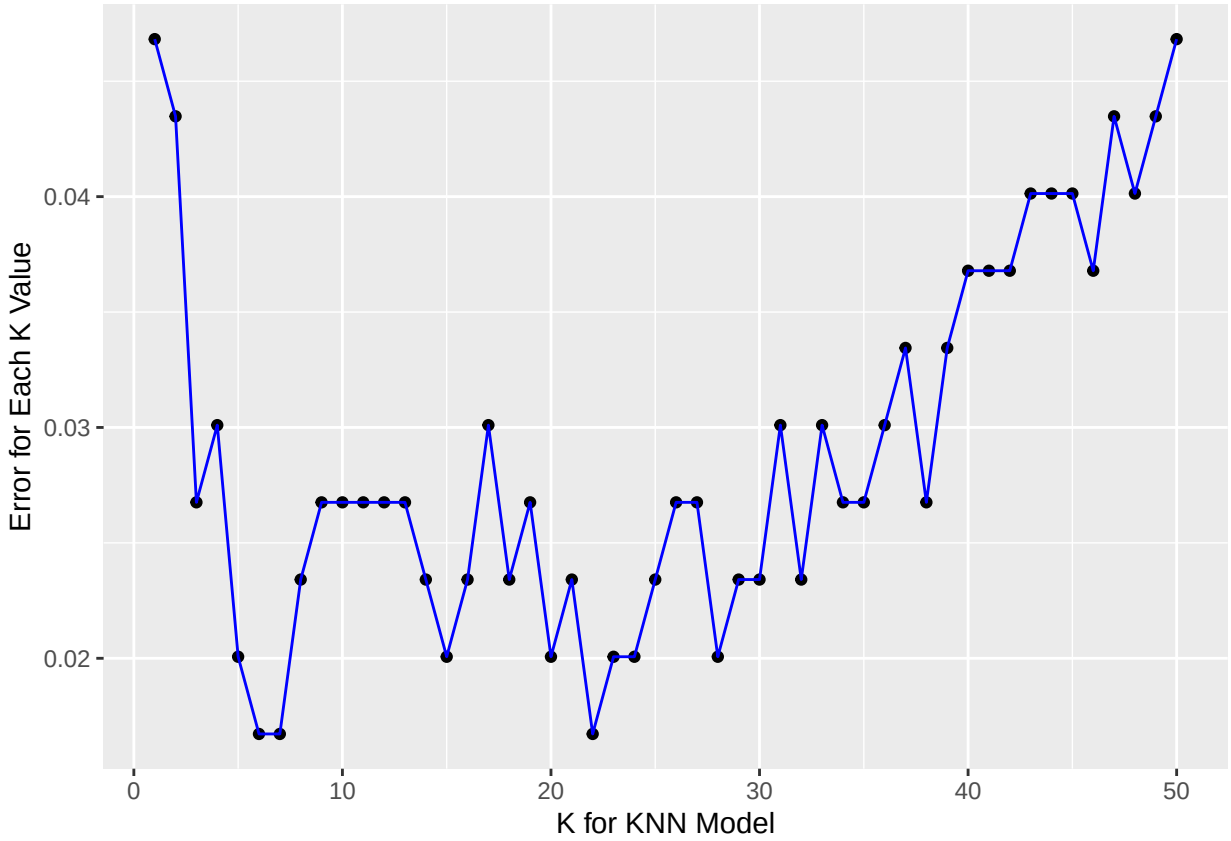
```
knn.model <- knn(training_data[2:3], test_data[2:3], training_data$label, k = 1)
mean(test_data$label != knn.model)
```

```
## [1] 0.04682274
```

```
predicted.values <- NULL
error.rate <- NULL
for(i in 1:50){

  predicted.values <- knn(training_data[2:3],test_data[2:3],training_data$label,k=i)
  error.rate[i] <- mean(test_data$label != predicted.values)
}
```

```
k.values <- 1:50
error.df <- data.frame(error.rate,k.values)
ggplot(error.df,aes(x=k.values,y=error.rate)) +
  geom_point() +
  geom_line(color='blue') +
  xlab("K for KNN Model") +
  ylab("Error for Each K Value")
```

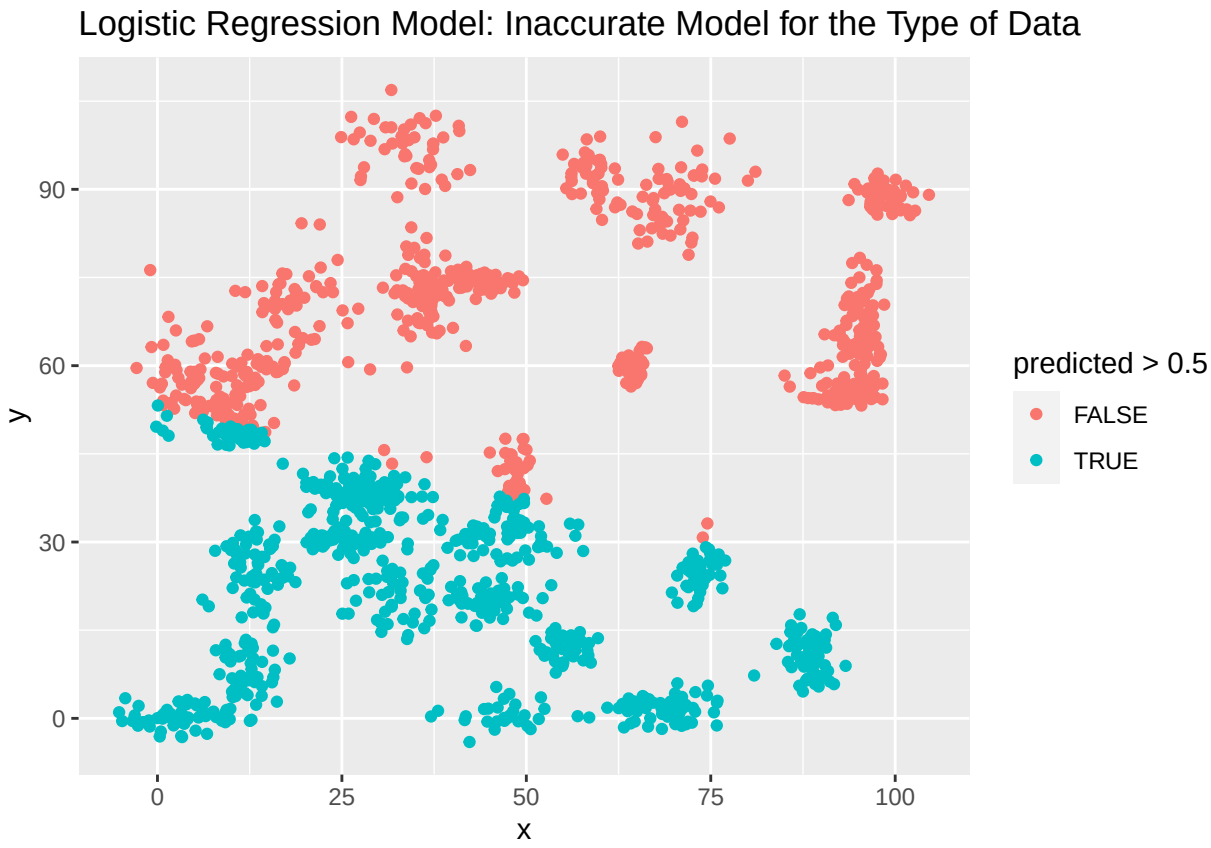


Our KNN model accuracy is much better than logistic regression model.

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

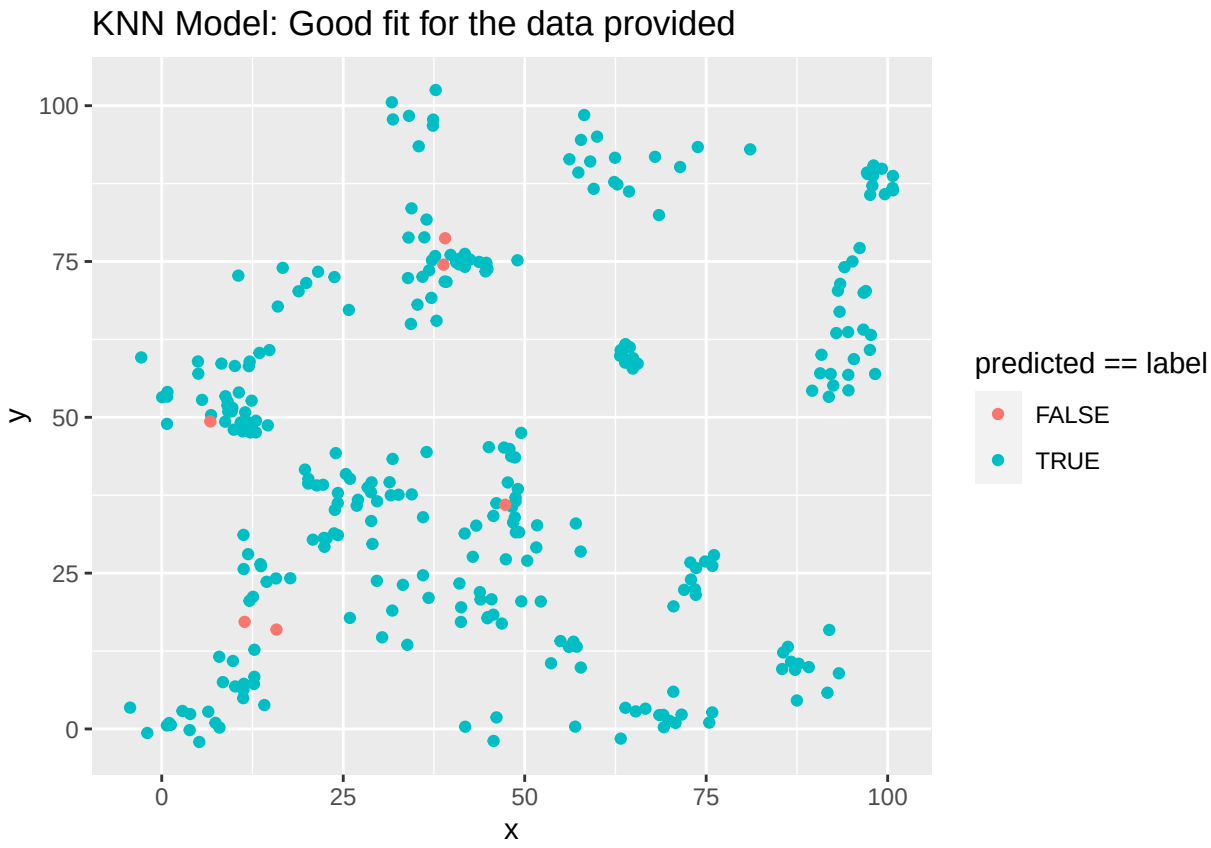
Logistic model only had two inputs, x and y and its difficult to get a line for our model. Plotting our estimates, we can see that the logistic regression attempted to cut a line through our data.

```
glm.model = glm(label ~ . , family = binomial(logit), data = bcd_df)
bcd_df$predicted = predict(glm.model, newdata=bcd_df, type="response")
ggplot(data = bcd_df, aes(x = x, y = y, color = predicted > 0.5)) +
  geom_point() +
  ggtitle("Logistic Regression Model: Inaccurate Model for the Type of Data")
```



Datapoints plot can't be splitted linearly hence clustering model would be more accurate.

```
test_data$predicted <- knn(training_data[2:3], test_data[2:3], training_data$label, k = 5)
ggplot(data = test_data, aes(x = x, y = y, color = predicted == label)) +
  geom_point() +
  ggtitle("KNN Model: Good fit for the data provided")
```


KNN Model: Good fit for the data provided
100
75
50
25
0
y
0
25
50
75
100
x
predicted == label
FALSE
TRUE