Assignment_10.2.2_Fit a Logistic Regression Model

Kausik Chattapadhyay

2022-11-02

Assignment

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

```
## Set the working directory to the root of your DSC 520 directory
setwd("/Users/kausik/desktop/MS Data Science/DSC 520/dsc520-stats-r-assignments")
library(caTools)
```

```
## Warning: package 'caTools' was built under R version 4.0.2
```

```
library('foreign')
```

```
## Warning: package 'foreign' was built under R version 4.0.5
```

```
set.seed(101)
data_df <- read.csv("data/binary-classifier-data.csv")
head(data_df)
```

```
##   label        x        y
## 1     0 70.88469 83.17702
## 2     0 74.97176 87.92922
## 3     0 73.78333 92.20325
## 4     0 66.40747 81.10617
## 5     0 69.07399 84.53739
## 6     0 72.23616 86.38403
```

```
# Split the data into train(80%) and test(20%).
split <- sample.split(data_df, SplitRatio = 0.80)
train <- subset(data_df, split == TRUE)
test <- subset(data_df, split == FALSE)

#logistic regression model with 80% train data
data_glm <- glm(label ~ x + y, data=train, family = binomial)

summary(data_glm)
```

```
##
## Call:
```

```
## glm(formula = label ~ x + y, family = binomial, data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3766  -1.1693  -0.9522   1.1648   1.3896
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.433172   0.143853   3.011 0.002602 **
## x           -0.002722   0.002231  -1.220 0.222475
## y           -0.008017   0.002286  -3.507 0.000453 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1384.3  on 998  degrees of freedom
## Residual deviance: 1368.0  on 996  degrees of freedom
## AIC: 1374
##
## Number of Fisher Scoring iterations: 4
```

```
# Predict the train and test data with model

res_train <- predict(data_glm, train, type="response")
res_test <- predict(data_glm, test, type="response")

# validate the model- confusion matrix

## Train Data confusion Matrix
confusion_mat_train <- table(Actual_Value=train$label,
                             Predicted_Value=res_train >0.5)
confusion_mat_train
```

```
##              Predicted_Value
## Actual_Value FALSE TRUE
##            0   283  229
##            1   190  297
```

```
## Test Data Confusion Matrix
confusion_mat_test <- table(Actual_Value=test$label,
                            Predicted_Value=res_test >0.5)
confusion_mat_test
```

```
##              Predicted_Value
## Actual_Value FALSE TRUE
##            0   142  113
##            1    96  148
```

```
## Train Accuracy
modelAccuracy_train <- (confusion_mat_train[[1,1]] + confusion_mat_train[[2,2]]) / sum(confusion_mat_tra
modelAccuracy_train
```

```
## [1] 0.5805806
```

```
## Test Accuracy
modelAccuracy_test <- (confusion_mat_test[[1,1]] + confusion_mat_test[[2,2]]) / sum(confusion_mat_test)
modelAccuracy_test
```

```
## [1] 0.5811623
```

Question A:

What is the accuracy of the logistic regression classifier?

Answer for A:

```
The accuracy came out to be 58% both for 80% train data and 20% set aside test data.
Accuracy is not good so the logistic regression doesn’t fit well with the data.
```