Assignment: 9.2 Exercise

Raghuwanshi, Prashant

2021-08-06

Question no 1 : Complete assignment09

Fit a Logistic Regression Model to Thoracic Surgery Binary Dataset

For this problem, you will be working with the thoracic surgery data set from the University of California Irvine machine learning repository.

This dataset contains information on life expectancy in lung cancer patients after surgery. The underlying thoracic surgery data is in ARFF format.

This is a text-based format with information on each of the attributes. You can load this data using a package such as foreign or by cutting and pasting the data section into a CSV file.

```
library(ggplot2)
theme_set(theme_minimal())
library(readxl)
### Set the working directory to the root of your DSC 520 directory

setwd("D:/MS_DataScience/DSC 520-Datastatiics.pdf/dsc520")

### Load the `data/ThoraricSurgery.csv` to
th_surgery_df <- read.csv("data/ThoraricSurgery.csv")
str(th_surgery_df)
```

```
## 'data.frame':    470 obs. of  17 variables:
##  $ DGN    : chr  "DGN2" "DGN3" "DGN3" "DGN3" ...
##  $ PRE4   : num  2.88 3.4 2.76 3.68 2.44 2.48 4.36 3.19 3.16 2.32 ...
##  $ PRE5   : num  2.16 1.88 2.08 3.04 0.96 1.88 3.28 2.5 2.64 2.16 ...
##  $ PRE6   : chr  "PRZ1" "PRZ0" "PRZ1" "PRZ0" ...
##  $ PRE7   : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ PRE8   : logi  FALSE FALSE FALSE FALSE TRUE FALSE ...
##  $ PRE9   : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ PRE10  : logi  TRUE FALSE TRUE FALSE TRUE TRUE ...
##  $ PRE11  : logi  TRUE FALSE FALSE FALSE TRUE FALSE ...
##  $ PRE14  : chr  "OC14" "OC12" "OC11" "OC11" ...
##  $ PRE17  : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ PRE19  : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ PRE25  : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ PRE30  : logi  TRUE TRUE TRUE FALSE TRUE FALSE ...
##  $ PRE32  : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ AGE    : int  60 51 59 54 73 51 59 66 68 54 ...
##  $ Risk1Yr: logi  FALSE FALSE FALSE FALSE TRUE FALSE ...
```

```
### Assignment Instructions:
```

Fit a binary logistic regression model to the data set that predicts whether or not the patient survived for one year (the Risk1Y variable) after the surgery.

Use the glm() function to perform the logistic regression. See Generalized Linear Models for an example. Include a summary using the summary() function in your results.

```
logistic_model <- glm(Risk1Yr ~ ., family = binomial(), th_surgery_df)
summary(logistic_model)
```

```
##
## Call:
## glm(formula = Risk1Yr ~ ., family = binomial(), data = th_surgery_df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6084  -0.5439  -0.4199  -0.2762   2.4929
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.655e+01  2.400e+03  -0.007  0.99450
## DGNDGN2      1.474e+01  2.400e+03   0.006  0.99510
## DGNDGN3      1.418e+01  2.400e+03   0.006  0.99528
## DGNDGN4      1.461e+01  2.400e+03   0.006  0.99514
## DGNDGN5      1.638e+01  2.400e+03   0.007  0.99455
## DGNDGN6      4.089e-01  2.673e+03   0.000  0.99988
## DGNDGN8      1.803e+01  2.400e+03   0.008  0.99400
## PRE4        -2.272e-01  1.849e-01  -1.229  0.21909
## PRE5        -3.030e-02  1.786e-02  -1.697  0.08971 .
## PRE6PRZ1    -4.427e-01  5.199e-01  -0.852  0.39448
## PRE6PRZ2    -2.937e-01  7.907e-01  -0.371  0.71030
## PRE7TRUE     7.153e-01  5.556e-01   1.288  0.19788
## PRE8TRUE     1.743e-01  3.892e-01   0.448  0.65419
## PRE9TRUE     1.368e+00  4.868e-01   2.811  0.00494 **
## PRE10TRUE    5.770e-01  4.826e-01   1.196  0.23185
## PRE11TRUE    5.162e-01  3.965e-01   1.302  0.19295
## PRE14OC12    4.394e-01  3.301e-01   1.331  0.18318
## PRE14OC13    1.179e+00  6.165e-01   1.913  0.05580 .
## PRE14OC14    1.653e+00  6.094e-01   2.713  0.00668 **
## PRE17TRUE    9.266e-01  4.445e-01   2.085  0.03709 *
## PRE19TRUE   -1.466e+01  1.654e+03  -0.009  0.99293
## PRE25TRUE   -9.789e-02  1.003e+00  -0.098  0.92227
## PRE30TRUE    1.084e+00  4.990e-01   2.172  0.02984 *
## PRE32TRUE   -1.398e+01  1.645e+03  -0.008  0.99322
## AGE         -9.506e-03  1.810e-02  -0.525  0.59944
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 395.61  on 469  degrees of freedom
## Residual deviance: 341.19  on 445  degrees of freedom
## AIC: 391.19
```

```
##
## Number of Fisher Scoring iterations: 15
```

```
### According to the summary, which variables had the greatest effect on the survival rate?
### ANS seems PRE14 variable is having greatest effect on survival rate
```

To compute the accuracy of your model, use the dataset to predict the outcome variable.

The percent of correct predictions is the accuracy of your model. What is the accuracy of your model?

Subsetting the data and keeping the required variables

```
surgery_df1 <- th_surgery_df[ ,c("Risk1Yr", "PRE14", "PRE17")]
### Checking the dim
dim(surgery_df1)
```

```
## [1] 470   3
```

```
### Converting to factor variables
surgery_df1$Risk1Yr <- as.factor(surgery_df1$Risk1Yr)
surgery_df1$PRE14 <- as.factor(surgery_df1$PRE14)
surgery_df1$PRE17 <- as.factor(surgery_df1$PRE17)
### Loading caret library
require(caret)
```

```
## Loading required package: caret
```

```
## Loading required package: lattice
```

```
### Splitting the data into train and test
index <- createDataPartition(surgery_df1$Risk1Yr, p = .70, list = FALSE)
train <- surgery_df1[index, ]
test <- surgery_df1[-index, ]
### Training the model
logistic_model1 <- glm(Risk1Yr ~ ., family = binomial(), train)
### Checking the model
summary(logistic_model1)
```

```
##
## Call:
## glm(formula = Risk1Yr ~ ., family = binomial(), data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0231  -0.5629  -0.5629  -0.4782   2.1097
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -2.1111     0.2880  -7.330  2.3e-13 ***
## PRE14OC12     0.3490     0.3524   0.990   0.3220
## PRE14OC13     1.4180     0.6767   2.095   0.0361 *
## PRE14OC14     1.3504     0.6779   1.992   0.0464 *
## PRE17TRUE     0.3864     0.5364   0.720   0.4714
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 276.93  on 328  degrees of freedom
## Residual deviance: 269.82  on 324  degrees of freedom
## AIC: 279.82
##
## Number of Fisher Scoring iterations: 4
```

```
### Converting from probability to actual output
train$pred_Risk1Yr <- ifelse(logistic_model1$fitted.values >= 0.5, "False", "True")
### Generating the classification table
ctab_train <- table(train$Risk1Yr, train$pred_Risk1Yr)
### Predicting in the test dataset
pred_prob <- predict(logistic_model1, test, type = "response")
### Generating the classification table
ctab_train <- table(train$Risk1Yr, train$pred_Risk1Yr)
### Converting from probability to actual output
test$pred_Risk1Yr <- ifelse(pred_prob >= 0.5, "False", "True")
# Generating the classification table
ctab_test <- table(test$Risk1Yr, test$pred_Risk1Yr)
ctab_test
```

```
##
##         True
##   FALSE  120
##   TRUE    21
```

Accuracy = (TP + TN)/(TN + FP + FN + TP)

Accuracy in Training dataset

```
accuracy_train <- sum(diag(ctab_train))/sum(ctab_train)*100
accuracy_train
```

```
## [1] 85.10638
```

```
# Accuracy in Test dataset
accuracy_test <- sum(diag(ctab_test))/sum(ctab_test)*100
accuracy_test
```

```
## [1] 85.10638
```

2 Fit a Logistic Regression Model

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

```
library(ggplot2)
theme_set(theme_minimal())
library(readxl)
### Set the working directory to the root of your DSC 520 directory

setwd("D:/MS_DataScience/DSC 520-Datastatiics.pdf/dsc520")

### Load the `data/ThoraricSurgery.csv` to
```

```r
binary_df <- read.csv("data/binary-classifier-data.csv")
str(binary_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 ...
```

```r
require(caret)
### Splitting the data into train and test
index_bin <- createDataPartition(binary_df$label, p = .70, list = FALSE)
train_bin <- binary_df[index, ]
test_bin <- binary_df[-index, ]
### Training the model
logistic_model_bin <- glm(label ~ x + y, family = binomial(), train_bin)
```

```
## Warning: glm.fit: algorithm did not converge
```

```r
### Checking the model
summary(logistic_model_bin)
```

```
##
## Call:
## glm(formula = label ~ x + y, family = binomial(), data = train_bin)
##
## Deviance Residuals:
##        Min          1Q      Median          3Q         Max
## -2.409e-06  -2.409e-06  -2.409e-06  -2.409e-06  -2.409e-06
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.657e+01  5.469e+04       0        1
## x            4.761e-16  6.922e+02       0        1
## y            8.362e-16  7.251e+02       0        1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 0.0000e+00  on 328  degrees of freedom
## Residual deviance: 1.9087e-09  on 326  degrees of freedom
## AIC: 6
##
## Number of Fisher Scoring iterations: 25
```

```r
### Converting from probability to actual output
train_bin$pred_label <- ifelse(logistic_model_bin$fitted.values >= 0.5, 0, 1)
### Generating the classification table
ctab_train_bin <- table(train_bin$label, train_bin$pred_label)
### Predicting in the test dataset
pred_prob_bin <- predict(logistic_model_bin, test_bin, type = "response")
### Converting from probability to actual output
### Accuracy = (TP + TN)/(TN + FP + FN + TP)
### Accuracy in Training dataset
accuracy_train_bin <- sum(diag(ctab_train_bin))/sum(ctab_train_bin)*100
accuracy_train_bin
```

```
## [1] 100
```

```
### Keep this assignment handy, as you will be comparing your results from this week to next week
```