Exercise 13: Fit a Logistic Regression Model to the Thoracic Surgery Binary DataSet

Patkhedkar Ninad

2020-10-27

Exercise 13: Fit a Logistic Regression Model to the Thoracic Surgery Binary Data

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.

a. 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.

```
library(knitr)
library(foreign)
library(caTools)
library(pander)

setwd("/cloud/project")
patient_df <- read.arff("data/ThoraricSurgery.arff")
head(patient_df)
```

```
##    DGN PRE4 PRE5 PRE6 PRE7 PRE8 PRE9 PRE10 PRE11 PRE14 PRE17 PRE19 PRE25 PRE30
## 1 DGN2 2.88 2.16 PRZ1    F    F    F     T     T  OC14     F     F     F     T
## 2 DGN3 3.40 1.88 PRZ0    F    F    F     F     F  OC12     F     F     F     T
## 3 DGN3 2.76 2.08 PRZ1    F    F    F     T     F  OC11     F     F     F     T
## 4 DGN3 3.68 3.04 PRZ0    F    F    F     F     F  OC11     F     F     F     F
## 5 DGN3 2.44 0.96 PRZ2    F    T    F     T     T  OC11     F     F     F     T
## 6 DGN3 2.48 1.88 PRZ1    F    F    F     T     F  OC11     F     F     F     F
##   PRE32 AGE Risk1Yr
## 1     F  60       F
## 2     F  51       F
## 3     F  59       F
## 4     F  54       F
## 5     F  73       T
## 6     F  51       F
```

```
sample <- sample.split(patient_df$Risk1Yr, SplitRatio = 0.70)
training_data = subset(patient_df, sample == TRUE)
```

```r
test_data = subset(patient_df, sample == FALSE)
model = glm(Risk1Yr ~ . -1 , family = binomial(logit), data = training_data)
model <- step(model, trace=FALSE);
summary(model)
```

```
##
## Call:
## glm(formula = Risk1Yr ~ DGN + PRE5 + PRE9 + PRE10 + PRE14 + PRE17 +
##     PRE30 - 1, family = binomial(logit), data = training_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7840  -0.5034  -0.4497  -0.2684   2.8028
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## DGNDGN1    -16.80536 1455.39770  -0.012  0.99079
## DGNDGN2     -3.36213    0.78194  -4.300 1.71e-05 ***
## DGNDGN3     -4.08725    0.70720  -5.779 7.49e-09 ***
## DGNDGN4     -3.44182    0.86963  -3.958 7.56e-05 ***
## DGNDGN5     -2.09194    0.89963  -2.325  0.02005 *
## DGNDGN6    -17.18461 1007.00553  -0.017  0.98638
## DGNDGN8     -0.02312    1.43103  -0.016  0.98711
## PRE5        -0.02530    0.01775  -1.425  0.15405
## PRE9T        1.71718    0.55069   3.118  0.00182 **
## PRE10T       0.82911    0.47371   1.750  0.08007 .
## PRE14OC12    0.23096    0.39039   0.592  0.55411
## PRE14OC13    1.88403    0.76632   2.459  0.01395 *
## PRE14OC14    1.15932    0.68601   1.690  0.09104 .
## PRE17T       1.10254    0.47107   2.340  0.01926 *
## PRE30T       1.07919    0.57696   1.870  0.06142 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 456.09  on 329  degrees of freedom
## Residual deviance: 233.46  on 314  degrees of freedom
## AIC: 263.46
##
## Number of Fisher Scoring iterations: 14
```

b. According to the summary, which variables had the greatest effect on the survival rate?

DGNDGN1
DGNDGN2
DGNDGN3
DGNDGN4
DGNDGN5
DGNDGN6
DGNDGN8
PRE5
PRE9T
PRE14OC12
PRE14OC13

PRE14OC14
PRE30T

c. 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?

```
library(pander)
test_data$predicted = predict(model, newdata=test_data, type="response")
pander(table(test_data$Risk1Yr, test_data$predicted> 0.5))
```

	FALSE	TRUE
F	116	4
T	20	1

The logistic model looks fine for predicting False values but not great for true values.