8.1 Assignment: Fit a Logistic Regression Model to the Thoracic Surgery Binary Dataset

Dan Siegel

July 22, 2018

# Assignment 8.1

## 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.

survived <- glm(Risk1Yr ~ DGN + PRE4 + PRE5 + PRE6 + PRE7 + PRE8 +   
 PRE9 + PRE10 + PRE11 + PRE14 + PRE17 + PRE19 +   
 PRE25 + PRE30 + PRE32 + AGE, data=surg)  
summary(survived)

##   
## Call:  
## glm(formula = Risk1Yr ~ DGN + PRE4 + PRE5 + PRE6 + PRE7 + PRE8 +   
## PRE9 + PRE10 + PRE11 + PRE14 + PRE17 + PRE19 + PRE25 + PRE30 +   
## PRE32 + AGE, data = surg)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -0.57641 -0.15677 -0.09273 -0.02625 0.96916   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.036968 0.381981 0.097 0.92294   
## DGNDGN2 0.157202 0.348148 0.452 0.65182   
## DGNDGN3 0.091518 0.344957 0.265 0.79090   
## DGNDGN4 0.147371 0.347883 0.424 0.67205   
## DGNDGN5 0.449192 0.356291 1.261 0.20806   
## DGNDGN6 0.044436 0.385765 0.115 0.90835   
## DGNDGN8 0.631123 0.422684 1.493 0.13611   
## PRE4 -0.024605 0.019695 -1.249 0.21222   
## PRE5 -0.002921 0.001466 -1.993 0.04689 \*   
## PRE6PRZ1 -0.047866 0.056115 -0.853 0.39412   
## PRE6PRZ2 -0.008813 0.098215 -0.090 0.92854   
## PRE7 0.068320 0.070272 0.972 0.33147   
## PRE8 0.020634 0.048506 0.425 0.67076   
## PRE9 0.194810 0.069078 2.820 0.00501 \*\*  
## PRE10 0.054956 0.052737 1.042 0.29795   
## PRE11 0.054024 0.050393 1.072 0.28428   
## PRE14OC12 0.041587 0.034555 1.204 0.22941   
## PRE14OC13 0.161951 0.086277 1.877 0.06116 .   
## PRE14OC14 0.270389 0.090386 2.991 0.00293 \*\*  
## PRE17 0.130185 0.061687 2.110 0.03538 \*   
## PRE19 -0.126556 0.244238 -0.518 0.60460   
## PRE25 0.012465 0.125005 0.100 0.92062   
## PRE30 0.093934 0.043322 2.168 0.03067 \*   
## PRE32 -0.065692 0.244281 -0.269 0.78812   
## AGE -0.001142 0.002049 -0.557 0.57751   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for gaussian family taken to be 0.1173851)  
##   
## Null deviance: 59.574 on 469 degrees of freedom  
## Residual deviance: 52.236 on 445 degrees of freedom  
## AIC: 353.23  
##   
## Number of Fisher Scoring iterations: 2

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

DGN8, pre9, pre14/Oc14

## 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?

set.seed(123)  
n<- nrow(surg)  
shuffled <- surg[sample(n),]  
train <- shuffled[1:round(0.7 \* n),]  
test <- shuffled[(round(0.7 \* n) + 1):n,]  
testing\_survived <- glm(Risk1Yr ~ DGN + PRE4 + PRE5 + PRE6 + PRE7 + PRE8 +   
 PRE9 + PRE10 + PRE11 + PRE14 + PRE17 + PRE19 +   
 PRE25 + PRE30 + PRE32 + AGE, data=train)  
predicted\_data <- predict(testing\_survived, newdata = test)  
x<- as.integer(predicted\_data)  
y <- test$Risk1Yr  
l <- union(x, y)  
Table2 <- table(factor(x, l), factor(y, l))

## Confusion Matrix and Statistics  
##   
##   
## 0 1  
## 0 122 19  
## 1 0 0  
##   
## Accuracy : 0.8652   
## 95% CI : (0.7976, 0.9169)  
## No Information Rate : 0.8652   
## P-Value [Acc > NIR] : 0.5608   
##   
## Kappa : 0   
## Mcnemar's Test P-Value : 3.636e-05   
##   
## Sensitivity : 1.0000   
## Specificity : 0.0000   
## Pos Pred Value : 0.8652   
## Neg Pred Value : NaN   
## Prevalence : 0.8652   
## Detection Rate : 0.8652   
## Detection Prevalence : 1.0000   
## Balanced Accuracy : 0.5000   
##   
## 'Positive' Class : 0   
##