---

title: "ASSIGNMENT 8"

author: "Rajasekhar Reddy Karna"

date: '2020-10-24'

output:

word\_document: default

html\_document: default

pdf\_document: default

bibliography: bibliography.bib

---

##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.

##Assignment Instructions:

##Include all of your answers in a R Markdown report. Here is an example R Markdown report that you can use as a guide.

##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.

***Code:***

```{r include=FALSE}

library(ggplot2)

library(readxl)

options(warn=-1)

library(tidyr)

library(readr)

library(foreign)

library(caTools)

setwd("C:/Users/vahin/Documents/GitHub/dsc520/")

thoracic\_surgery\_df <- read.arff("data/ThoraricSurgery.arff")

head(thoracic\_surgery\_df)

str(thoracic\_surgery\_df)

split<-sample.split(thoracic\_surgery\_df, SplitRatio=0.8)

str(split)

Split\_True <- subset(thoracic\_surgery\_df, split="TRUE")

str(Split\_True)

Split\_False <- subset(thoracic\_surgery\_df, split="FALSE")

str(Split\_False)

regression\_all\_variables<-glm(Risk1Yr ~ DGN + PRE4 + PRE5 + PRE6 + PRE7 + PRE8 + PRE9 + PRE10 +PRE14+ PRE11 + PRE17 + PRE19 + PRE25 + PRE30 + PRE32 + AGE, data = train, family = "binomial")

summary(regression\_all\_variables)

exp(regression\_all\_variables$coefficients)

regression\_selected\_variables<-glm(Risk1Yr ~ DGN + PRE5 + PRE9 + PRE11 + PRE14+ PRE17 + PRE30, data = train, family = "binomial")

summary(regression\_selected\_variables)

```

**Code Output:**

> library(ggplot2)

> library(readxl)

> options(warn=-1)

> library(tidyr)

> library(readr)

> library(foreign)

> library(caTools)

> setwd("C:/Users/vahin/Documents/GitHub/dsc520/")

The working directory was changed to C:/Users/vahin/Documents/GitHub/dsc520 inside a notebook chunk. The working directory will be reset when the chunk is finished running. Use the knitr root.dir option in the setup chunk to change the working directory for notebook chunks.> thoracic\_surgery\_df <- read.arff("data/ThoraricSurgery.arff")

> head(thoracic\_surgery\_df)

> str(thoracic\_surgery\_df)

'data.frame': 470 obs. of 17 variables:

$ DGN : Factor w/ 7 levels "DGN1","DGN2",..: 2 3 3 3 3 3 3 2 3 3 ...

$ 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 : Factor w/ 3 levels "PRZ0","PRZ1",..: 2 1 2 1 3 2 2 2 3 2 ...

$ PRE7 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 1 1 1 ...

$ PRE8 : Factor w/ 2 levels "F","T": 1 1 1 1 2 1 1 1 1 1 ...

$ PRE9 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 1 1 1 ...

$ PRE10 : Factor w/ 2 levels "F","T": 2 1 2 1 2 2 2 2 2 2 ...

$ PRE11 : Factor w/ 2 levels "F","T": 2 1 1 1 2 1 1 1 2 1 ...

$ PRE14 : Factor w/ 4 levels "OC11","OC12",..: 4 2 1 1 1 1 2 1 1 1 ...

$ PRE17 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 2 1 1 1 ...

$ PRE19 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 1 1 1 ...

$ PRE25 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 2 1 1 ...

$ PRE30 : Factor w/ 2 levels "F","T": 2 2 2 1 2 1 2 2 2 2 ...

$ PRE32 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 1 1 1 ...

$ AGE : num 60 51 59 54 73 51 59 66 68 54 ...

$ Risk1Yr: Factor w/ 2 levels "F","T": 1 1 1 1 2 1 2 2 1 1 ...

>

> split<-sample.split(thoracic\_surgery\_df, SplitRatio=0.8)

> str(split)

logi [1:17] TRUE FALSE TRUE TRUE TRUE TRUE ...

> Split\_True <- subset(thoracic\_surgery\_df, split="TRUE")

> str(Split\_True)

'data.frame': 470 obs. of 17 variables:

$ DGN : Factor w/ 7 levels "DGN1","DGN2",..: 2 3 3 3 3 3 3 2 3 3 ...

$ 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 : Factor w/ 3 levels "PRZ0","PRZ1",..: 2 1 2 1 3 2 2 2 3 2 ...

$ PRE7 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 1 1 1 ...

$ PRE8 : Factor w/ 2 levels "F","T": 1 1 1 1 2 1 1 1 1 1 ...

$ PRE9 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 1 1 1 ...

$ PRE10 : Factor w/ 2 levels "F","T": 2 1 2 1 2 2 2 2 2 2 ...

$ PRE11 : Factor w/ 2 levels "F","T": 2 1 1 1 2 1 1 1 2 1 ...

$ PRE14 : Factor w/ 4 levels "OC11","OC12",..: 4 2 1 1 1 1 2 1 1 1 ...

$ PRE17 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 2 1 1 1 ...

$ PRE19 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 1 1 1 ...

$ PRE25 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 2 1 1 ...

$ PRE30 : Factor w/ 2 levels "F","T": 2 2 2 1 2 1 2 2 2 2 ...

$ PRE32 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 1 1 1 ...

$ AGE : num 60 51 59 54 73 51 59 66 68 54 ...

$ Risk1Yr: Factor w/ 2 levels "F","T": 1 1 1 1 2 1 2 2 1 1 ...

> Split\_False <- subset(thoracic\_surgery\_df, split="FALSE")

> str(Split\_False)

'data.frame': 470 obs. of 17 variables:

$ DGN : Factor w/ 7 levels "DGN1","DGN2",..: 2 3 3 3 3 3 3 2 3 3 ...

$ 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 : Factor w/ 3 levels "PRZ0","PRZ1",..: 2 1 2 1 3 2 2 2 3 2 ...

$ PRE7 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 1 1 1 ...

$ PRE8 : Factor w/ 2 levels "F","T": 1 1 1 1 2 1 1 1 1 1 ...

$ PRE9 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 1 1 1 ...

$ PRE10 : Factor w/ 2 levels "F","T": 2 1 2 1 2 2 2 2 2 2 ...

$ PRE11 : Factor w/ 2 levels "F","T": 2 1 1 1 2 1 1 1 2 1 ...

$ PRE14 : Factor w/ 4 levels "OC11","OC12",..: 4 2 1 1 1 1 2 1 1 1 ...

$ PRE17 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 2 1 1 1 ...

$ PRE19 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 1 1 1 ...

$ PRE25 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 2 1 1 ...

$ PRE30 : Factor w/ 2 levels "F","T": 2 2 2 1 2 1 2 2 2 2 ...

$ PRE32 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 1 1 1 ...

$ AGE : num 60 51 59 54 73 51 59 66 68 54 ...

$ Risk1Yr: Factor w/ 2 levels "F","T": 1 1 1 1 2 1 2 2 1 1 ...

> regression\_all\_variables<-glm(Risk1Yr ~ DGN + PRE4 + PRE5 + PRE6 + PRE7 + PRE8 + PRE9 + PRE10 +PRE14+ PRE11 + PRE17 + PRE19 + PRE25 + PRE30 + PRE32 + AGE, data = train, family = "binomial")

> summary(regression\_all\_variables)

Call:

glm(formula = Risk1Yr ~ DGN + PRE4 + PRE5 + PRE6 + PRE7 + PRE8 +

PRE9 + PRE10 + PRE14 + PRE11 + PRE17 + PRE19 + PRE25 + PRE30 +

PRE32 + AGE, family = "binomial", data = train)

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

PRE7T 7.153e-01 5.556e-01 1.288 0.19788

PRE8T 1.743e-01 3.892e-01 0.448 0.65419

PRE9T 1.368e+00 4.868e-01 2.811 0.00494 \*\*

PRE10T 5.770e-01 4.826e-01 1.196 0.23185

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 \*\*

PRE11T 5.162e-01 3.965e-01 1.302 0.19295

PRE17T 9.266e-01 4.445e-01 2.085 0.03709 \*

PRE19T -1.466e+01 1.654e+03 -0.009 0.99293

PRE25T -9.789e-02 1.003e+00 -0.098 0.92227

PRE30T 1.084e+00 4.990e-01 2.172 0.02984 \*

PRE32T -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

> exp(regression\_all\_variables$coefficients)

(Intercept) DGNDGN2 DGNDGN3 DGNDGN4 DGNDGN5

6.481698e-08 2.511211e+06 1.440574e+06 2.209615e+06 1.301120e+07

DGNDGN6 DGNDGN8 PRE4 PRE5 PRE6PRZ1

1.505091e+00 6.785355e+07 7.967257e-01 9.701510e-01 6.422903e-01

PRE6PRZ2 PRE7T PRE8T PRE9T PRE10T

7.454996e-01 2.044884e+00 1.190456e+00 3.928338e+00 1.780613e+00

PRE14OC12 PRE14OC13 PRE14OC14 PRE11T PRE17T

1.551720e+00 3.251796e+00 5.222483e+00 1.675616e+00 2.525890e+00

PRE19T PRE25T PRE30T PRE32T AGE

4.317676e-07 9.067446e-01 2.956473e+00 8.455364e-07 9.905394e-01

> regression\_selected\_variables<-glm(Risk1Yr ~ DGN + PRE5 + PRE9 + PRE11 + PRE14+ PRE17 + PRE30, data = train, family = "binomial")

> summary(regression\_selected\_variables)

Call:

glm(formula = Risk1Yr ~ DGN + PRE5 + PRE9 + PRE11 + PRE14 + PRE17 +

PRE30, family = "binomial", data = train)

Deviance Residuals:

Min 1Q Median 3Q Max

-1.4667 -0.5583 -0.4617 -0.2863 2.5340

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) -17.05284 1455.39766 -0.012 0.99065

DGNDGN2 13.98984 1455.39759 0.010 0.99233

DGNDGN3 13.47962 1455.39755 0.009 0.99261

DGNDGN4 13.82213 1455.39761 0.009 0.99242

DGNDGN5 15.63840 1455.39766 0.011 0.99143

DGNDGN6 0.45620 1623.40830 0.000 0.99978

DGNDGN8 16.91476 1455.39832 0.012 0.99073

PRE5 -0.02428 0.01731 -1.403 0.16059

PRE9T 1.35551 0.46854 2.893 0.00382 \*\*

PRE11T 0.50303 0.33762 1.490 0.13624

PRE14OC12 0.45340 0.32471 1.396 0.16261

PRE14OC13 1.31605 0.60232 2.185 0.02889 \*

PRE14OC14 1.77128 0.59355 2.984 0.00284 \*\*

PRE17T 0.98455 0.43089 2.285 0.02232 \*

PRE30T 1.10136 0.49490 2.225 0.02605 \*

---

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: 346.61 on 455 degrees of freedom

AIC: 376.61

Number of Fisher Scoring iterations: 14

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

##Output Analysis: Seems PRE9 has highest P-value with positive correlation and it is having highest impact on the model.

***Code:***

```{r include=FALSE}

#Calculating accuracy for model with all variables

result <- predict(regression\_all\_variables, Split\_False, type="response")

result <- predict(regression\_all\_variables, Split\_True, type="response")

confusion\_matrix <- table(Actual\_Value=Split\_True$Risk1Yr, Predicted\_Value= result >0.5)

confusion\_matrix

#Accuracy calculation based on confusion matrix

accuracy = (confusion\_matrix[[1,1]] + confusion\_matrix[[2,2]])/sum(confusion\_matrix) \* 100

accuracy

#Calculating accuracy for the

result <- predict(regression\_selected\_variables, Split\_False, type="response")

result <- predict(regression\_selected\_variables, Split\_True, type="response")

confusion\_matrix <- table(Actual\_Value=Split\_True$Risk1Yr, Predicted\_Value= result >0.5)

confusion\_matrix

#Accuracy calculation based on confusion matrix

accuracy = (confusion\_matrix[[1,1]] + confusion\_matrix[[2,2]])/sum(confusion\_matrix) \* 100

accuracy

```

***Code Output:***

> #Calculating accuracy for model with all variables

> result <- predict(regression\_all\_variables, Split\_False, type="response")

> result <- predict(regression\_all\_variables, Split\_True, type="response")

> confusion\_matrix <- table(Actual\_Value=Split\_True$Risk1Yr, Predicted\_Value= result >0.5)

> confusion\_matrix

Predicted\_Value

Actual\_Value FALSE TRUE

F 390 10

T 67 3

> #Accuracy calculation based on confusion matrix

> accuracy = (confusion\_matrix[[1,1]] + confusion\_matrix[[2,2]])/sum(confusion\_matrix) \* 100

> accuracy

[1] 83.61702

> #Calculating accuracy for the

> result <- predict(regression\_selected\_variables, Split\_False, type="response")

> result <- predict(regression\_selected\_variables, Split\_True, type="response")

> confusion\_matrix <- table(Actual\_Value=Split\_True$Risk1Yr, Predicted\_Value= result >0.5)

> confusion\_matrix

Predicted\_Value

Actual\_Value FALSE TRUE

F 390 10

T 64 6

> #Accuracy calculation based on confusion matrix

> accuracy = (confusion\_matrix[[1,1]] + confusion\_matrix[[2,2]])/sum(confusion\_matrix) \* 100

> accuracy

[1] 84.25532

##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?

##Output Analysis: As per confusion matrix and accuracy calculation for both the models the best fit model has increased model accuracy by 84.25 - 83.61 = 0.64%.