

```

## read in data required
mydata <- read.csv("https://stats.idre.ucla.edu/stat/data/binary.csv")

## view the first few rows of the data
head(mydata)

##   admit gre gpa rank
## 1     0 380 3.61    3
## 2     1 660 3.67    3
## 3     1 800 4.00    1
## 4     1 640 3.19    4
## 5     0 520 2.93    4
## 6     1 760 3.00    2

summary(mydata)

##      admit           gre            gpa          rank
## Min.   :0.0000   Min.   :220.0   Min.   :2.260   Min.   :1.000
## 1st Qu.:0.0000  1st Qu.:520.0   1st Qu.:3.130   1st Qu.:2.000
## Median :0.0000  Median :580.0   Median :3.395   Median :2.000
## Mean   :0.3175  Mean   :587.7   Mean   :3.390   Mean   :2.485
## 3rd Qu.:1.0000  3rd Qu.:660.0   3rd Qu.:3.670   3rd Qu.:3.000
## Max.   :1.0000  Max.   :800.0   Max.   :4.000   Max.   :4.000

## split training set and testing test
train_sub = sample(nrow(mydata), 0.75*nrow(mydata))

train_data = mydata[train_sub,]
test_data = mydata[-train_sub,]

## Build logistic regression model
admission_logistic <- glm(admit ~ gre + gpa + rank, data = train_data, family = "binomial")

summary(admission_logistic)

## 
## Call:
## glm(formula = admit ~ gre + gpa + rank, family = "binomial",
##      data = train_data)
## 
## Deviance Residuals:
##    Min      1Q      Median      3Q      Max 
## -1.4644 -0.8976 -0.6364  1.1436  2.1198 
## 
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)    
## (Intercept) -3.632240  1.316228 -2.760 0.005788 ** 
## gre          0.001695  0.001295  1.309 0.190385    
## gpa          0.904206  0.380442  2.377 0.017467 *  
## rank         -0.518033  0.140814 -3.679 0.000234 *** 
## ---                                                 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

##  

## (Dispersion parameter for binomial family taken to be 1)  

##  

## Null deviance: 376.12 on 299 degrees of freedom  

## Residual deviance: 346.66 on 296 degrees of freedom  

## AIC: 354.66  

##  

## Number of Fisher Scoring iterations: 3

```

## ROC curve and AUC value

```

## make predictions for test set
fitted.results <- predict(admission_logistic,newdata=subset(test_data,type='response'))
fitted.results <- ifelse(fitted.results > 0.5,1,0)
misClasificError <- mean(fitted.results != test_data$admit)
print(paste('Accuracy',1-misClasificError))

```

```
## [1] "Accuracy 0.68"
```

```
#output confusion matrix
table(test_data$admit,fitted.results,dnn=c("true value","prediction value"))
```

```

##           prediction value
## true value  0   1
##             0 68  1
##             1 31  0

```

```
#plot ROC curve
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```

##  

## Attaching package: 'pROC'
```

```

## The following objects are masked from 'package:stats':  

##  

## cov, smooth, var
```

```
logistic_roc <- roc(test_data$admit,fitted.results)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
plot(logistic_roc, print.auc=TRUE, auc.polygon=TRUE, grid=c(0.1, 0.2),grid.col=c("green", "red"), max.a
```

## Logistic Regression ROC Curve

