

# R Notebook

```
# Installing the package  
install.packages("dplyr")
```

```
## Installing package into '/opt/R-3.5.3'  
## (as 'lib' is unspecified)  
  
## Warning in install.packages("dplyr"): installation of package 'dplyr' had  
## non-zero exit status
```

```
# For Logistic regression  
install.packages("caTools")
```

```
## Installing package into '/opt/R-3.5.3'  
## (as 'lib' is unspecified)  
  
## Warning in install.packages("caTools"): installation of package 'caTools' had  
## non-zero exit status
```

```
# For ROC curve to evaluate model  
install.packages("ROCR")
```

```
## Installing package into '/opt/R-3.5.3'  
## (as 'lib' is unspecified)  
  
## Warning in install.packages("ROCR"): installation of package 'ROCR' had non-zero  
## exit status
```

```
# Installing the package
```

```
# For Logistic regression  
install.packages("caTools")
```

```
## Installing package into '/opt/R-3.5.3'  
## (as 'lib' is unspecified)  
  
## Warning in install.packages("caTools"): installation of package 'caTools' had  
## non-zero exit status
```

```
# For ROC curve to evaluate model  
install.packages("ROCR")
```

```
## Installing package into '/opt/R-3.5.3'  
## (as 'lib' is unspecified)  
  
## Warning in install.packages("ROCR"): installation of package 'ROCR' had non-zero  
## exit status
```

```
# Loading package  
library(dplyr)
```

```
##  
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union

# Summary of dataset in package
summary(mtcars)

##           mpg           cyl           disp           hp
##  Min.    :10.40   Min.    :4.000   Min.    : 71.1   Min.    : 52.0
## 1st Qu.:15.43   1st Qu.:4.000   1st Qu.:120.8   1st Qu.: 96.5
## Median :19.20   Median :6.000   Median :196.3   Median :123.0
## Mean   :20.09   Mean    :6.188   Mean    :230.7   Mean    :146.7
## 3rd Qu.:22.80   3rd Qu.:8.000   3rd Qu.:326.0   3rd Qu.:180.0
## Max.   :33.90   Max.    :8.000   Max.    :472.0   Max.    :335.0
##           drat           wt           qsec           vs
##  Min.    :2.760   Min.    :1.513   Min.    :14.50   Min.    :0.0000
## 1st Qu.:3.080   1st Qu.:2.581   1st Qu.:16.89   1st Qu.:0.0000
## Median :3.695   Median :3.325   Median :17.71   Median :0.0000
## Mean   :3.597   Mean    :3.217   Mean    :17.85   Mean    :0.4375
## 3rd Qu.:3.920   3rd Qu.:3.610   3rd Qu.:18.90   3rd Qu.:1.0000
## Max.   :4.930   Max.    :5.424   Max.    :22.90   Max.    :1.0000
##           am           gear           carb
##  Min.    :0.0000   Min.    :3.000   Min.    :1.000
## 1st Qu.:0.0000   1st Qu.:3.000   1st Qu.:2.000
## Median :0.0000   Median :4.000   Median :2.000
## Mean   :0.4062   Mean    :3.688   Mean    :2.812
## 3rd Qu.:1.0000   3rd Qu.:4.000   3rd Qu.:4.000
## Max.   :1.0000   Max.    :5.000   Max.    :8.000

# Loading package
library(caTools)
library(ROCR)

# Loading package
library(caTools)
library(ROCR)

# Splitting dataset
split <- sample.split(mtcars, SplitRatio = 0.8)
split

## [1] FALSE TRUE FALSE TRUE TRUE TRUE TRUE FALSE TRUE TRUE TRUE

train_reg <- subset(mtcars, split == "TRUE")
test_reg <- subset(mtcars, split == "FALSE")

# Training model
logistic_model <- glm(vs ~ wt + disp,
                      data = train_reg,
                      family = "binomial")

logistic_model
```

```
##
## Call: glm(formula = vs ~ wt + disp, family = "binomial", data = train_reg)
##
## Coefficients:
## (Intercept)          wt          disp
##      1.2207      1.9727     -0.0344
##
## Degrees of Freedom: 22 Total (i.e. Null);  20 Residual
## Null Deviance:      31.84
## Residual Deviance: 13.93    AIC: 19.93
```

#### *# Summary*

```
summary(logistic_model)
```

```
##
## Call:
## glm(formula = vs ~ wt + disp, family = "binomial", data = train_reg)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.7949  -0.2682  -0.1373   0.4412   1.7611
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.22066    2.97912   0.410  0.6820
## wt          1.97271    1.79416   1.100  0.2715
## disp        -0.03440    0.01639  -2.099  0.0358 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 31.841  on 22  degrees of freedom
## Residual deviance: 13.929  on 20  degrees of freedom
## AIC: 19.929
##
## Number of Fisher Scoring iterations: 6
```

```
predict_reg <- predict(logistic_model,
                      test_reg, type = "response")
predict_reg
```

```
##      Mazda RX4      Datsun 710      Merc 240D      Merc 450SE
##      0.70787408      0.88915062      0.92179968      0.44076319
##      Merc 450SLC      Honda Civic      AMC Javelin Pontiac Firebird
##      0.30785813      0.85844918      0.07865001      0.00700244
##      Ferrari Dino
##      0.84514197
```

#### *# Changing probabilities*

```
predict_reg <- ifelse(predict_reg > 0.5, 1, 0)
```

#### *# Evaluating model accuracy*

##### *# using confusion matrix*

```
table(test_reg$vs, predict_reg)
```

```
## predict_reg
## 0 1
## 0 4 2
## 1 0 3
```

```
missing_classerr <- mean(predict_reg != test_reg$vs)
print(paste('Accuracy =', 1 - missing_classerr))
```

```
## [1] "Accuracy = 0.777777777777778"
```

```
# ROC-AUC Curve
```

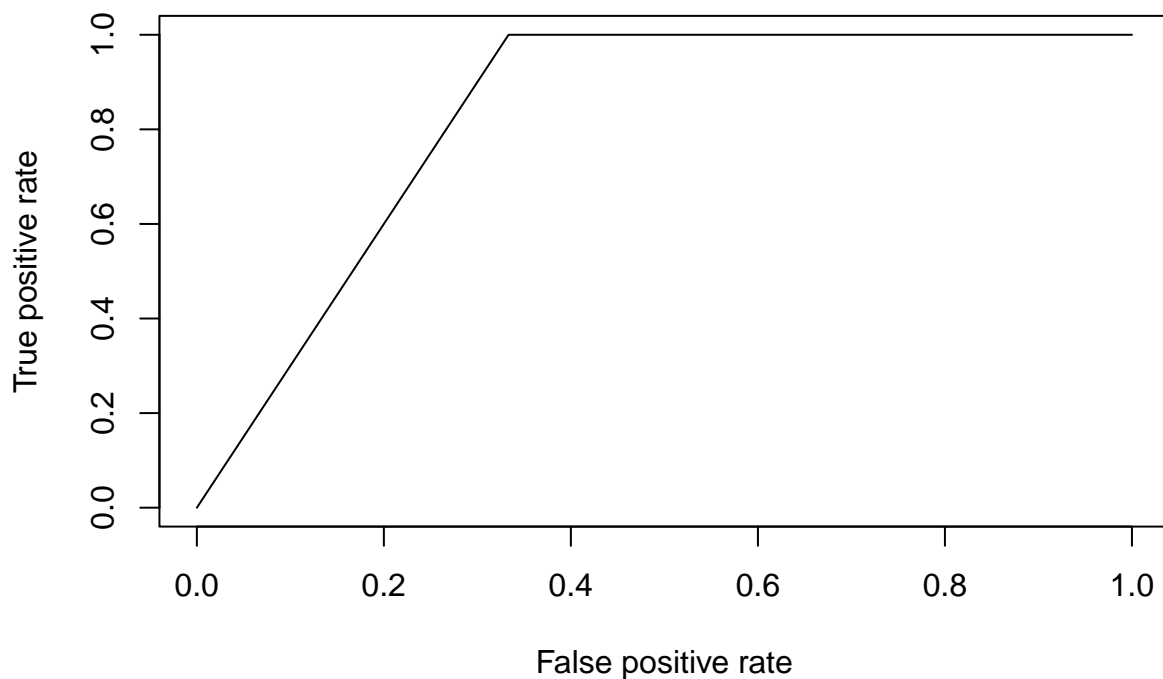
```
ROCPred <- prediction(predict_reg, test_reg$vs)
ROCPer <- performance(ROCPred, measure = "tpr",
                      x.measure = "fpr")
```

```
auc <- performance(ROCPer, measure = "auc")
auc <- auc@y.values[[1]]
auc
```

```
## [1] 0.8333333
```

```
# Plotting curve
```

```
plot(ROCPer)
```



```
plot(ROCPer, colorize = TRUE,
     print.cutoffs.at = seq(0.1, by = 0.1),
     main = "ROC CURVE")
abline(a = 0, b = 1)
```

```
auc <- round(auc, 4)
legend(.6, .4, auc, title = "AUC", cex = 1)
```

