Muthkumar&Rajagopal-Data621-Homework2

Muthukumar Srinivasan & Rajagopal Srinivasan

May 7, 2017

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
library("pROC")
```

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

##
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
library("caret")
```

```
## Loading required package: lattice

## Loading required package: ggplot2
```

```
########MUTHUKUMAR SRINIVASAN & RAJAGOPAL SRINIVASAN ##########
######## WEEK5 HOMEWORKD 2 - SUBMISSION#########################

####  Instruction : 1 - Week5-Homework 2
#### - Downloaded data and uploaded them into our GitHub. got the raw data and used through http protoco

data<-read.csv("https://raw.githubusercontent.com/muthukumars/DATA-621/master/Week5-Homework2/classifica
head(data)
```

```
##   pregnant glucose diastolic skinfold insulin  bmi pedigree age class
## 1        7     124        70       33     215 25.5    0.161  37     0
## 2        2     122        76       27     200 35.9    0.483  26     0
## 3        3     107        62       13      48 22.9    0.678  23     1
## 4        1      91        64       24       0 29.2    0.192  21     0
## 5        4      83        86       19       0 29.3    0.317  34     0
## 6        1     100        74       12      46 19.5    0.149  28     0
##   scored.class scored.probability
## 1            0         0.32845226
## 2            0         0.27319044
## 3            0         0.10966039
## 4            0         0.05599835
## 5            0         0.10049072
## 6            0         0.05515460
```

```
summary(data)
```

```
##     pregnant         glucose        diastolic        skinfold
##  Min.   : 0.000   Min.   : 57.0   Min.   : 38.0   Min.   : 0.0
##  1st Qu.: 1.000   1st Qu.: 99.0   1st Qu.: 64.0   1st Qu.: 0.0
##  Median : 3.000   Median :112.0   Median : 70.0   Median :22.0
##  Mean   : 3.862   Mean   :118.3   Mean   : 71.7   Mean   :19.8
##  3rd Qu.: 6.000   3rd Qu.:136.0   3rd Qu.: 78.0   3rd Qu.:32.0
##  Max.   :15.000   Max.   :197.0   Max.   :104.0   Max.   :54.0
##     insulin            bmi           pedigree           age
##  Min.   :  0.00   Min.   :19.40   Min.   :0.0850   Min.   :21.00
##  1st Qu.:  0.00   1st Qu.:26.30   1st Qu.:0.2570   1st Qu.:24.00
##  Median :  0.00   Median :31.60   Median :0.3910   Median :30.00
##  Mean   : 63.77   Mean   :31.58   Mean   :0.4496   Mean   :33.31
##  3rd Qu.:105.00   3rd Qu.:36.00   3rd Qu.:0.5800   3rd Qu.:41.00
##  Max.   :543.00   Max.   :50.00   Max.   :2.2880   Max.   :67.00
##      class         scored.class    scored.probability
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.02323
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.11702
##  Median :0.0000   Median :0.0000   Median :0.23999
##  Mean   :0.3149   Mean   :0.1768   Mean   :0.30373
##  3rd Qu.:1.0000   3rd Qu.:0.0000   3rd Qu.:0.43093
##  Max.   :1.0000   Max.   :1.0000   Max.   :0.94633
```

```
####Table of Scored vs Class
####  Instruction : 2 - Week5-Homework 2
#### -User table function to get raw data to table

tab<-table(data$class,data$scored.class)
colnames(tab)<-c("Real Negative","Real Positive")
rownames(tab)<-c("Model Negative","Model Positive")
tab
```

```
##
##                  Real Negative Real Positive
##   Model Negative           119             5
##   Model Positive            30            27
```

```
head(tab)
```

```
##
##                  Real Negative Real Positive
##   Model Negative           119             5
##   Model Positive            30            27
```

```
tail(tab)
```

```
##
##                  Real Negative Real Positive
##   Model Negative           119             5
##   Model Positive            30            27
```

```r
summary(tab)
```

```
## Number of cases in table: 181
## Number of factors: 2
## Test for independence of all factors:
##  Chisq = 50.39, df = 1, p-value = 1.261e-12
```

```r
####  Instruction : 3 - Week5-Homework 2
####All Metrics Function (Problems 3-8)

allmetrics<-function(data,predictMethod){

  tab <- table(data$class,data$scored.class)
  tn<-tab[1,1]
  tp<-tab[2,2]
  fn<-tab[2,1]
  fp<-tab[1,2]

  ####All Metrics Function Problems 3
  if (predictMethod=='Accuracy'){
    calcAccuracy<-(tp+tn)/(tp+tn+fn+fp)
    print ("Solution for Problem 3:")
    return(calcAccuracy)
  }

  ####All Metrics Function Problems 4
  if (predictMethod=='ErrorRate'){
    calcErrorRate<-(fp+fn)/(tp+tn+fn+fp)
    print ("Solution for Problem 4:")
    return(calcErrorRate)

  }

  ####All Metrics Function Problems 5
  if (predictMethod=='Precision'){
    calcPrecision<-(tp)/(tp+fp)
    print ("Solution for Problem 5:")
    return(calcPrecision)
  }

  ####All Metrics Function Problems 6
  if (predictMethod=='Sensitivity'){
    calcSensitivity<-(tp)/(tp+fn)
    print ("Solution for Problem 6:")
    return(calcSensitivity)
  }

  ####All Metrics Function Problems 7
  if (predictMethod=='Specificity'){
    calcSpecificity<-(tn)/(tn+fp)
    print ("Solution for Problem 7:")
    return(calcSpecificity)
  }
```

```
}

allmetrics(data,'Accuracy')
```

```
## [1] "Solution for Problem 3:"

## [1] 0.8066298
```

```
allmetrics(data,'ErrorRate')
```

```
## [1] "Solution for Problem 4:"

## [1] 0.1933702
```

```
allmetrics(data,'Precision')
```

```
## [1] "Solution for Problem 5:"

## [1] 0.84375
```

```
allmetrics(data,'Sensitivity')
```

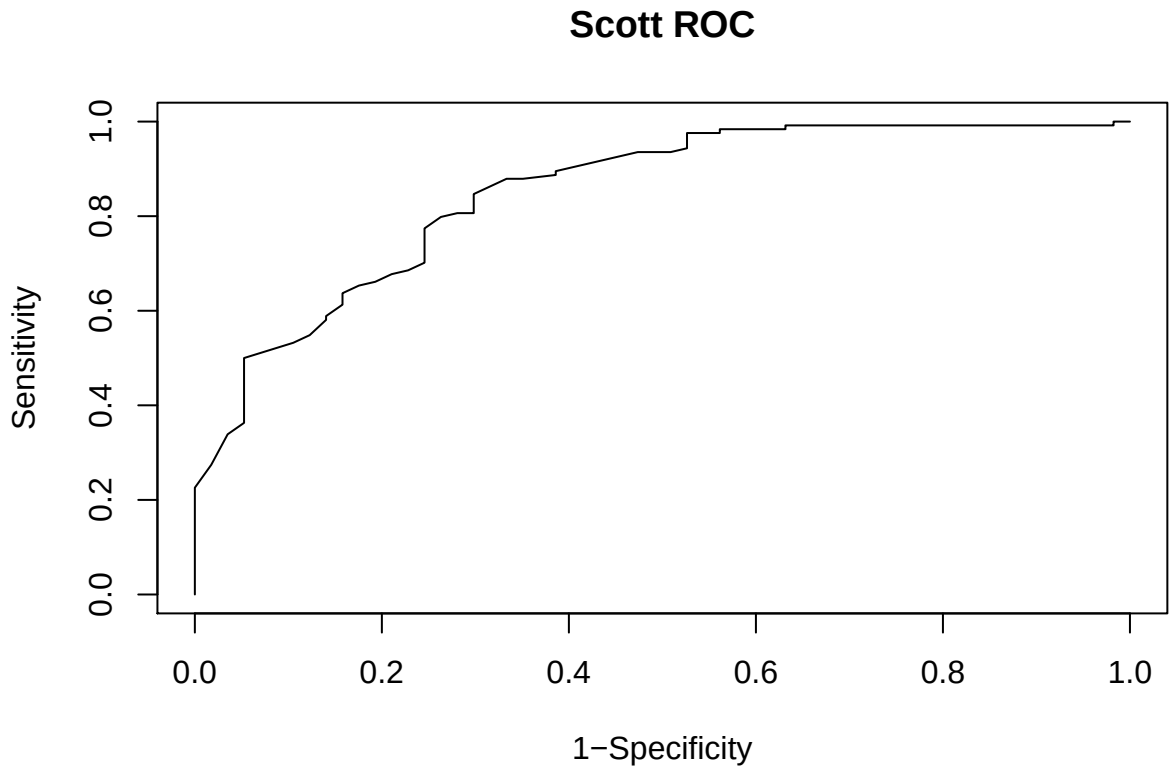
```
## [1] "Solution for Problem 6:"

## [1] 0.4736842
```

```
allmetrics(data,'Specificity')
```

```
## [1] "Solution for Problem 7:"

## [1] 0.9596774
```

```
####  Instruction : 3 - Week5-Homework 2
####All Metrics Function Problems 10

  ROC_Scott<- function(data,t) {

  se<-0
  sp<-0
  a<-0
  for (i in 1:round(1/t))
  {

  se[i]<-sensitivity(reference=as.factor(data$class),data=as.factor(as.numeric(data$scored.probability >
  sp[i]<-specificity(reference=as.factor(data$class),data=as.factor(as.numeric(data$scored.probability >
  a[i]<-t/2*(sp[i+1]+se[i])
  }
```

```
  ## Area of rectangles
  b1<-se[-1]
  b2<-se[-round(1/t)]
  x1<-sp[-1]
  x2<-sp[-round(1/t)]

  auc<-sum(b1*(x2-x1))
  plot(y=se,x=1-sp,xlab="1-Specificity",ylab="Sensitivity",main = "Scott ROC",type="l")
  return(paste0("AUC=",round(auc,3)))
}

ROC_Scott(data,t=0.01)
```




```
## [1] "AUC=0.854"
```

```
####  Instruction : 3 - Week5-Homework 2
####All Metrics Function Problems 11
##########################################PROBLEM 11***************
ACCU<-allmetrics(data,'Accuracy')
```

```
## [1] "Solution for Problem 3:"
```

```
ERROR<-allmetrics(data,'ErrorRate')
```

```
## [1] "Solution for Problem 4:"
```

```
PREC<-allmetrics(data,'Precision')
```

```
## [1] "Solution for Problem 5:"
```

```
SENS<-allmetrics(data,'Sensitivity')
```

```
## [1] "Solution for Problem 6:"
```

```
SPEC<-allmetrics(data,'Specificity')
```

```
## [1] "Solution for Problem 7:"
```

```
F1<-2*PREC*SENS/(PREC+SENS)

print(paste0("Accurancy Value->>>>>: ", ACCU))
```

```
## [1] "Accurancy Value->>>>>: 0.806629834254144"
```

```
print(paste0("Classification Error Rate->>>>>: ", ERROR))
```

```
## [1] "Classification Error Rate->>>>>: 0.193370165745856"
```

```
print(paste0("Precision Value->>>>>: ", ACCU))
```

```
## [1] "Precision Value->>>>>: 0.806629834254144"
```

```
print(paste0("Sensitivity Value->>>>>: ", SENS))
```

```
## [1] "Sensitivity Value->>>>>: 0.473684210526316"
```

```
print(paste0("Specificity Value->>>>>: ", SPEC))
```

```
## [1] "Specificity Value->>>>>: 0.959677419354839"
```

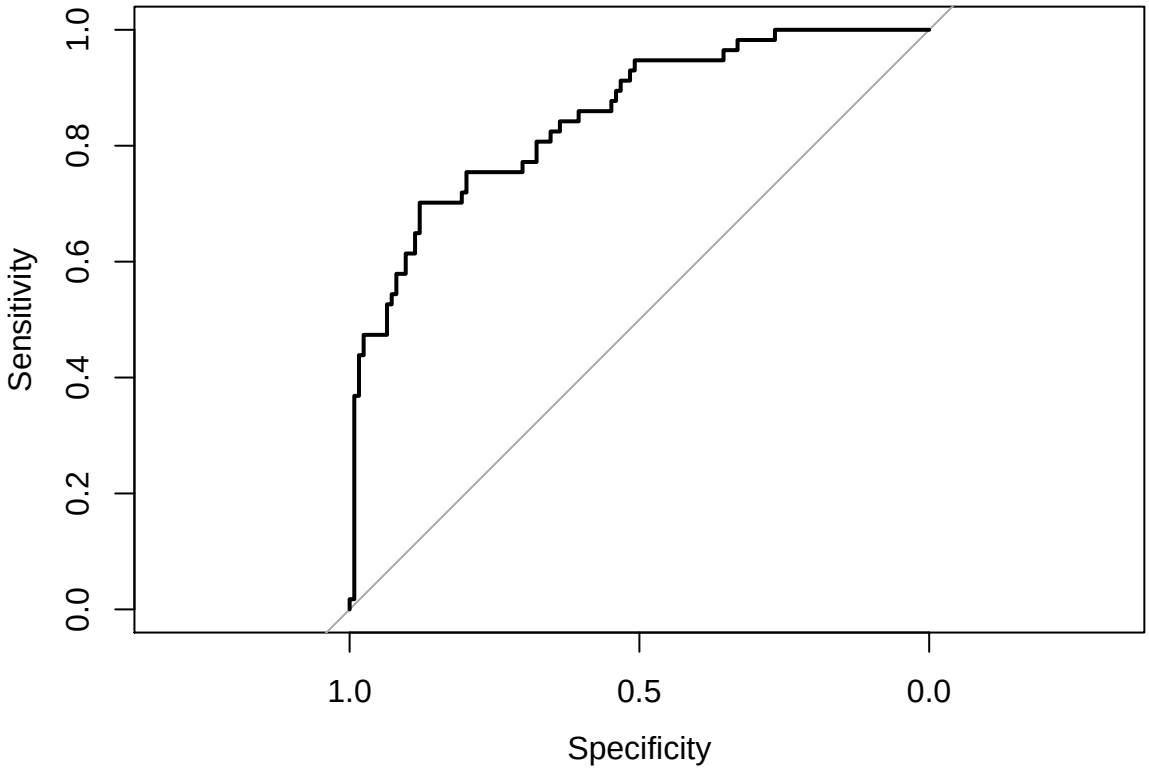
```
print(paste0("F1 SCORE->>>>>: ", F1))
```

```
## [1] "F1 SCORE->>>>>: 0.606741573033708"
```

```
####  Instruction : 3 - Week5-Homework 2
####All Metrics Function Problems 12
##########################################PROBLEM 11***************
confusionMatrix(data=data$scored.class,reference = data$class)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 119  30
##          1   5  27
##
##                Accuracy : 0.8066
##                  95% CI : (0.7415, 0.8615)
##     No Information Rate : 0.6851
##     P-Value [Acc > NIR] : 0.0001712
##
##                   Kappa : 0.4916
##  Mcnemar's Test P-Value : 4.976e-05
##
##             Sensitivity : 0.9597
##             Specificity : 0.4737
##          Pos Pred Value : 0.7987
##          Neg Pred Value : 0.8438
##              Prevalence : 0.6851
##          Detection Rate : 0.6575
##    Detection Prevalence : 0.8232
##       Balanced Accuracy : 0.7167
##
##        'Positive' Class : 0
##
```

```
####  Instruction : 3 - Week5-Homework 2
####All Metrics Function Problems 13
##########################################PROBLEM 11***************

roc(data$class, data$scored.probability,plot=TRUE)
```




```
##
## Call:
## roc.default(response = data$class, predictor = data$scored.probability,     plot = TRUE)
##
## Data: data$scored.probability in 124 controls (data$class 0) < 57 cases (data$class 1).
## Area under the curve: 0.8503
```

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.