Linear regression with ROC

Anton Antonov

10/10/2016

Introduction

This document demonstrates how to do in R linear regression (easily using the built-in function `lm`) and to tune the binary classification with the derived model through the so called Receiver Operating Characteristic (ROC) framework, [5, 6].

The data used in this document is from [1] and it has been analyzed in more detail in [2]. In this document we only show to how to ingest and do very basic analysis of that data before proceeding with the linear regression model and its tuning. The package ROCR, [3], (introduced with [4]) provides the needed ROC functionalities.

Libraries needed to run the Rmd file:

```
library(plyr)
library(ROCR)
```

```
## Loading required package: gplots

##
## Attaching package: 'gplots'

## The following object is masked from 'package:stats':
##
##     lowess
```

```
library(lattice)
library(reshape2)
library(ggplot2)
```

Data ingestion

The code below imports the data from [1].

```
data <- read.table( "~/Datasets/adult/adult.data", sep = ",", stringsAsFactors = FALSE )
testData <- read.table( "~/Datasets/adult/adult.test", fill = TRUE, sep = ",", stringsAsFactors = FALSE
testData <- testData[-1,]
testData[,1] <- as.numeric(testData[,1])

columnNames<-
  strsplit(paste0("age,workclass,fnlwgt,education,education.num,marital.status,occupation,",
                  "relationship,race,sex,capital.gain,capital.loss,hours.per.week,native.country,income

names(data) <- columnNames
names(testData) <- columnNames

data$income <- gsub( pattern = "\\s", replacement = "", data$income )
```

```r
testData$income <- gsub( pattern = "\\s", replacement = "", testData$income )
testData$income <- gsub( pattern = ".", replacement = "", testData$income, fixed = TRUE )
```

Assignment of training and tuning data

As usual in classification and regression problems we work with two data sets: a training data set and a testing data set. Here we split the original training set into two sets a training set and a tuning set. The tuning set is going to be used to find a good value of a tuning parameter through ROC.

```r
trainingInds <- sample( 1:nrow(data), ceiling( 0.8*nrow(data) ) )
tuningInds <- setdiff( 1:nrow(data), trainingInds )
trainingData <- data[ trainingInds, ]
tuningData <- data[ tuningInds, ]
```

Basic data analysis

Before doing regression it is a good idea to do some preliminary analysis of the data.

Here is the summary of the training data:

```r
summary(as.data.frame(unclass(data)))
```

```
##       age                    workclass         fnlwgt       
##  Min.   :17.00    Private          :22696   Min.   :  12285  
##  1st Qu.:28.00    Self-emp-not-inc: 2541   1st Qu.: 117827  
##  Median :37.00    Local-gov        : 2093   Median : 178356  
##  Mean   :38.58    ?                : 1836   Mean   : 189778  
##  3rd Qu.:48.00    State-gov        : 1298   3rd Qu.: 237051  
##  Max.   :90.00    Self-emp-inc     : 1116   Max.   :1484705  
##                  (Other)          :  981                    
##          education     education.num                 marital.status 
##   HS-grad     :10501   Min.   : 1.00    Divorced             : 4443  
##   Some-college: 7291   1st Qu.: 9.00    Married-AF-spouse    :   23  
##   Bachelors   : 5355   Median :10.00    Married-civ-spouse   :14976  
##   Masters     : 1723   Mean   :10.08    Married-spouse-absent:  418  
##   Assoc-voc   : 1382   3rd Qu.:12.00    Never-married        :10683  
##   11th        : 1175   Max.   :16.00    Separated            : 1025  
##  (Other)      : 5134                    Widowed              :  993  
##             occupation           relationship  
##   Prof-specialty :4140    Husband       :13193  
##   Craft-repair   :4099    Not-in-family : 8305  
##   Exec-managerial:4066    Other-relative:  981  
##   Adm-clerical   :3770    Own-child     : 5068  
##   Sales          :3650    Unmarried     : 3446  
##   Other-service  :3295    Wife          : 1568  
##  (Other)         :9541                          
##                  race            sex         capital.gain  
##   Amer-Indian-Eskimo:  311    Female:10771   Min.   :    0  
##   Asian-Pac-Islander: 1039    Male  :21790   1st Qu.:    0  
##   Black             : 3124                   Median :    0  
##   Other             :  271                   Mean   : 1078  
##   White             :27816                   3rd Qu.:    0  
##                                              Max.   :99999  
```

```
##
##   capital.loss    hours.per.week        native.country    income
##  Min.   :   0.0   Min.   : 1.00    United-States:29170   <=50K:24720
##  1st Qu.:   0.0   1st Qu.:40.00    Mexico       :  643   >50K : 7841
##  Median :   0.0   Median :40.00    ?            :  583
##  Mean   :  87.3   Mean   :40.44    Philippines  :  198
##  3rd Qu.:   0.0   3rd Qu.:45.00    Germany      :  137
##  Max.   :4356.0   Max.   :99.00    Canada       :  121
##                                   (Other)       : 1709
```

And here is the summary of the test data:

```
summary(as.data.frame(unclass(testData)))
```

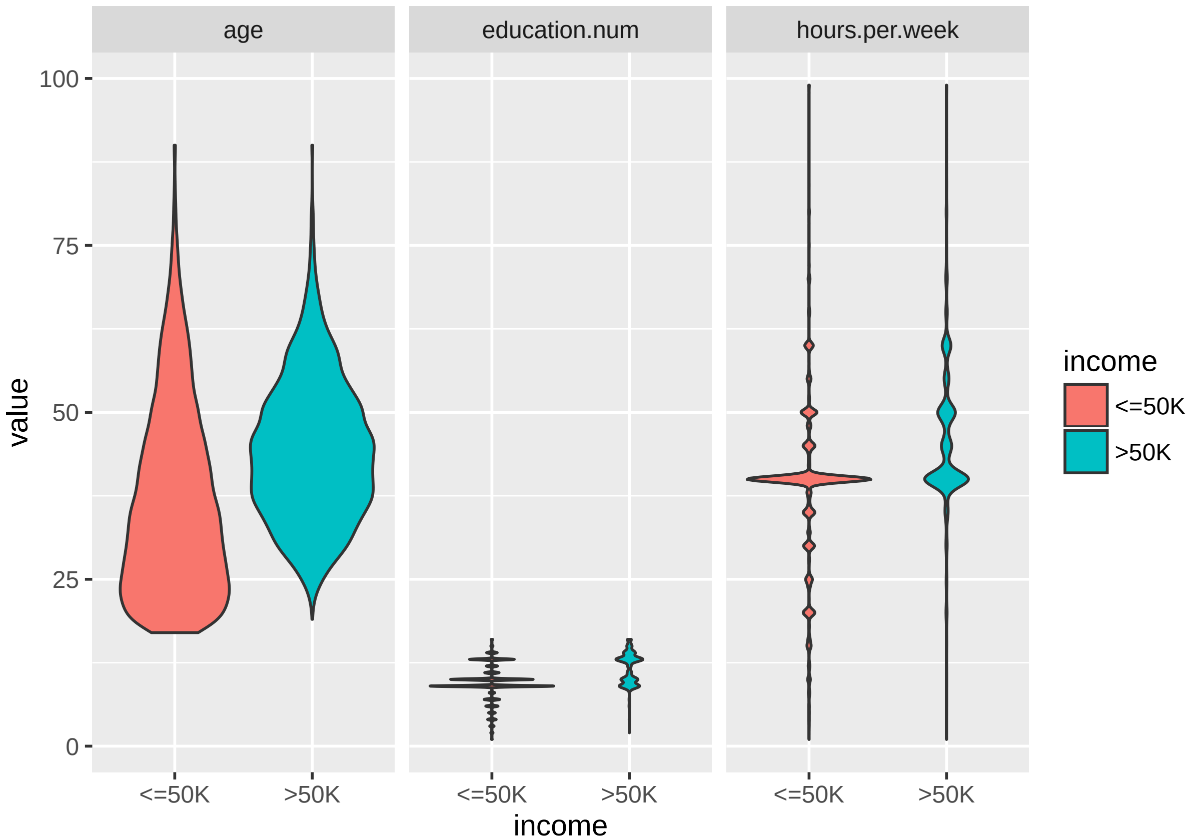
```
##       age                    workclass         fnlwgt
##  Min.   :17.00    Private         :11210   Min.   :  13492
##  1st Qu.:28.00    Self-emp-not-inc: 1321   1st Qu.: 116736
##  Median :37.00    Local-gov       : 1043   Median : 177831
##  Mean   :38.77    ?               :  963   Mean   : 189436
##  3rd Qu.:48.00    State-gov       :  683   3rd Qu.: 238384
##  Max.   :90.00    Self-emp-inc    :  579   Max.   :1490400
##                  (Other)          :  482
##          education     education.num                   marital.status
##   HS-grad     :5283   Min.   : 1.00    Divorced             :2190
##   Some-college:3587   1st Qu.: 9.00    Married-AF-spouse    :  14
##   Bachelors   :2670   Median :10.00    Married-civ-spouse   :7403
##   Masters     : 934   Mean   :10.07    Married-spouse-absent: 210
##   Assoc-voc   : 679   3rd Qu.:12.00    Never-married        :5434
##   11th        : 637   Max.   :16.00    Separated            : 505
##  (Other)      :2491                    Widowed              : 525
##            occupation          relationship
##   Prof-specialty :2032    Husband       :6523
##   Exec-managerial:2020    Not-in-family :4278
##   Craft-repair   :2013    Other-relative: 525
##   Sales          :1854    Own-child     :2513
##   Adm-clerical   :1841    Unmarried     :1679
##   Other-service  :1628    Wife          : 763
##  (Other)         :4893
##                  race            sex         capital.gain
##   Amer-Indian-Eskimo:  159    Female: 5421   Min.   :    0
##   Asian-Pac-Islander:  480    Male  :10860   1st Qu.:    0
##   Black             : 1561                   Median :    0
##   Other             :  135                   Mean   : 1082
##   White             :13946                   3rd Qu.:    0
##                                              Max.   :99999
##
##   capital.loss    hours.per.week        native.country    income
##  Min.   :   0.0   Min.   : 1.00    United-States:14662   <=50K:12435
##  1st Qu.:   0.0   1st Qu.:40.00    Mexico       :  308   >50K : 3846
##  Median :   0.0   Median :40.00    ?            :  274
##  Mean   :  87.9   Mean   :40.39    Philippines  :   97
##  3rd Qu.:   0.0   3rd Qu.:45.00    Puerto-Rico  :   70
##  Max.   :3770.0   Max.   :99.00    Germany      :   69
##                                   (Other)       :  801
```

For the code below we are going to use the following variables

```r
columnNameResponseVar <- "income"
columnNamesExplanatoryVars <- c("age", "education.num", "hours.per.week")
columnNamesForAnalysis <- c( columnNamesExplanatoryVars, columnNameResponseVar )
```

With this plot we can see that `age, education.num, hours.per.week` correlate (can explain) with `income`:

```r
dataLong <- melt( data = data[, columnNamesForAnalysis], id.vars = columnNameResponseVar  )
ggplot(dataLong, aes(x = income, y = value, fill = income)) + geom_violin() + facet_wrap( ~variable, nc
```



On the plot above we see that higher values of `age, education.num, hours.per.week` are associated closer with ">50K". For more detailed analysis see [2].

Linear regression

```r
dataReg <- trainingData[,columnNamesForAnalysis]
unique(dataReg$income)
```

```
## [1] "<=50K" ">50K"
```

```r
dataReg$income <- ifelse( dataReg$income == ">50K", 1, 0 )

lmRes <- lm( income ~ age + education.num + hours.per.week, data = dataReg )
```

Linear regression with ROC

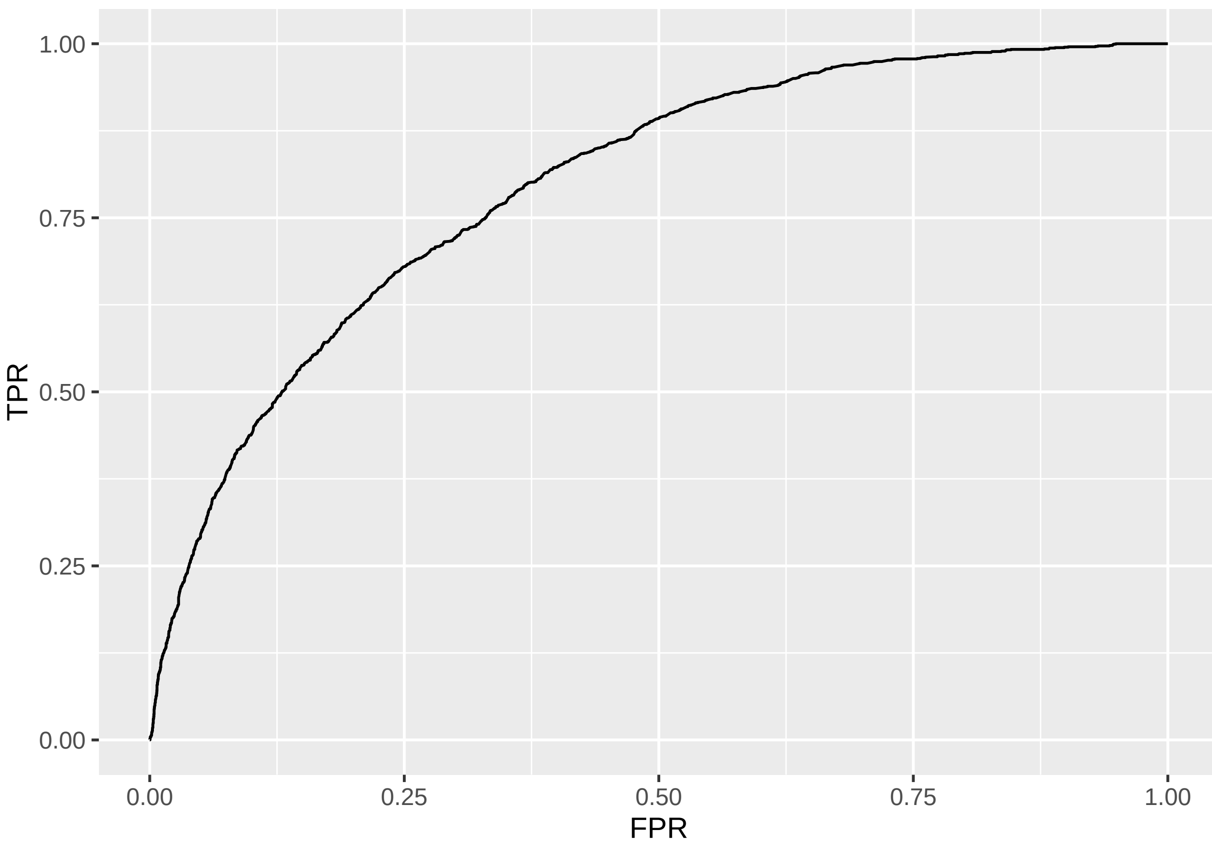
In this section we take a systematic approach of determining the best threshold to be used to separate the regression model values.

We will consider ">50" to be the more important class label for the classifiers built below. As a result, we are going to call *positive* the income values ">50K" and *negative* the income values "<=50K".

The used ROC functionalities are employed through the package [3].

Computations to find the best threshold

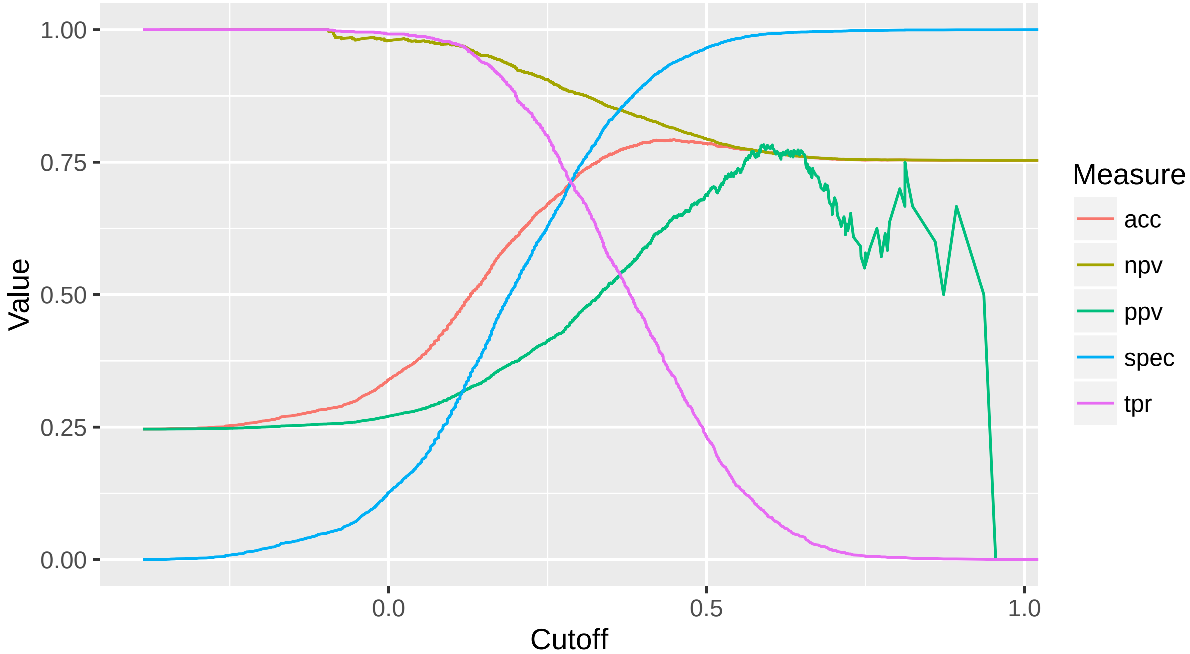
```
modelValues <- predict(lmRes, newdata = tuningData[, columnNamesExplanatoryVars], type="response")

## unique(tuningData$income)

pr <- prediction( modelValues, ifelse( tuningData$income == ">50K", 1, 0) )
prf <- performance(pr, measure = "tpr", x.measure = "fpr")
ggplot( data.frame( FPR = prf@x.values[[1]], TPR = prf@y.values[[1]] ) ) + aes( x = FPR, y = TPR) + geo
```



After looking at "" we can come up with the following code that plots the ROC functions "PPV", "NPV", "TPR", "ACC", and "SPC"/"SPEC".

```
rocDF <-
  ldply( c("ppv", "npv", "tpr", "acc", "spec"), function(x) {
    res <- performance(pr, measure = x, x.measure = "cutoff")
    data.frame( Measure = x, Cutoff = as.numeric(res@x.values[[1]]), Value = as.numeric(res@y.values[[1
  })
```

```r
rocDF <- rocDF[ !is.na(rocDF$Value), ]
ggplot(rocDF) + aes( x = Cutoff, y = Value, color = Measure) + geom_line() + coord_fixed(ratio = 1/1.2)
```



From the plot we can select the best cutoff value, in this case ≈ 0.3.

Accuracy over the test data

We split the original training data into two parts for training and tuning. Using the found threshold, let us use evaluate the classification process over the test data.

```r
modelValues <- predict(lmRes, newdata = testData[, columnNamesExplanatoryVars], type="response")

threshold <- 0.3
classDF <- data.frame( Actual = testData[, columnNameResponseVar], Predicted = ifelse( modelValues >= tl
```

Here is the overall accuracy:

```r
mean( classDF$Actual == classDF$Predicted)
```

```
## [1] 0.7220687
```

And here is the confusion matrix

```r
xtabs( ~ Actual + Predicted, classDF )
```

```
##        Predicted
## Actual  <=50K >50K
##   <=50K  9119 3316
##   >50K   1209 2637
```

Here are the corresponding frequencies:

```r
xtabs( ~ Actual + Predicted, classDF ) / count( classDF, .(Actual))[,2]
```

```
##        Predicted
## Actual      <=50K      >50K
```

```
##   <=50K 0.7333333 0.2666667
##   >50K  0.3143526 0.6856474
```

References

[1] Bache, K. & Lichman, M. (2013). UCI Machine Learning Repository. Irvine, CA: University of California, School of Information and Computer Science. Census Income Data Set, URL: http://archive.ics.uci.edu/ml/datasets/Census+Income .

[2] Anton Antonov, "Classification and association rules for census income data", (2014), MathematicaForPrediction at WordPress.com , URL: https://mathematicaforprediction.wordpress.com/2014/03/30/classification-and-association-rules-for-census-income-data/ .

[3] [ROCR web site](http://rocr.bioinf.mpi-sb.mpg.de) http://rocr.bioinf.mpi-sb.mpg.de.

[4] Tobias Sing, Oliver Sander, Niko Beerenwinkel, Thomas Lengauer. ROCR: visualizing classifier performance in R, (2005), Bioinformatics 21(20):3940-3941.

[5] Wikipedia entry, Receiver operating characteristic. URL: http://en.wikipedia.org/wiki/Receiver_operating_characteristic .

[6] Tom Fawcett, An introduction to ROC analysis, (2006), Pattern Recognition Letters, 27, 861–874. (Link to PDF.)