Kaggle Competition, The Analytics Edge

Morozov Gleb

2 2015 .

Machine Learning, Kaggle.com. “The Analytics Edge” “Massachusetts Institute of Technology”.

. , Apple iPad , eBay.

:

- eBayiPadTrain.csv - . 1861 .
- eBayiPadTest.csv -

, .

```
library(dplyr) #
library(readr) #
```

.

```
eBayTrain <-  read_csv("eBayiPadTrain.csv")
eBayTest <-  read_csv("eBayiPadTest.csv")
```

.

```
summary(eBayTrain)
```

```
##  description           biddable        startprice      condition
##  Length:1861        Min.   :0.0000   Min.   :  0.01   Length:1861
##  Class :character   1st Qu.:0.0000   1st Qu.: 80.00   Class :character
##  Mode  :character   Median :0.0000   Median :179.99   Mode  :character
##                     Mean   :0.4498   Mean   :211.18
##                     3rd Qu.:1.0000   3rd Qu.:300.00
##                     Max.   :1.0000   Max.   :999.00
##    cellular           carrier             color
##  Length:1861        Length:1861        Length:1861
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##    storage           productline             sold           UniqueID
##  Length:1861        Length:1861        Min.   :0.0000   Min.   :10001
```

```
##  Class :character   Class :character   1st Qu.:0.0000   1st Qu.:10466
##  Mode  :character   Mode  :character   Median :0.0000   Median :10931
##                                        Mean   :0.4621   Mean   :10931
##                                        3rd Qu.:1.0000   3rd Qu.:11396
##                                        Max.   :1.0000   Max.   :11861
```

```
str(eBayTrain)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    1861 obs. of  11 variables:
##  $ description: chr  "iPad is in 8.5+ out of 10 cosmetic condition!" "Previously used, please read de
##  $ biddable   : int  0 1 0 0 0 1 1 0 1 1 ...
##  $ startprice : num  159.99 0.99 199.99 235 199.99 ...
##  $ condition  : chr  "Used" "Used" "Used" "New other (see details)" ...
##  $ cellular   : chr  "0" "1" "0" "0" ...
##  $ carrier    : chr  "None" "Verizon" "None" "None" ...
##  $ color      : chr  "Black" "Unknown" "White" "Unknown" ...
##  $ storage    : chr  "16" "16" "16" "16" ...
##  $ productline: chr  "iPad 2" "iPad 2" "iPad 4" "iPad mini 2" ...
##  $ sold       : int  0 1 1 0 0 1 1 0 1 1 ...
##  $ UniqueID   : int  10001 10002 10003 10004 10005 10006 10007 10008 10009 10010 ...
```

11 :

- description - ,
- biddable - (= 1) (= 0)

- startprice - (biddable=1) (biddable=0)
- condition - (, / ..)
- cellular - (= 1) (= 0)

- carrier - (cellular = 1)
- color -

- storage -

- productline -
- sold - (= 1) (=0). .

- UniqueID -

: description, startprice - .

```
table(eBayTrain$description == "")
```

```
##
## FALSE  TRUE
##   790  1071
```

, , . , 1, , 0, .

```
eBayTrain$is_descr = as.factor(eBayTrain$description == "")
table(eBayTrain$description == "", eBayTrain$is_descr)
```

```
##
##         FALSE TRUE
##   FALSE   790    0
##   TRUE      0 1071
```

. tm.

```
library(tm) ##
```

```
## Loading required package: NLP
```

```
 ##    ,
 CorpusDescription <-  Corpus(VectorSource(c(eBayTrain$description, eBayTest$description)))
 ##
 CorpusDescription <-  tm_map(CorpusDescription, content_transformer(tolower))
 CorpusDescription <-  tm_map(CorpusDescription, PlainTextDocument)
 ##
 CorpusDescription <-  tm_map(CorpusDescription, removePunctuation)
 ##    -, .. ,
 CorpusDescription <-  tm_map(CorpusDescription, removeWords, stopwords("english"))
 ##  , ..
 CorpusDescription <-  tm_map(CorpusDescription, stemDocument)
 ##
 dtm <-  DocumentTermMatrix(CorpusDescription)
 ##
 sparse <-  removeSparseTerms(dtm, 0.97)

 ##     data.frame
DescriptionWords = as.data.frame(as.matrix(sparse))
colnames(DescriptionWords) = make.names(colnames(DescriptionWords))
DescriptionWordsTrain = head(DescriptionWords, nrow(eBayTrain))
DescriptionWordsTest = tail(DescriptionWords, nrow(eBayTest))
```

factor, . , . magnittr

```
library(magrittr)
eBayTrain %<>% mutate(condition = as.factor(condition), cellular = as.factor(cellular),
        carrier = as.factor(carrier), color = as.factor(color),
        storage = as.factor(storage), productline = as.factor(productline), sold = as.factor(sold)) %>%
        select(-description, -UniqueID ) %>% cbind(., DescriptionWordsTrain)
```

.

```
str(eBayTrain)
```

```
## 'data.frame':    1861 obs. of  30 variables:
##  $ biddable   : int  0 1 0 0 0 1 1 0 1 1 ...
##  $ startprice : num  159.99 0.99 199.99 235 199.99 ...
##  $ condition  : Factor w/ 6 levels "For parts or not working",..: 6 6 6 4 5 6 3 3 6 6 ...
##  $ cellular   : Factor w/ 3 levels "0","1","Unknown": 1 2 1 1 3 2 1 1 2 1 ...
##  $ carrier    : Factor w/ 7 levels "AT&T","None",..: 2 7 2 2 6 1 2 2 6 2 ...
##  $ color      : Factor w/ 5 levels "Black","Gold",..: 1 4 5 4 4 3 3 5 5 5 ...
##  $ storage    : Factor w/ 5 levels "128","16","32",..: 2 2 2 2 5 3 2 2 4 3 ...
##  $ productline: Factor w/ 12 levels "iPad 1","iPad 2",..: 2 2 4 9 12 9 8 10 1 4 ...
##  $ sold       : Factor w/ 2 levels "0","1": 1 2 2 1 1 2 2 1 2 2 ...
##  $ is_descr   : Factor w/ 2 levels "FALSE","TRUE": 1 1 2 2 1 2 2 2 2 2 ...
##  $ box        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ condit     : num  1 0 0 0 0 0 0 0 0 0 ...
##  $ cosmet     : num  1 0 0 0 0 0 0 0 0 0 ...
##  $ devic      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ excel      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ fulli      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ function.  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ good       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ great      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ includ     : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ ipad       : num  1 0 0 0 0 0 0 0 0 0 ...
##  $ item       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ light      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ minor      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ new        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ scratch    : num  0 1 0 0 0 0 0 0 0 0 ...
##  $ screen     : num  0 1 0 0 0 0 0 0 0 0 ...
##  $ use        : num  0 2 0 0 0 0 0 0 0 0 ...
##  $ wear       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ work       : num  0 0 0 0 0 0 0 0 0 0 ...
```

. , . AUC. . . AUC 1.0, - 0.5.

, , . caTools.

```
set.seed(1000) ##
library(caTools)
split <- sample.split(eBayTrain$sold, SplitRatio = 0.7)
train  <- filter(eBayTrain, split == T)
test <- filter(eBayTrain, split == F)
```

```
model_glm1 <- glm(sold ~ ., data = train, family = binomial)
```

```
summary(model_glm1)
```

```
##
## Call:
## glm(formula = sold ~ ., family = binomial, data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6620  -0.7308  -0.2450   0.6229   3.5600
##
## Coefficients:
##                                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)                       1.435e+01  6.194e+02   0.023 0.981520
## biddable                          1.523e+00  1.694e-01   8.987  < 2e-16
## startprice                       -1.153e-02  1.122e-03 -10.274  < 2e-16
## conditionManufacturer refurbished  9.276e-01  5.941e-01   1.562 0.118394
## conditionNew                       6.479e-01  3.845e-01   1.685 0.091964
## conditionNew other (see details)   9.838e-01  5.031e-01   1.956 0.050517
## conditionSeller refurbished       -3.144e-02  4.068e-01  -0.077 0.938388
## conditionUsed                      4.382e-01  2.717e-01   1.613 0.106767
## cellular1                         -1.314e+01  6.194e+02  -0.021 0.983079
## cellularUnknown                   -1.351e+01  6.194e+02  -0.022 0.982603
## carrierNone                       -1.326e+01  6.194e+02  -0.021 0.982921
## carrierOther                       1.252e+01  6.223e+02   0.020 0.983951
## carrierSprint                      8.900e-01  6.992e-01   1.273 0.203098
## carrierT-Mobile                    2.578e-02  8.932e-01   0.029 0.976973
## carrierUnknown                    -4.390e-01  4.168e-01  -1.053 0.292296
## carrierVerizon                     1.565e-01  3.634e-01   0.431 0.666625
## colorGold                          1.076e-01  5.356e-01   0.201 0.840755
## colorSpace Gray                   -1.304e-01  3.066e-01  -0.425 0.670564
## colorUnknown                      -1.447e-01  2.083e-01  -0.695 0.487307
## colorWhite                        -3.924e-02  2.300e-01  -0.171 0.864523
## storage16                         -1.097e+00  5.054e-01  -2.171 0.029933
## storage32                         -1.145e+00  5.186e-01  -2.207 0.027315
## storage64                         -5.065e-01  5.035e-01  -1.006 0.314474
## storageUnknown                    -2.930e-01  6.339e-01  -0.462 0.643867
## productlineiPad 2                  3.336e-01  2.846e-01   1.172 0.241026
## productlineiPad 3                  7.190e-01  3.460e-01   2.078 0.037694
## productlineiPad 4                  8.195e-01  3.651e-01   2.244 0.024801
## productlineiPad 5                  2.893e+00  1.080e+03   0.003 0.997863
## productlineiPad Air                2.152e+00  4.029e-01   5.341 9.22e-08
## productlineiPad Air 2              3.053e+00  5.083e-01   6.005 1.91e-09
## productlineiPad mini               4.068e-01  3.058e-01   1.330 0.183456
## productlineiPad mini 2             1.591e+00  4.174e-01   3.811 0.000138
## productlineiPad mini 3             2.191e+00  5.346e-01   4.099 4.16e-05
## productlineiPad mini Retina        3.225e+00  1.120e+00   2.879 0.003993
## productlineUnknown                 3.822e-01  3.922e-01   0.974 0.329891
## is_descrTRUE                       1.721e-01  2.562e-01   0.672 0.501722
## box                               -7.867e-01  4.813e-01  -1.635 0.102134
```

```
## condit                            -4.848e-01  2.914e-01  -1.664 0.096198
## cosmet                             1.438e-01  4.409e-01   0.326 0.744385
## devic                             -2.439e-01  4.101e-01  -0.595 0.552027
## excel                              8.378e-01  4.710e-01   1.779 0.075268
## fulli                             -5.841e-01  6.604e-01  -0.884 0.376464
## function.                         -3.029e-01  5.914e-01  -0.512 0.608555
## good                               7.870e-01  3.390e-01   2.321 0.020275
## great                              4.625e-01  3.895e-01   1.188 0.235003
## includ                             4.163e-01  4.295e-01   0.969 0.332421
## ipad                              -3.198e-01  2.442e-01  -1.310 0.190295
## item                              -8.037e-02  3.503e-01  -0.229 0.818501
## light                              3.290e-01  4.019e-01   0.819 0.412963
## minor                             -2.794e-01  3.760e-01  -0.743 0.457462
## new                                8.576e-02  3.844e-01   0.223 0.823479
## scratch                            2.037e-02  2.649e-01   0.077 0.938712
## screen                             1.437e-01  2.816e-01   0.510 0.609773
## use                                1.477e-01  2.181e-01   0.677 0.498243
## wear                              -5.187e-02  4.093e-01  -0.127 0.899154
## work                              -2.566e-01  2.944e-01  -0.871 0.383509
##
## (Intercept)
## biddable                          ***
## startprice                        ***
## conditionManufacturer refurbished
## conditionNew                      .
## conditionNew other (see details)  .
## conditionSeller refurbished
## conditionUsed
## cellular1
## cellularUnknown
## carrierNone
## carrierOther
## carrierSprint
## carrierT-Mobile
## carrierUnknown
## carrierVerizon
## colorGold
## colorSpace Gray
## colorUnknown
## colorWhite
## storage16                         *
## storage32                         *
## storage64
## storageUnknown
## productlineiPad 2
## productlineiPad 3                 *
## productlineiPad 4                 *
## productlineiPad 5
## productlineiPad Air               ***
## productlineiPad Air 2             ***
## productlineiPad mini
## productlineiPad mini 2            ***
## productlineiPad mini 3            ***
## productlineiPad mini Retina       **
```

```
## productlineUnknown
## is_descrTRUE
## box
## condit                          .
## cosmet
## devic
## excel                           .
## fulli
## function.
## good                            *
## great
## includ
## ipad
## item
## light
## minor
## new
## scratch
## screen
## use
## wear
## work
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1798.8  on 1302  degrees of freedom
## Residual deviance: 1168.8  on 1247  degrees of freedom
## AIC: 1280.8
##
## Number of Fisher Scoring iterations: 13
```

,

AUC . ROCR

```
library(ROCR)
```

```
## Loading required package: gplots
##
## Attaching package: 'gplots'
##
## The following object is masked from 'package:stats':
##
##     lowess
```

```
predict_glm <- predict(model_glm1, newdata = test, type = "response" )
ROCRpred = prediction(predict_glm, test$sold)
as.numeric(performance(ROCRpred, "auc")@y.values)
```

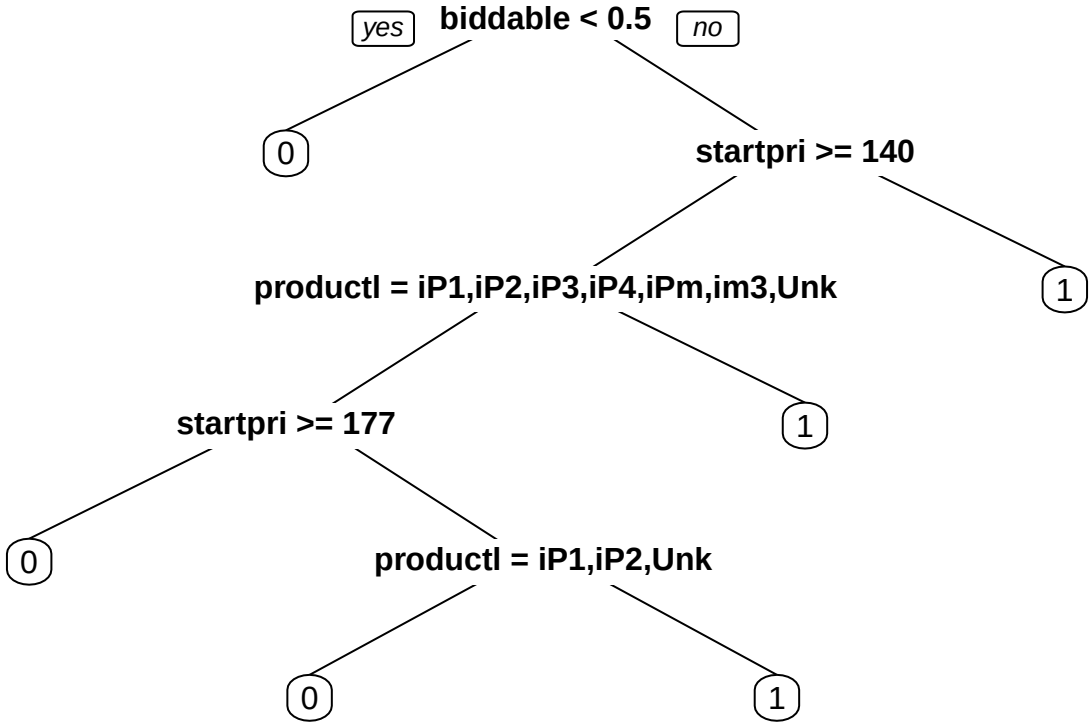
```
## [1] 0.8592183
```

, , .

(CART model)

CART

```
library(rpart)
library(rpart.plot)
model_cart1 <- rpart(sold ~ ., data = train, method = "class")
prp(model_cart1)
```




```
predict_cart <- predict(model_cart1, newdata = test, type = "prob")[,2]
ROCRpred = prediction(predict_cart, test$sold)
as.numeric(performance(ROCRpred, "auc")@y.values)
```

```
## [1] 0.8222028
```

. cross-validation. cp,

```
library(caret)
```

```
## Loading required package: lattice
## Loading required package: ggplot2
##
## Attaching package: 'ggplot2'
##
```

```
## The following object is masked from 'package:NLP':
##
##     annotate
```

```
library(e1071)
tr.control = trainControl(method = "cv", number = 10)
cpGrid = expand.grid( .cp = seq(0.0001,0.01,0.002))
train(sold ~ ., data = train, method = "rpart", trControl = tr.control, tuneGrid = cpGrid )$bestTune
```

```
##       cp
## 3 0.0041
```

```
bestcp <- train(sold ~ ., data = train, method = "rpart", trControl = tr.control, tuneGrid = cpGrid )$b
```

```
bestcp <- train(sold ~ ., data = train, method = "rpart", trControl = tr.control, tuneGrid = cpGrid )$b
model_cart2 <- rpart(sold ~ ., data = train, method = "class", cp = bestcp)
predict_cart <- predict(model_cart2, newdata = test, type = "prob")[,2]
ROCRpred = prediction(predict_cart, test$sold)
as.numeric(performance(ROCRpred, "auc")@y.values)
```

```
## [1] 0.8024935
```

Random Forest

, - Random Forest

```
library(randomForest)
```

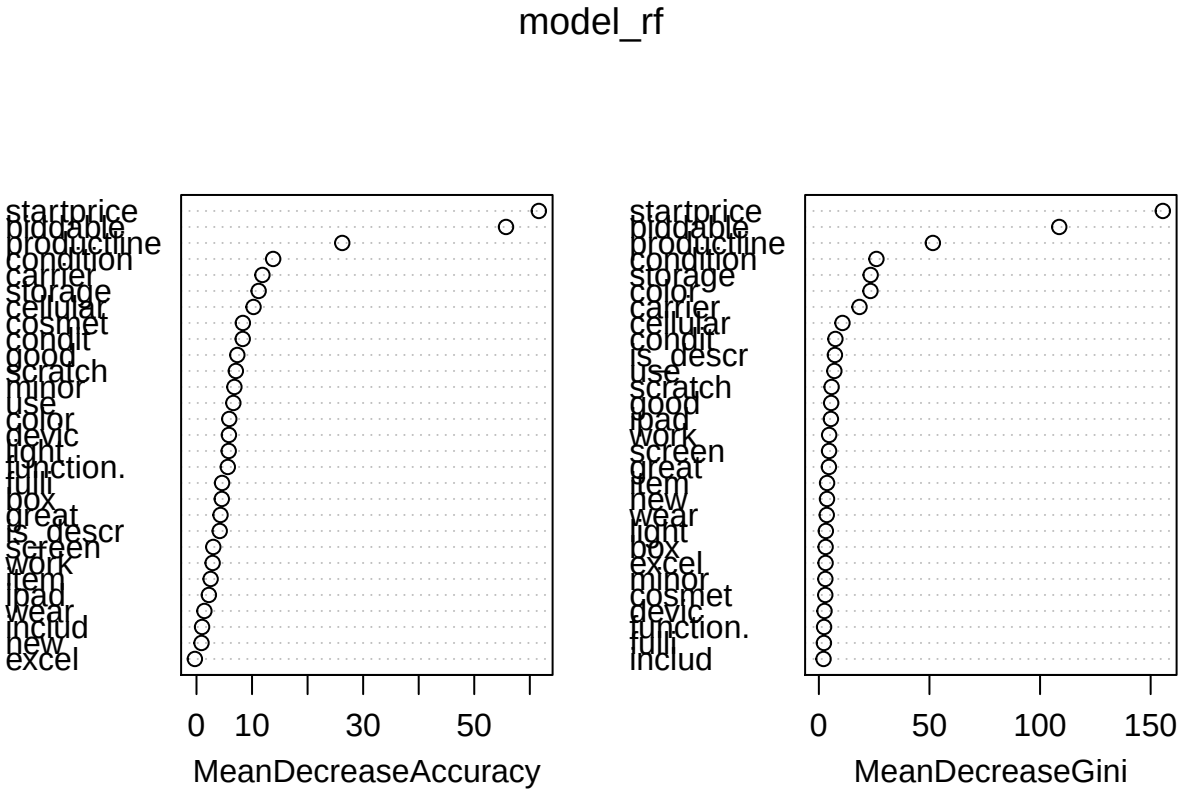
```
## randomForest 4.6-10
## Type rfNews() to see new features/changes/bug fixes.
##
## Attaching package: 'randomForest'
##
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
set.seed(1000)
model_rf <- randomForest(sold ~ ., data = train, importance = T)
predict_rf  <- predict(model_rf, newdata = test, type = "prob")[,2]
ROCRpred = prediction(predict_rf, test$sold)
as.numeric(performance(ROCRpred, "auc")@y.values)
```

```
## [1] 0.8608979
```

, . . .

```
varImpPlot(model_rf)
```



, , . .

```
set.seed(1000)
model_rf2 <- randomForest(sold ~ .-excel, data = train, importance = T)
predict_rf  <- predict(model_rf2, newdata = test, type = "prob")[,2]
ROCRpred = prediction(predict_rf, test$sold)
as.numeric(performance(ROCRpred, "auc")@y.values)
```

```
## [1] 0.8587726
```

, , , , , excel , , , ( ) . , .
180 1500 .