eBay-shoe sales

2023-04-02

```
knitr::opts_chunk$set(echo = TRUE)

#Predicting Sales on eBay
##This analysis aims to predict shoe sales on eBay using relevant variables such as starting/listed pri

##Let's first have a look at the data.

ebay = read.csv("ebayA.csv")

summary(ebay)
```

```
##     biddable           sold          startprice       saleprice
##  Min.   :0.0000   Min.   :0.0000   Min.   :   0.0   Min.   :   0.0
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.: 280.0   1st Qu.: 200.0
##  Median :1.0000   Median :0.0000   Median : 449.0   Median : 325.0
##  Mean   :0.5911   Mean   :0.2105   Mean   : 472.3   Mean   : 372.7
##  3rd Qu.:1.0000   3rd Qu.:0.0000   3rd Qu.: 600.0   3rd Qu.: 500.0
##  Max.   :1.0000   Max.   :1.0000   Max.   :4500.0   Max.   :3995.0
##                                                     NA's   :2997
##   condition              size           heel              style
##  Length:3796        Min.   : 4.000   Length:3796        Length:3796
##  Class :character   1st Qu.: 7.000   Class :character   Class :character
##  Mode  :character   Median : 8.000   Mode  :character   Mode  :character
##                     Mean   : 7.933
##                     3rd Qu.: 9.000
##                     Max.   :12.000
##                     NA's   :68
##     color              material           snippit          description
##  Length:3796        Length:3796        Length:3796        Length:3796
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
```

```
##Based on the summary, we can see that "saleprice" and "size" have 2997 and 68 missing values respecti

##What proportion of all shoes were sold?
mean(ebay$sold)
```

```
## [1] 0.2104847
```

```
##Around 21.04% of the listed shoes are sold.

##What is the most common shoe size in the data set?
```

```
table(ebay$size)
```

```
##
##    4  4.5    5  5.5    6  6.5    7  7.5    8  8.5    9  9.5   10 10.5   11 11.5
##   14   28   87  154  264  356  402  413  442  385  397  298  235  139   79   21
##   12
##   14
```

```
##We can see that size 8 is the most frequently listed size.

##It is a good idea to create a copy of the original data set before we make any changes.

ebay_1 = ebay

##Before we start building a model, let's change the variable types to categorical/factor for some of t


cat_cols = c('biddable', 'sold', 'condition', 'heel', 'style', 'color')
ebay[cat_cols] = lapply(ebay[cat_cols], factor)
summary(ebay)
```

```
##  biddable sold       startprice       saleprice                condition
##  0:1552   0:2997   Min.   :   0.0   Min.   :   0.0   New with box    :1092
##  1:2244   1: 799   1st Qu.: 280.0   1st Qu.: 200.0   New with defects:  80
##                    Median : 449.0   Median : 325.0   New without box : 257
##                    Mean   : 472.3   Mean   : 372.7   Pre-owned       :2367
##                    3rd Qu.: 600.0   3rd Qu.: 500.0
##                    Max.   :4500.0   Max.   :3995.0
##                                     NA's   :2997
##       size            heel                style                color
##  Min.   : 4.000         : 961   Open Toe     : 398   Beige        : 411
##  1st Qu.: 7.000   Flat  :  24   Other/Missing: 486   Black        :1330
##  Median : 8.000   High  :2610   Platform     : 622   Brown        : 174
##  Mean   : 7.933   Low   :  38   Pump         :1719   Other/Missing:1689
##  3rd Qu.: 9.000   Medium: 163   Slingback    : 342   Red          : 192
##  Max.   :12.000                 Stiletto     : 229
##  NA's   :68
##    material           snippit          description
##  Length:3796        Length:3796        Length:3796
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
```

```
##Now, let's move on to splitting the data in train and test data sets to perform logistic regression.

set.seed(123)
library(caTools)

##We have chosen to make a 70-30 divide for train-test sets.

spl = sample.split(ebay$sold, 0.7)
ebay_train = subset(ebay, spl == TRUE)
```

```r
ebay_test = subset(ebay, spl == FALSE)

##Fit the logistic regression model to classify if a shoes into two categories, sold = 0 and sold = 1.

##For the variable 'size', while there are 68 missing values, it is still usable in the model. So, we w

##Model 1 - without the variable 'size':

glmebay1 = glm(sold~biddable + startprice + condition + heel + style + color + material, data=ebay_trai

summary(glmebay1)
```

```
##
## Call:
## glm(formula = sold ~ biddable + startprice + condition + heel +
##     style + color + material, family = "binomial", data = ebay_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6974  -0.6858  -0.4890  -0.1962   6.1573
##
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 1.0083756  0.3367056   2.995  0.00275 **
## biddable1                   0.0287520  0.1128902   0.255  0.79896
## startprice                 -0.0047507  0.0003126 -15.198  < 2e-16 ***
## conditionNew with defects -0.0263347  0.3653528  -0.072  0.94254
## conditionNew without box  -0.3563656  0.2364984  -1.507  0.13185
## conditionPre-owned        -0.5780679  0.1414537  -4.087 4.38e-05 ***
## heelFlat                  -1.2529858  0.8081024  -1.551  0.12101
## heelHigh                   0.0842399  0.1369821   0.615  0.53857
## heelLow                   -1.5350902  0.6436781  -2.385  0.01709 *
## heelMedium                -0.5198310  0.2618301  -1.985  0.04710 *
## styleOther/Missing         0.2792610  0.2133055   1.309  0.19046
## stylePlatform             -0.3990969  0.2114349  -1.888  0.05908 .
## stylePump                  0.3160880  0.1802039   1.754  0.07942 .
## styleSlingback            -0.6105096  0.2574217  -2.372  0.01771 *
## styleStiletto              0.7868008  0.2577974   3.052  0.00227 **
## colorBlack                 0.1815069  0.1763139   1.029  0.30327
## colorBrown                -0.5050025  0.2866906  -1.761  0.07816 .
## colorOther/Missing        -0.3865016  0.1819704  -2.124  0.03367 *
## colorRed                  -0.1595420  0.2812716  -0.567  0.57057
## materialOther/Missing     -0.0749871  0.1544320  -0.486  0.62727
## materialPatent Leather     0.0236551  0.1450325   0.163  0.87044
## materialSatin             -0.9392876  0.3131491  -2.999  0.00270 **
## materialSnakeskin          0.4026441  0.3435472   1.172  0.24119
## materialSuede             -0.1998894  0.1798604  -1.111  0.26641
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2733.9  on 2656  degrees of freedom
## Residual deviance: 2342.0  on 2633  degrees of freedom
```

```
## AIC: 2390
##
## Number of Fisher Scoring iterations: 5
```

```
##Based on the model 1 summary, we can see that there are 9 significant variables in total (including t

##Also, the residual deviance is 2363.7 with AIC of 2411.7.

##Model 2 - including the variable 'size':

glmebay2 = glm(sold~biddable + startprice + condition + size + heel + style + color + material, data=eba

summary(glmebay2)
```

```
##
## Call:
## glm(formula = sold ~ biddable + startprice + condition + size +
##     heel + style + color + material, family = "binomial", data = ebay_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7747  -0.6843  -0.4880  -0.2017   6.1983
##
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 0.3359285  0.4502248   0.746  0.45559
## biddable1                   0.0099339  0.1155798   0.086  0.93151
## startprice                 -0.0047647  0.0003237 -14.721  < 2e-16 ***
## conditionNew with defects -0.0287031  0.3644993  -0.079  0.93723
## conditionNew without box  -0.3587012  0.2445830  -1.467  0.14249
## conditionPre-owned        -0.5938155  0.1444778  -4.110 3.96e-05 ***
## size                       0.0805886  0.0349811   2.304  0.02124 *
## heelFlat                  -1.1522064  0.8089656  -1.424  0.15436
## heelHigh                   0.1266880  0.1403524   0.903  0.36672
## heelLow                   -1.4663350  0.6473245  -2.265  0.02350 *
## heelMedium                -0.4920847  0.2667711  -1.845  0.06510 .
## styleOther/Missing         0.2523681  0.2201862   1.146  0.25173
## stylePlatform             -0.4381921  0.2141719  -2.046  0.04076 *
## stylePump                  0.3265594  0.1809778   1.804  0.07117 .
## styleSlingback            -0.6001858  0.2583467  -2.323  0.02017 *
## styleStiletto              0.7942435  0.2585313   3.072  0.00213 **
## colorBlack                 0.2201951  0.1769794   1.244  0.21343
## colorBrown                -0.4512851  0.2879151  -1.567  0.11702
## colorOther/Missing        -0.3641413  0.1837822  -1.981  0.04755 *
## colorRed                  -0.1407536  0.2837186  -0.496  0.61982
## materialOther/Missing     -0.0513600  0.1566626  -0.328  0.74303
## materialPatent Leather     0.0124869  0.1459659   0.086  0.93183
## materialSatin             -0.9582342  0.3136408  -3.055  0.00225 **
## materialSnakeskin          0.4170982  0.3460541   1.205  0.22809
## materialSuede             -0.2722752  0.1849989  -1.472  0.14108
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 2659.5  on 2611  degrees of freedom
## Residual deviance: 2282.0  on 2587  degrees of freedom
##   (45 observations deleted due to missingness)
## AIC: 2332
##
## Number of Fisher Scoring iterations: 5
```

```
##Based on the model 1 summary, we can see that there are 8 significant variables in total (including t

##Also, the residual deviance is 2310 with AIC of 2360. So, the variable 'biddable = 0' has become less

##Now, let's first remove observations with missing entries in 'size'. (In this particular case, imputi

ebay_train = ebay_train[!is.na(ebay_train[,"size"]),]

##Model 3 - after removing NAs from 'size':

glmebay3 = glm(sold~biddable + startprice + condition + size + heel + style + color + material, data=eba
summary(glmebay3)
```

```
##
## Call:
## glm(formula = sold ~ biddable + startprice + condition + size +
##     heel + style + color + material, family = "binomial", data = ebay_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7747  -0.6843  -0.4880  -0.2017   6.1983
##
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 0.3359285  0.4502248   0.746  0.45559
## biddable1                   0.0099339  0.1155798   0.086  0.93151
## startprice                 -0.0047647  0.0003237 -14.721  < 2e-16 ***
## conditionNew with defects  -0.0287031  0.3644993  -0.079  0.93723
## conditionNew without box   -0.3587012  0.2445830  -1.467  0.14249
## conditionPre-owned         -0.5938155  0.1444778  -4.110 3.96e-05 ***
## size                        0.0805886  0.0349811   2.304  0.02124 *
## heelFlat                   -1.1522064  0.8089656  -1.424  0.15436
## heelHigh                    0.1266880  0.1403524   0.903  0.36672
## heelLow                    -1.4663350  0.6473245  -2.265  0.02350 *
## heelMedium                 -0.4920847  0.2667711  -1.845  0.06510 .
## styleOther/Missing          0.2523681  0.2201862   1.146  0.25173
## stylePlatform              -0.4381921  0.2141719  -2.046  0.04076 *
## stylePump                   0.3265594  0.1809778   1.804  0.07117 .
## styleSlingback             -0.6001858  0.2583467  -2.323  0.02017 *
## styleStiletto               0.7942435  0.2585313   3.072  0.00213 **
## colorBlack                  0.2201951  0.1769794   1.244  0.21343
## colorBrown                 -0.4512851  0.2879151  -1.567  0.11702
## colorOther/Missing         -0.3641413  0.1837822  -1.981  0.04755 *
## colorRed                   -0.1407536  0.2837186  -0.496  0.61982
## materialOther/Missing      -0.0513600  0.1566626  -0.328  0.74303
## materialPatent Leather      0.0124869  0.1459659   0.086  0.93183
## materialSatin              -0.9582342  0.3136408  -3.055  0.00225 **
## materialSnakeskin           0.4170982  0.3460541   1.205  0.22809
```

```
## materialSuede             -0.2722752  0.1849989  -1.472  0.14108
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2659.5  on 2611  degrees of freedom
## Residual deviance: 2282.0  on 2587  degrees of freedom
## AIC: 2332
##
## Number of Fisher Scoring iterations: 5
```

```
##As can be seen in the output, the result is exactly the same as Model 2. So, we will use the Model 2

glmpred = predict(glmebay2, newdata=ebay_test, type="response")

##Confusion Matrix:
(confu_mat = table(ebay_test$sold, glmpred >= 0.5))
```

```
##
##     FALSE TRUE
##   0   870   17
##   1   186   43
```

```
(pred_acc = (confu_mat[1,1] + confu_mat[2,2])/sum(confu_mat))
```

```
## [1] 0.8181004
```

```
##Model 2 has a prediction accuracy of 82.41%.

##Now, let's move on to building a CART model to check the variables of importance.

library(rpart)
library(rpart.plot)

CARTebay = rpart(formula = sold ~ biddable + startprice + condition + size + heel + style + color + mat

##cp= 0.005 was chosen instead of default 0.01 because we want a deeper tree to get a better idea.

##Based on the output, we can see that the variable 'startprice' is the most important variable followe

##Now, let's fit a RandomForest model and see if this variable importance order still holds.

library(randomForest)
```

```
## randomForest 4.7-1.1
```

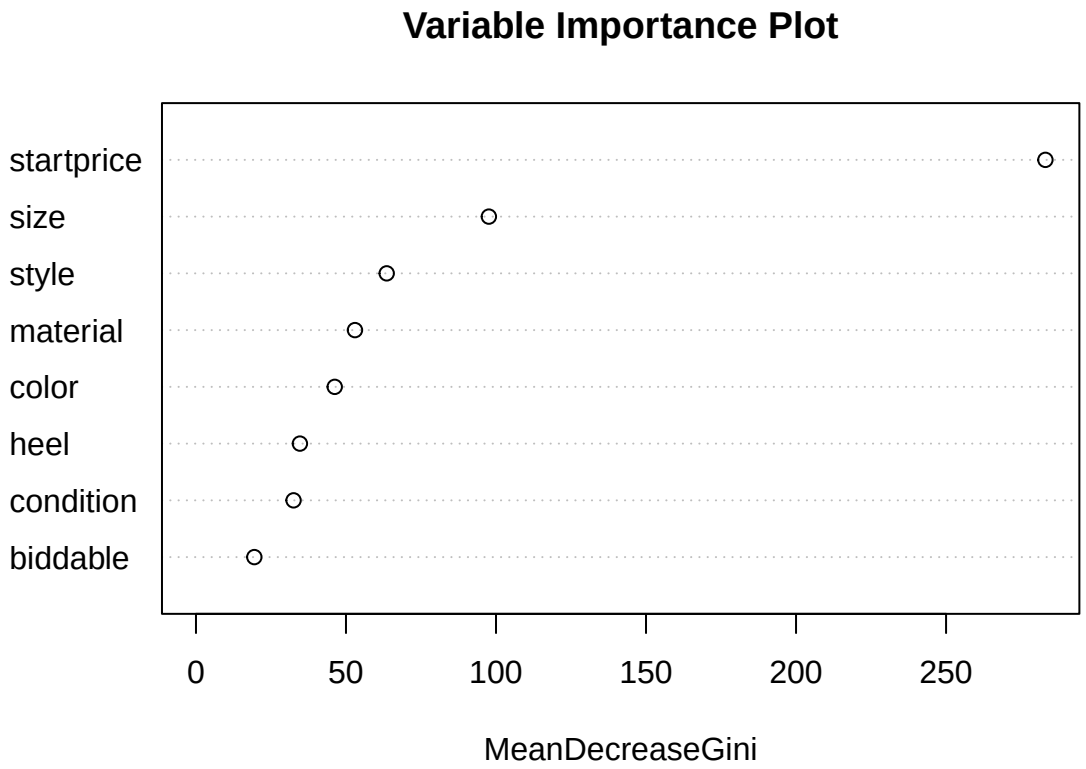
```
## Type rfNews() to see new features/changes/bug fixes.
```

```
ebay_rf = randomForest(sold~biddable + startprice + condition + size + heel + style + color + material,
ebay_rf
```

```
##
## Call:
##  randomForest(formula = sold ~ biddable + startprice + condition +      size + heel + style + color
##                Type of random forest: classification
##                      Number of trees: 500
```

```
## No. of variables tried at each split: 2
##
##         OOB estimate of  error rate: 17.38%
## Confusion matrix:
##      0   1 class.error
## 0 2022  51  0.02460203
## 1  403 136  0.74768089
```

```
varImpPlot(ebay_rf, main = "Variable Importance Plot")
```




```
##From the plot, 'startprice' comes out on the top as the most important variable in predicting the cla

##After this analysis, we can make following remarks regarding the shoe sales on eBay:

##Start price of the product matters the most when selling shoes. If it is more than 104, there is prac

##Following combinations of the variables make up most of the sold shoes: a. Price= 22-104, style = Oth
```