Classification

Benton Fariss

2023-02-18

#Dataset from: https://archive.ics.uci.edu/ml/datasets/Online+Shoppers+Purchasing+Intention+Dataset ### How do linear models for classification work? Logistic regression is used to model the relationship between a categorical dependent variable and a set of independent variables. We want to find linear boundaries between classes of data that allow us to predict the probability the probability of independent variables belonging to a specific class. Linear models for classification are best fit for large data sets and are very efficient and easily observable. #load data

```
shoppers <- read.csv("C:/Users/setup/Downloads/online_shoppers_intention.csv")
```

A. Split the Data into 80 train and 20 test

```
set.seed(1234)
i <- sample(1:nrow(shoppers), nrow(shoppers)*.8, replace=FALSE)
train <- shoppers[i,]
test <- shoppers[-i,]
```

B. Use 5 R functions for data exploration

This displays the structures of objects.

```
str(train)
```

```
## 'data.frame':    9864 obs. of  18 variables:
##  $ Administrative         : int  4 1 0 0 3 11 0 4 0 0 ...
##  $ Administrative_Duration: num  95.8 6.5 0 0 423 ...
##  $ Informational          : int  2 0 0 0 0 4 0 0 0 0 ...
##  $ Informational_Duration : num  35.7 0 0 0 0 ...
##  $ ProductRelated         : int  14 10 1 2 24 397 16 21 1 5 ...
##  $ ProductRelated_Duration: num  380 511 0 17 1204 ...
##  $ BounceRates            : num  0 0 0.2 0 0 ...
##  $ ExitRates              : num  0.01111 0.00741 0.2 0.1 0.01111 ...
##  $ PageValues             : num  0 7.85 0 0 0 ...
##  $ SpecialDay             : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Month                  : chr  "Oct" "Dec" "Jul" "Dec" ...
##  $ OperatingSystems       : int  2 2 1 2 4 1 2 3 2 2 ...
##  $ Browser                : int  4 5 1 2 1 1 5 2 2 2 ...
##  $ Region                 : int  7 3 3 1 1 3 7 1 1 3 ...
##  $ TrafficType            : int  2 8 4 10 4 3 1 2 10 1 ...
##  $ VisitorType            : chr  "New_Visitor" "New_Visitor" "Returning_Visitor" "Returning_Visitor"
##  $ Weekend                : logi  FALSE TRUE TRUE FALSE FALSE TRUE ...
##  $ Revenue                : logi  FALSE TRUE FALSE FALSE FALSE TRUE ...
```

This displays the first parts of the train dataset.

```r
head(train)
```

```
##      Administrative Administrative_Duration Informational
## 7452              4                  95.800             2
## 8016              1                   6.500             0
## 7162              0                   0.000             0
## 8086              0                   0.000             0
## 7269              3                 423.000             0
## 9196             11                 298.082             4
##      Informational_Duration ProductRelated ProductRelated_Duration BounceRates
## 7452                   35.7             14                380.2667 0.000000000
## 8016                    0.0             10                511.2500 0.000000000
## 7162                    0.0              1                  0.0000 0.200000000
## 8086                    0.0              2                 17.0000 0.000000000
## 7269                    0.0             24               1203.5333 0.000000000
## 9196                  138.5            397              11940.0165 0.006959671
##        ExitRates PageValues SpecialDay Month OperatingSystems Browser Region
## 7452 0.011111111   0.000000          0   Oct                2       4      7
## 8016 0.007407407   7.848539          0   Dec                2       5      3
## 7162 0.200000000   0.000000          0   Jul                1       1      3
## 8086 0.100000000   0.000000          0   Dec                2       2      1
## 7269 0.011111111   0.000000          0   Oct                4       1      1
## 9196 0.009804129   9.221243          0   Nov                1       1      3
##      TrafficType       VisitorType Weekend Revenue
## 7452           2       New_Visitor   FALSE   FALSE
## 8016           8       New_Visitor    TRUE    TRUE
## 7162           4 Returning_Visitor    TRUE   FALSE
## 8086          10 Returning_Visitor   FALSE   FALSE
## 7269           4       New_Visitor   FALSE   FALSE
## 9196           3 Returning_Visitor    TRUE    TRUE
```

This displays the last parts of the train dataset.

```r
tail(train)
```

```
##       Administrative Administrative_Duration Informational
## 6451               7                  667.25             0
## 5698               7                  199.85             0
## 6493               0                    0.00             0
## 6796               0                    0.00             0
## 10797              0                    0.00             0
## 4529               0                    0.00             0
##       Informational_Duration ProductRelated ProductRelated_Duration BounceRates
## 6451                       0              4                101.8000 0.000000000
## 5698                       0             39               1267.4533 0.004761905
## 6493                       0              2                 16.2000 0.000000000
## 6796                       0              2                  0.0000 0.200000000
## 10797                      0              5                 99.0000 0.000000000
## 4529                       0             44                831.6083 0.000000000
##        ExitRates PageValues SpecialDay Month OperatingSystems Browser Region
## 6451  0.01000000    0.00000          0   Jul                3       2      6
## 5698  0.02063492   17.92759          0   Nov                2       4      1
## 6493  0.10000000    0.00000          0   Oct                1       1      1
## 6796  0.20000000    0.00000          0   Oct                3       2      1
## 10797 0.04000000    0.00000          0   Nov                2       4      1
```

```
## 4529  0.01317829    0.00000          0   May                2       2      2
##       TrafficType       VisitorType Weekend Revenue
## 6451            2 Returning_Visitor    TRUE   FALSE
## 5698            4 Returning_Visitor   FALSE   FALSE
## 6493            2       New_Visitor   FALSE   FALSE
## 6796            1 Returning_Visitor    TRUE   FALSE
## 10797           3 Returning_Visitor   FALSE   FALSE
## 4529            2 Returning_Visitor   FALSE   FALSE
```

This displays the names of the objects in the train dataset.

```
names(train)
```

```
##  [1] "Administrative"          "Administrative_Duration"
##  [3] "Informational"           "Informational_Duration"
##  [5] "ProductRelated"          "ProductRelated_Duration"
##  [7] "BounceRates"             "ExitRates"
##  [9] "PageValues"              "SpecialDay"
## [11] "Month"                   "OperatingSystems"
## [13] "Browser"                 "Region"
## [15] "TrafficType"             "VisitorType"
## [17] "Weekend"                 "Revenue"
```

This displays the dimensions of the train dataset.

```
dim(train)
```

```
## [1] 9864   18
```

This displays the number of rows in the train dataset.

```
nrow(train)
```

```
## [1] 9864
```

This displays the number of columns in the train dataset.

```
ncol(train)
```

```
## [1] 18
```

This displays the summaries of all of the values data frames.

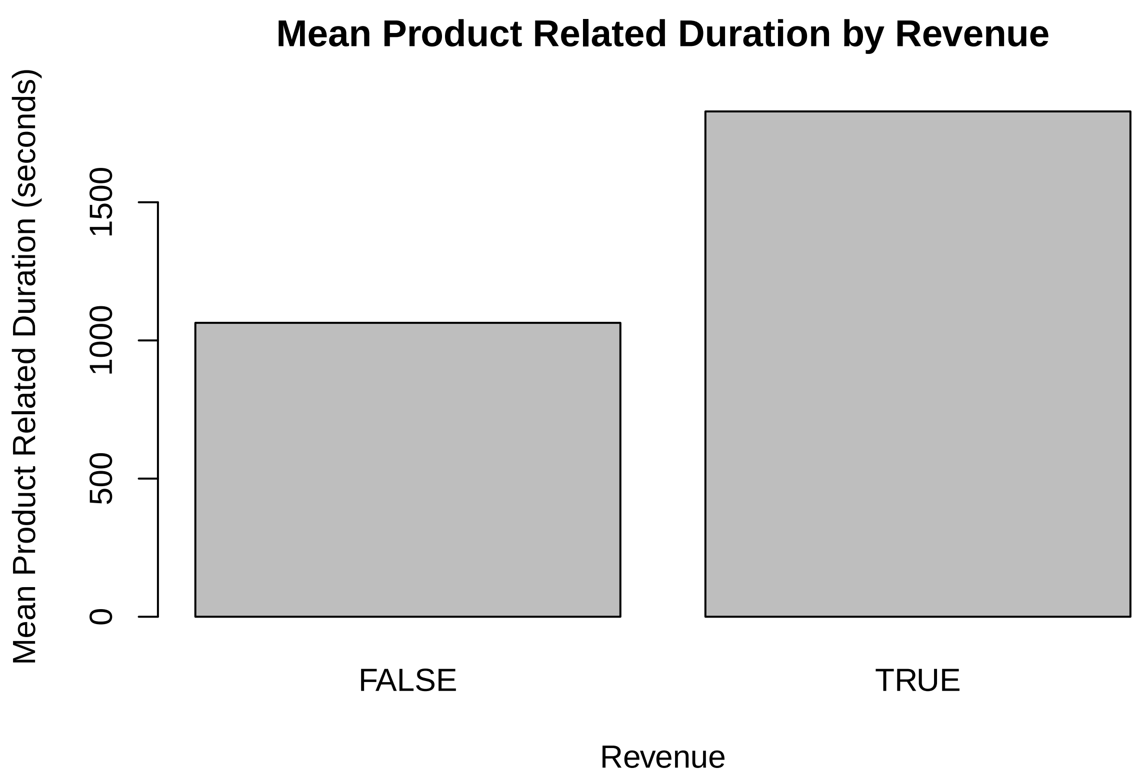
```
summary(train)
```

```
##  Administrative   Administrative_Duration Informational
##  Min.   : 0.000   Min.   :   0.00         Min.   : 0.0000
##  1st Qu.: 0.000   1st Qu.:   0.00         1st Qu.: 0.0000
##  Median : 1.000   Median :   7.00         Median : 0.0000
##  Mean   : 2.322   Mean   :  82.32         Mean   : 0.5022
##  3rd Qu.: 4.000   3rd Qu.:  94.60         3rd Qu.: 0.0000
##  Max.   :27.000   Max.   :3398.75         Max.   :24.0000
##  Informational_Duration ProductRelated   ProductRelated_Duration
##  Min.   :   0.00        Min.   :  0.00   Min.   :    0.0
##  1st Qu.:   0.00        1st Qu.:  7.00   1st Qu.:  181.2
##  Median :   0.00        Median : 17.00   Median :  590.9
##  Mean   :  33.84        Mean   : 31.42   Mean   : 1182.4
##  3rd Qu.:   0.00        3rd Qu.: 37.00   3rd Qu.: 1438.0
##  Max.   :2549.38        Max.   :705.00   Max.   :63973.5
##   BounceRates          ExitRates          PageValues        SpecialDay
##  Min.   :0.000000   Min.   :0.00000   Min.   :  0.000   Min.   :0.00000
```

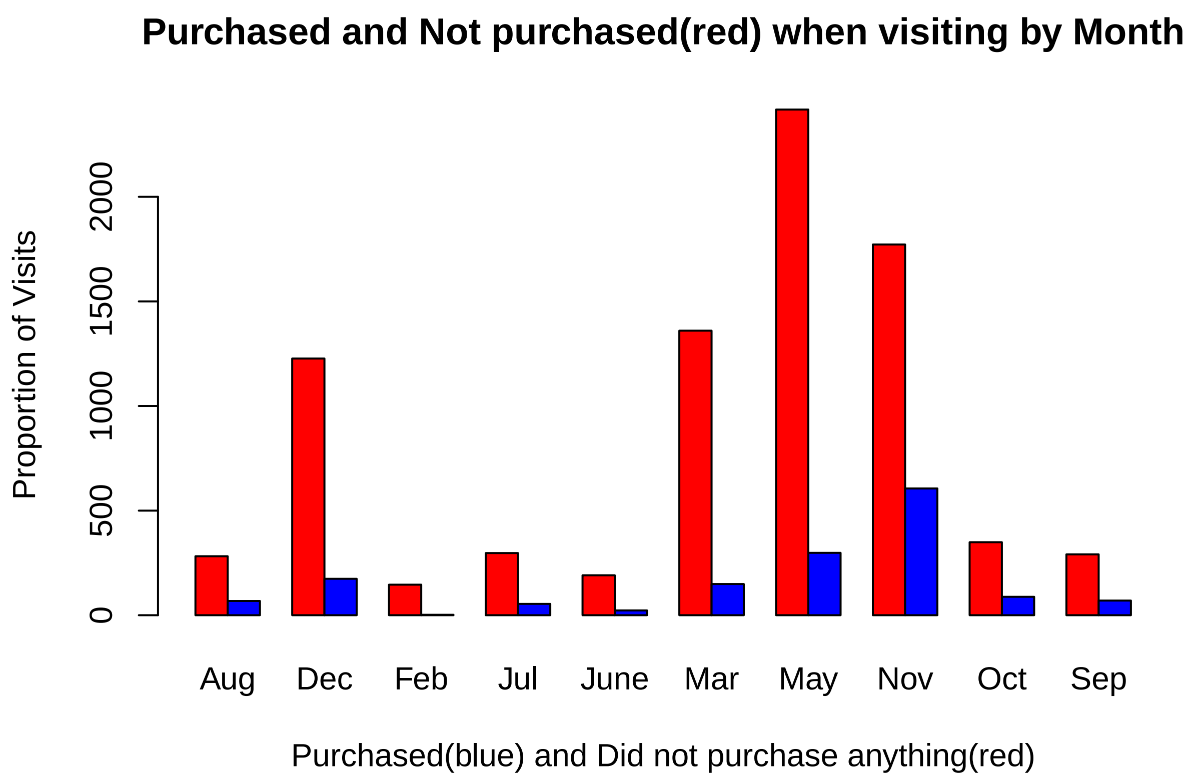
```
##  1st Qu.:0.000000   1st Qu.:0.01429   1st Qu.:  0.000   1st Qu.:0.00000
##  Median :0.003074   Median :0.02500   Median :  0.000   Median :0.00000
##  Mean   :0.022441   Mean   :0.04332   Mean   :  5.816   Mean   :0.06251
##  3rd Qu.:0.016667   3rd Qu.:0.05000   3rd Qu.:  0.000   3rd Qu.:0.00000
##  Max.   :0.200000   Max.   :0.20000   Max.   :360.953   Max.   :1.00000
##     Month           OperatingSystems    Browser            Region
##  Length:9864        Min.   :1.000    Min.   : 1.000   Min.   :1.000
##  Class :character   1st Qu.:2.000    1st Qu.: 2.000   1st Qu.:1.000
##  Mode  :character   Median :2.000    Median : 2.000   Median :3.000
##                     Mean   :2.127    Mean   : 2.354   Mean   :3.161
##                     3rd Qu.:3.000    3rd Qu.: 2.000   3rd Qu.:4.000
##                     Max.   :8.000    Max.   :13.000   Max.   :9.000
##   TrafficType     VisitorType          Weekend         Revenue
##  Min.   : 1.000   Length:9864        Mode :logical   Mode :logical
##  1st Qu.: 2.000   Class :character   FALSE:7594      FALSE:8332
##  Median : 2.000   Mode  :character   TRUE :2270      TRUE :1532
##  Mean   : 4.052
##  3rd Qu.: 4.000
##  Max.   :20.000
```

C. Create informative graphs using the training data

```
# Mean Product Related Duration by Revenue
mean_dur_by_revenue <- tapply(train$ProductRelated_Duration, train$Revenue, mean)
barplot(mean_dur_by_revenue, main = "Mean Product Related Duration by Revenue",
        xlab = "Revenue", ylab = "Mean Product Related Duration (seconds)")
```



```r
# Revenue by Month
revenue_by_month <- table(train$Revenue, train$Month)
barplot(revenue_by_month, beside = TRUE, main = "Purchased and Not purchased(red) when visiting by Montl
        xlab = "Purchased(blue) and Did not purchase anything(red)", ylab = "Proportion of Visits", col
```



D. Build a logistic regression model, output summary, and explain

```r
glm1 <- glm(Revenue~ProductRelated_Duration, data=train, family="binomial")
summary(glm1)
```

```
##
## Call:
## glm(formula = Revenue ~ ProductRelated_Duration, family = "binomial",
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -4.2882  -0.5691  -0.5356  -0.5224   2.0318
##
## Coefficients:
##                           Estimate Std. Error z value Pr(>|z|)
## (Intercept)             -1.9282885  0.0343721  -56.10   <2e-16 ***
## ProductRelated_Duration  0.0001739  0.0000134   12.97   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8518.8  on 9863  degrees of freedom
## Residual deviance: 8346.4  on 9862  degrees of freedom
## AIC: 8350.4
##
## Number of Fisher Scoring iterations: 4
```

Deviance Residuals show the difference between the observed and predicted models. This means that when these numbers are lower, there is a low different in the predicted and actual probability. Coefficients show the change in the log odds of y for every 1 unit predictor change. The p-value is only acceptable if it is below .05. Null deviance is the measure of the response variable's entire variability. This is done using only the model's intercept. Residual deviance is the measure of the response variable's unexplained variability . Our Null and Residual deviance are not amazing, but they are okay.

E. Build a naïve Bayes model, output what the model learned, and explain

```
library(e1071)
nb_model <- naiveBayes(Revenue~., data=train)
nb_model
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##     FALSE      TRUE
## 0.8446878 0.1553122
##
## Conditional probabilities:
##        Administrative
## Y           [,1]     [,2]
##   FALSE 2.126380 3.234706
##   TRUE  3.388381 3.719238
##
##        Administrative_Duration
## Y             [,1]     [,2]
##   FALSE  75.19256 174.2223
##   TRUE  121.06403 207.6091
##
##        Informational
## Y             [,1]     [,2]
##   FALSE 0.4470715 1.207828
##   TRUE  0.8022193 1.552283
##
##        Informational_Duration
## Y           [,1]     [,2]
##   FALSE 29.36686 130.2973
##   TRUE  58.17652 171.1713
##
##        ProductRelated
## Y           [,1]     [,2]
```

```
##   FALSE 28.50132 41.15672
##   TRUE  47.31593 55.70893
##
##        ProductRelated_Duration
## Y            [,1]     [,2]
##   FALSE 1063.578 1834.423
##   TRUE  1828.680 2158.020
##
##        BounceRates
## Y               [,1]       [,2]
##   FALSE 0.025613451 0.05246309
##   TRUE  0.005188415 0.01247293
##
##        ExitRates
## Y              [,1]       [,2]
##   FALSE 0.04764557 0.05175772
##   TRUE  0.01977131 0.01663370
##
##        PageValues
## Y             [,1]     [,2]
##   FALSE  1.993063  8.89963
##   TRUE  26.610956 34.23888
##
##        SpecialDay
## Y              [,1]      [,2]
##   FALSE 0.06985118 0.2102598
##   TRUE  0.02258486 0.1215472
##
##        Month
## Y               Aug         Dec         Feb         Jul        June         Mar
##   FALSE 0.033845415 0.147263562 0.017522804 0.035645703 0.022923668 0.163226116
##   TRUE  0.044386423 0.113577023 0.001305483 0.035248042 0.015013055 0.097258486
##        Month
## Y               May         Nov         Oct         Sep
##   FALSE 0.290086414 0.212674028 0.041886702 0.034925588
##   TRUE  0.194516971 0.395561358 0.057441253 0.045691906
##
##        OperatingSystems
## Y            [,1]      [,2]
##   FALSE 2.134061 0.9100232
##   TRUE  2.087467 0.9293387
##
##        Browser
## Y            [,1]     [,2]
##   FALSE 2.337494 1.680413
##   TRUE  2.443211 1.869704
##
##        Region
## Y            [,1]     [,2]
##   FALSE 3.178709 2.406651
##   TRUE  3.062663 2.408886
##
##        TrafficType
## Y            [,1]     [,2]
```

```
##   FALSE 4.058689 4.011900
##   TRUE  4.016319 4.018783
##
##        VisitorType
## Y        New_Visitor       Other Returning_Visitor
##   FALSE 0.122059530 0.006721075       0.871219395
##   TRUE  0.219973890 0.008485640       0.771540470
##
##        Weekend
## Y            FALSE      TRUE
##   FALSE 0.7744839 0.2255161
##   TRUE  0.7447781 0.2552219
```

```
summary(nb_model)
```

```
##           Length Class  Mode
## apriori    2     table  numeric
## tables    17     -none- list
## levels     2     -none- character
## isnumeric 17     -none- logical
## call       4     -none- call
```

This information tells us is the prior probabilities of Revenue were 84.47% False, and 15.53% True. This information allows us to much more easily make predictions on our new data.

F. Using these two classifications models models, predict and evaluate on the test data using all of the classification metrics from class. Compare and explain results.

```
probs <- predict(glm1, newdata=test, type="response")
pred <- ifelse(probs>.5,2,1)
acc1 <- mean(pred==as.integer(test$Revenue))
err1 <- 1-acc1
print(paste("glm1 accuracy= ", acc1))
```

```
## [1] "glm1 accuracy=  0.14963503649635"
```

```
print(paste("glm1 error: ", err1))
```

```
## [1] "glm1 error:  0.85036496350365"
```

```
table(pred, as.integer(test$Revenue))
```

```
##
## pred    0    1
##    1 2084  369
##    2    6    7
```

```
probs2 <- predict(nb_model, newdata=test, type="class")
acc2 <- mean(probs2==test$Revenue)
err2 <- 1-acc2
print(paste("nb_model accuracy:", acc2))
```

```
## [1] "nb_model accuracy: 0.823195458231955"
```

```
print(paste("nb_model error: ", err2))
```

```
## [1] "nb_model error:  0.176804541768045"
```

```
table(probs2, test$Revenue)
```

```
##
## probs2  FALSE TRUE
##   FALSE  1758  104
##   TRUE    332  272
```

The higher accuracy comes from the Naive Bayes model compared to the logistic regression. The accuracy of the logistic regression was .1496 and the accuracy of the Naive Bayes model was .8232. The logistic regression model did very poorly at about 15%. Our table shows us that this method should not even be consider, especially when comparing it to the vast difference in accuracy when using the NB model.

G. Write a paragraph explaining the strengths and weaknesses of the Naïve Bayes and Logistic Regression

The problem with Naive Bayes is that it assumes that all features are independent, which can limit the performance of the algorithm. It also does not work as well with larger data sets as something like logistic regression would. The last primary weakness is the guesses that Naive Bayes makes in the test set that do not happen in the training. The positives of it is that it is a great classifier for smaller data sets, it is simple, and it is great to use for multidimensional tasks. The problem with logistic regression is that it is not work well with non-linear data, which leads to lower accuracy. The positives of logistic regression is that it works very well with linear data and is very cheap to run.

H. Write a paragraph listing the benefits, drawbacks of each of the classification metrics used

Accuracy measures the percentage of time that the classifier is predicting correctly. This is a great metric for the user to understand the data. Error rate shows us the percentage of time that the classifier is predicting incorrectly which can easily be calculated by the user, however it is important to see. This shows us the inaccuracies in the classifier and can help us decipher if it is significant. The confusion matrix shows a more complex performance analyzation using the true and false positives/negatives. This can also show us the accuracy and can help compute other metrcs such as sensitivity, specificity, and precision.