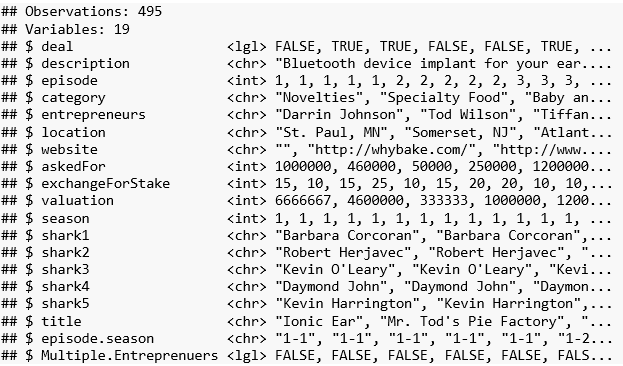
Introduction:

The data is loaded into the R console and the summary of all the variables present and data types of each variables is shown below.



Step 1 :

Creating a subset of the dataframe with description,deal and askedFor as column variables. As mentioned in the assignment question,a dataframe is created with description,deal and askedFor as column variables

Using the following code:-

*shark = read.csv("Shark Tank Companies.csv", stringsAsFactors=FALSE)*

*table(shark$deal)*

*install.packages('randomForest')*

*library(tm)*

*library(SnowballC)*

*library(wordcloud)*

*corpusShark = Corpus(VectorSource(shark$description))*

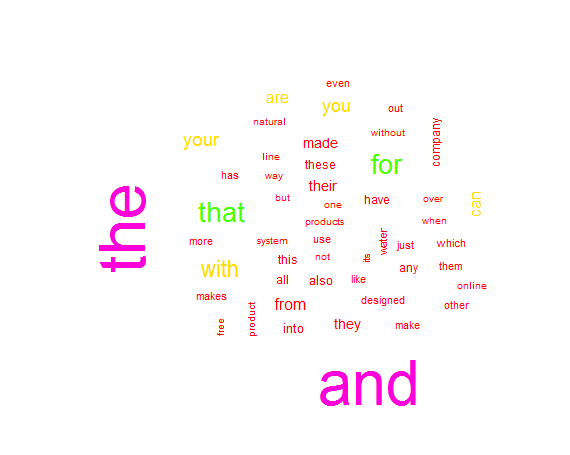
*wordcloud(corpusShark,colors=rainbow(7),max.words=50)*

*# Convert to lower-case*

*corpusShark = tm\_map(corpusShark, tolower)*

*corpusShark = tm\_map(corpusShark, removePunctuation)*

Wordcloud of words when deal is struck .



We have a subset of the data with deal and askedFor and the description column.An individual wordcloud representing the frequent terms if a deal is struck (deal =TRUE) and not struck (deal=FALE) is obtained as seen.

Creating Document Term Matrix

# Convert to lower-case

corpusShark = tm\_map(corpusShark, tolower)

corpusShark = tm\_map(corpusShark, removePunctuation)

# Remove stopwords

corpusShark = tm\_map(corpusShark, removeWords, c(stopwords("english")))

# Stem document

corpusShark = tm\_map(corpusShark, stemDocument)

frequenciesShark = DocumentTermMatrix(corpusShark)

sparseShark = removeSparseTerms(frequenciesShark, 0.995)

SharkSparse = as.data.frame(as.matrix(sparseShark))

# Make all variable names R-friendly

colnames(SharkSparse) = make.names(colnames(SharkSparse))

# Add dependent variable

SharkSparse$DV = shark$deal

SharkSparse$DV=as.factor(SharkSparse$DV)

Output

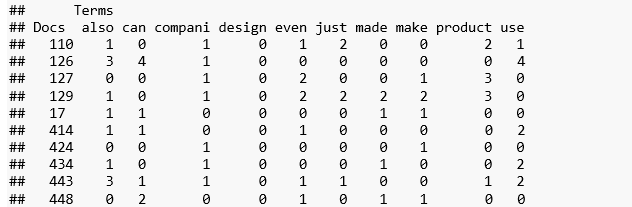
<<DocumentTermMatrix (documents: 495, terms: 3492)>>

Non-/sparse entries: 9519/1719021

Sparsity : 99%

Maximal term length: 21

Weighting : term frequency (tf)



Building the CART model

A dataframe is created with the document term matrix having deal as the dependent variable.

CART model is built on the data.

*# Build a CART model*

*library(rpart)*

*library(rpart.plot)*

*SharkCART = rpart(DV ~ ., data=SharkSparse, method="class")*

*prp(SharkCART,extra=2)*

*Output*

n= 495

node), split, n, loss, yval, (yprob)

\* denotes terminal node

1) root 495 244 TRUE (0.4929293 0.5070707)

2) shape< 0.5 485 241 FALSE (0.5030928 0.4969072)

4) weight< 0.5 478 234 FALSE (0.5104603 0.4895397)

8) roll< 0.5 471 227 FALSE (0.5180467 0.4819533)

16) easi< 0.5 450 211 FALSE (0.5311111 0.4688889)

32) made>=0.5 57 18 FALSE (0.6842105 0.3157895) \*

33) made< 0.5 393 193 FALSE (0.5089059 0.4910941)

66) regular< 0.5 386 186 FALSE (0.5181347 0.4818653)

132) easier< 0.5 377 178 FALSE (0.5278515 0.4721485)

264) children< 0.5 362 167 FALSE (0.5386740 0.4613260)

528) student< 0.5 355 161 FALSE (0.5464789 0.4535211)

1056) packag< 0.5 348 155 FALSE (0.5545977 0.4454023)

2112) new< 0.5 337 147 FALSE (0.5637982 0.4362018)

4224) turn< 0.5 329 141 FALSE (0.5714286 0.4285714)

8448) fun>=0.5 9 1 FALSE (0.8888889 0.1111111) \*

8449) fun< 0.5 320 140 FALSE (0.5625000 0.4375000)

16898) design< 0.5 286 120 FALSE (0.5804196 0.4195804) \*

16899) design>=0.5 34 14 TRUE (0.4117647 0.5882353) \*

4225) turn>=0.5 8 2 TRUE (0.2500000 0.7500000) \*

2113) new>=0.5 11 3 TRUE (0.2727273 0.7272727) \*

1057) packag>=0.5 7 1 TRUE (0.1428571 0.8571429) \*

529) student>=0.5 7 1 TRUE (0.1428571 0.8571429) \*

265) children>=0.5 15 4 TRUE (0.2666667 0.7333333) \*

133) easier>=0.5 9 1 TRUE (0.1111111 0.8888889) \*

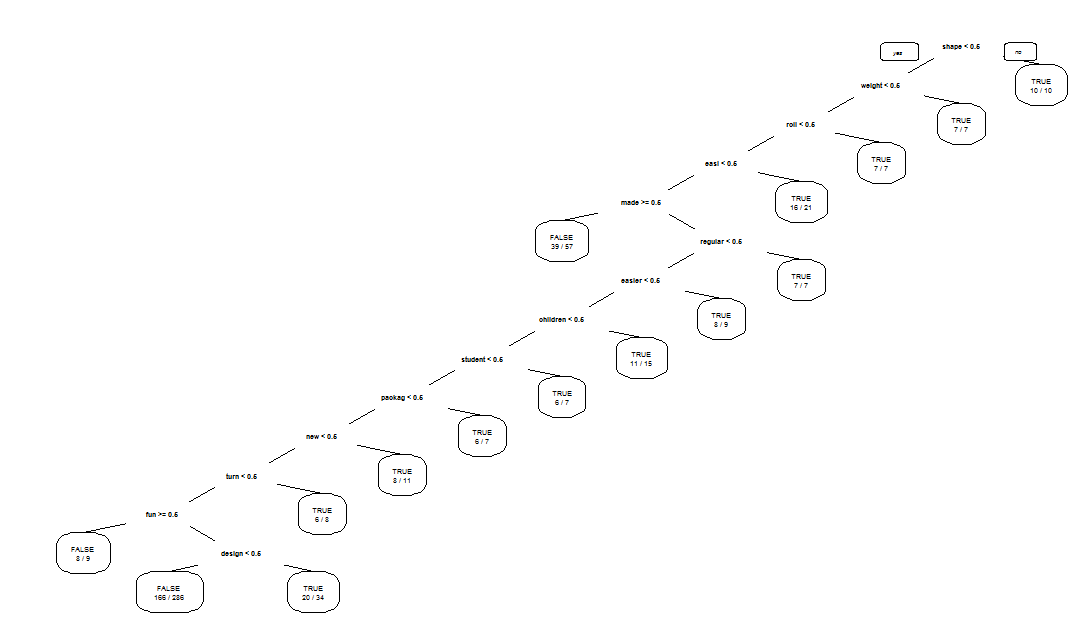
67) regular>=0.5 7 0 TRUE (0.0000000 1.0000000) \*

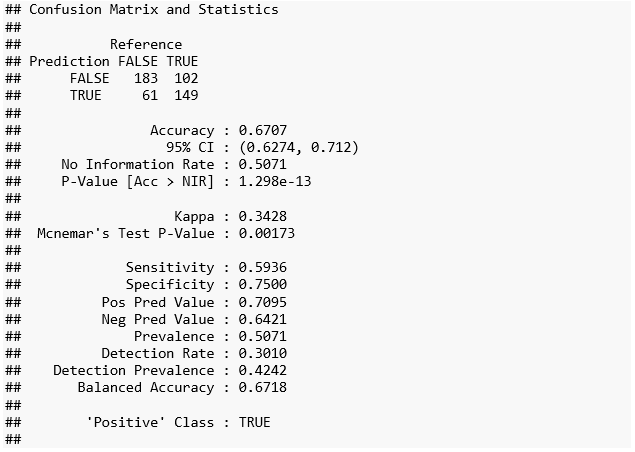
17) easi>=0.5 21 5 TRUE (0.2380952 0.7619048) \*

9) roll>=0.5 7 0 TRUE (0.0000000 1.0000000) \*

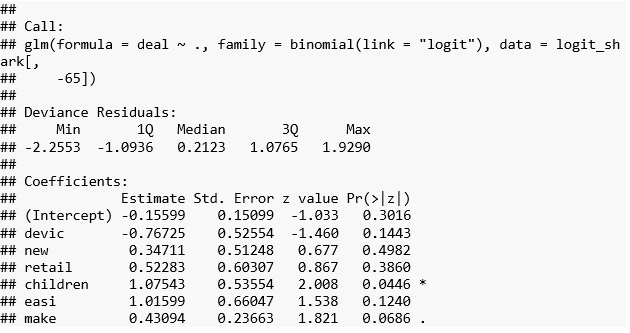
5) weight>=0.5 7 0 TRUE (0.0000000 1.0000000) \*

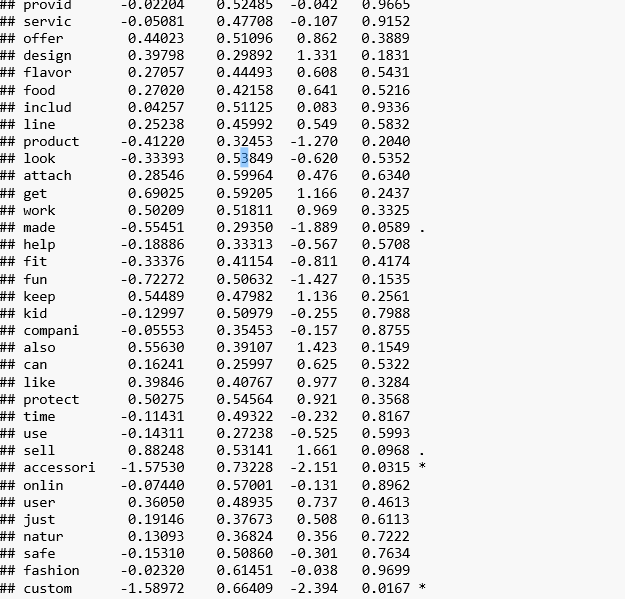
3) shape>=0.5 10 0 TRUE (0.0000000 1.0000000) \*

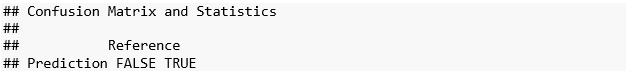


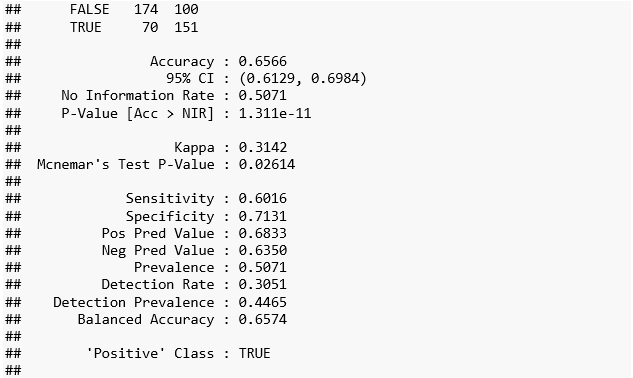


***The logistic regression model is build over the dataset having deal as binary dependent variable while the other terms being the independent variable.The summary of the model is as obtained below:***









**Building a Random Forest Model**

*library(randomForest)*

*SharkTankRF=randomForest(DV~.,data=SharkSparse)*

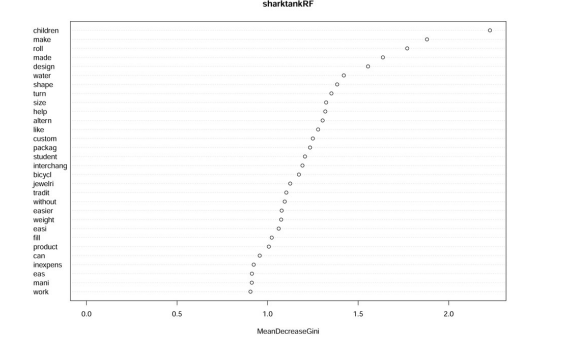
*varImpPlot(SharkTankRF)*

*SharkLogit=glm(DV~.,data=SharkSparse,family="binomial")*

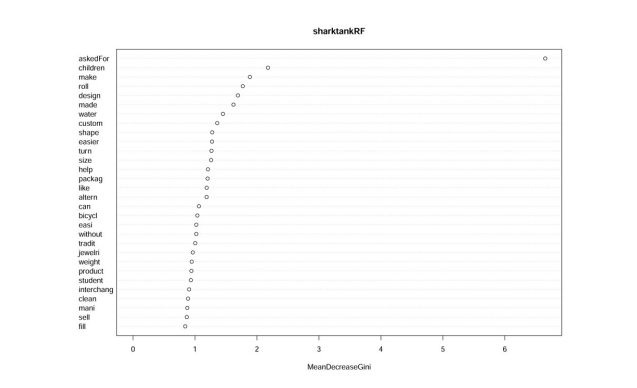
*SharkPred=predict(SharkLogit,data=SharkSparse,type="response")*

*table(SharkSparse$DV,SharkPred>0.5)*

*Random Forest (before)­­­*



*Random Forest (after)*



**Confusion Matrix Analysis**

Accuracy (TP + TN)/total- **Before**: 0.5051 || **After**:0.9939

Misclassification Rate (FP + FN)/total- **Before**: 0.4949||**After**: 0.0061

True Positive Rate TP/actual yes-**Before**: 0.4582||**After**: 0.9920

False Positive Rate FP/actual no-**Before**: 0.4467 ||**After**:0.0041

Specificity TN/actual no -**Before**:0.5533||**After**: 0.9959

Precision TP/predicted yes-**Before**: 0.5134||**After**: 0.9960

Prevalence actual yes/total -**Before**:0.5071||**After**: 0.5071

Classification Error (FP + FN)/(TP + TN + FP + FN) -0.4949

Interpretations to Follow

* The confusion matrix is most fundamental tool in assessing a two-category prediction model. The purpose of the confusion matrix is to compare the predictions of model with the known outcomes. Accuracy of the test depends on how well the test separates the group being tested True or False.
* Accuracy - can be measured by the area under the ROC curve. An area of 1 represents a perfect test; an area of .5 represents a worthless test. In some applications of ROC curves, you want the point closest to the TPR of ‘1’ and FPR of ‘0’. This cut point is “optimal” in the sense it weighs both sensitivity and specificity equally.
* *From the summary of the logistic model, only the word custom, accessori, sell, made, make, children are significant in building the model while others are insignificant.*
* *The Variable importance plot output of the random forest algorithm tells us how important a variable is in classifying the data. All the variables involved under study are arranged top-to-bottom as most-to-least important. The Before and After plots clearly shows that post inclusion of the ratio variable as one of the independent variables in this random forest supervised learning algorithm. From the Random Forest plot, we could infer that the words “make”, “roll”, “children’, ‘made’ and “design” formed top five identifiers to identify if it is a deal or not.*