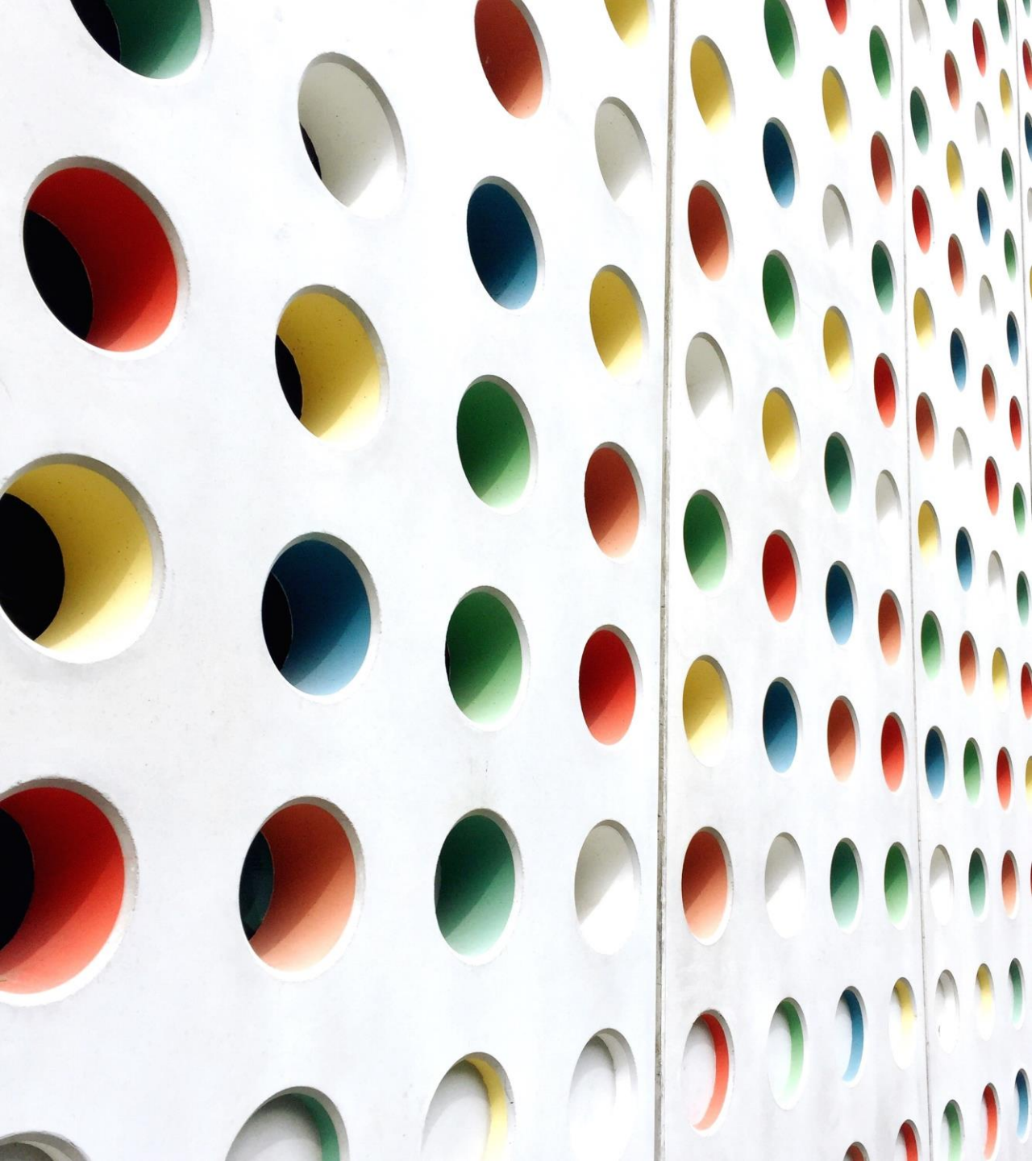
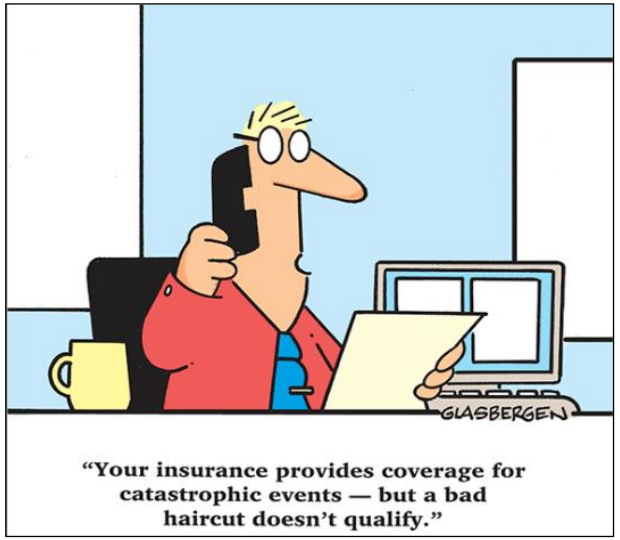


CROSS-SELL MODEL

FINAL PROJECT

T3- LINEAR BOOST

What is Cross-Selling?



Cross-selling in insurance is the act of promoting products that are related or complementary to the one(s) your current customers already own or use. It is one of the most effective methods of marketing.

General smart question

Which factors suggest whether an existing customer would be interested in vehicle insurance?

About the Dataset

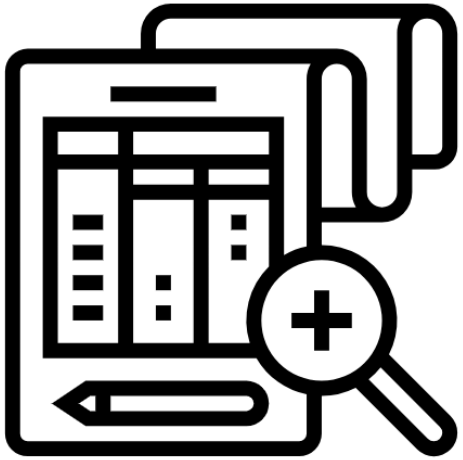
Whether a customer would be interested in an additional insurance service like vehicle Insurance is extremely helpful for the company because it can then accordingly plan its communication strategy to reach out to those customers and optimize its business model and revenue. We have following information to assist our analysis: demographics (gender, age, region code type), Vehicles (Vehicle Age, Damage), Policy (Premium, sourcing channel) etc.

Variable	Definition
id	Unique ID for the customer
Gender	Gender of the customer
Age	Age of the customer
Driving_License	0 : Customer does not have DL, 1 : Customer already has DL
Region_Code	Unique code for the region of the customer
Previously_Insured	1 : Customer already has Vehicle Insurance, 0 : Customer doesn't have Vehicle Insurance
Vehicle_Age	Age of the Vehicle
Vehicle_Damage	1 : Customer got his/her vehicle damaged in the past. 0 : Customer didn't get his/her vehicle damaged in the past.
Annual_Premium	The amount customer needs to pay as premium in the year
Policy_Sales_Channel	Anonymised Code for the channel of outreaching to the customer ie. Different Agents, Over Mail, Over Phone, In Person, etc.
Vintage	Number of Days, Customer has been associated with the company
Response	1 : Customer is interested, 0 : Customer is not interested

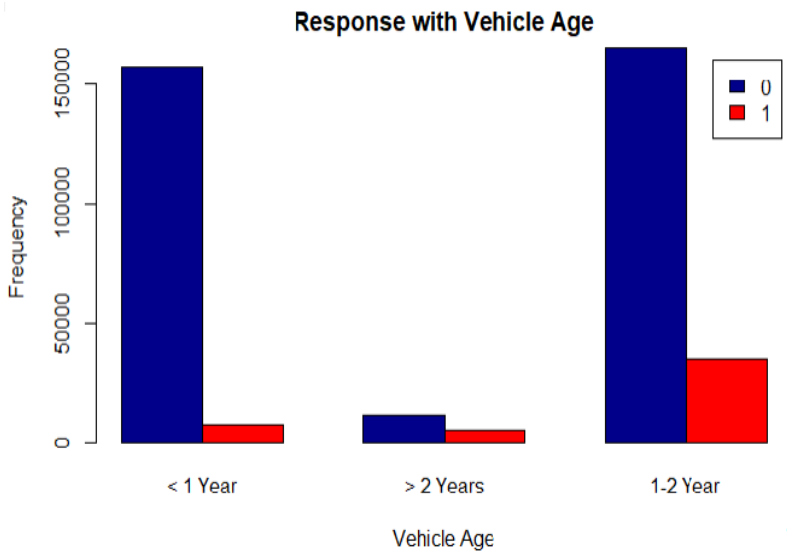
Summary of Dataset

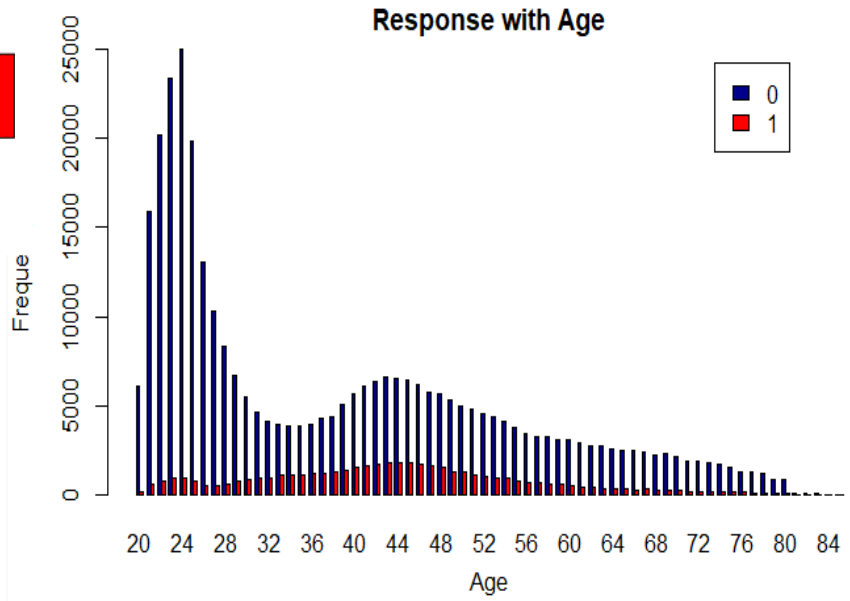
```
      id          Gender          Age        Driving_License    Region_Code     Previously_Insurec
 Min.   :     1   0:206089   Min.   :20.0   0:   812        Hogwarts : 22135   0:206481
 1st Qu.: 95278   1:175020   1st Qu.:25.0   1:380297        Midwest  : 56094   1:174628
 Median :190555              Median :36.0                   Northeast: 37321
 Mean   :190555              Mean   :38.8                   South    :162233
 3rd Qu.:285832              3rd Qu.:49.0                   West     :103326
 Max.   :381109              Max.   :85.0
 Vehicle_Age Vehicle_Damage Annual_Premium   Policy_Sales_Channel    Vintage    Response
 0:164786    0:188696       Min.   :  2630   Min.   :  1          Min.   : 10   0:334399
 1:200316    1:192413       1st Qu.: 24405   1st Qu.: 29          1st Qu.: 82   1: 46710
 2: 16007                   Median : 31669   Median :133          Median :154
                            Mean   : 30564   Mean   :112          Mean   :154
                            3rd Qu.: 39400   3rd Qu.:152          3rd Qu.:227
                            Max.   :540165   Max.   :163          Max.   :299
      prob
 Min.   :0.0036
 1st Qu.:0.0036
 Median :0.1408
 Mean   :0.1226
 3rd Qu.:0.2670
 Max.   :0.2940
```

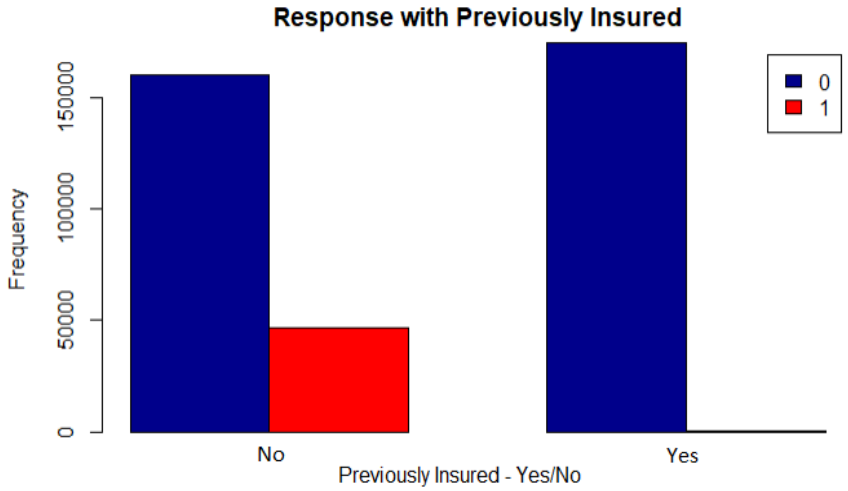
Exploratory Data Analysis



How vehicle age, age of a person and previously insured flag impacts the response?

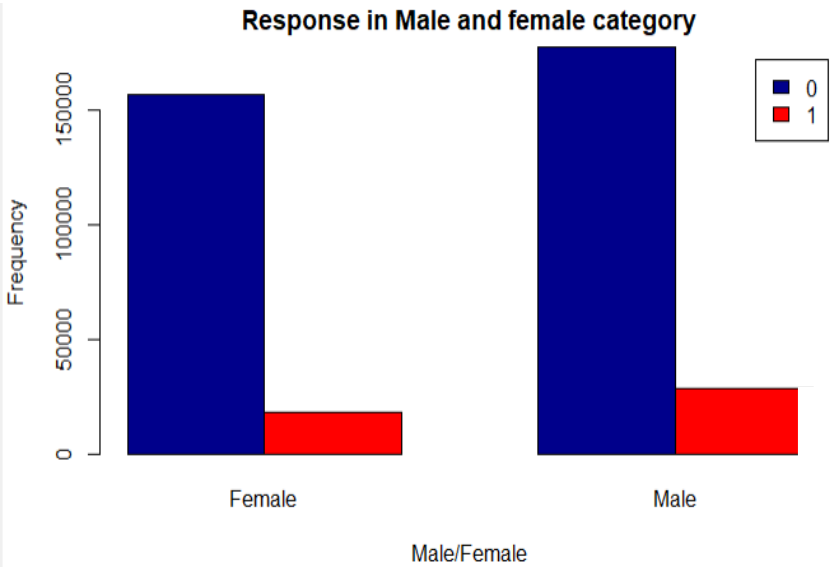


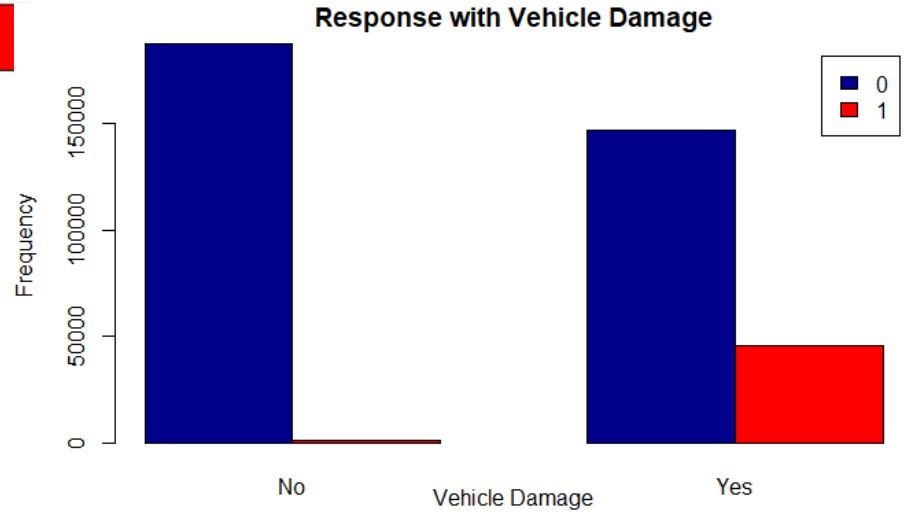


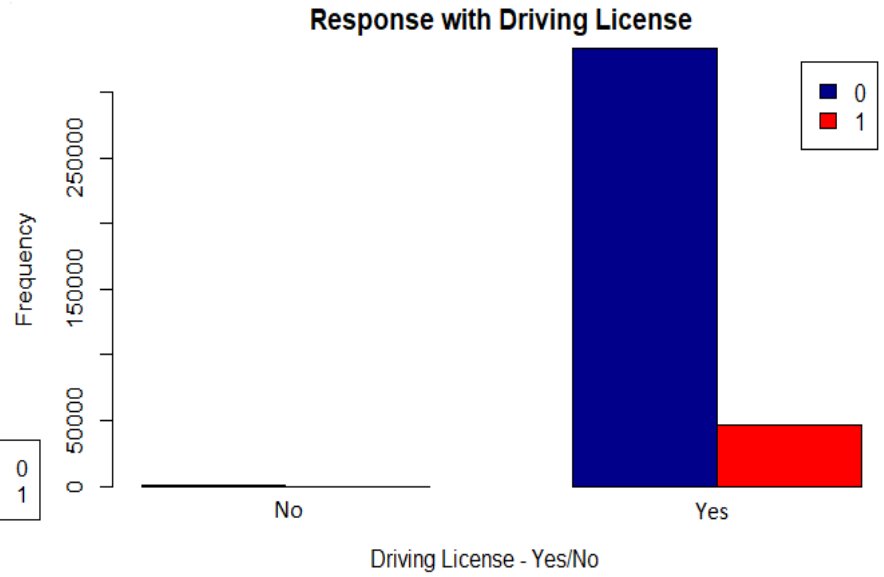


/*0=Reject the insurance, 1 = Accept the Insurance

Do males/females, vehicle damage flag and driving license flag impact the respond?

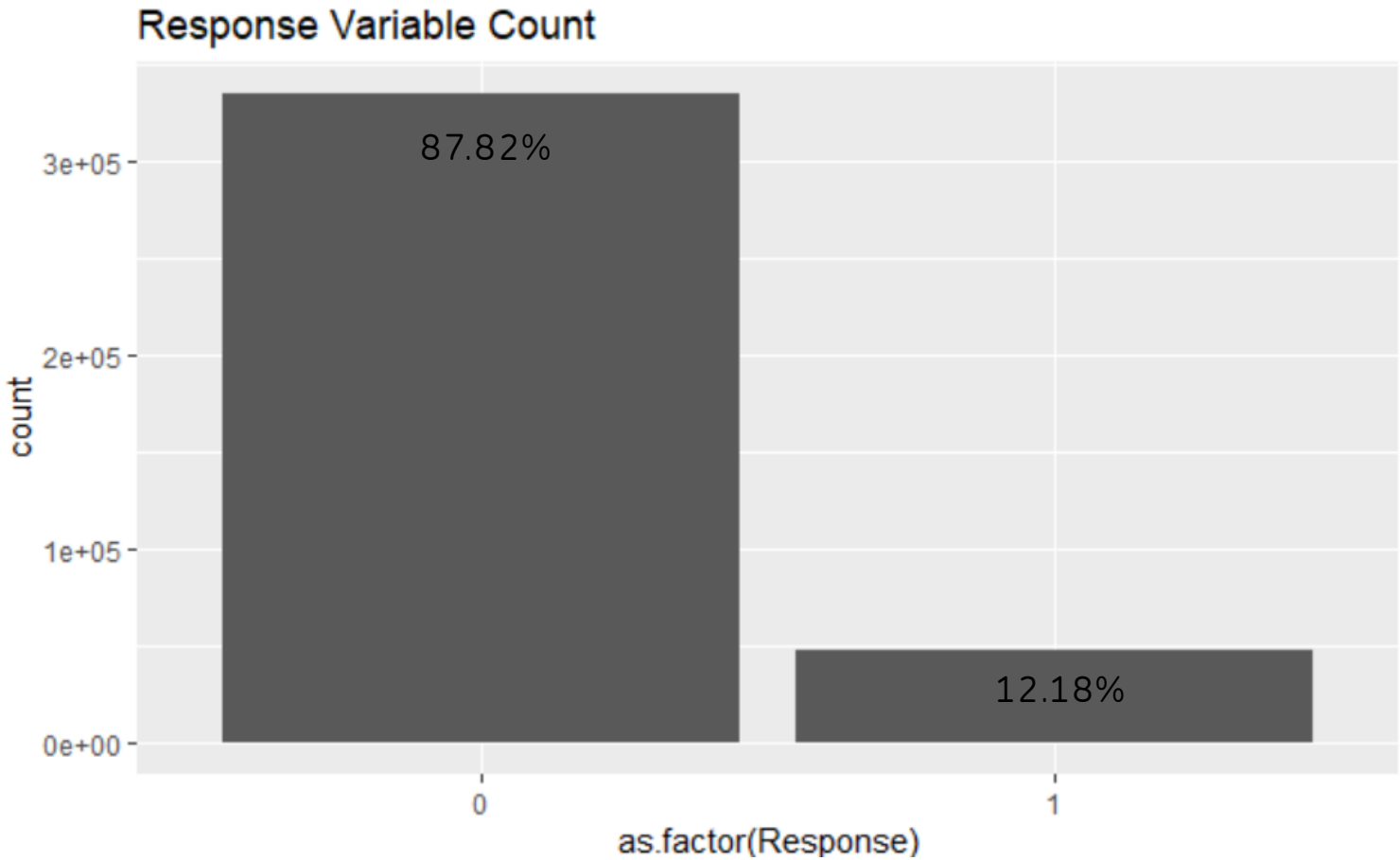






/*0=Reject the insurance, 1 = Accept the Insurance

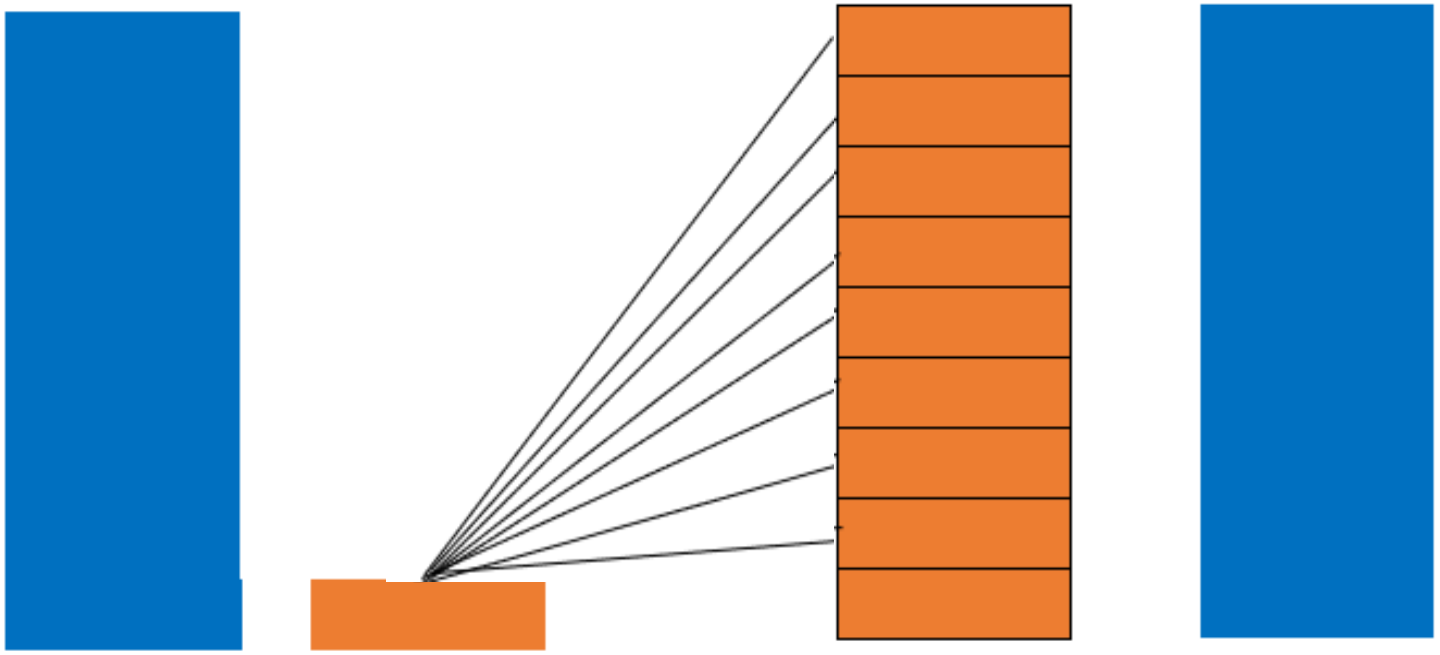
Target Variable Imbalance



/*0=Reject the insurance, 1 = Accept the Insurance

Over Sampling

Adding samples to minority class.



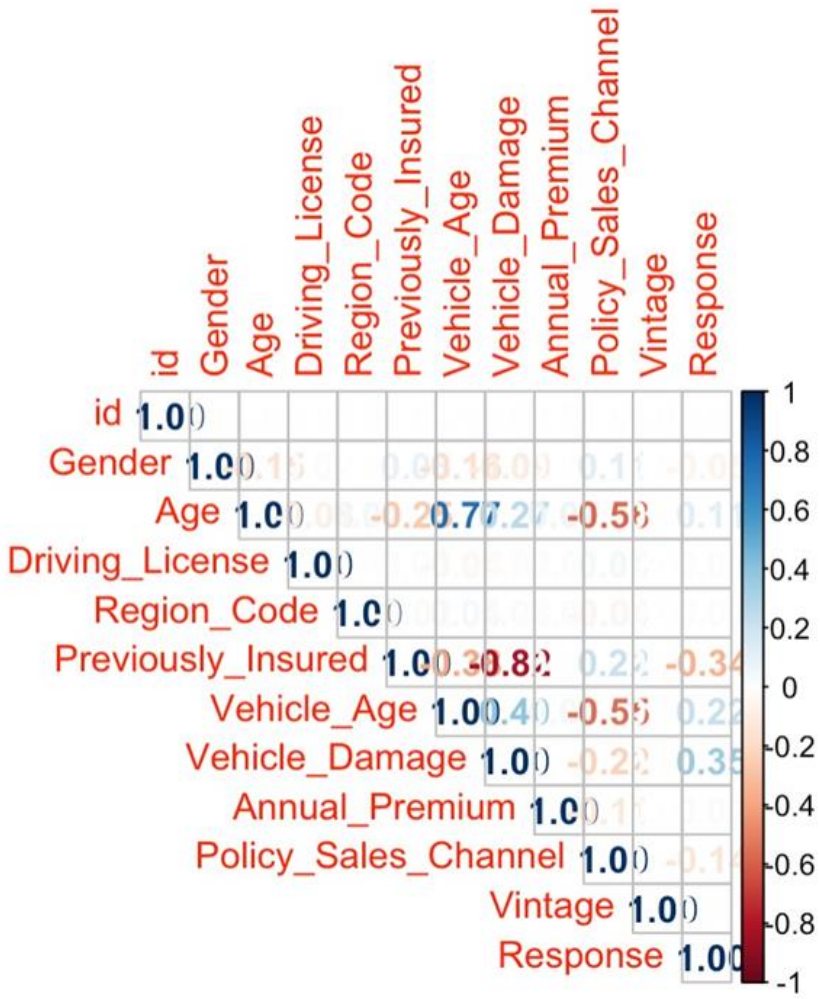
Original dataset

Does correlation concur?

A correlation matrix is simply a table which displays the correlation coefficients for different variables. The matrix depicts the correlation between all the possible pairs of values in a table.

- For our data, vehicle_damage, previously_insured and vehicle_age have high correlation.

- Positive Correlation: vehicle_damage and vehicle_age
- negative correlation: previously_insured

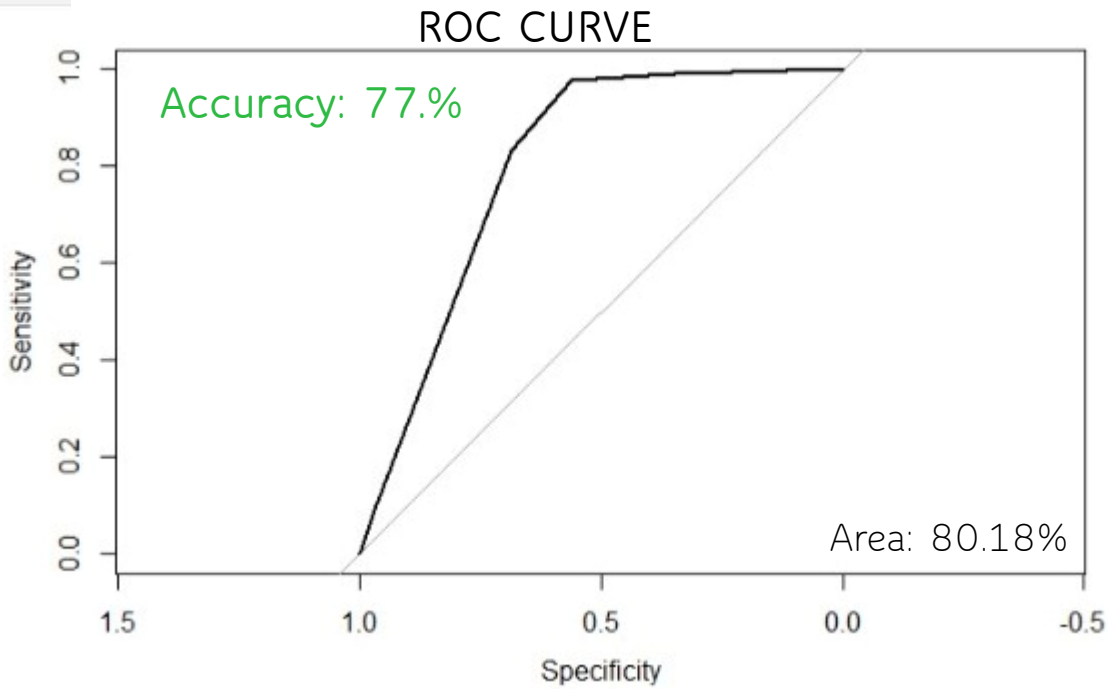


New smart question:

How each feature impact on the Response?

Logistic

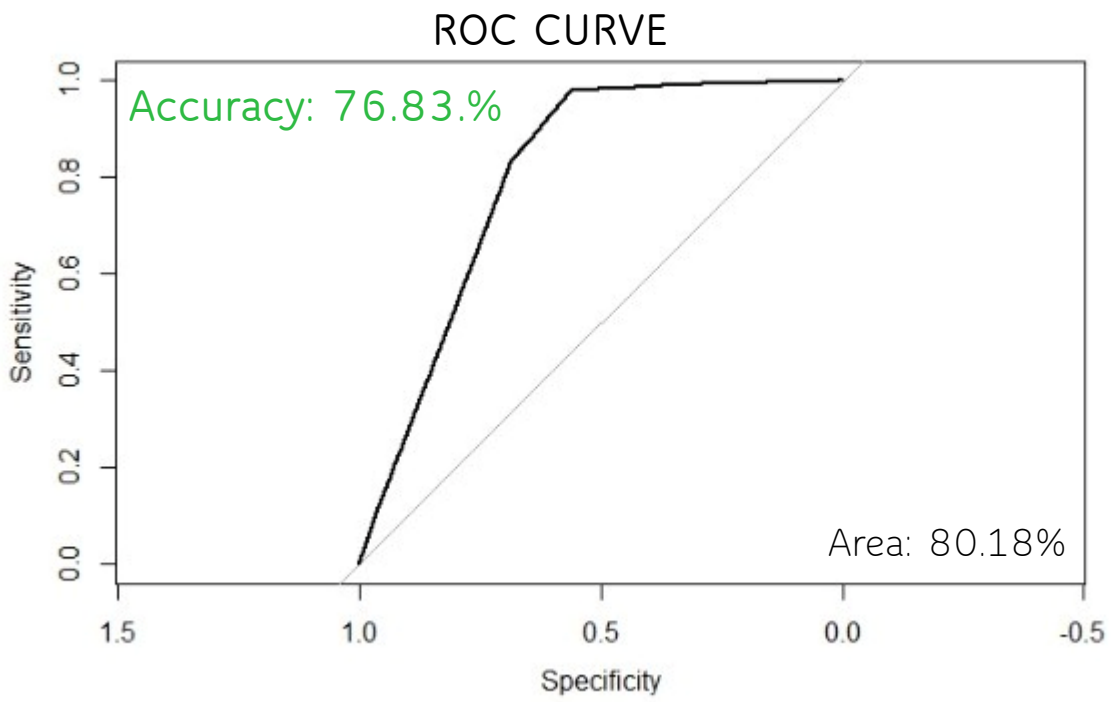
```
Call:
glm(formula = Response ~ vehicle_Age + vehicle_Damage, family = binomial(link = "logit"),
    data = vehicle)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6608  -0.3349   0.2679   0.8052   2.7131

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)     -3.655025   0.012805 -285.43   <2e-16 ***
vehicle_Age1     0.802181   0.007347  109.18   <2e-16 ***
vehicle_Age2     0.931132   0.012646   73.63   <2e-16 ***
vehicle_Damage1  3.812843   0.012614  302.26   <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: 927151  on 668797  degrees of freedom
Residual deviance: 635112  on 668794  degrees of freedom
AIC: 635120

Number of Fisher Scoring iterations: 6
```



Is a process of modelling the probability of a discrete outcome given an input variable. The most common logistic regression models a binary outcome; something that can take two values such as true/false, yes/no, and so on. Logistic regression is a useful analysis method for classification problems, where you are trying to determine if a new sample fits best into a category.

Random Forest

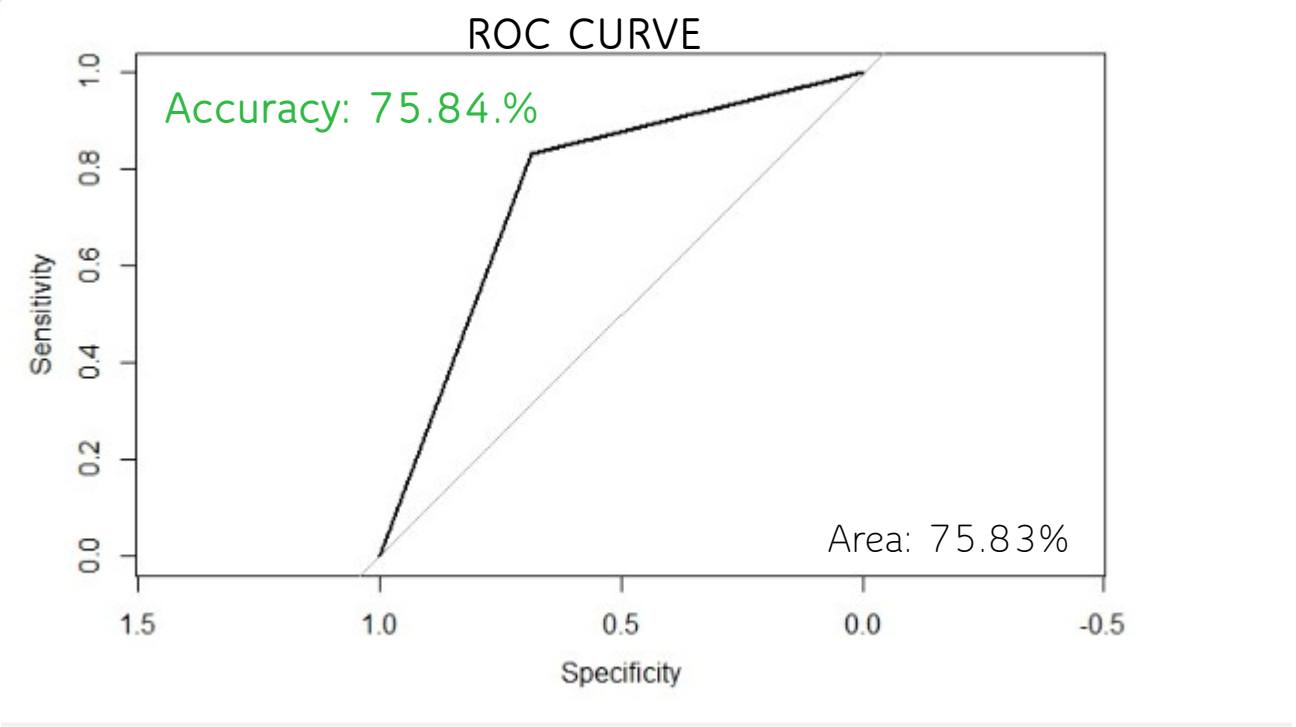
```
                Length Class  Mode
call                 4 -none- call
type                 1 -none- character
predicted       285831 -none- numeric
mse                500 -none- numeric
rsq                500 -none- numeric
oob.times       285831 -none- numeric
importance           2 -none- numeric
importanceSD         0 -none- NULL
localImportance      0 -none- NULL
proximity            0 -none- NULL
ntree                1 -none- numeric
mtry                 1 -none- numeric
forest              11 -none- list
coefs                0 -none- NULL
y               285831 -none- numeric
test                 0 -none- NULL
inbag                0 -none- NULL
terms                3 terms  call
```



A supervised learning algorithm that is based on the ensemble learning method and many Decision Trees. Random Forest uses a Bagging technique, so all calculations are run in parallel and there is no interaction between the Decision Trees when building them.

Naïve Bayes

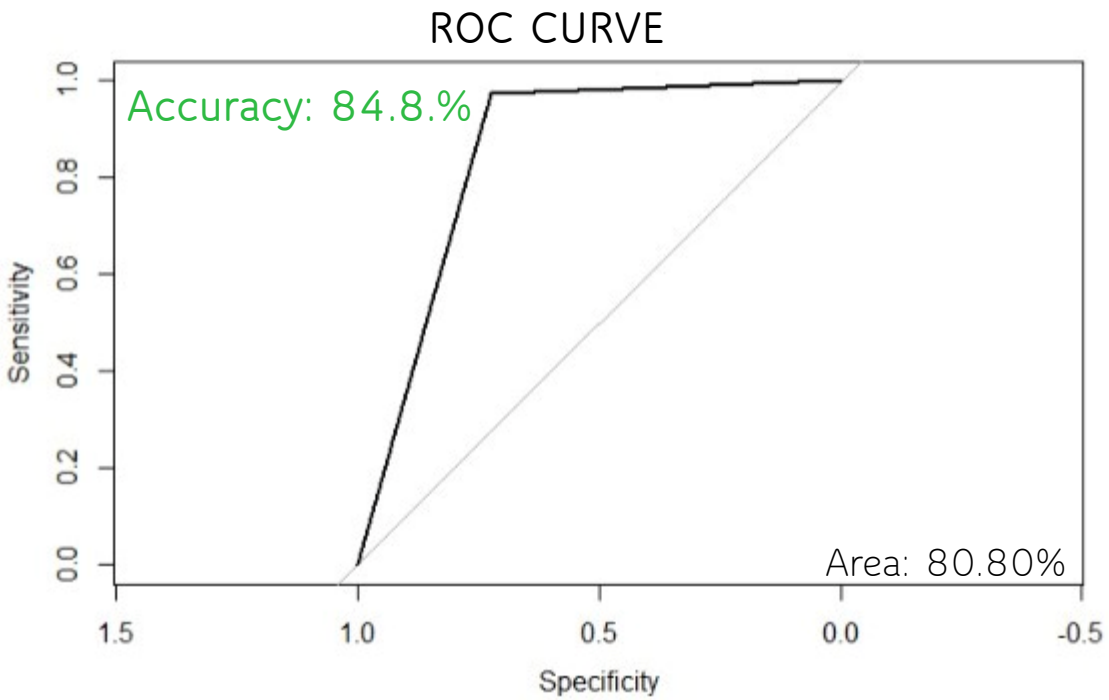
```
Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
        0        1
0.500439 0.499561

Conditional probabilities:
   Vehicle_Age
Y            0          1          2
  0 0.47196363 0.49385151 0.03418486
  1 0.15426683 0.74536716 0.10036601

   Vehicle_Damage
Y            0          1
  0 0.56135049 0.43864951
  1 0.02158427 0.97841573
```



It is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.

Xgboost

```
##### xgb.Booster
raw: 32.2 Mb
call:
  xgb.train(params = params, data = xgb.train, nrounds = 500, watchlist = list(val1 =
xgb.train,
    val2 = xgb.test), verbose = 0, early_stopping_rounds = 10,
    nthreads = 1)
params (as set within xgb.train):
  booster = "gbtree", eta = "0.1", max_depth = "10", gamma = "3", subsample = "0.75",
colsample_bytree = "0.75", objective = "multi:softprob", eval_metric = "merror",
num_class = "2", nthreads = "1", validate_parameters = "TRUE"
xgb.attributes:
  best_iteration, best_msg, best_ntreelimit, best_score, niter
callbacks:
  cb.evaluation.log()
  cb.early.stop(stopping_rounds = early_stopping_rounds, maximize = maximize,
    verbose = verbose)
# of features: 11
niter: 500
best_iteration : 500
best_ntreelimit : 500
best_score : 0.152022
best_msg : [500]        val1-merror:0.131855    val2-merror:0.152022
nfeatures : 11
evaluation_log:
```



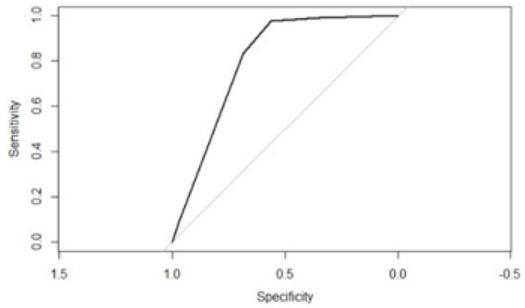
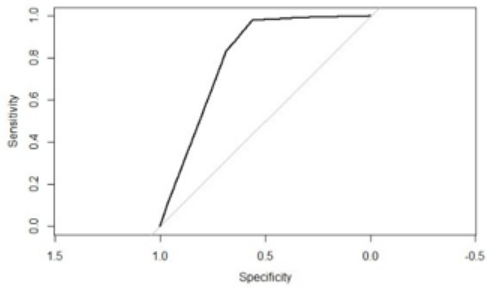
Refers to a class of ensemble machine learning algorithms constructed from decision tree models. Models are fit using any arbitrary differentiable loss function and gradient descent optimization algorithm. This gives the technique its name, “gradient boosting,” as the loss gradient is minimized as the model is fit.

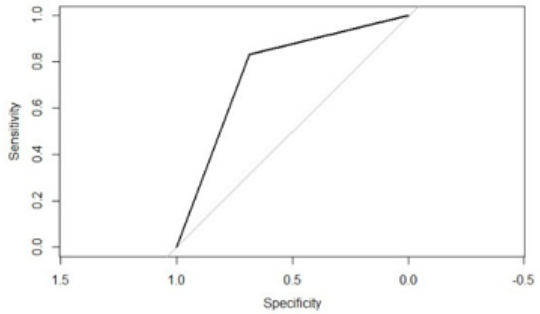
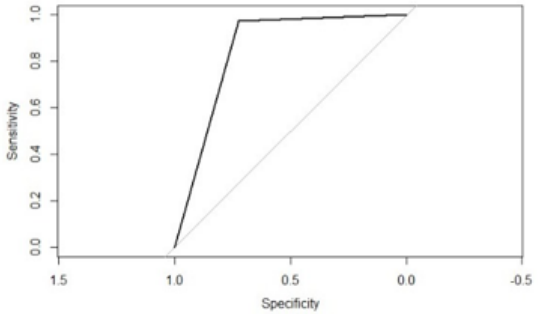
Other models

Decision Tree : 77.7%

KNN:72.05%

Conclusion

	Accuracy	ROC
Logistic	77.00%	 80.18%
Random Forest	76.83%	 80.08%

	Accuracy	ROC
Naïve Bayes	75.84%	 75.83%
Xgboost	84.80%	 84.80%