Health Insurance Cross Sell Prediction

Jinping Bai, Joshua Dalphy, Choongil Kim and Gouri Kulkarni

Tuesday, October 13th 2020

Contents

1 Business Introduction

1.1 Abstract

Cross-selling is a sales technique used to get a customer to spend more by purchasing a product that's related to what's being bought already.It is a sales technique used by Insurance companies to market new products to their existing customer base.Machine learning models can replace the manual task of sifting through customer files saving time and money.Implementing a machine learning model comes with it's challenges and we aim to address a few in this project.Data is a business critical asset and customer's privacy is of utmost importance.The team follows the Ethical ML framework and conduct the data analysis and model by using the methodology of CRISP-DM.

1.2 Background

ABC Insurance Ltd. , a leading Life Insurance Agent in the Town of XYZ has a large book of business comprising 381,109 households in the area.They are negotiating a deal with a major Auto Insurance carrier, Aplus Auto Insurance and plan to offer their product. Aplus Auto Insurance requires from ABC Insurance a report showing what percentage of their clients base would most likely purchase Auto insurance.Instead of manually sifting through their customer base, ABC Insurance Ltd. decided to approach the CAML1000 team for a solution to develop a machine learning system that will help them predict not just once, but over time, the households in their book of business that would be most likely candidates for Auto Insurance.The machine learning system would understand the customer base, and given the demographics of a prospect, be able to "predict" whether that prospect is a "good" or "bad" prospect. With this , the sales team at ABC Insurance will be better equipped in their cross-selling activity. They will be able to focus on the most lucrative leads making the most optimal use of their marketing dollars.

The solution would serve many purposes:

- gives Aplus auto an idea of the amount of business ABC Insurance can generate from the existing their book of business
- give ABC Insurance staff a tool that can help them prioritize clients for focused marketing campaigns
- let ABC Insurance project future revenues
- in realtime, provide notifications of possible future cross-sells

The client was unaware that the data they hold has the answers they seek.Some of their concerns and questions are :

- What do you need from us to deliver us the solution ?
- Will the system classify customers accurately?
- What if a customer who should have qualified for Auto Insurance is not selected by the system?

1.3 Objective

The objective of this project is to come up with a model or models that would provide ABC Insurance Ltd. with an initial report for Aplus Insurance and develop a tool that can predict if a customer with certain characteristics is a suitable candidate for their upcoming cross selling campaign.

1.4 Business Understanding:

Understand and identify business problems:

- Identify the key target variables that have to be predicted (good candidate/ bad candidate)
- What are the metrics associated with the target variable ?
- Understand the project objectives and requirements
- Formulate relevant and specific questions
- what identifies a good prospect from a bad prospect?
- what is the current practice in place used by sales to indentify good prospects from bad ones ?
- Define a success metric for the project
- Identify data sources that contain answers to questions
- which data would be an accurate measure of the model target and the features of interest?
- does the existing system need to collect and log additional kinds of data to address the problem and achieve the project goals?
- are external data sources needed or do systems need to be updated to to collect new data?

We then convert the knowledge into a data mining problem definition and develop a preliminary plan designed to achieve the objectives

1.5 Data Understanding:

After understanding the business statement, we conclude that ABC Insurance could have chosen to market Auto Insurance to all their customers,yet that was not the optimal use of their marketing dollars.It is better to target to those customers who are more likely to respond to the Aplus Auto campaign This targeted campaign not only can save them marketing dollars but also will not disturb those customers who have no interest in the new product.If we have historical data with the reactions of customers to past campaigns, we can use the data to build a model to predict which customer is a good prospect or not.We proceed to collect relevant data , identify data quality problems, discover first insights into the data, detect interesting subjects from the data.

The data for this project was downloaded from Kaggle, weblink: https://www.kaggle.com/anmolkumar/health-insurance-cross-sell-prediction. For privacy and security, the customer names have been masked with an id.The data is in csv format.

1.6 Data Description :

Variable Name	Variable Description
id	Unique ID for the customer

Variable Name	Variable Description
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's vehicle damaged in the past. 0 : Customer"s vehicle no damaged in the past.
Annual_Premium	The premium insurec paid in the year
Policy_Sales_Channel	Anonymized Code for outreach channels for sales.
Vintage	Number of Days, Customer has been associated with. the company
Response	1 : Customer is interested 0 : Customer is not interested

2 Data Exploration

2.1 Raw data understaing

Determine the dimension of our dataset:

```
## [1] 381109     12
```

View the contents and structure of our dataset:

```
##   id Gender Age Driving_License Region_Code Previously_Insured Vehicle_Age
## 1  1   Male  44               1          28                   0   > 2 Years
## 2  2   Male  76               1           3                   0    1-2 Year
## 3  3   Male  47               1          28                   0   > 2 Years
## 4  4   Male  21               1          11                   1    < 1 Year
## 5  5 Female  29               1          41                   1    < 1 Year
## 6  6 Female  24               1          33                   0    < 1 Year
##   Vehicle_Damage Annual_Premium Policy_Sales_Channel Vintage Response
## 1            Yes          40454                   26     217        1
## 2             No          33536                   26     183        0
## 3            Yes          38294                   26      27        1
## 4             No          28619                  152     203        0
## 5             No          27496                  152      39        0
## 6            Yes           2630                  160     176        0
```

View the main structure of the raw data

```
## 'data.frame':    381109 obs. of  12 variables:
##  $ id                 : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Gender             : chr  "Male" "Male" "Male" "Male" ...
##  $ Age                : int  44 76 47 21 29 24 23 56 24 32 ...
##  $ Driving_License    : int  1 1 1 1 1 1 1 1 1 1 ...
```

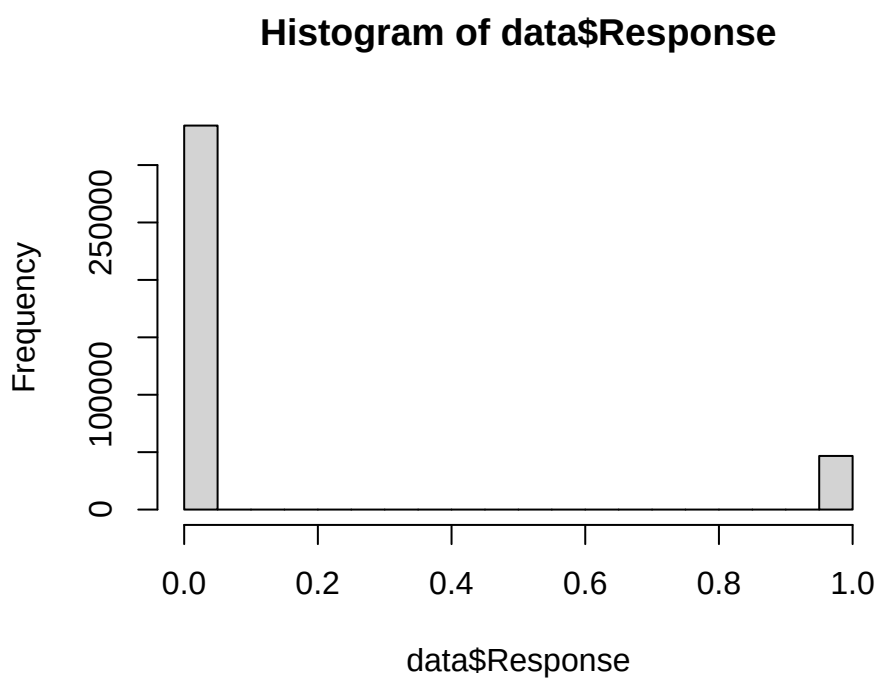
```
##  $ Region_Code         : num  28 3 28 11 41 33 11 28 3 6 ...
##  $ Previously_Insured  : int  0 0 0 1 1 0 0 0 1 1 ...
##  $ Vehicle_Age         : chr  "> 2 Years" "1-2 Year" "> 2 Years" "< 1 Year" ...
##  $ Vehicle_Damage      : chr  "Yes" "No" "Yes" "No" ...
##  $ Annual_Premium      : num  40454 33536 38294 28619 27496 ...
##  $ Policy_Sales_Channel: num  26 26 26 152 152 160 152 26 152 152 ...
##  $ Vintage             : int  217 183 27 203 39 176 249 72 28 80 ...
##  $ Response            : int  1 0 1 0 0 0 0 1 0 0 ...
```

After inspecting the dataset, the variables can be divided into two categories: categorical and numeric. The are divided as follows:

- Categorical variables: Gender, Driving_License, Previously_Insured, Vehicle_Age, Vehicle_Damage and Response (target variable)
- Numeric variables: id, Age, Region_Code, Annual_Premium, Policy_Sales_Channel and Vintage

Currently, certain of the categorical variables are being represented in numeric form (1 or 0), these will be transformed to Yes/No in the dataset. The former applies to Driving_License, Previously_Insured, Vehicle_Damage and Response. Additionally, upon further inspection, there are variables currently being represented as numeric which would be better suited as categorical, this applies to: Region_code and Policy_Sales_Channel (need explanation).

The objective of this project is to predict whether a customer is likely to purchase vehicle insurance. The target variable in this analysis is Response. Let's look at it's distribution:



To better understand the target variable, calculate the proportion tables

Table 2: Proportion of positive/negative responses

Var1	Freq
0	0.8774366
1	0.1225634

2.1.1 Categorical Variable Exploration

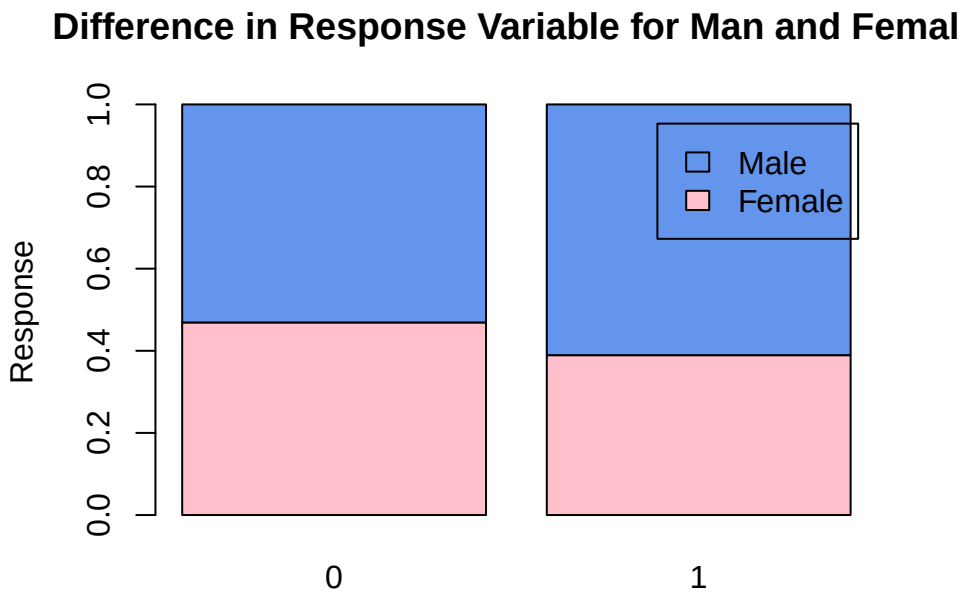
In this section we will explore each categorical variable.

Here is the frequency table for the Gender variable

Table 3: Gender Frequency

Gender	Freq
Female	175020
Male	206089

Investigate the proportion of men and women with respect to the target variable



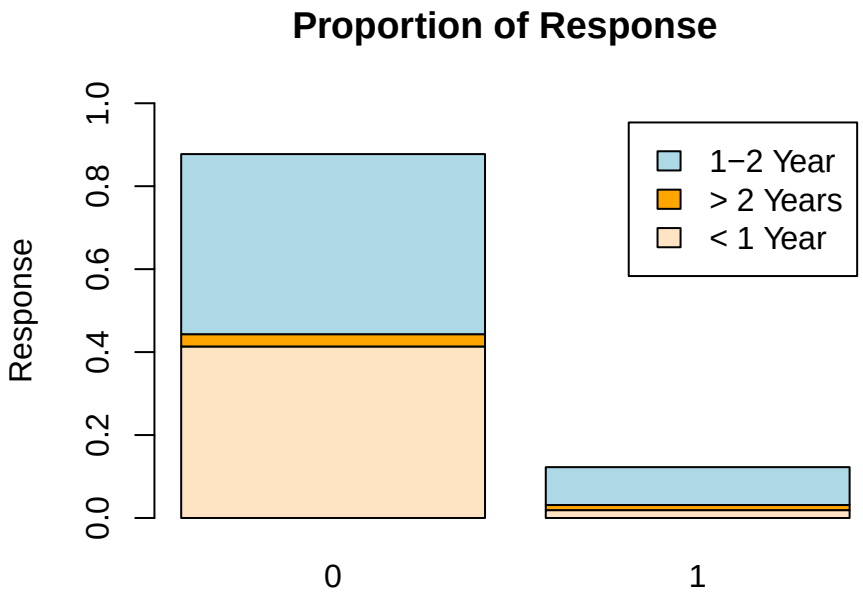
Here is the frequency table for the Vehicle Age variable

Table 4: Vehicle Age Frequency

Vehicle_Age	Freq
< 1 Year	164786
> 2 Years	16007

Vehicle_Age	Freq
1-2 Year	200316

Investigate the proportion of vehicle classes with respect to the target variable



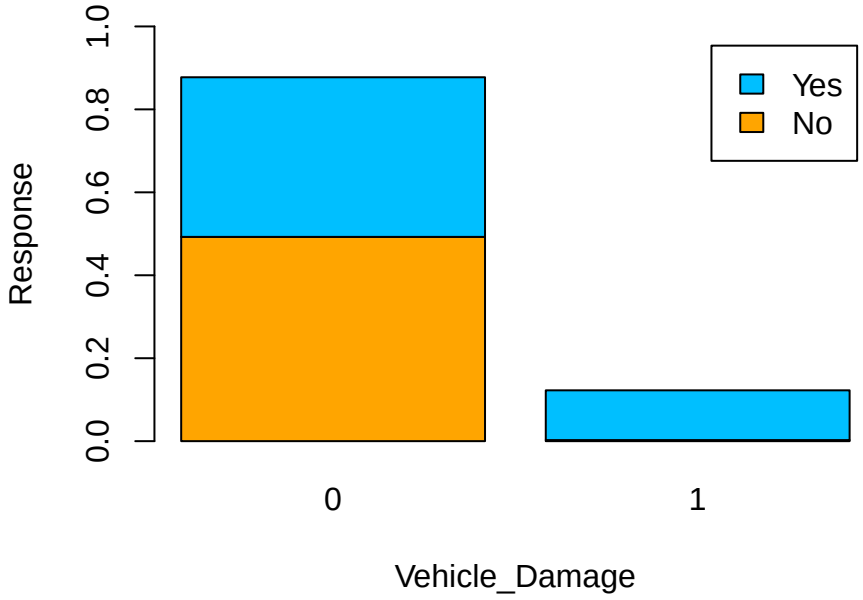
Vehicle Damage

Here is the frequency table for the Vehicle Damage variable

Table 5: Vehicle Damage Frequency

Vehicle_Damage	Freq
No	188696
Yes	192413

Investigate the proportion of vehicle damage with respect to the target variable



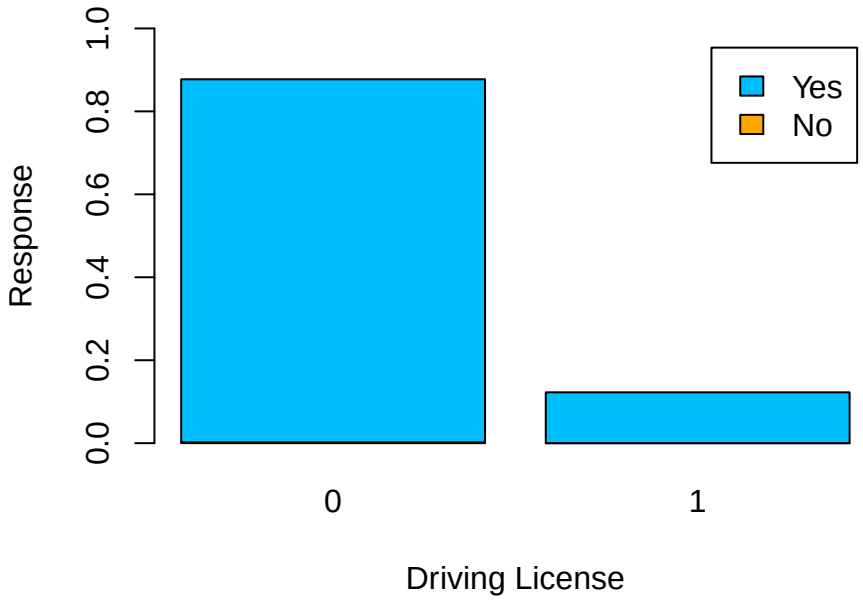
Driving License

Here is the frequency table for the Driving License variable

Table 6: Driving License Frequency

Driving_License	Freq
0	812
1	380297

Investigate the proportion of Driving License with respect to the target variable



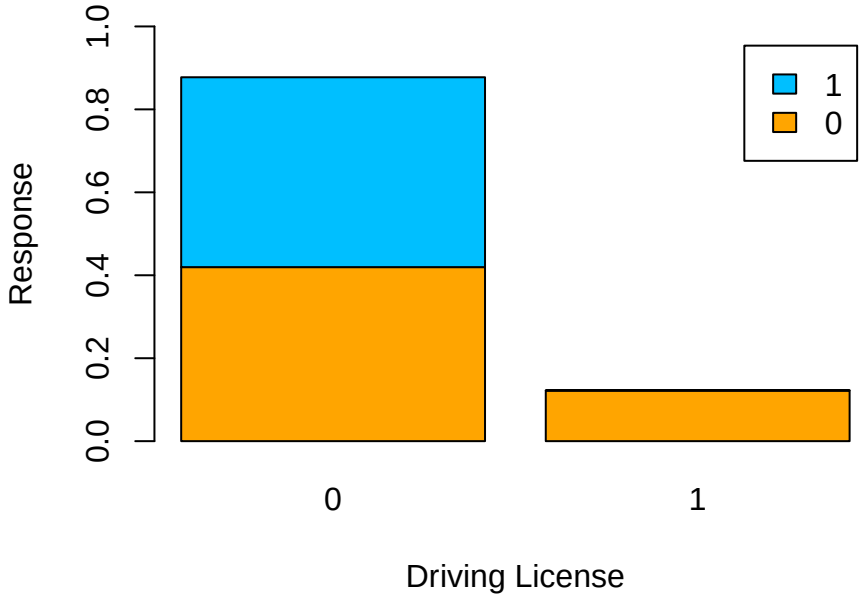
Previously Insured

Here is the frequency table for the Previously Insured variable

Table 7: Previously Insured Frequency

Previously_Insured	Freq
0	206481
1	174628

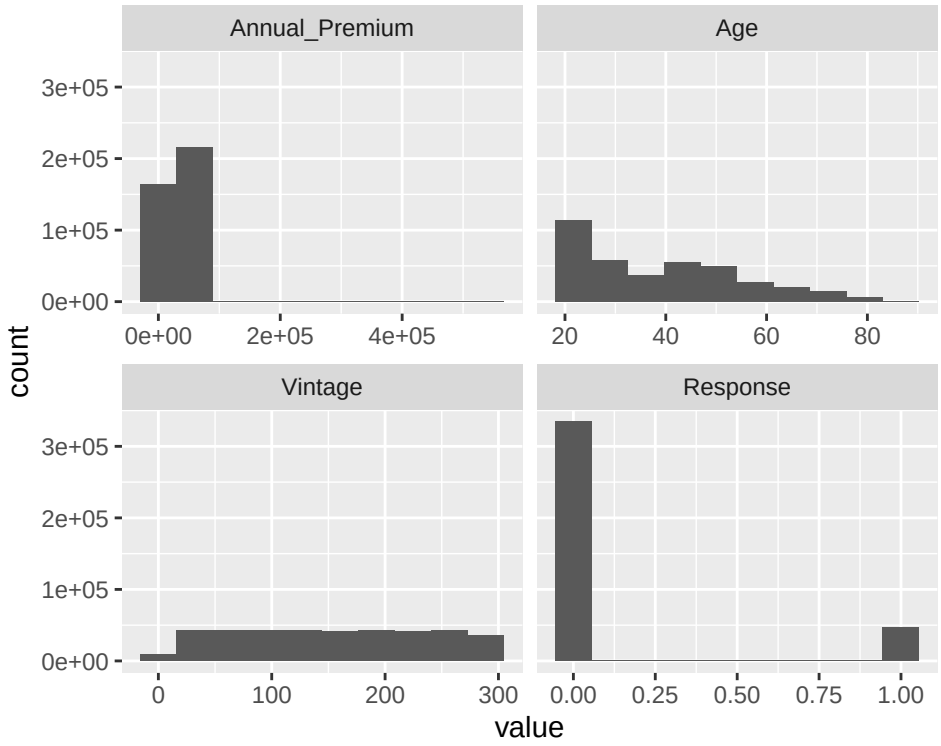
Investigate the proportion of Previously Insured with respect to the target variable



2.1.2 Numeric Variable Exploration

In this section data exploration is conducted on the numeric variables. These variables are: Age, Region_code, Annual_Premium, Policy_Sales_Channel and Vintage

Determine and visualize the distribution of the numeric variables:



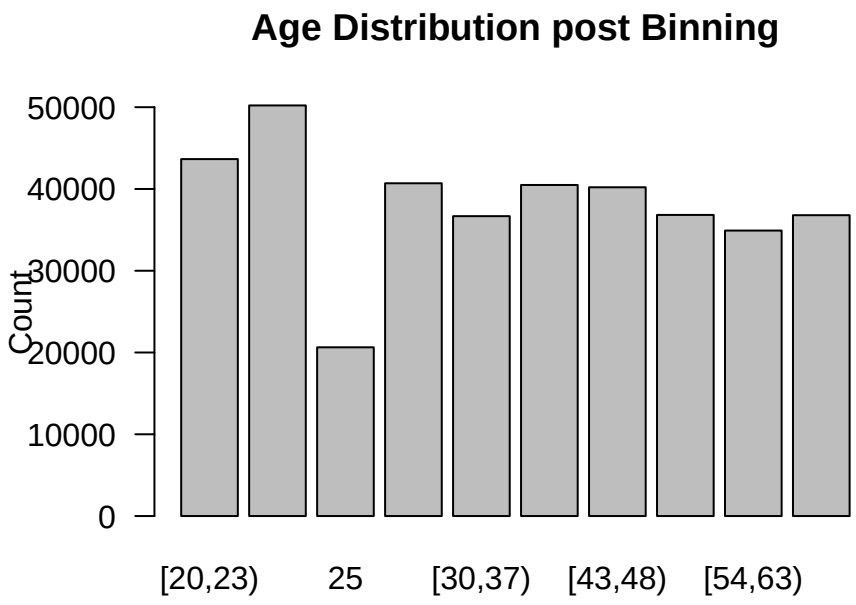
Approach to explore numeric variables Vs binary target(Response). First Bin numeric variables and then create table that shows average value of target by bin and visualize on a graph

First create a categorical version of target variable .

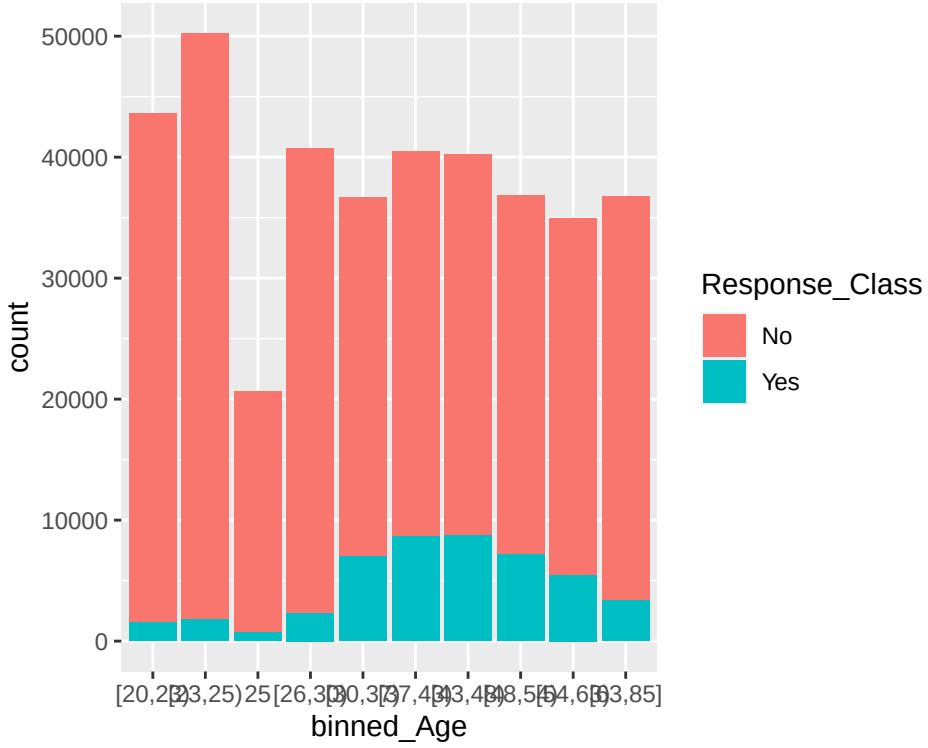
Binning Numeric Variables for a better visualization about the relationship with the target variable, Response

Create factor for Ages from minumum to max by using the bin function divided into 10 group base on the average age of each group.

Now view the histogram for Age after the binning process

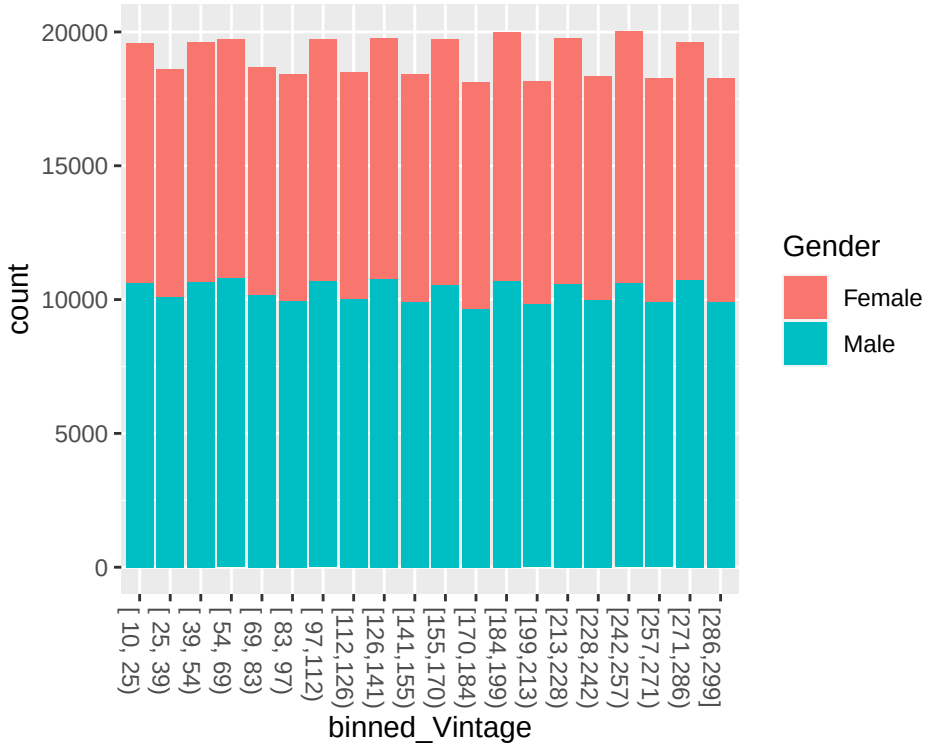


Let's visualize the relationship between binned_Age and Response_Class.



Age in the range of 37 to 48 responsed most.

Use the same way to bin Vintage from shortest days to longest days of staying in the insurance.



In general, male response more then that of female does.

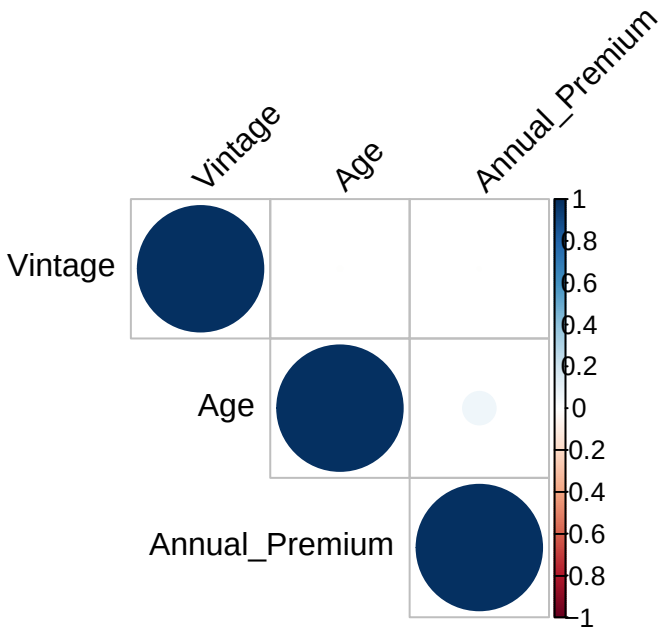
2.1.2.1 Numeric variables correlation Analysis

check the independent numeric variables relationship with each others by using correlation matrix to check the correlation coeficient between independent variables.

Remove ‘id’ and ‘Response’ columns before doing correlation matrix

Correlation matrix

Correlation Matrix with statistical signiticance, R square



No multilinearity between numeric variables.

2.2 Data Preparation

In this section we will import the original data to do data preparation for modeling

2.2.1 Missing Value Treatment

The missing values can be determined:

```
##                   id               Gender                  Age
##                    0                    0                    0
##      Driving_License          Region_Code   Previously_Insured
##                    0                    0                    0
##          Vehicle_Age       Vehicle_Damage       Annual_Premium
##                    0                    0                    0
## Policy_Sales_Channel              Vintage             Response
##                    0                    0                    0
```

There are no missing values in our dataset

Remove “id” column

2.2.2 Encoding categorical data and factor levels grouping

Convert Gender, Vehicle_Age, Vehicle_Damage from categorical variables to factors

A categorical variable can be divided into nominal categorical variable and ordinal categorical variable.Continuous class variables are the default value in R. They are stored as numeric or integer. Driving_License and Previously_Insured are nominal cateforical variables but labeled as intergers. We need to convert them into factors.

Convert numeric variables to levels of factors

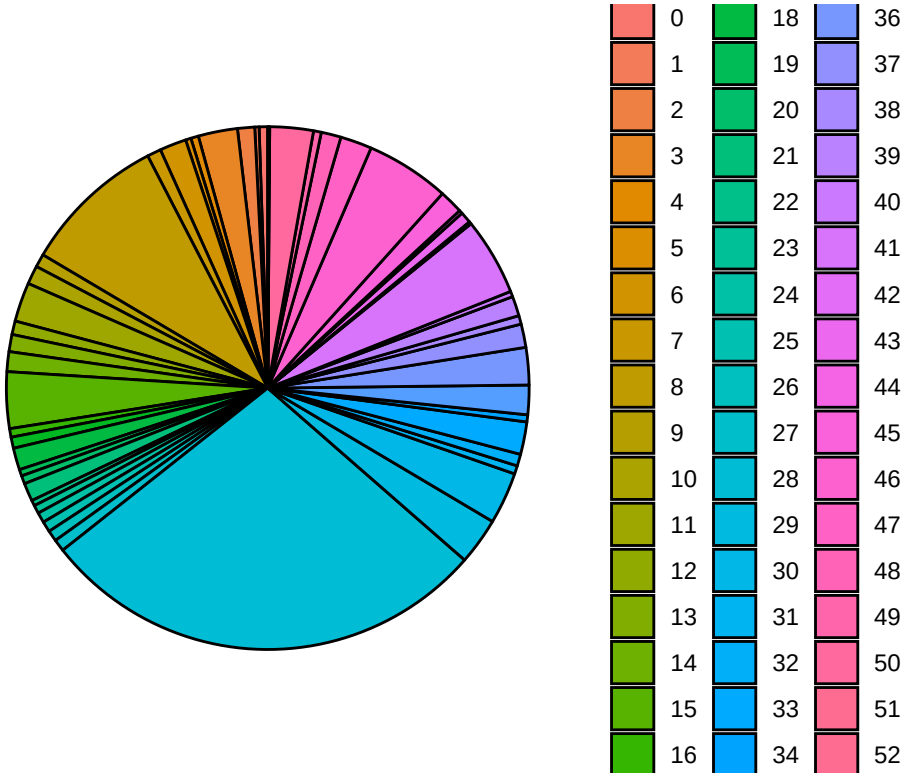
"Region_code's variables and Policy_Sales_Channel's variables are in the format of numeric. However those code number or channel numbers are characters. Region_Code are the unique code for the region of the customer; Policy_Sales_Channel are the anonymized Code for the channel of outreaching to the customer ie. Different Agents, Over Mail, Over Phone, In Person, etc. So we need to convert those numerics to charactors and then group them by the frequency.

Check how many levels of Region_Code

```
##  [1] "0"  "1"  "2"  "3"  "4"  "5"  "6"  "7"  "8"  "9"  "10" "11" "12" "13" "14"
## [16] "15" "16" "17" "18" "19" "20" "21" "22" "23" "24" "25" "26" "27" "28" "29"
## [31] "30" "31" "32" "33" "34" "35" "36" "37" "38" "39" "40" "41" "42" "43" "44"
## [46] "45" "46" "47" "48" "49" "50" "51" "52"
```

There are 53 levels(0 - 52) in Region_Code. We need check the order of the frequency and group them into less levels to avoid the issue of too manu levels of factors in one attribute when we do the modeling.

Check the frequency of each level in Region_Code



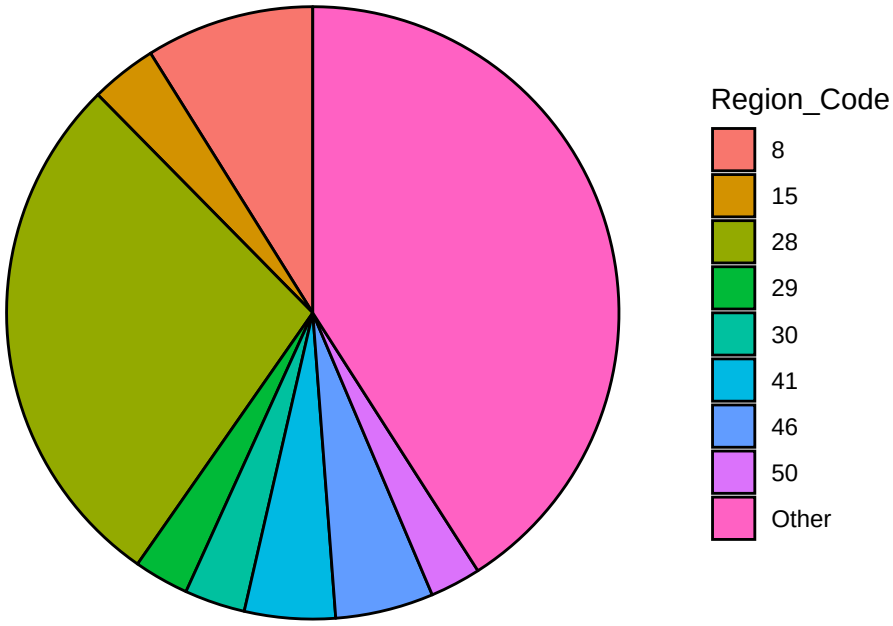

```
##
##     28      8     46     41     15     30     29     50      3     11     36
## 106415  33877  19749  18263  13308  12191  11065  10243   9251   9232   8797
##     33     47     35      6     45     37     18     48     14     39     10
##   7654   7436   6942   6280   5605   5501   5153   4681   4678   4644   4374
##     21      2     13      7     12      9     27     32     43     17     26
```

```
##    4266    4038    4036    3279    3198    3101    2823    2787    2639    2617    2587
##      25      24      38       0      16      23      31      20      49       4      34
##    2503    2415    2026    2021    2007    1960    1960    1935    1832    1801    1664
##      19      22      40       5       1      44      42      52      51
##    1535    1309    1295    1279    1008     808     591     267     183
```

The top 8 frequency Region_Code are “28”, “8”, “46”, “41”, “15”, “30”,“29”,“50”. Base on above plot and sort table we can group the Region_Code by the frequency into 9 groups including "other’ group.

```
## [1] "8"     "15"    "28"    "29"    "30"    "41"    "46"    "50"    "Other"
```

We get 9 levels of Region_Code.

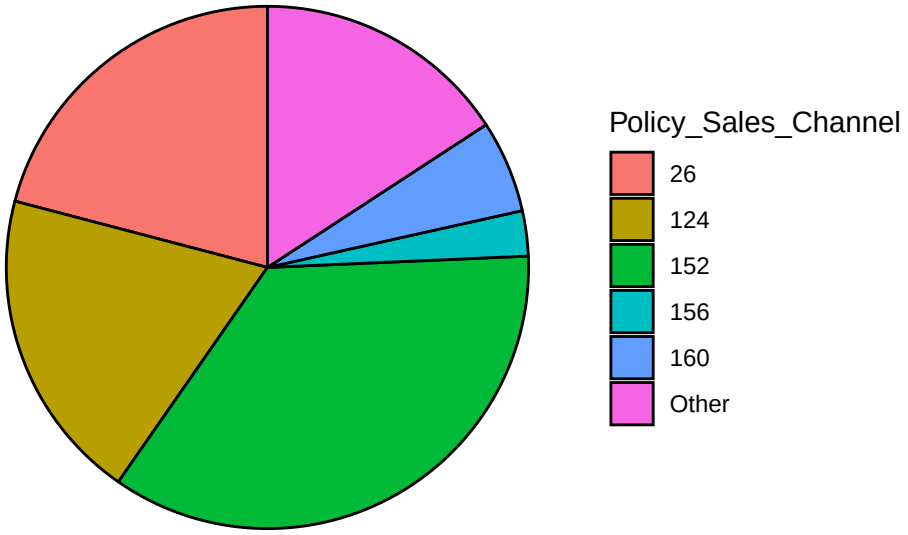


Relabel the factor levers of Region_Code

```
## [1] "1" "2" "3" "4" "5" "6" "7" "8" "9"
```

Using forcats method check the order of frequency in Policy_Sales_Channel

Base on above plot, that we can group the Policy_Sales_Channel by the frequency into 6 groups including one "Other" group.



Relabel the levels of Policy_Sales_Channel

```
## [1] "1" "2" "3" "4" "5" "6"
```

2.2.3 Outlier Treatment

Using Capping method to treat the Annual_Premium outliers issue.

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    2630   24405   31669   29898   39400   55176
```

2.2.4 Solve overfitting train data issues

There is an article in a website "If you choose too large of a training set you run the risk of overfitting your model. Overfitting is a classic mistake people make when first entering the field of machine learning." https://machinelearningmastery.com/arent-results-good-thought-youre-probably-overfitting/

We have 381,109.00 observations we will going to only use 10% of the raw data as a model data and split the 10% into train/test datasets.

Using the Partition method to get a new dataset and use the new data as a sample data to do the medolling.We will use the10% observations to do the data modeling

```
## [1] 38111    11
```

```
## [1] 342998     11
```

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:reshape2':
##
##     dcast, melt
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

convert all sampleDate factor levels to numeric so that we can scale the data to do the modelling.

convert Response to factor variables.

2.2.5 Split sample data to training set and testing set

Split the sampleDate to generate train and test dataset.We only use 20% of the sampleData as the training set.

```
## [1] 7622   11
```

```
## [1] 30489    11
```

Comparing the train dataset and original dataset.

```
##
##      0      1
## 334399  46710
```

```
##
##         0         1
## 0.8774366 0.1225634
```

```
##
##    0    1
## 6701  921
```

```
##
##         0         1
## 0.8791656 0.1208344
```

Both yhe Percentage of customer who have positive response"1" is 12% in the original data and the train data. So that the small sample of train set can represent the original data. We will use the train dataset to do our model.

2.2.6 Features scaling

We only need to scale continues numeric in both train dataset and test dataset.

```
## 'data.frame':    7622 obs. of  11 variables:
##  $ Gender              : num  1 1 1 1 2 1 1 1 1 1 ...
##  $ Age                 : num  0.0325 0.7735 -1.18 0.5041 0.4367 ...
##  $ Driving_License     : num  2 2 2 2 2 2 2 2 2 2 ...
##  $ Region_Code         : num  1 4 9 2 2 9 2 2 2 1 ...
##  $ Previously_Insured  : num  1 1 2 1 1 1 1 1 2 2 ...
##  $ Vehicle_Age         : num  2 2 3 2 2 2 2 2 2 3 ...
##  $ Vehicle_Damage      : num  1 1 2 1 1 2 1 1 2 2 ...
##  $ Annual_Premium      : num  0.518 0.281 -0.112 1.663 -0.292 ...
##  $ Policy_Sales_Channel: num  2 1 5 1 1 6 2 2 2 6 ...
##  $ Vintage             : num  -1.554 0.919 -0.903 -0.3 -0.819 ...
##  $ Response            : Factor w/ 2 levels "0","1": 1 2 1 2 1 1 1 2 1 1 ...
```

```
## 'data.frame':    30489 obs. of  11 variables:
##  $ Gender              : num  2 2 2 2 2 2 1 1 2 1 ...
##  $ Age                 : num  -0.446 -1.179 -0.912 1.552 -0.779 ...
##  $ Driving_License     : num  2 2 2 2 2 2 2 2 2 2 ...
##  $ Region_Code         : num  9 2 2 2 9 8 2 2 9 9 ...
##  $ Previously_Insured  : num  2 2 2 1 2 2 1 1 1 2 ...
##  $ Vehicle_Age         : num  3 3 3 1 3 2 2 2 3 3 ...
##  $ Vehicle_Damage      : num  2 2 2 1 2 2 1 1 1 2 ...
##  $ Annual_Premium      : num  -0.0774 1.6802 1.6802 0.2594 0.3713 ...
##  $ Policy_Sales_Channel: num  3 3 3 1 3 3 6 6 3 3 ...
##  $ Vintage             : num  -0.894 -0.991 -0.566 -0.286 -0.517 ...
##  $ Response            : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

2.3 Create Models to analize feature sellection

2.3.1 Logistic regression classifier model

```
##
## Call:
## glm(formula = Response ~ ., family = binomial, data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2085  -0.6619  -0.0482  -0.0358   3.7399
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)         -17.813093 469.444647  -0.038    0.970
## Gender               -0.113227   0.078937  -1.434    0.151
## Age                  -0.277957   0.057112  -4.867 1.13e-06 ***
## Driving_License      12.322855 234.722146   0.052    0.958
## Region_Code          -0.059059   0.012808  -4.611 4.00e-06 ***
## Previously_Insured   -3.797389   0.526951  -7.206 5.75e-13 ***
## Vehicle_Age          -0.758397   0.096425  -7.865 3.69e-15 ***
## Vehicle_Damage       -1.920117   0.243432  -7.888 3.08e-15 ***
## Annual_Premium       -0.021589   0.038601  -0.559    0.576
## Policy_Sales_Channel -0.023989   0.021601  -1.111    0.267
## Vintage               0.001752   0.038465   0.046    0.964
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5618.7  on 7621  degrees of freedom
## Residual deviance: 4192.7  on 7611  degrees of freedom
## AIC: 4214.7
##
## Number of Fisher Scoring iterations: 13
```

2.3.2 Features selection

Gender, Driving_License, Annual_Premium , Policy_Sales_Channel and Vintage have P_valua are much more than 0.05. which means they do not infleunce the traget variable, Response, much. We remove these 5 features from both the train dataset and test dataset.

```
## [1] 7622    6
```

```
## [1] 30489     6
```

```
## 'data.frame':    7622 obs. of  6 variables:
##  $ Age               : num  0.0325 0.7735 -1.18 0.5041 0.4367 ...
##  $ Region_Code       : num  1 4 9 2 2 9 2 2 2 1 ...
##  $ Previously_Insured: num  1 1 2 1 1 1 1 1 2 2 ...
##  $ Vehicle_Age       : num  2 2 3 2 2 2 2 2 2 3 ...
##  $ Vehicle_Damage    : num  1 1 2 1 1 2 1 1 2 2 ...
##  $ Response          : Factor w/ 2 levels "0","1": 1 2 1 2 1 1 1 2 1 1 ...
```

```
## 'data.frame':    30489 obs. of  6 variables:
##  $ Age               : num  -0.446 -1.179 -0.912 1.552 -0.779 ...
##  $ Region_Code       : num  9 2 2 2 9 8 2 2 9 9 ...
##  $ Previously_Insured: num  2 2 2 1 2 2 1 1 1 2 ...
##  $ Vehicle_Age       : num  3 3 3 1 3 2 2 2 3 3 ...
##  $ Vehicle_Damage    : num  2 2 2 1 2 2 1 1 1 2 ...
##  $ Response          : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...

## 'data.frame':    30489 obs. of  6 variables:
##  $ Age               : num  -0.446 -1.179 -0.912 1.552 -0.779 ...
##  $ Region_Code       : num  9 2 2 2 9 8 2 2 9 9 ...
##  $ Previously_Insured: num  2 2 2 1 2 2 1 1 1 2 ...
##  $ Vehicle_Age       : num  3 3 3 1 3 2 2 2 3 3 ...
##  $ Vehicle_Damage    : num  2 2 2 1 2 2 1 1 1 2 ...
##  $ Response          : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

2.3.3 New Logistic Regression model after feature selection

Use the new glm model to do the probability prediction.

Change prob_pred percentage of probability to "1", "0" binimial number.

Convert "y_pred" list vector to atomic vector matching with the test$Response for comparison

```
##    y_pred
##        0     1
##   0 26780    24
##   1  3669    16
```

2.3.3.1 new logistic regression statistic analysis

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction     0     1
##          0 26780  3669
##          1    24    16
##
##                Accuracy : 0.8789
##                  95% CI : (0.8752, 0.8825)
##     No Information Rate : 0.8791
##     P-Value [Acc > NIR] : 0.5602
##
##                   Kappa : 0.006
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.0043419
##             Specificity : 0.9991046
##          Pos Pred Value : 0.4000000
##          Neg Pred Value : 0.8795034
##              Prevalence : 0.1208633
##          Detection Rate : 0.0005248
```

```
##    Detection Prevalence : 0.0013119
##       Balanced Accuracy : 0.5017233
##
##        'Positive' Class : 1
##
```

Accuracy can be a misleading metric for imbalanced data sets. Consider a sample with 95 negative and 5 positive values. Classifying all values as negative in this case gives 0.95 accuracy score.

Same issue as our original file. Althogh we got 0.8789 accuracy , however the Sensitivity is only 0.004. That means the model detect customers did not response very well, however did not do good job at detecting those customers who are interested in the cross sell. Our purpose of the project is to help client find out who are those customer have more likely to purchase the vehicles. There is strong imbalance clissificaton issues in the original data.

What is Imbalanced Classification ?

"Imbalanced classification is a supervised learning problem where one class outnumbers other class by a large proportion. This problem is faced more frequently in binary classification problems than multi-level classification problems." For more information about imbalande classification, check link: https://www.analyticsvidhya.com/blog/2016/03/practical-guide-deal-imbalanced-classification-problems/

2.3.3.2 Solve the imbalance classification

We use oversampling method to deal with the imbalanced classification issues.

```
##
##    0    1
## 6701  921
```

```
## [1] 13402
```

```
##       Age            Region_Code    Previously_Insured  Vehicle_Age
##  Min.   :-1.1800   Min.   :1.000   Min.   :1.000      Min.   :1.000
##  1st Qu.:-0.7758   1st Qu.:2.000   1st Qu.:1.000      1st Qu.:2.000
##  Median : 0.1673   Median :6.000   Median :1.000      Median :2.000
##  Mean   : 0.1343   Mean   :5.585   Mean   :1.263      Mean   :2.249
##  3rd Qu.: 0.7735   3rd Qu.:9.000   3rd Qu.:2.000      3rd Qu.:3.000
##  Max.   : 2.0534   Max.   :9.000   Max.   :2.000      Max.   :3.000
##  Vehicle_Damage  Response
##  Min.   :1.000   0:6701
##  1st Qu.:1.000   1:6701
##  Median :1.000
##  Mean   :1.287
##  3rd Qu.:2.000
##  Max.   :2.000
```

2.3.3.3 Make another new logistic regression model by using the treated imbalance training data as the trainning dataset.

2.3.3.4 Use Confusion Matrix to analize the glm model after oversampling

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction     0     1
##          0 15820    80
##          1 10984  3605
##
##                Accuracy : 0.6371
##                  95% CI : (0.6317, 0.6425)
##     No Information Rate : 0.8791
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.2498
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.9783
##             Specificity : 0.5902
##          Pos Pred Value : 0.2471
##          Neg Pred Value : 0.9950
##              Prevalence : 0.1209
##          Detection Rate : 0.1182
##    Detection Prevalence : 0.4785
##       Balanced Accuracy : 0.7843
##
##        'Positive' Class : 1
##
```

We got 0.97 Sensitivity rate. That means this model can predict 97% of those customer who are intersted the cross sell. So far we got a good model. Let try other models to see which one is fit the data most. We will focus on the model Sensitivity value, which indicate how much the percentage accuracy the model catched for those customer who is interested in the cross sell.

3 Apply the treated training set to other models

3.1 Random Forest Prediction

Random Forest is a classification algorithm used in supervised machine learning and consists of constructing multiple decision trees during training and outpus the mode of the predicted variable of each decision tree <Hashmat's notes>. For the current application, the predicted variable is Response and consists of a Yes/No value. The function allows the user to customize multiple input parameters, including among other, the number of trees, number of features, tree depth and the minimum leaf size. This Random Forest model uses the default parameters available in R, with the sole exception being the number of trees. The former was set to 100, as numbers above started to affect processing time. The results of the model are summarized in the confusion matrix below:

Predict using the test set

Save the solution to a dataframe with the Response (prediction)

Table 8: Random Forest Confusion Matrix

	0	1
0	17646	9158
1	282	3403

Using the information provided in the matrix, the accuracy and sensitivity of the model can be calculated using the following two equations:

Accuracy

(n_truePositive + n_trueNegative)/(n_truePositive + n_trueNegative + n_falsePositive + n_falseNegative)

Sensitivity

n_truePositive/(n_truePositive + n_falsePositive)

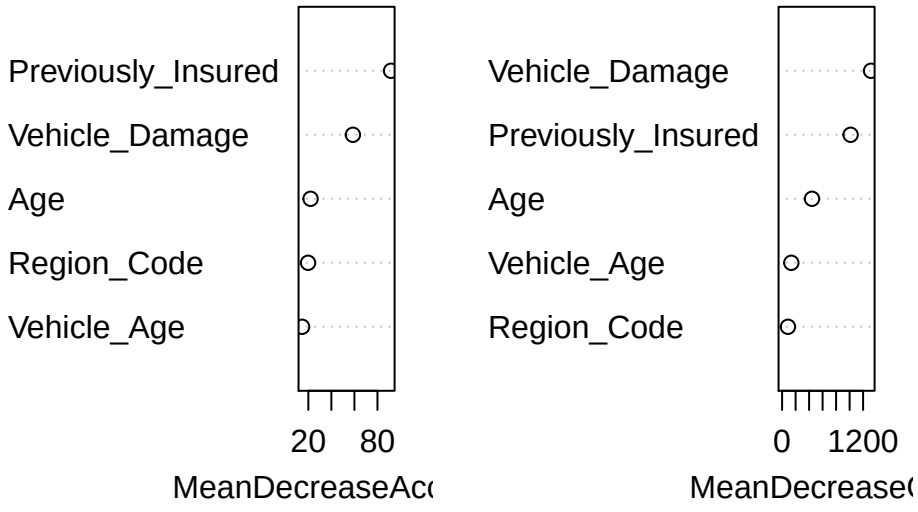
where: n_truePositive is the number true positive occurances, n_trueNegative is the number of true negative occurances, n_falsePositive is the number of false positive occurances and lastly, n_falsePositive is the number of false positives occurances present in the confusion matrix.

```
## [1] 0.69
```

```
## [1] 0.98
```

Based on the equation above, the accuracy and sensitivity of the Random Forest model were determined to be 0.69 and 0.98, respectively. An additional advantage of using the Random Forest algorithm is that it can be used to assess the relative importance of each feature, this is shown in the figure below.

Get features importance



The left figure above, is the important features order of Random Forest.it appears that the feature could be grouped into three categories: high importance, moderate importance and low importance. Previously_Insured and Vehicle_Damage would be categorized as the most important features when predicting response. Age, Vehicle_Age and Region_code would fall under moderate importance. The right figure is the important features order of the model of logistic regression which using the Gini importance method.

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction     0     1
##          0 17643   282
##          1  9161  3403
##
##                Accuracy : 0.6903
##                  95% CI : (0.6851, 0.6955)
##     No Information Rate : 0.8791
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.2853
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.9235
##             Specificity : 0.6582
##          Pos Pred Value : 0.2709
##          Neg Pred Value : 0.9843
##              Prevalence : 0.1209
##          Detection Rate : 0.1116
##    Detection Prevalence : 0.4121
##       Balanced Accuracy : 0.7908
##
##        'Positive' Class : 1
##
```

Sensitivity is 0.9189

3.2 Support Vector Classification (SVM_Classification)

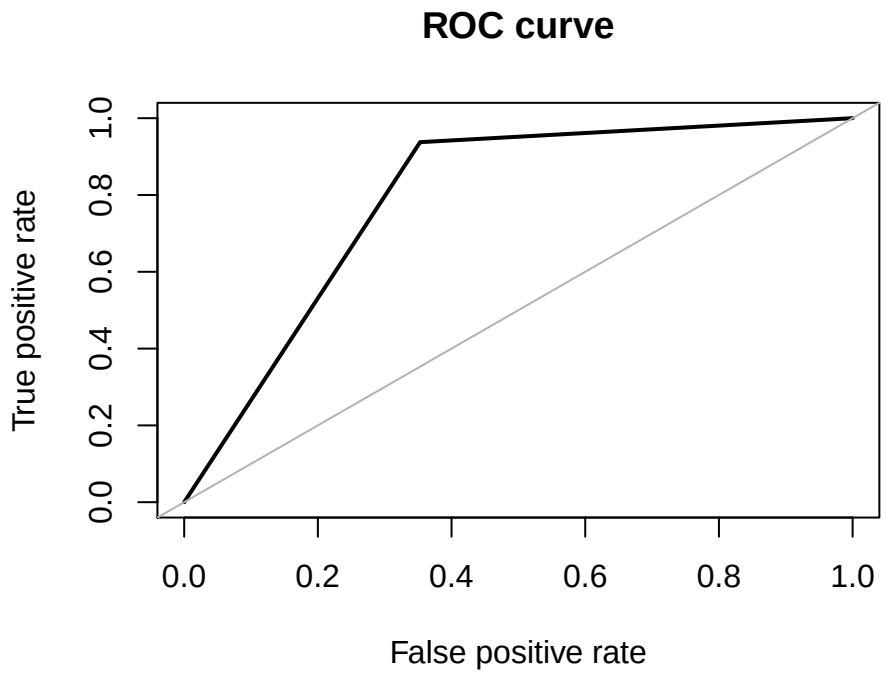
SVM or Support Vector Machine is a linear model for classification and regression problems. It can solve linear and non-linear problems and work well for many practical problems. The idea of SVM is simple: The algorithm creates a line or a hyperplane which separates the data into classes.However, the main idea is always the same: to minimize error, individualizing the hyperplane which maximizes the margin, keeping in mind that part of the error is tolerated. For more information link http://ocdevel.com/mlg/13

There are two types of SVM. We will use the SVClassification of SVM for classification algorithum.

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction     0     1
##          0 17342   230
##          1  9462  3455
##
```

```
##                Accuracy : 0.6821
##                  95% CI : (0.6769, 0.6873)
##     No Information Rate : 0.8791
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.281
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.9376
##             Specificity : 0.6470
##          Pos Pred Value : 0.2675
##          Neg Pred Value : 0.9869
##              Prevalence : 0.1209
##          Detection Rate : 0.1133
##    Detection Prevalence : 0.4237
##       Balanced Accuracy : 0.7923
##
##        'Positive' Class : 1
##
```

Sensitivity is 0.93, close to the one of Random Forest.




```
## Area under the curve (AUC): 0.792
```

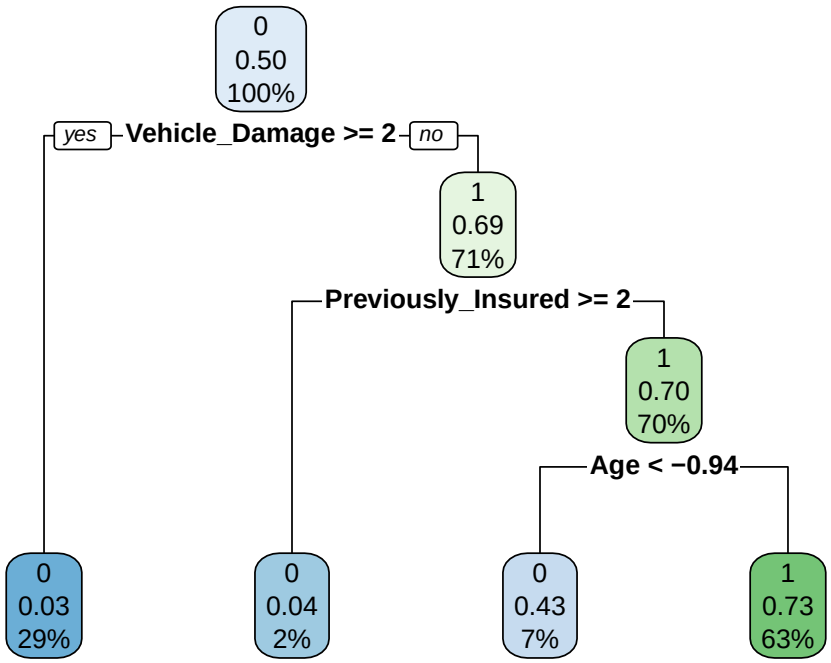
3.3 Naive Bayes Model

```
## Confusion Matrix and Statistics
```

```
##
##           Reference
## Prediction     0     1
##          0 15812    76
##          1 10992  3609
##
##                Accuracy : 0.637
##                  95% CI : (0.6316, 0.6424)
##     No Information Rate : 0.8791
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.25
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.9794
##             Specificity : 0.5899
##          Pos Pred Value : 0.2472
##          Neg Pred Value : 0.9952
##              Prevalence : 0.1209
##          Detection Rate : 0.1184
##    Detection Prevalence : 0.4789
##       Balanced Accuracy : 0.7846
##
##        'Positive' Class : 1
##
```

Sensitivity score is 0.9794. Same as logistic regression.

3.4 Decision Tree



Decision Tree Model Evaluation

Making the Confusion Matrix

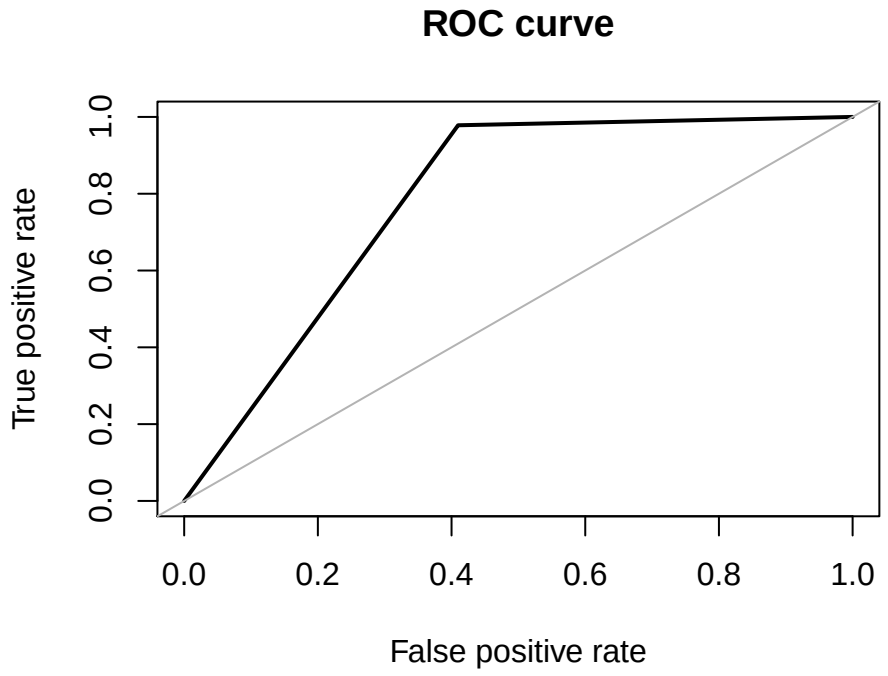
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction     0     1
##          0 15820    80
##          1 10984  3605
##
##                Accuracy : 0.6371
##                  95% CI : (0.6317, 0.6425)
##     No Information Rate : 0.8791
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.2498
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.9783
##             Specificity : 0.5902
##          Pos Pred Value : 0.2471
##          Neg Pred Value : 0.9950
##              Prevalence : 0.1209
##          Detection Rate : 0.1182
##    Detection Prevalence : 0.4785
##       Balanced Accuracy : 0.7843
```

```
##
##        'Positive' Class : 1
##
```

```
##
## Call:
## accuracy.meas(response = test$Response, predicted = y_predTree)
##
## Examples are labelled as positive when predicted is greater than 0.5
##
## precision: 0.247
## recall: 0.978
## F: 0.197
```

These metrics provide an interesting interpretation. With threshold value as 0.5, Precision = 0.247 says there are no false positives. Recall = 0.978 is very much high and indicates that we have lower number of false negatives as well. Threshold values can be altered also. F = 0.197 measn we have very accuracy of this model.

Recall in this context is also referred to as the true positive rate or sensitivity, and precision is also referred to as positive predictive value (PPV); other related measures used in classification include true negative rate and accuracy.True negative rate is also called specificity.




```
## Area under the curve (AUC): 0.784
```

4 Model Deployment

4.1 Shiny.app pipeline

After we have chosen a model, we deploy the model with a data pipeline to a production or production-like environment for final user acceptance.This makes ready for integration with the client's existing applications.Our model was deployed on R Shiny and presented to ABC Insurance Ltd.

From the Business understanding, we gathered that the client wishes to benefit from this project in the following ways :

- give Aplus auto an idea of the amount of business ABC Insurance can generate from the existing their book of business
- give ABC Insurance staff a tool that can help them prioritize clients for focused marketing campaigns
- let ABC Insurance project future revenues
- in realtime, provide notifications of possible future cross-sells

A pipleline can be developed for each of these requirements and is outside the scope of this project. Predictions can be made in realtime ir batch basis.We have deployed the model to predict a cross sell based on customer input provided by the user , the Gender, Age, Region Code, Policy Sales Channel,Vehicle Age and Vehicle Damage.Another deployment provides data file upload capability allowing predictions for entire datasets

shiny.app linkhttps://ml-lab.shinyapps.io/HealthInsCrsSellPredictor/

4.2 Publish our models in Github

We also published our data anlysis and modeling in the Github.Link https://github.com/csml1000groupc/HealthInsuranceCrossSellPredictionMLProject

5 Conclusion

We have done the data exploration and visulization to have a basic statistc backgroud information of our raw data. Then we did some data preparation for modeling, including check missing data, convert data variables for modeling, treat outliers issues. When we do the first model, logistic regression, I found out that the model had overfitting issues and imbalanced classification. After solving these two big issues, we are able to generate several applicable models which all have more the 93% Sensitivity rate(recall rate, true positive).We have Decision tree, Naive bayes and logistic Regresion have the highest True Positive Rate (Sensitivity rate). We recommend the Insurance company use the logistic regression model due to the other two models may cost more on the daily usage in the field of business management and technical maintaining.