Health Insurance Cross Sell Prediction:
Predict Health Insurance Owner's who will be interested in Vehicle Insurance

Executive Summary

The client, an insurance company requires a suitable model to predict whether its existing Health Insurance customers would be interested to buy its Vehicle Insurance service. In this report, all the variables that would influence this prediction are analysed and identified by informative graphs. Two different models were applied to predict the results and critically considered. As for feature analysis, the customer's response on this varies by these attributes: whether the customer already has a vehicle insurance, the customer's age, the vehicle age and whether it has been damaged, and what is the approach the company reached the customer.

Introduction

Cross selling refers to selling an existing customer an additional product or service in addition to what they initially wanted. This selling technique is important for companies nowadays and brings benefits in different ways. A high success rate in cross selling will not only increase a company's profitability but will also help a customer solve a problem and thus increase the customer's loyalty.

Goal

The client is an insurance company with a variety of insurance services. The company is hoping to cross sell vehicle insurance service to existing health insurance customers. So, the goal is to predict whether a customer will be interested in purchasing a Vehicle Insurance knowing that they are currently joined in the company's Health Insurance.

Dataset

The dataset provided by the client contains 12 variables including relative information of customers' when they signed up the Health Insurance and their responses of whether being interested in purchasing the Vehicle Insurance. All the variables are list as: *"id", "Gender", "Age", "Driving_Licence", "Region_Code", "Previously_Insured", "Vehicle_Age", "Vehicle_Damage", "Annual_Premium", "Policy_Sales_Channel", "Vintage", "Response"*. Most of the variables are easy to understand except for "*PolicySalesChannel*" and "*Vintage*".*"PolicySalesChannel*" is an anonymized code for the channel of outreaching to the customer, for example Different Agents, Over Mail, Over Phone, In Person, etc. "Vintage" refers to the number of days, customer has been associated with the company. "Response" is the dependent variable that consists of only 1 and 0 which stand for "interested" and "not interested".

In this report, two models have been used to predict a customer's response as "interested" in R studio. There are several modelling techniques for predicting such a variable, the ones I choose to use are Logistic Regression and Decision Tree. Depending on the layout of the results, both models have pros and cons in different circumstances. The whole data analysis process contains 5 steps: 1. Data Cleaning; 2. Data Analyse; 3. Train and test set split; 4. Balance Data; 5. Modelling Data; 6. Model Selection.

Process

Data cleaning

Data cleaning is an essential step that needs to be carried out before it can be analysed. Data cleaning ensures the data used for analysing is correct, consistent and useable. Provided is a large dataset with 381,109 rows and 12 columns. After loaded the original CSV file into R environment, the missing values is checked as the first step. The code below returns the total number of missing values in this dataset named “insurance” which is 0.

```r
sum(is.na(insurance))
```

```
[1] 0
```

Figure1. Check missing values.

The second step is to check whether there are special values in the dataset. It might be some cases that numeric variables are endowed with several formalized special values including ±Inf, NA and NaN. However, they should be excluded before generating a statistical statement. The following codes detect the total number of special values. Fortunately, there are not any special values nor missing values, the dataset is decently cleaned.

```r
is.special <- function(x){
if (is.numeric(x)) !is.finite(x) else is.na(x)
}
```

```r
sum(sapply(insurance, is.special))
```

```
[1] 0
```

Figure2. Check special values.

Next, obvious inconsistencies will be checked by using “summary()” function. An obvious inconsistency happens when a record contains a value or combination of values that cannot correspond to a real-world situation. Some rules or constraints can be created to detect obvious inconsistencies. Below shows the first line of the dataset with all variables. In this context, a customer’s age or a vehicle age should be above 0, an under-aged person cannot possess a driver’s license, and other numeric variables such as Vintage, Annual Premium paid should also be positive numbers. “summary(insurance)” summarizes the range of all the variables in the insurance dataset. As it can be seen in the Figure4, all the records are in the acceptable range, for example, the minimum age is 20 which exceeds the legitimate age for driving a vehicle.

id	Gender	Age	Driving_License	Region_Code	Previously_Insured	Vehicle_Age	Vehicle_Damage
<int>	<chr>	<int>	<int>	<dbl>	<int>	<chr>	<chr>
1	Male	44	1	28	0	> 2 Years	Yes

Annual_Premium	Policy_Sales_Channel	Vintage	Response
<dbl>	<dbl>	<int>	<int>
40454	26	217	1

Figure3. Screenshot the first line of that data.

```
      id              Gender
Min.   :     1   Female:175020
1st Qu.: 95278   Male  :206089
Median :190555
Mean   :190555
3rd Qu.:285832
Max.   :381109

     Age         Driving_License
Min.   :20.00   0:   812
1st Qu.:25.00   1:380297
Median :36.00
Mean   :38.82
3rd Qu.:49.00
Max.   :85.00

 Region_Code      Previously_Insured
28      :106415   0:206481
8       : 33877   1:174628
46      : 19749
41      : 18263
15      : 13308
30      : 12191
(Other):177306
   Vehicle_Age      Vehicle_Damage
< 1 Year :164786   No :188696
> 2 Years: 16007   Yes:192413
1-2 Year :200316

Annual_Premium   Policy_Sales_Channel
Min.   :  2630   152    :134784
1st Qu.: 24405   26     : 79700
Median : 31669   124    : 73995
Mean   : 30564   160    : 21779
3rd Qu.: 39400   156    : 10661
Max.   :540165   122    :  9930
                 (Other): 50260

   Vintage      Response
Min.   : 10.0   0:334399
1st Qu.: 82.0   1: 46710
Median :154.0
Mean   :154.3
3rd Qu.:227.0
Max.   :299.0
```

Figure4. Output by “summary()” function to check consistency.

Data analyse

The aim of this project is to find effective predictors for customer’s response on buying a Vehicle Insurance. So, the relationship between “Response” and other variables should be explored to understand ideal customer portrait. To help gain a better understanding of the graph plotted, I revalued factors 0 and 1 in “Response” to “Interested” and “Not_Interested”.

One important thing needs to be considered before exploring the data is to identify confound variables that can cause bias for the analysis. In data science, a confound variable or lurking variable is a variable influence both the dependent variable and independent variable causing a spurious association or a coincidental correlation. Having inspected all the independent variables with the dependent variable “Response”, it was found that “*Previously_Insured*” is a lurking variable.

To help visualize the relationship between variables in a simple and clear way, I have mainly used bar chart and fill the “Interested” response region with green while the rest is in orange.

“*Previously_Insured*” has two factors 1 and 0, which stand for 1: customer already has Vehicle Insurance; 0: customer has no Vehicle Insurance. Below is a bar chart plot of two types of responses within “*Previously_Insured*”. This is understandable that only the customers who have not insured their vehicle would be interested in engaging a new vehicle insurance. Thus, this makes the “*Response*” highly dependent on “*Previously_Insured*”.

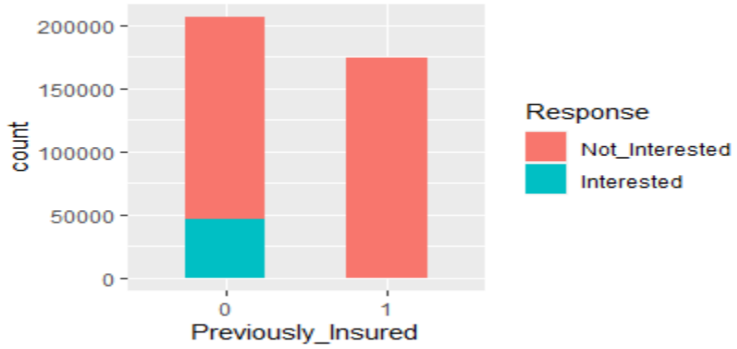


Figure5. Response in “Previously_Insured”.

However, this gives a precondition that a customer could be “Interested” if only if he/she has not owned a Vehicle Insurance. So, there will be cases that customers in a variable shows a very low percentage of “Interested”, and it is simply because the customers have insured their vehicle previously. Take “*Vehicle_Age*” for example, the vehicle age less than one year has a very low proportion of customer interesting in buying a Vehicle insurance. However, if we take out the observations which “*Previously_Insured*” is 1 and only consider the customers whose vehicle is not previously insured, then, that proportion has increased nearly 5 times as it shows in the Figure 6. This is because about 70 percent of customers who own vehicle less than one year have already purchased Vehicle Insurance (Figure7). Therefore, this “Interested” percentage was influenced by “Previously_Insured” rather than “Vehicle_Age”. Therefore, the observations with “Previously_Insured” equals 1 is suggested to be excluded before analysing the other variables with “Response”.

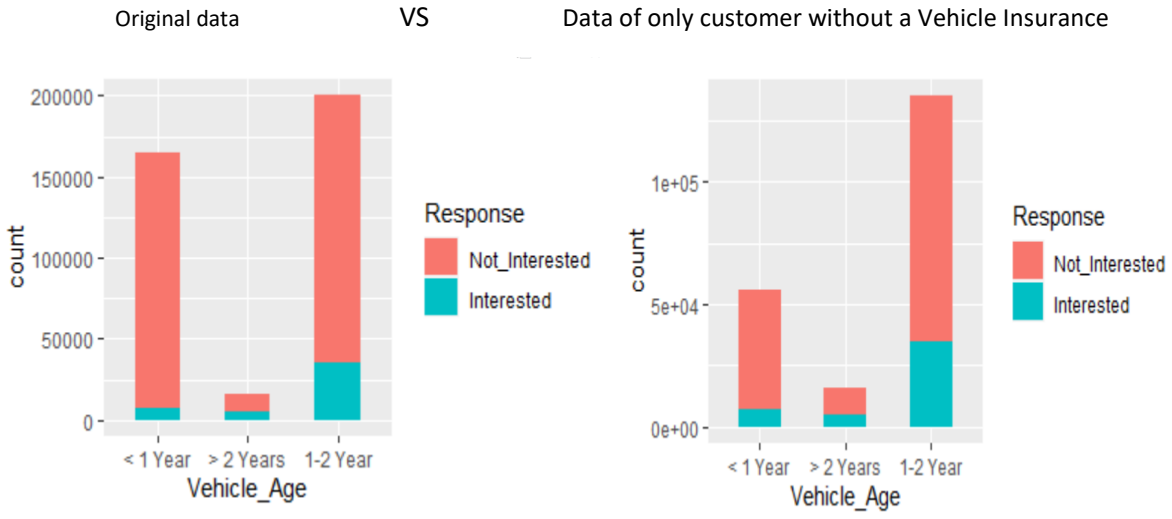


Figure6. Responses in “*Vehicle_Age*” in dataset with and without “Previously_Insured” equals 1.

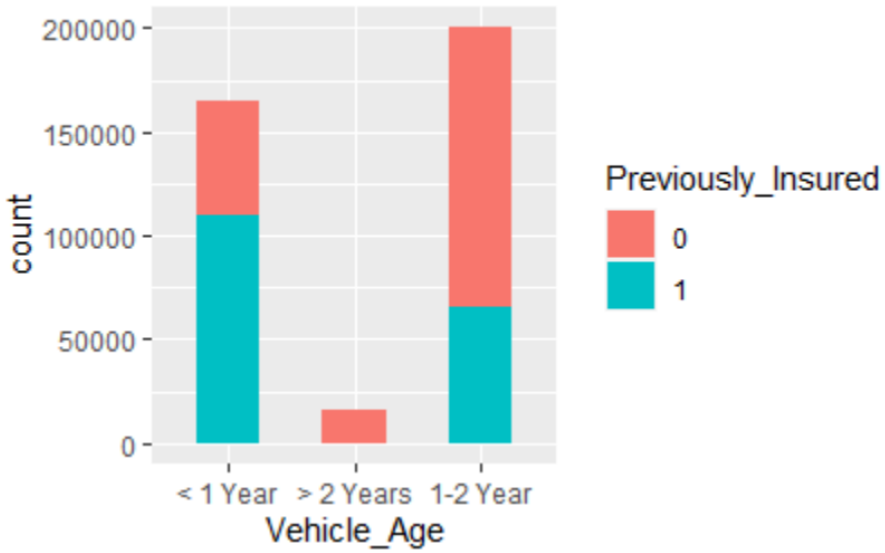


Figure7. Customers previously insured their vehicle by different vehicle age

Having excluded the confounding observations, the relationships between “Response” and other variables are examined one at a time. Firstly, a histogram is plotted to find how age influences a customer’s response. This shows that under age 25, the proportion of customers who are interested in buying the Vehicle Insurance is less than the rest who aged over 25.

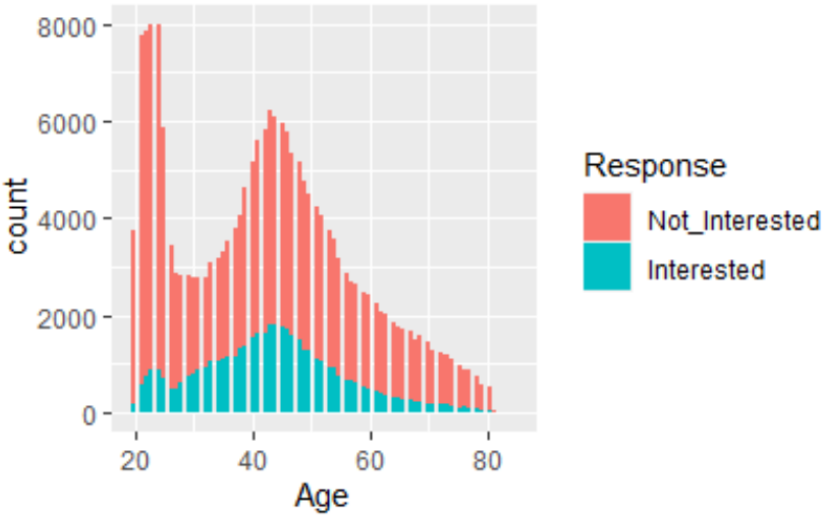


Figure8. Responses by customer in different age.

Next, we find how a customer’s gender will influence the response. Though it shows that the interested area in male is larger than it is in female, but the total number of male customer also exceeds that in females on a same scale. This may indicate that “Gender” is a not a good predictor and has no influence on the final model.

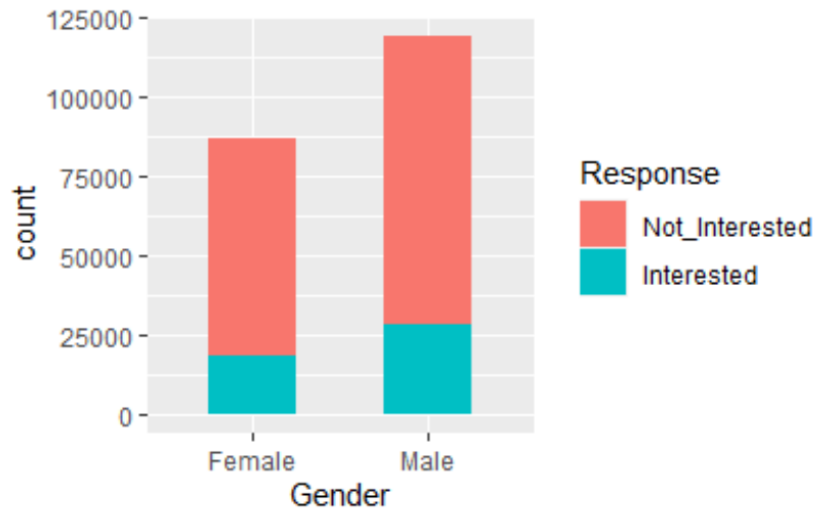


Figure9. Responses by customer in different gender.

Another less effective predictor is “Driving_License”. Over 99.5 percent of the customers are using type 1 driving license. So, it can be assumed that nearly all the customers would have one same Driving license. Therefore, “Driving_License” will not influence the prediction at all.

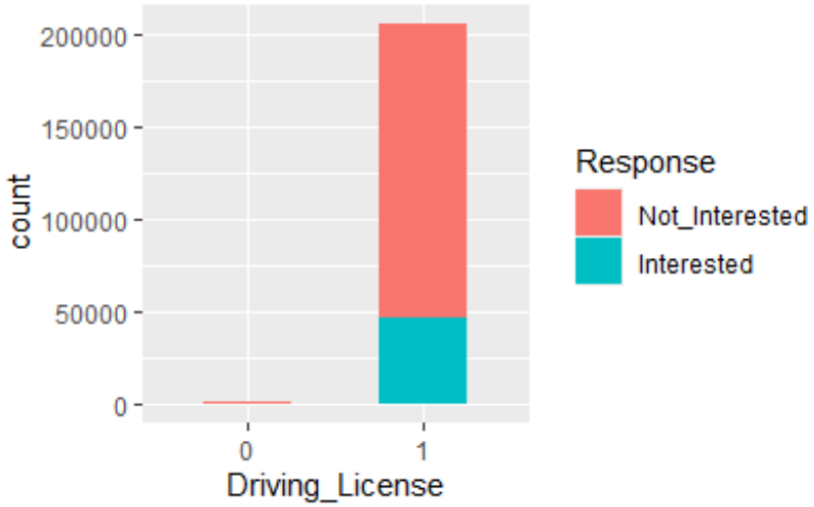


Figure10. Responses in 2 types of Driving license.

Also, “Region_Code” would not be considered as a good predictor. Despite the numbers of interested customers varies from region to region but, the proportion of “Interested” responses appears to be equal among all regions.

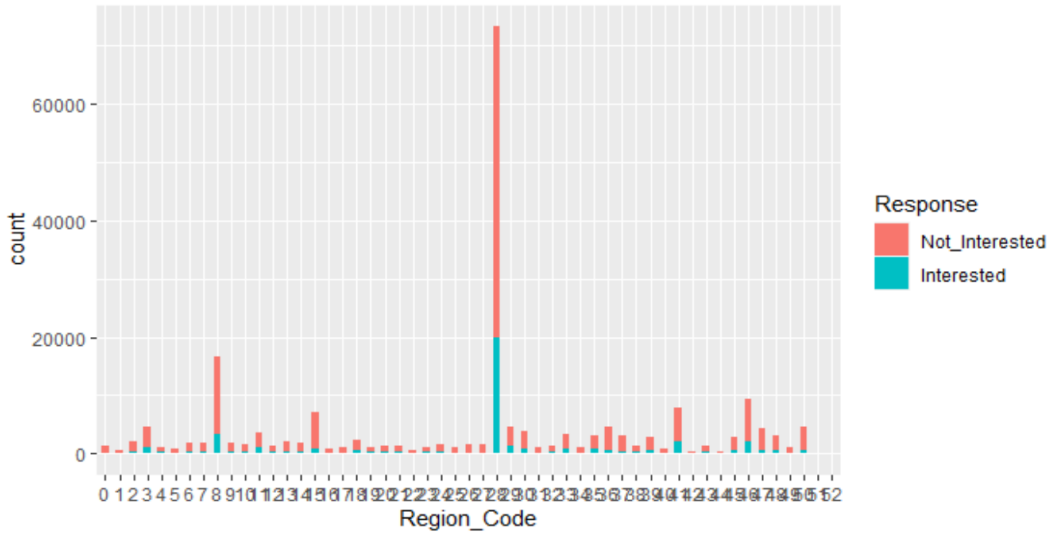


Figure11. Reponses among different regions.

The “Interested” percentage of vehicles in 3 different ages (less than one year; between 1 and 2 years; more than 2 years) differs. Vehicle Aged less than 1 year has the lowest “Interested” percentage of 10% whereas the vehicle older than 2 years has a highest percentage of 30%. As the vehicle gets older, a customer is more likely to be interested in buying an insurance for his/her vehicle.

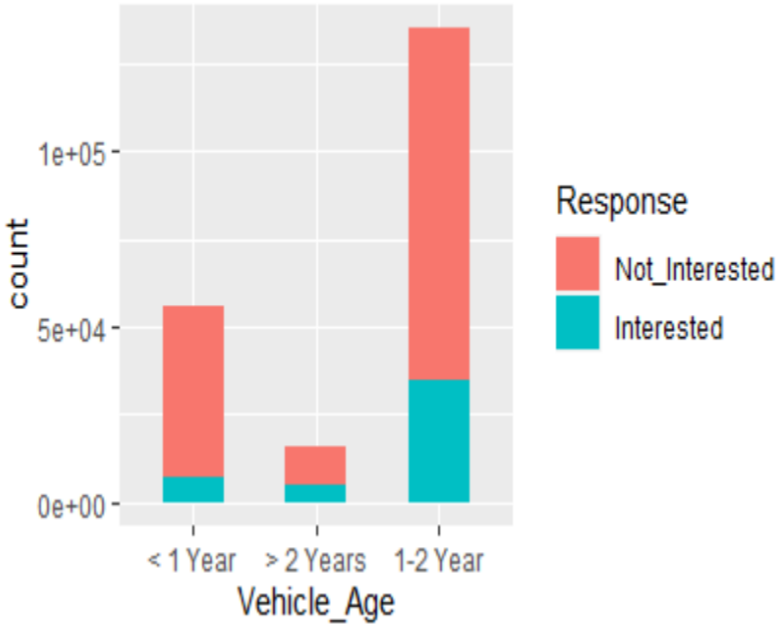


Figure12. Responses distribution in 3 levels of vehicle age.

It is often the case that after a vehicle with no insurance covered has been damaged, the owner would start to consider buying a Vehicle insurance. The chart below has explained it clearly, that most of the customers tend to be interested in insuring their vehicles is after they have been damaged. Only a tiny proportion of customers would buy a Vehicle Insurance before the damage has taken place.

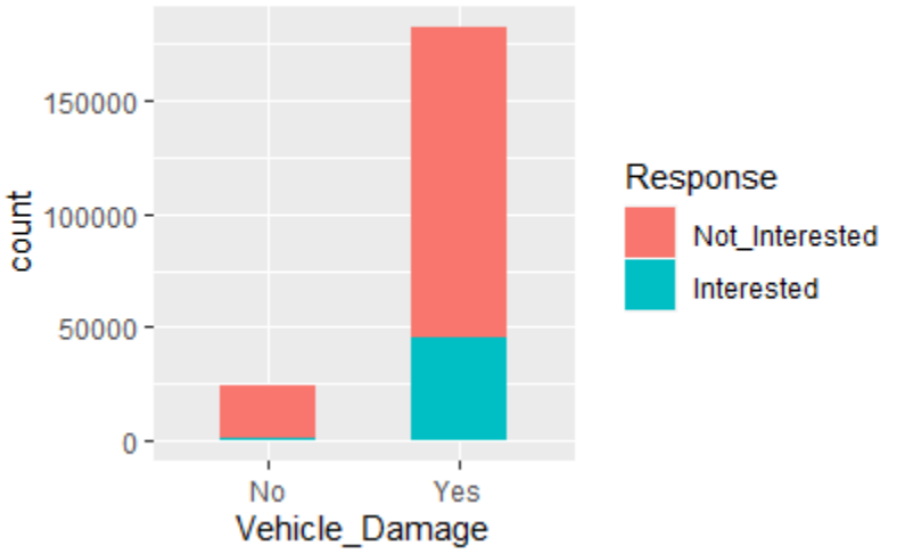


Figure13. Response distribution in vehicle damaged or not.

The histogram of annual premium paid by customers all stacked up over on the left side, and this indicates that there are some outliers in this vector. This can be clearly seen that the outliers are the values exceed 1e+05, and they only account for 0.5% of the whole data. After filtering the outliers, we get a better view of the plot. The distribution of the interested customers follows a same pattern of its total numbers. In other words, the number of customers interested is only influenced by the total customer number. Therefore, "Annual_Premium" is less likely to have influence on the final model.

Annual Premium with outliers VS Annual premium without outliers

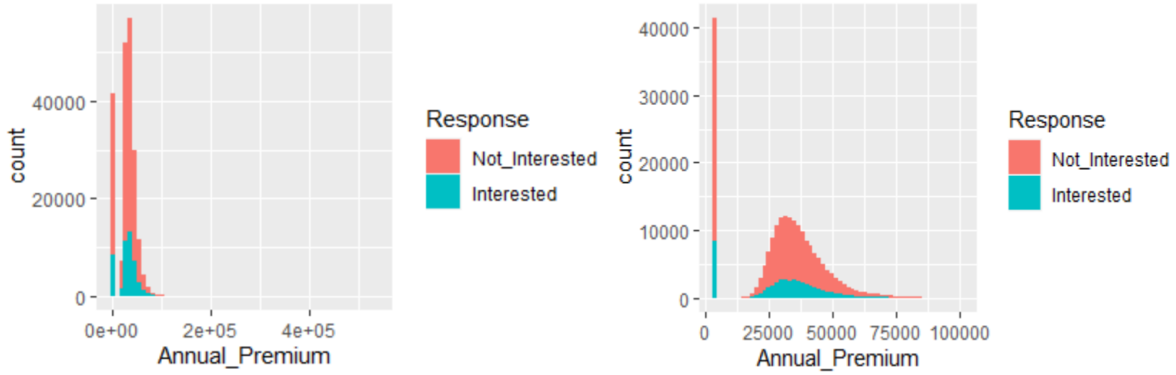


Figure14. Responses distribution among different levels annual premium.

"Policy_Sales_Channel" is a categorical variable with more than 160 levels. The majority of customers are from 5 main channels. The distribution of interested customers is various from channel to channel. This variable will have influence on the final prediction to some extent.

Figure15. Responses distribution in different levels of sales channels.

“Vintage” as mentioned before, is the number of days the customer has been associated with the company. Clearly, this variable would not have too much impact on the final model because the proportion of customer interested is equally distributed from the start to the end.

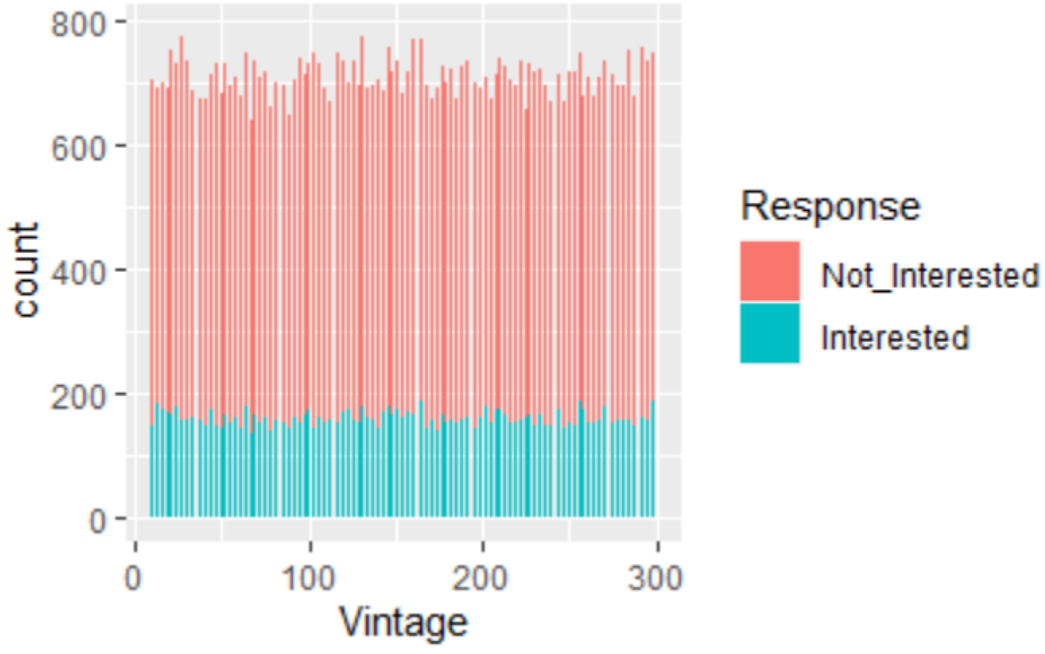


Figure16. Responses change by how long the customer has been stayed with the company.

In conclusion, having analysed and visualized the relationship between customer’s response with every other variable except id, it can be found that “Previously_Insured”, “Age”, “Vehicle_Age”, “Vehicle_Damage” and “Policy_Sale_Channel” will influence a customer’s interest on buying a Vehicle Insurance.

Train and test split

This original dataset has a large number records of 381109 with no missing value or special characters. It should be split into training and test sets before constructing our models for several reasons. The training set is used to fit the model whereas the test set is used to evaluate how that model fits. So, we should only train our models on the training set and test them later on test set. If a model is tested on some examples that are used as part of its training, the accuracy of the prediction will be high, but it is meaningless because this is an example of overfitting problem. Therefore, the dataset is randomly selected into training and test sets on a ratio of 8:2. Because we have such a large dataset, so both sets are large enough to yield statistically meaningful result.

Data balance

The proportion of "Interested" to "Not_Interested" response is about 1 to 9. This indicates that this dataset is unbalanced and that can be a problem for modelling. If we train our model on such an unbalanced dataset, 9 out of 10 times, the prediction will say "Not_Interested" and there is nothing wrong about it. Though this model has a high accuracy, but it ensures nothing because it is obviously biased. In our application, the client company is seeking customers who would be interested in buying their Vehicle Insurance so, the company can take some actions on these customers to increase the success rate in this cross selling.(the company can take advantage of this result and increase the chance of selling this cross sell.) Therefore, the company cares more about why a customer would be interested or what are the variables could have impact on this result. In this case, the train dataset should be balanced by its "Response" ratio before training.

There are several techniques to balance a dataset, I choose to use SMOTE (Synthetic Minority Oversampling Technique). In this method, the new examples oversampled are synthesized from the existing examples by randomly sampling from feature set. Thus, it will not add any new information to the model. Moreover, it is interesting to find that the SMOTE balanced data has also solved the confounding variable issue. The bar chart below shows the percentage of two responses in "Previously_Insured" from the balanced train set. The "Interested" response among the customers who already has a Vehicle insurance is increased from nearly 0 to more than 25% by comparing with the imbalanced dataset result. This also reveals that there was a number of customers interested in the Vehicle Insurance even they already had one. However, due to the small proportion of the "Interested" response, this information could not be discovered.

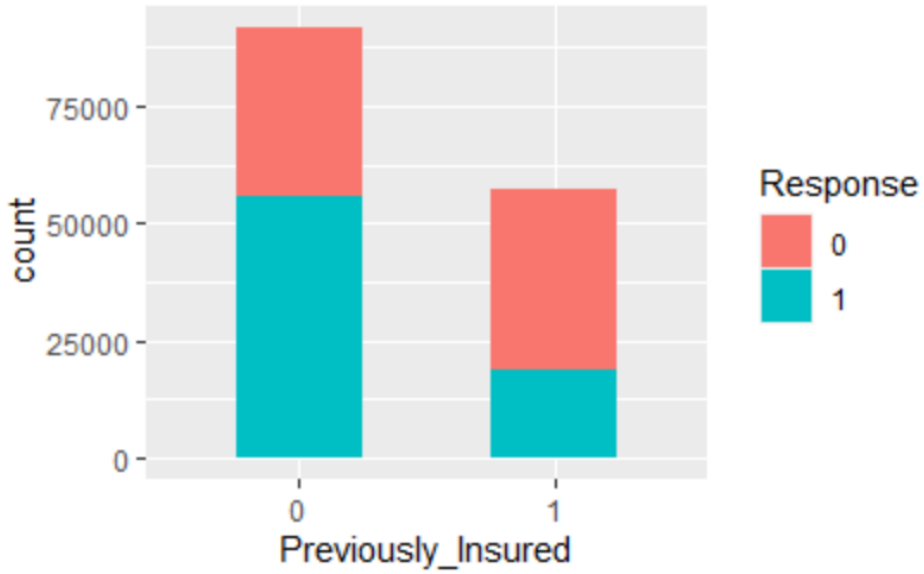


Figure17. Responses in “Previously_Insured” after data is balanced.

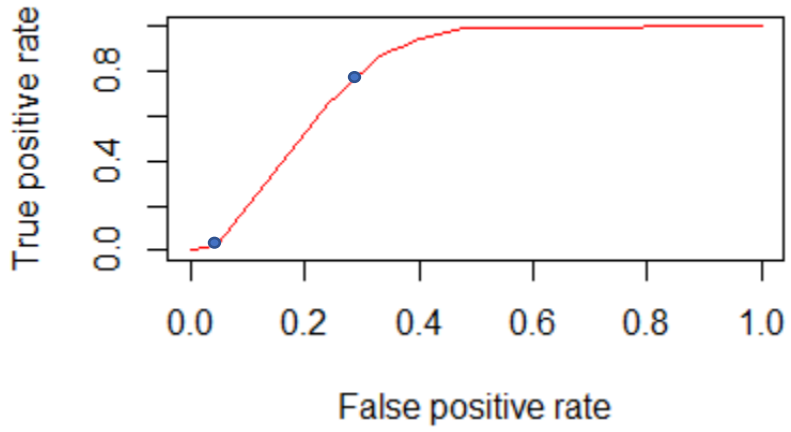
Another thing to be noticed is that the data should not be balanced before the train test split so that the test data would not be balanced. Because, the test data should be represented as real word data, and it is not supposed to be balanced.

Train models

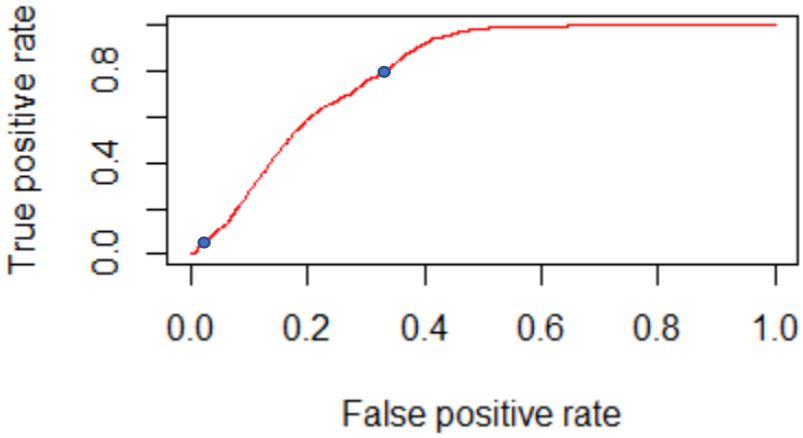
Two different models have been trained using different method. The first model is built by R library(rpart) for Decision Trees, while the second model applied Logistic Regression. The 5 previously concluded predictors: “Previously_Insured”, “Age”, “Vehicle_Age”, “Vehicle_Damage” and “Policy_Sale_Channel” have been used to predict the dependent variable “Response” on both models. After testing the imbalanced test data, ROC (Receiver Operating Characteristic) curve and AUC (Area Under ROC Curve) metrics are selected as a measurement for the two models. An ROC curve is a graph showing the performance of a classification model at all classification thresholds by plotting two parameters: TPR (True Positive Rate) and FPR (False Positive Rate) and AUC measures the area underneath that entire ROC curve.

Model selection

From the result, it shows that the AUC value of Logistic Regression model is 0.01 higher than the Decision Tree model overall. However, a slightly larger AUC value does not necessarily mean the Logistic Regression is the better model. Depending on the domain and the way we intend to use the model, each model has its own advantages and disadvantages. I used blue dots to mark two points where the true-positive rate equals 0.05 and 0.8. When the requirement of true-positive rate is on a very low threshold as 5%, Logistic Regression has a slightly lower false-positive rate, approximately 0.1 compare to that in Decision tree which is just above 0.2. On the other hand, if the model is required to have a true-positive rate over 80%, then the Decision Tree can achieve a lower false-positive rate as 28% compare to the 35 % in Logistic Regression. From the client company’s perspective, the goal is to discover all the customers that would like to purchase its Vehicle Insurance and would not want to miss a potential sale. Therefore, a high true-positive rate of at least 80% is likely to be the requirement of the company so that makes Decision Tree model the better selection.



ROC curve for Decision Tree model with AUC equals 0.79.



ROC curve for Logistic Regression model with AUC equals 0.80.

Conclusion

By successfully carrying out each step in the data analysis process from data cleaning to model selection, an ideal model is determined and can be used for the client company to predict whether a customer would be interested in its Vehicle Insurance. The relations between customer's response with the other variables have been clearly visualised and analysed. The train, test dataset split and data balance were taken out carefully with consideration to the dataset's feature. Finally, appropriate metrics such as ROC curve and AUC are used to test the results for the two models developed. The Decision Tree is selected to be the better model because it fits the company's intention that it is able to achieve a 80% true-positive rate and lower false-positive rate.

References

R. Itzikovitch, “7 Things You Should Know about ROC AUC,” *Medium*, 14-Oct-2019. [Online]. Available: https://medium.com/hiredscore-engineering/7-things-you-should-know-about-roc-auc-b4389ea2b2e3. [Accessed: 22-Oct-2020].

Aniruddha BhandariI am on a journey to becoming a data scientist. I love to unravel trends in data, “AUC-ROC Curve in Machine Learning Clearly Explained,” *Analytics Vidhya*, 20-Jul-2020. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/06/auc-roc-curve-machine-learning/. [Accessed: 21-Oct-2020].

J. Brownlee, “SMOTE for Imbalanced Classification with Python,” *Machine Learning Mastery*, 20-Aug-2020. [Online]. Available: https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/. [Accessed: 23-Oct-2020].

J. Brownlee, “A Gentle Introduction to Model Selection for Machine Learning,” *Machine Learning Mastery*, 25-Sep-2019. [Online]. Available: https://machinelearningmastery.com/a-gentle-introduction-to-model-selection-for-machine-learning/. [Accessed: 23-Oct-2020].

E. D. Jonge and M. V. D. Loo, “Statistical Data Cleaning with Applications in R,” 2018.