Health Insurance Cross Sell Prediction

Pawar Sapana,
Data science trainees,
AlmaBetter, Bangalore

Abstract:

Data science and machine learning are everywhere, even when we do not realize it. Less known to the public, but the insurance industry heavily relies on these. In this project, we apply the modern machine learning techniques on the insurance policyholders' data to analyze and predict their behavior. Using Python language, our approach to the data resulted in exciting insights to help the insurance companies in modeling their businesses.

Keywords: Machine Learning, Classification, Health Insurance

1.Problem Statement

Our client is an Insurance company that has provided Health Insurance to its customers now they need your help in building a model to predict whether the policyholders (customers) from past year will also be interested in Vehicle Insurance provided by the company.

An insurance policy is an arrangement by which a company undertakes to provide a guarantee of compensation for specified loss, damage, illness, or death in return for the payment of a specified premium. A premium is a sum of money that the customer needs to pay regularly to an insurance company for this guarantee.

Just like medical insurance, there is vehicle insurance where every year customer needs to pay a premium of certain amount to insurance provider company so that in case of unfortunate accident by the vehicle, the insurance provider company will provide a compensation (called 'sum assured') to the customer.

Building a model to predict whether a customer would be interested in Vehicle Insurance is extremely helpful for the company because it can then accordingly plan its communication strategy to reach out to those customers and optimise its business model and revenue.

Now, in order to predict, whether the customer would be interested in Vehicle insurance, we have information about demographics (gender, age, region code type), Vehicles (Vehicle Age, Damage), Policy (Premium, sourcing channel) etc.

1. **id :** Unique ID for the customer
2. **Gender :** Gender of the customer
3. **Age :** Age of the customer
4. **Driving License 0 :** Customer does not have DL, **Driving License 1 :** Customer already has DL
5. **Region Code :** Unique code for the region of the customer
6. **Previously Insured : 1** : Customer already has Vehicle Insurance, **0 :** Customer doesn't have Vehicle Insurance
7. **Vehicle Age :** Age of the Vehicle
8. **Vehicle Damage :1** : Customer got his/her vehicle damaged in the past. **0 :** Customer didn't get his/her vehicle damaged in the past.

9. **Annual Premium** : The amount customer needs to pay as premium in the year
10. **Policy Sales Channel :** Anonymized Code for the channel of outreaching to the customer i.e. Different Agents, Over Mail, Over Phone, In Person, etc.
11. **Vintage :** Number of Days, Customer has been associated with the company
12. **Response : 1 :** Customer is interested, **0 :** Customer is not interested

3. What is Cross Sell Prediction

It is important to understand the problem domain and key terms used in the definition of a problem before beginning a project. In the financial services industry, cross-selling is a popular term.

Cross-selling involves selling complementary products to existing customers. It is one of the highly effective techniques in the marketing industry.

To understand better, suppose you are a bank representative and you try to sell a mutual fund or insurance policy to your existing customer. The main objective behind this method is to increase sales revenue and profit from the already acquired customer base of a company.

Cross-selling is perhaps one of the easiest ways to grow the business as they have already established a relationship with the client. Further, it is more profitable as the cost of acquiring a new customer is comparatively higher.

4. Introduction

An insurance policy is an arrangement by which a company undertakes to provide a guarantee of compensation for specified loss, damage, illness, or death in return for the payment of a specified premium. A premium is a sum of money that the customer needs to pay regularly to an insurance company for this guarantee.

For this project, we are using the dataset that is about an Insurance company that has provided Health Insurance to its customers in past year and is now interested in providing Vehicle Insurance to its policy holders.

5. Objective

To predict if an insurance policy holder would be interested to buy a vehicle insurance as well. Building a model to predict whether a customer would be interested in 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.

The aim of this project is to leverage the machine learning algorithms such as Logistic Regression and Random Forest to create a predictive model using statistically significant variables from the given data set.

Model accuracy will be assessed using different techniques such as ROC (Receiver operating characteristic), AUC (Area under the ROC curve) and Confusion Matrix.

Hypothesis Generation for Cross-Sell Prediction

The structured thinking approach will help us here. Let me state some hypotheses from our problem statement.

1. Male customers are more tend to buy vehicle insurance than females.
2. The middle-aged customers would be more interested in the insurance offer.
3. Customers having a driving license are more prone to convert.

4. Those with new vehicles would be more interested in getting insurance.
5. The customers who already have vehicle insurance won’t be interested in getting another.
6. If the Customer got his/her vehicle damaged in the past, they would be more interested in buying insurance.

6. Steps involved:

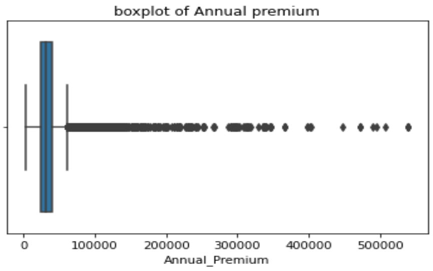
- **Exploratory Data Analysis**

After loading the dataset we performed this method by comparing our target variable that is Response with other independent variables. This process helped us figuring out various aspects and relationships among the target and the independent variables. It gave us a better idea of which feature behaves in which manner compared to the target variable.

- **Null values Treatment**

Our dataset does not contains any null value and also there are no duplicates present in the dataset.

- **Outliers**

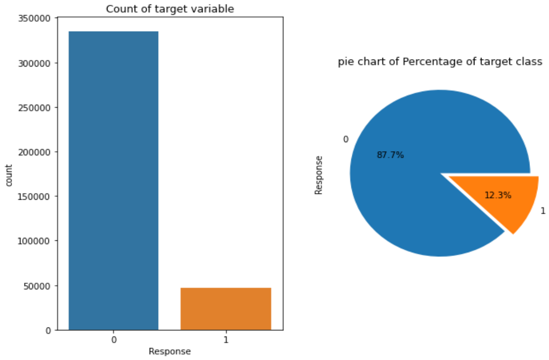


After plotting boxplot we can see dataset contain only one column having dataset, After that we performed IQR method to remove outliers.

- **Data Visualization:**

Target Variable:

We had plotted the count plot and percentage plot of target variable Response.

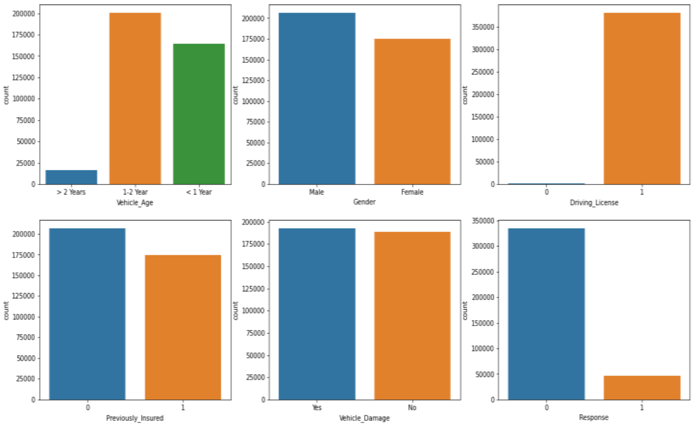


The target variable is highly imbalanced.

By the plot we can say that this is imbalance binary classification problem.

The customers interested in insurance is 87 % as compared to the other one.

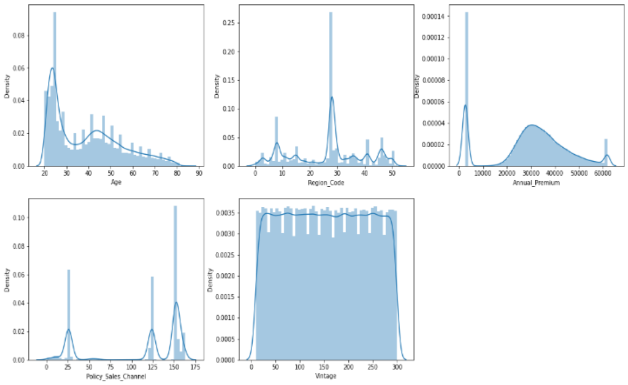
Univariate Analysis Of Categorical Variables



- The plot shows that most of the vehicle taken in this study is 1-2 years old. There is very less number of customers with vehicle age less than 2 years.
- The gender variable in the dataset is almost equally distributed.
- Almost all people have driving license.

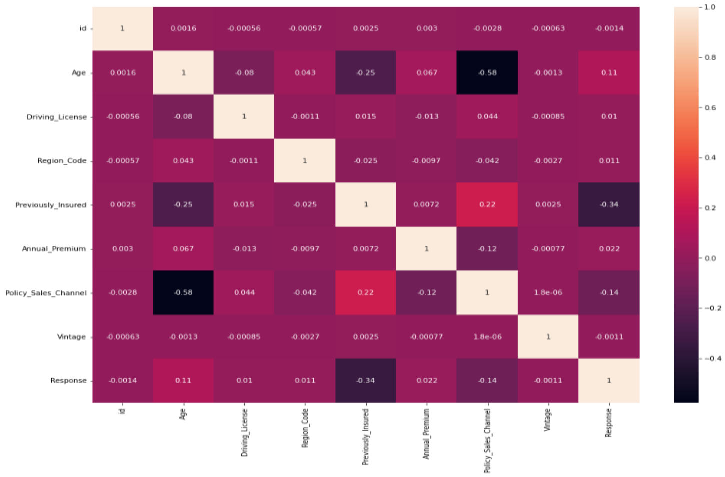
- There are more number of people who have no previous insurance policy than the other one.
- Customers with vehicle damage Yes and NO are equally distributed.
- We can say that most of the customers are not interested in vehicle insurance policy.

Univariate Analysis Of Continuous Variables



- The Column Age is highly skewed towards right.
- The Column Region Code is randomly distributed. The individuals with region code 28 the highest as compared to the other ones
- The column Annual Premium normally distributed with little right skewed.
- The variable Policy Sales Channel is randomly distributed with hueness.
- The variable Vintage is uniformly distributed.

Heatmap:



- ➢ **Encoding of categorical columns**

We used Label Encoding usually deal with datasets which contains multiple labels in one or more than one columns. These labels can be in the form of words or numbers. To make the data understandable or in human readable form, the training data is often labeled in words. Label Encoding refers to converting the labels into numeric form so as to convert it into the machine-readable form.

- ➢ **Handling Imbalanced Data**

As from the distribution of target variables in the EDA section, we knew it was an imbalance problem. The imbalance datasets could have their own challenge. For example, a disease prediction model may have an accuracy of 99% but it is of no use if it can not classify a patient successfully.

So to handle such a problem, we could resample the data. For this problem we used Random Oversampling.

- ➢ **Standardization of features**

Our main motive through this step was to scale our data into a uniform format that would allow us to utilize the data in a better way while performing fitting and applying different algorithms to it.

- **Fitting different models**

For modelling we tried various classification algorithms like:

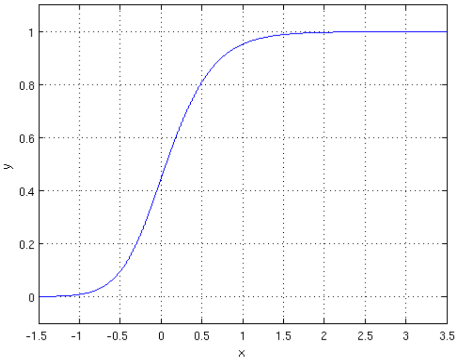
1. **Logistic Regression**
2. **Random Forest Classifier**
3. **XG Boost classifier**
4. **Naive Bayes Classifier**

7. Algorithms:

1. Logistic Regression:

Logistic Regression is actually a classification algorithm that was given the name regression due to the fact that the mathematical formulation is very similar to linear regression.
The function used in Logistic Regression is sigmoid function or the logistic function given by:

f(x)= 1/1+e ^(-x)

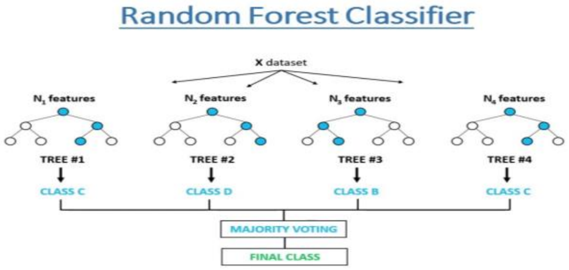


The optimization algorithm used is: Maximum Log Likelihood. We mostly take log likelihood in Logistic:

$$\ln L(\mathbf{y},\boldsymbol{\beta}) = \ln \prod_{i=1}^{n} f_i(y_i) = \sum_{i=1}^{n}\left[y_i \ln\left(\frac{\pi_i}{1-\pi_i}\right)\right] + \sum_{i=1}^{n} \ln(1-\pi_i)$$

2. Random Forest Classifier:

Random Forest is a bagging type of Decision Tree Algorithm that creates a number of decision trees from a randomly selected subset of the training set, collects the labels from these subsets and then averages the final prediction depending on the most number of times a label has been predicted out of all.

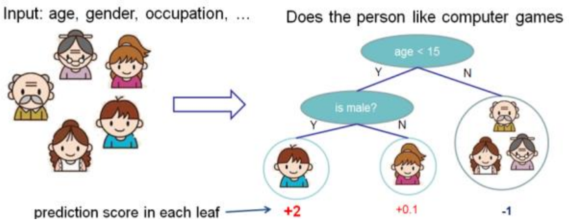


3. XG Boost-

To understand XGBoost we have to know gradient boosting beforehand.

Gradient Boosting-

Gradient boosted trees consider the special case where the simple model is a decision tree



Input: age, gender, occupation, ...
Does the person like computer games
age < 15
Y
N
is male?
Y
N
prediction score in each leaf
+2
+0.1
-1

In this case, there are going to be 2 kinds of parameters P: the weights at each leaf, w, and the number of leaves T in each tree (so that in the above example, T=3 and w=[2, 0.1, -1]).

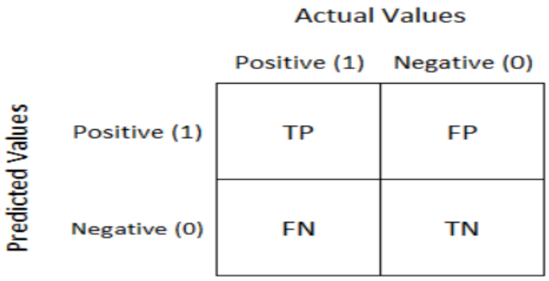
When building a decision tree, a challenge is to decide how to split a current leaf. For instance, in the above image, how could I add another layer to the (age > 15) leaf? A 'greedy' way to do this is to consider every possible split on the remaining features (so, gender and occupation), and calculate the new loss for each split; you could then pick the tree which most reduces your loss.

XGBoost is one of the fastest implementations of gradient boosting. trees. It does this by tackling one of the major inefficiencies of gradient boosted trees: considering the potential loss for all possible splits to create a new branch (especially if you consider the case where there are thousands of features, and therefore thousands of possible splits). XGBoost tackles this inefficiency by looking at the distribution of features across all data points in a leaf and using this information to reduce the search space of possible feature splits.

Model performance:

Model can be evaluated by various metrics such as:

Confusion Matrix: It is a performance measurement for machine learning classification problem where output can be two or more classes. It is a table with 4 different combinations of predicted and actual values.



		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Let's understand TP, FP, FN, TN.

True Positive:

Interpretation: You predicted positive and it's true.

True Negative:

Interpretation: You predicted negative and it's true.

False Positive:

Interpretation: You predicted positive and it's false.

False Negative:

Interpretation: You predicted negative and it's false.

It is extremely useful for measuring Recall, Precision, Specificity, Accuracy, and most importantly AUC-ROC curves.

Precision: It shows how many of the values we guess as Positive are actually Positive. And it is given by:

TP/TP+FP

Recall: It is a metric that shows how many of the operations we need to predict positive. And it is given by:

TP/FN+TP

F1 score: The F1 Score value shows us the harmonic mean of the Precision and Recall values. And it is given by:

$$F1\ score = 2 * \frac{Precision\ * Recall}{Precision + Recall}$$

Accuracy:

Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. In terms of the confusion matrix, it is given by:

TP+TN/TP+TN+FP+FN

Area under ROC Curve(AUC):

ROC curves use a combination of the true positive rate (the proportion of positive examples predicted correctly, defined exactly as recall) and false positive rate (the proportion of negative examples predicted incorrectly) to build up a summary picture of the classification performance.

Hyper parameter tuning:

Hyperparameters are sets of information that are used to control the way of learning an algorithm. Their definitions impact parameters of the models, seen as a way of learning, change from the new hyperparameters. This set of values affects performance, stability and interpretation of a model. Each algorithm requires a specific hyperparameters grid that can be adjusted according to the business problem. Hyperparameters alter the way a model learns to trigger this training algorithm after parameters to generate outputs.

We used Grid Search CV, Randomized Search CV and Bayesian Optimization for hyperparameter tuning. This also results in cross validation and in our case we divided the dataset into different folds. The best performance improvement among the three was by Bayesian Optimization.

1. **Grid Search CV-**Grid Search combines a selection of hyperparameters established by the scientist and runs through all of them to evaluate the model’s performance. Its advantage is that it is a simple technique that will go through all the programmed combinations. The biggest disadvantage is that it traverses a specific region of the parameter space and cannot understand which movement or which region of the space is important to optimize the model.

2. **Randomized Search CV-** In Random Search, the hyperparameters are chosen at random within a range of values that it can assume. The advantage of this method is that there is a greater chance of finding regions of the cost minimization space with more suitable hyperparameters, since the choice for each iteration is random. The disadvantage of this method is that the combination of hyperparameters is beyond the scientist’s control

8. Conclusion:

That's it! We reached the end of our exercise.
Starting with loading the data so far we have done EDA , null values treatment, encoding of categorical columns, feature selection and then model building.
In all of these models our accuracy revolves in the range of 70 to 74%.
And there is no such improvement in accuracy score even after hyperparameter tuning.
So the accuracy of our best model is 73% which can be said to be good for this large dataset. This performance could be due to various reasons like: no proper pattern of data, too much data, not enough relevant features.

References-

1. MachineLearningMastery
2. GeeksforGeeks
3. Analytics Vidhya