Supervised ML(classification) - Health Insurance Cross sell prediction

Technical documentation

—

Mohammed Akifuddin
Kashif
akifkashif007@gmail.com

Introduction

Often we see people purchasing Health Insurance for themselves and their family. Health Insurance is crucial for betterment of their lives. Vehicle Insurance is also important too but not as much as Health Insurance. 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. We are required to a dataset of customers who purchased Health Insurance in the previous year.

Problem statement

We are tasked with predicting whether a customer who previously purchased a company's Health Insurance will opt for Vehicle Insurance or not. This is a supervised machine learning classification problem with many independent variables and dependent variable namely Response which comprises responses of customers regarding vehicle insurance. Our model helps understand the behaviors of customers and build a model around that behavior.

Overview of data

We are given the following columns in our data:

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, 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.   **PolicySalesChannel** : Anonymized Code for the channel of outreaching to the customer ie. 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

Steps involved

I. Performing EDA (exploratory data analysis)

    A. Extracting the hail and tail of the dataset.

    B. Extracting the description and info of the dataset to check for mean, min, max values and non-null count.

    C. Getting the shape of the data to know the number of rows and columns

    D. Checking the presence and sum of null values of all columns(there weren't any null values)

    E. Applying the IQR method to find the outliers in the Annual Premium Column and removing them by limiting the values and then visualizing it using boxplots.

    F. Extracting correlation heatmap and calculating VIF to remove correlated and multicollinear variables.

    G. Implemented resampling to remove the class imbalance of dependent variable values i.e, Response column

II. Drawing conclusions from the data

- There is a great disparity among positive and negative responses from customers.
- Among the positive responses, males were more interested in purchasing Vehicle Insurance.
- PolicyHolders between age groups 27-45 were most interested in vehicle insurance
- Most negative responses are from the age group 23 and 24 years
- The customers who possess driving licenses almost always purchase vehicle insurance.
- People who don't already have a vehicle insurance policy opt in for vehicle Insurance.
- If the customers' vehicle is damaged, they definitely will buy vehicle insurance as seen in the data.
- There is substantial response from Area code 28 followed by codes 8 and 46.

III. Training the model

- Assigning the dependent and independent variables
- Splitting the model into train and test sets.
- Transforming data using StandardScaler.
- Fitting logistic regression on train set.
- Getting the predicted dependent variable values from the model.

IV. Evaluating metrics of model

A. Getting Accuracy, Precision, Recall, F1 scores for different models used.

    a. **PRECISION:** The precision score is a useful measure of the **success of prediction when the classes are very imbalanced.** Mathematically, it represents the ratio of true positive to the sum of true positive and false positive.**Precision Score = TP / (FP + TP).**

    b. RECALL : Model recall score represents the model's ability to correctly predict the positives out of actual positives. This is unlike precision which measures how many predictions made by models are actually positive out of all positive predictions made. The higher the recall score, the better the machine learning model is at identifying both positive and negative examples. Recall is also known as sensitivity or the true positive rate. Mathematically, it represents the ratio of true positive to the sum of true positive and false negative. **Recall Score = TP / (FN + TP).**

    c. ACCURACY : Model accuracy is a machine learning classification model performance metric that is defined as the ratio of true positives and true negatives to all positive and negative observations. **Accuracy Score = (TP + TN)/ (TP + FN + TN + FP)**

    d. F1 SCORE : Model F1 score represents the model score as a function of precision and recall score. F-score is a machine learning model performance metric that gives equal weight to both the Precision and

Recall for measuring its performance in terms of accuracy, making it an alternative

e. to Accuracy metrics. **F1 Score = 2* Precision Score * Recall Score/ (Precision Score + Recall Score/)**

Models used

Logistic regression:

Classification models are used in classification problems to predict the target class of the data sample. The classification model predicts the probability that each instance belongs to one class or another. It is important to evaluate the performance of the classifications model in order to reliably use these models in production for solving real-world problems. Performance measures in machine learning classification models are used to assess how well machine learning classification models perform in a given context. These performance metrics include **accuracy, precision, recall, and F1-score**. Because it helps us understand the strengths and limitations of these models when making predictions in new situations, model performance is essential for machine learning.

Logistic Regression Assumptions:

- First, binary logistic regression requires the dependent variable to be binary and ordinal logistic regression requires the dependent variable to be ordinal.
- Second, logistic regression requires the observations to be independent of each other. In other words, the observations should not come from repeated measurements or matched data.

- Third, logistic regression requires there to be little or no multicollinearity among the independent variables. This means that

- the independent variables should not be too highly correlated with each other.
- Fourth, logistic regression assumes linearity of independent variables and log odds. Although this analysis does not require the dependent and independent variables to be related linearly, it requires that the independent variables are linearly related to the log odds.
- Finally, logistic regression typically requires a large sample size. A general guideline is that you need at least 10 cases with the least frequent outcome for each independent variable in your model. For example, if you have 5 independent variables and the expected probability of your least frequent outcome is .10, then you would need a minimum sample size of 500 (10*5 / .10).
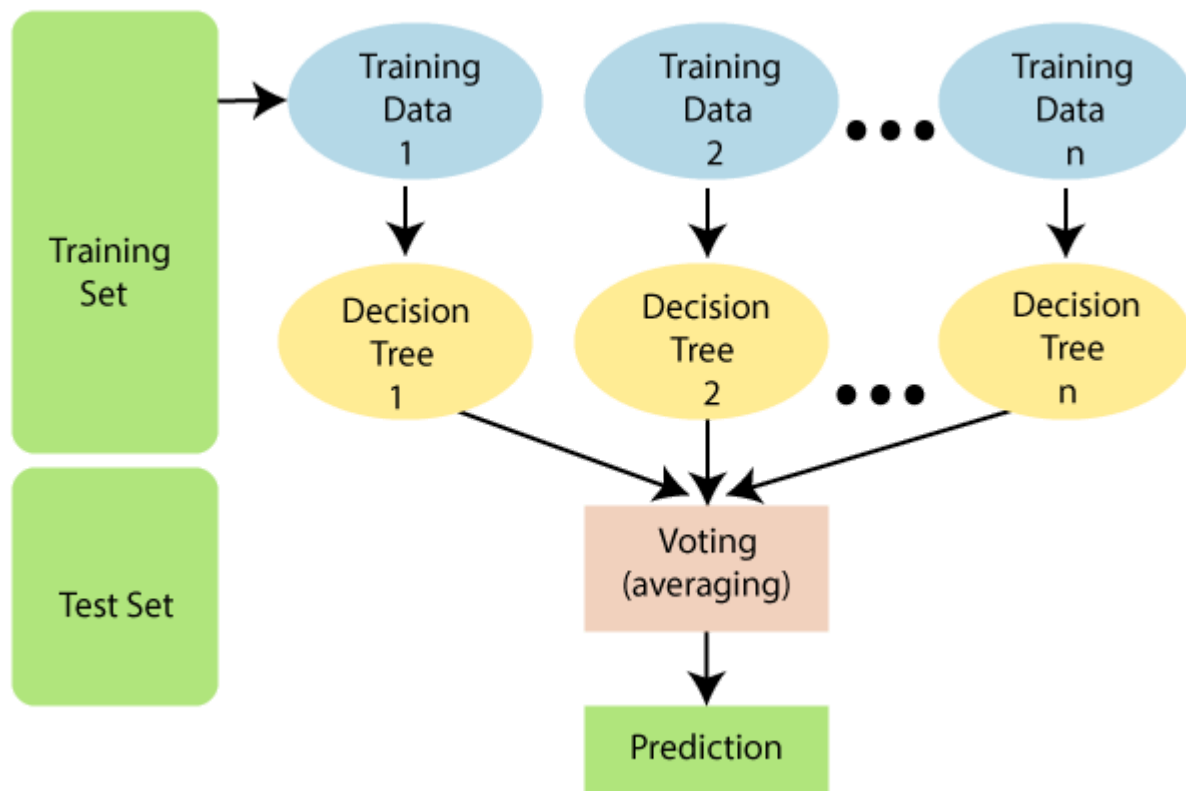
Random Forest Classification Model:

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of **ensemble learning,** which is a process of *combining multiple classifiers to solve a complex problem and to improve the performance of the model.*

As the name suggests, **"Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset**

and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

Training Set

Test Set

Training Data 1 → Decision Tree 1

Training Data 2 → Decision Tree 2

● ● ●

Training Data n → Decision Tree n

● ● ●

Voting (averaging)

Prediction

XGBoost Classification Model:

XGBoost is an algorithm that has recently been dominating applied machine learning and Kaggle competitions for structured or tabular data.

XGBoost is an implementation of gradient boosted decision trees designed for speed and performance. The library is laser focused on computational speed and model performance, as such there are few frills. Nevertheless, it does offer a number of advanced features. The implementation of the model supports the features of the scikit-learn and R implementations, with new additions like regularization. Three main forms of gradient boosting are supported:

- **Gradient Boosting** algorithm also called gradient boosting machine including the learning rate.
- **Stochastic Gradient Boosting** with sub-sampling at the row, column and column per split levels.
- **Regularized Gradient Boosting** with both L1 and L2 regularization.

Naïve Bayes Classifier Algorithm:

- Naïve Bayes algorithm is a supervised learning algorithm, which is based on **Bayes theorem** and used for solving classification problems.

- It is mainly used in *text classification* that includes a high-dimensional training dataset.

- Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.

- It is a probabilistic classifier, which means it predicts on the basis of the probability of an object.

- Some popular examples of Naïve Bayes Algorithm are spam filtration, Sentimental analysis, and classifying articles.

- Bayes' theorem is also known as **Bayes' Rule** or **Bayes' law**, which is used to determine the probability of a hypothesis with prior knowledge. It depends on the conditional probability.

- The formula for Bayes' theorem is given as:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Challenges faced

1. Pre-processing the data was one of the challenges we faced which includes removing unimportant columns from the data so as to not hinder the performance of our classification model.
2. Feature Engineering was one of the challenges that needed to be carefully done as it might increase or decrease the performance of our model and sensible features should be added.
3. Selecting the appropriate classification models was also a challenge as we never know which model performs the best on a given dataset and also there are a wide range of models in the Machine Learning Community.

Conclusion

We are finally at the conclusion of our project! Coming from the beginning we did EDA on the dataset and also cleaned the data according to our needs.After that we were able to draw relevant conclusions from the given data and then we trained our model on logistic regression and other models .Out of all models used , with the XGBoost classification model we were able to get the F1-score of 0.80.The model which performed poorly was Naive Bayes Classification model with F1-score of 0.73. Given the size of data and the amount of irrelevance in the data , the above score is good.