**Health Insurance Cross Sell**

**Indugopal Maity**

**Data science trainees**

**Alma Better, Bangalore**

**Abstract:**

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.

Health insurance is **a type of insurance that covers medical expenses that arise due to an illness**. These expenses could be related to hospitalization costs, cost of medicines or doctor consultation fees.

Health insurance encompasses two types - **Indemnity plans and Definite Benefit Plan**. The indemnity plans are traditional health covers which cover hospitalization costs from the sum assured. Definite benefit plans offer lump sum payment on detection of illness.

***Keywords: machine learning, health Insurance, Response Count, Driving License, Gender, Age, Previously Insurance and Vehicle\_Age.***

**1.Problem Statement**

Data, “TRAIN-HEALTH INSURANCE CROSS SELL PREDICTION.csv”, provided by an Alma Better. 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.

* Driving License available or not
* How different parameters are related (Age, Gender, Previously Insured and etc.)
* Events (Vehicle Damaged or Not)
* Type of Vehicle (Vintage or not)

**2. Introduction**

Vehicle insurance is insurance for cars, trucks, motorcycles, and other road vehicles. Its primary use is to provide financial protection against physical damage or bodily injury resulting from traffic collisions and against liability that could also arise from incidents in a vehicle.

### Our goal here is to build a predictive model, which could help us in predicting whether the Health insured are also interested for a vehicle Insurance or not.

**3. Steps involved:**

* **Exploratory Data Analysis**

After loading the dataset, we performed this method by comparing our target variable that is Reponses Count 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 contains no missing values, so it`s a good data set to go forward.

* **Correlation Matrix**

By the help of correlation feature we can understand that vintage is not having any role in terms responses, so we can drop that correlated value.

* **Feature Selection**

In this steps we used algorithms like “ExtraTreeClassifier” classifier to check the results of each feature i.e. which feature is more important compared to our model and which is of less importance. We can drop the less important features like, “Driving\_License” and “Gender”.

* **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.

The basic goal was to enforce a level of consistency or uniformity to certain practices or operations within the selected environment.

* **Fitting different models**

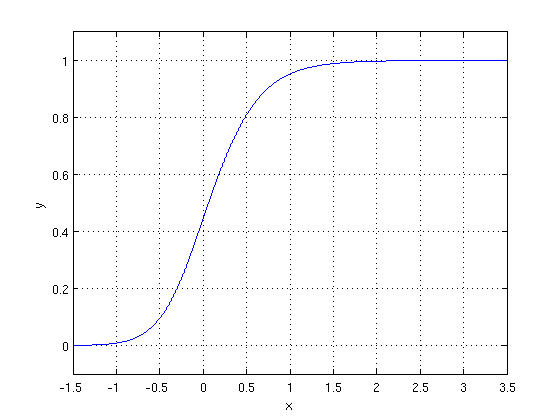
For modelling we tried various classification algorithms like:

1. **Logistic Regression**
2. **Random Forest Classifier**
3. **XGBoost classifier**

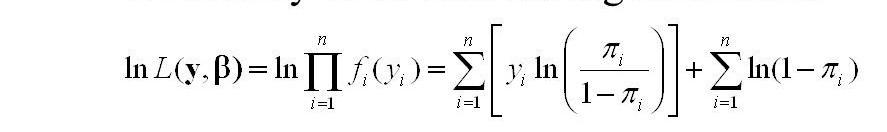
**5.1. Algorithms:**

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)

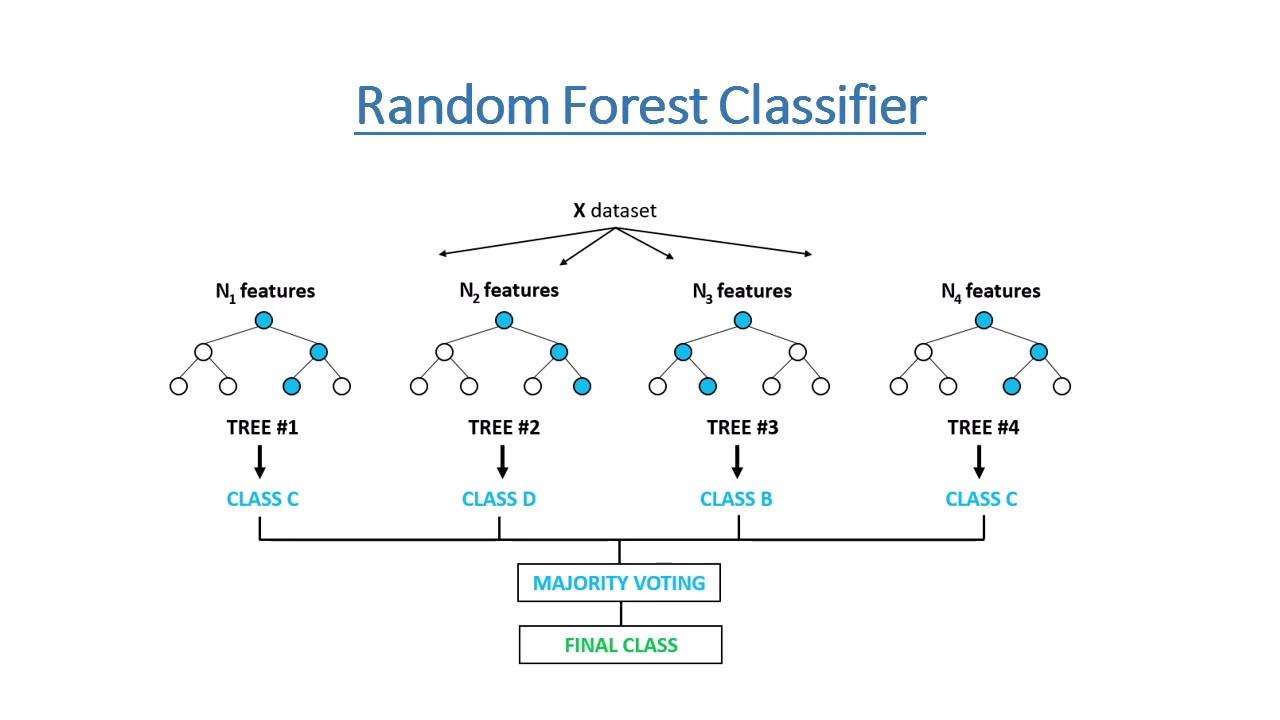


The optimization algorithm used is: Maximum Log Likelihood. We mostly take log likelihood in Logistic:



1. **Random Forest Classifier:**

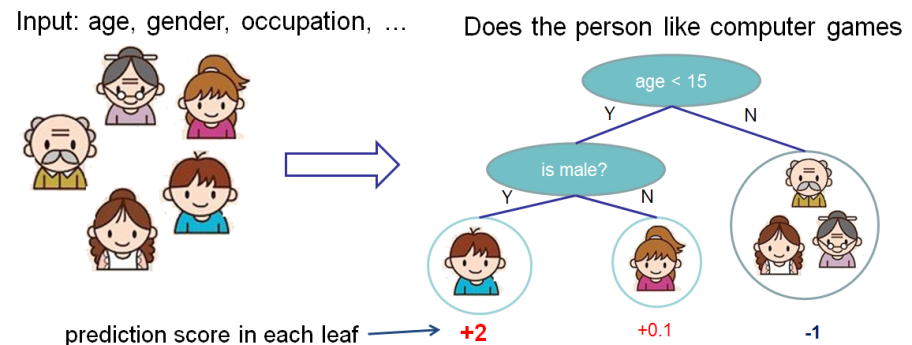
Random Forest is a bagging type of Decision Tree Algorithm that creates a number of decision trees from a randomly selected subset of the training set, collects the labels from these subsets and then averages the final prediction depending on the most number of times a label has been predicted out of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



1. **XGBoost-**

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.

****

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.

**5.2. Model performance:**

Model can be evaluated by various metrics such as:

**2. Precision/Recall**- Precision is the ratio of correct positive predictions to the overall number of positive predictions: TP/TP+FP

Recall is the ratio of correct positive predictions to the overall number of positive examples in the set: TP/FN+TP

1. **Accuracy**-

Accuracy is given by the number of correctly classified examples divided by the total number of classified examples. In terms of the confusion matrix, it is given by: TP+TN/TP+TN+FP+FN

**5.3. 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.

1. **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, feature selection and then model building. In all of these models our accuracy revolves in the range of 90% to 99%.

**References-**

1. Google.