**Technical Report**

This report aims at analysing the customer usage statistics dataset by collating various master and transaction tables and doing statistical analysis on them to develop probabilistic models which can predict the consumer propensity for various marketing schemes offered by the bank (Loan, Credit Cards and Mutual Funds) and maximize the revenue generated through them.

**Dataset and Statistical Analysis**

The data is divided into 4 main tables (Soc\_Dem, Inflow\_Outflow, Product\_ActBalance and Sales\_Revenue). As a part of the analysis a view was created in the database to consolidate the results of these tables. The entries of the view were first downloaded as CSV to build the dataset.

This dataset contains information about the customer such as age, tenure with the bank, account balance statistics, number of accounts (saving/current), number of debit/credit transactions etc. It also contains the target variable and the revenue targeted for each customer for a particular scheme offered by the bank (Loan, Credit Card and Mutual Funds).

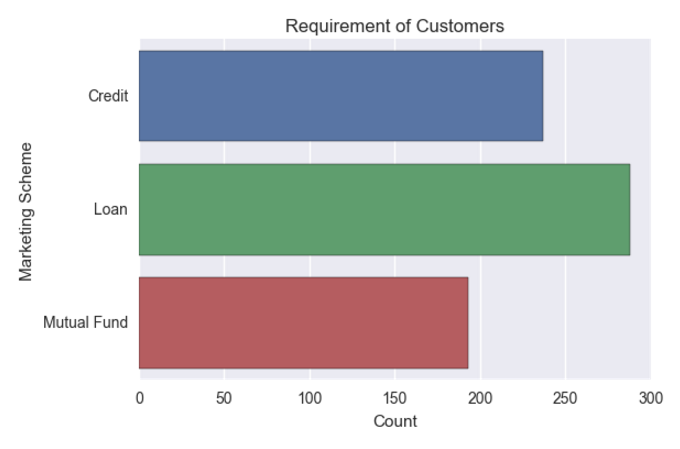
Total records: 1615

Total columns: 36

Although there were 1615 records, only 969 records were relevant as the target sale was available only for 60 % of the clients rendering the other records irrelevant for modelling.

Consequently, the records having null values for Sales Target were removed as a part of the cleansing process.

Furthermore, as the marketing offers were to be made only to the customers having a Target Sales output of 1 (Positive sales target), a horizontal bar plot was plotted to count the number of such targets across each category



This plot shows that most customers show tendency to opt for a consumer loan.

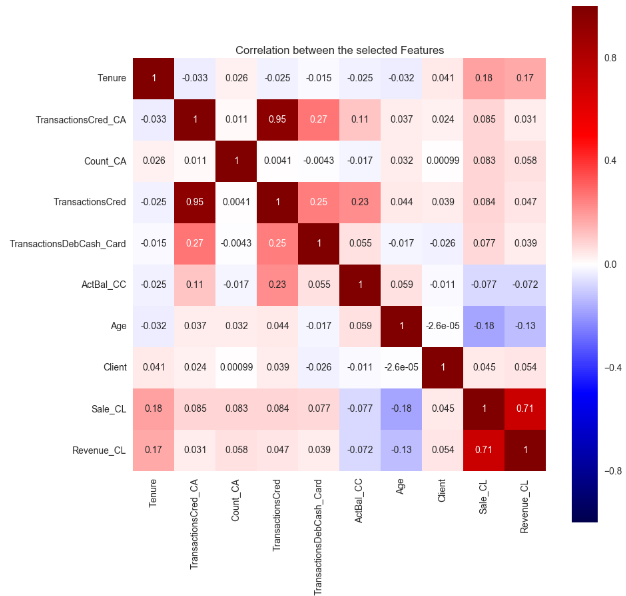
**Feature Engineering**

As the target variable is discrete, the whole analysis can be seen as a classification problem which can be modelled using one of many available techniques such as KNN, Logistic Regression, SVM, Random Forests etc.

A key part of the modelling process is the feature selection which involves determining the features to be fed to the model for effective predictions.

One way of achieving this, is by calculating the correlation of the target variable with all available fields and picking the features which are correlated. A Correlation score is between [-1,1] and a correlation of ~0 denotes that it is not related to the target.

The correlation matrix can be created by using the corr() function of the Panda library. For better understanding it can be visualized by plotting the matrix using heatmaps. One such example is denoted below which shows the correlation matrix score with respect to Sale\_CL.



As can be seen, the features such as Tenure, Age and Transaction Cred shows good correlation with Sale\_CL and hence can be used for modelling.

**Models Developed**

As there are three different Sales Target to be predicted, it requires three different probabilistic models.

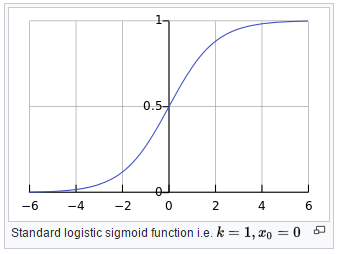
1. Propensity of Customers to buy Consumer Loans

As discussed in the previous section, the key to modelling is to feed the model with the correct features which can be determined by calculating the correlation scores.

The features selected for determining the Consumer Loans are 'Tenure', 'TransactionsCred\_CA',

'Count\_CA','TransactionsCred','TransactionsDebCash\_Card','ActBal\_CC' and 'Age'.

The **Logistic Regression** algorithm was chosen as it works well with Binary values. This algorithm predicts the likelihood of an event by fitting it into a logit function. A logit function is a sigmoid function given by the equation

Rather than minimizing the sum of squared errors, like in case of Linear regression, it maximizes the likelihood of occurrence of the target value. By fitting it into a curve it tries to replicate the step function.

Coming to the implementation part, the dataset was first split into training and testing data sets with a test size of 0.15 (as the bank can contact only 15 % of the customers) using train\_test\_split method of the cross\_validation package of sklearn. This method was used to avoid overfitting.

Using the linear\_model sub-package within the sklearn package, the LogisticRegression() method was loaded to fit the training data. The classifier which was used to fit the data was then used to determine the accuracy and predict the probabilities of the testing dataset.

**Results:**

The model had an accuracy of 62.93 %. The Client\_Loan dataset was updated with the predicted probabilities of the testing data, which was then exported as a csv file.



1. Propensity of Customers to buy Credit Cards

As with the previous case, the features were first identified by sorting the Correlation dataframe by Sale\_CC field (the target attribute for Credit Card Sale) and picking the features with higher correlation scores.

**Random Forest Classifier** was used to model this scenario. Random Forest in general works well with both regression and classification problems. It works by taking multiple decision trees as input during training time and its output is determined by the mean of the predictions of individual decision trees. It is an improvement over the decision trees as it tends to avoid overfitting the model. Since it takes multiple Decision trees to compute the result, it is a form of Ensemble learning.

Since Random forest is a very powerful algorithm and provides accurate predictions without overfitting, this method was chosen to predict the Target Sales of Credit cards.

The classifier was fitted by importing the RandomForestClassifier method from ensemble sub-package of sklearn library. The training data was then passed to the classifier for training and this classifier was finally used to make predictions on the testing data.

**Results:**

The model had an accuracy of 79 %. The Client\_Credit dataset was updated with the predicted probabilities of the testing data, which was then exported as a csv file.

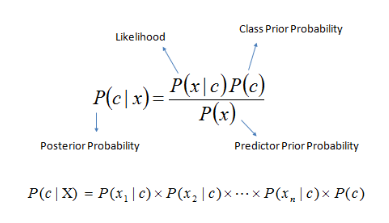


1. Propensity of Customers to buy Mutual Funds

After determining the features using the correlation matrix, the Sales target value for Mutual Funds was modelled using **Naïve Bayes Classifier**.

It is based on the Bayes theorem and assumes that there is complete independence between the predictors, i.e. the presence of a feature is assumed to be independent of other features.

Since the correlation values were very low it was safe to apply Naïve Bayes assumption in our case to determine the Sales target for Mutual funds.

Naïve Bayes is based on the formula,

It uses the prior probability to compute the posterior probability of an event. These algorithms are very simple to build because of its easy design. It has many methods such as GaussianNB(),MultinomialNB() and BernoulliNB().

In our case Naïve Bayes was chosen as it works particularly well with Binary choices ( like Spam/Not Spam, Positive/Negative etc ).

As far as implementation is concerned, GaussianNB() function is imported from the naïve\_bayes sub-package of Sklearn library.

The method is then assigned to a classifier which trains the model using the training datasets created using train\_test\_split method.

Once the model is trained, it can make predictions and compute the probability of testing data using the method predict\_proba.

**Results**

The model had an accuracy of 71.32 %. The Client\_mutual\_fund dataset was updated with the predicted probabilities of the testing data, which was then exported as a csv file.



**Revenue Determination using the predicted models**

Two strategies were employed to calculate the revenue generated by the market schemes.

1. Best Case Scenario
2. Expected Scenario

In the best case scenario, it was assumed that based on the predictions made by the models, suitable clients can be targeted. This included all clients who had a positive propensity and a Sales Target Value of 1.

The Revenue corresponding to those clients were then added the total best case revenue.

**Results**

Revenue From sale of Credit Card = 265.634285708

Revenue From sale of Consumer Loan = 497.7578571720001

Revenue From sale of Mutual Fund = 265.49410713699996

Total best case Revenue = 1028.886250017

The drawback of the above analysis is that, it assumes that all clients with a positive propensity score will buy the schemes offered to them if the Sales target Value = 1. In real world scenarios, it is impossible to realize such a proposition.

From the bank perspective, while contacting the customers there should be a reasonable probability that the customer should be interested in the offering. As a part of the strategy to maximize revenue in an efficient manner the propensity limit was set to 0.6 ,i.e. contact only those customers out of the testing data (the 15 % lot that the bank can contact) who are likely to buy the scheme ( Sale\_XX target variable = 1) and Propensity >= 0.6 (likely).

The list of such customers was then exported to a CSV file ‘CallTheseClients’, it has details about the client number, Target Sale value, Revenue and the propensity to buy the offer.



By employing such a method, it was observed that the number of customers to be contacted came down from 102 to 33. For that samples the following was the expected revenue for each group.

**Results**

Expected Revenue From sale of Credit Card = 63.93

Expected Revenue From sale of Consumer Loan = 221.583928589

Expected Revenue From sale of Mutual Fund = 32.973392861

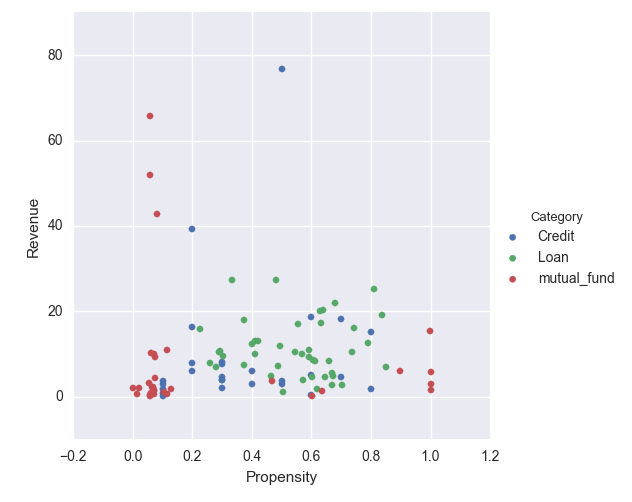
Total Expected Revenue = 318.48732144999997

The strategy to maximise the Revenue is therefore to contact customers having a higher propensity scores (so that the likelihood of the sale increases).

**Visualizations**

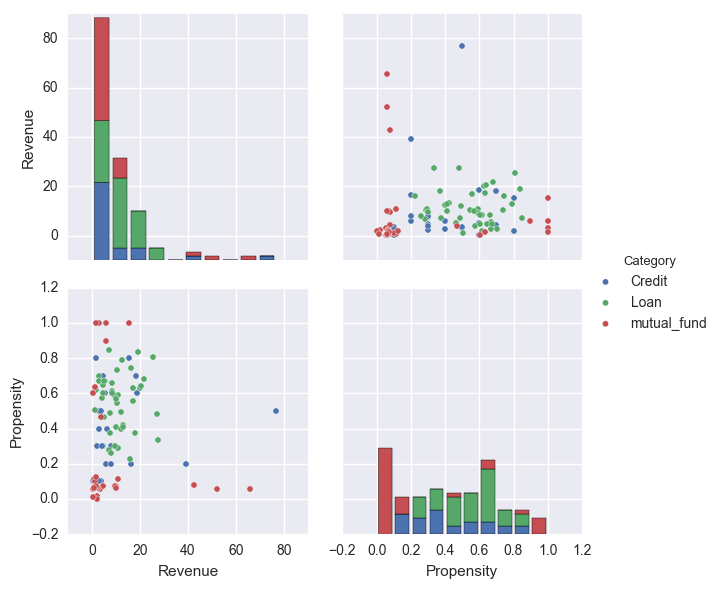
The results of the models were finally visualized using the seaborn library of python.

1. Spread of the Revenue with respect to the propensity for the three categories of marketing offer.



This visualization mainly depicts that most customers have a higher propensity to buy Loans, and it also tells us that revenue for most offers are under 40 (except for a few cases).

1. Pairplot tells us how the pair of attributes Revenue and Propensity vary with each other and itself for the three different categories ( Loan, Credit Cards and Mutual Funds).



This plot reveals a lot of information on how the different categories differ in terms of the probability of the customer to buy a particular offer and the revenue expected when a sale is made from a particular category.

We can clearly see that very few customers generate high revenue and the bulk of the revenue is nominal.

**Assumptions**

1. As the data was split into training and testing datasets with a test size of 0.15, it is assumed that the decisions will be made based on the pool of customers coming from the randomized generation of the test data.
2. All customers having the target sale attribute of 1 can be offered a marketing scheme regardless of the propensity score.
3. Training data is not used to target the best clients to maximize the revenue.

**Limitations**

1. The models could have been trained better with more data and better feature engineering.
2. A very low correlation was observed between the target and the features which impacted the fitness of the model.

**Conclusion**

With the advancement of technologies and the variety of schemes on offers, banks need to become smarter on their product offerings. With the digitalisation of data and cheap storage options, these data can be used to model algorithms which are capable of predicting the needs of a customer. This is a win-win situation for both the clients and the vendors, as, while the clients benefit from meaningful recommendations on marketing schemes, the vendors can also efficiently target the right clients with the right offers, thereby maximising their profits. The models developed in this report can further be improved by feeding more data to the machine to learn and by doing better feature engineering (including dimension reduction and Principal Component Analysis) to improve the predictive capability of the models.