

# PGP-Data Science & Engineering

## Capstone Project - Final Report

# Project Title

## ***Predicting the behaviour of customer towards loyalty program***



## Capstone Project Team

# Ankit Mathur

# Rinesh Rajan

# Apurv Slunke

**Vardharajan G P**

# Raghul Sharan R

## Capstone Project Mentor

## Mr. Ankush Bansal

**Date: 10th November 2020**

**Batch: DSE Feb 2020 Pune - Group 5**

**Group: Group 2**

# Acknowledgements

The development of this project was a result of collaborative efforts. I would like to express my appreciation to Mr. Ankush Bansal for his advice, support and encouragement. Furthermore, I would like to thank Great Lakes for conceding the opportunity to develop my capstone project. I owe a special thanks to my team members for their cooperation and assistance with this project. A heartfelt appreciation goes out to my family and friends who inspired me to try harder and do better.

# Abstract

Nowadays, the implementation of new and innovative promotional campaigns to retain as well as gain new customers has become an important aspect for many organisations. This makes organisation to create more profitable marketing strategies, by customizing offers where they can understand various segmented customer groups as each customer is an individual in themselves and they show different patterns or behaviour. One of such strategies is issuing discount shopping coupons which has become one of the popular ways to boost the sales of an organisation.

The groundwork of this project is set upon the retail market, in which is the creation of loyalty programs which typically includes giving away free products, coupons, discount codes, early access to new products and services, and so on, to the most loyal customers, which in turn allows the collection and treatment of demographic, customer behaviour and transactional data, which allows the implementation and creation of new marketing strategies accordingly. Using this information, customized and segmented campaigns are created for different customer groups based on the business strategies and directives. In this project the focus is set upon promotional coupon direct marketing campaigns, sent to customers through various channels. The implementation of this sort of campaign urges the need to predict the percentage of customer adhesion to these campaigns, since this knowledge provides strategic decision support regarding stock management, strategic outline and sales forecast. However, predicting the percentage of redeemed coupons is the main objectives of this project which would be achieved by implementing various machine learning algorithms.

# Table of Contents

1.Introduction .....	1
1.1 Problem Statement .....	1
1.2 Findings and Expectations.....	2
2. Overview of Methodology Used.....	2
3. Problem Solving Approach .....	3
3.1 Exploratory Data Analysis .....	3
3.1.1 Descriptive Statistics .....	4
3.1.2 EDA - Data Visualization Techniques.....	9
3.2 Feature Engineering.....	19
3.3 Merging Techniques Used.....	21
3.4 Model Building Approach.....	23
4.Model Evaluation .....	24
5.Model Tuning.....	31
6.Data Visualizations .....	33
7.Limitations.....	40
8.Closing Reflections.....	41
9. References.....	42

# 1.Introduction

The fast evolution the retail industry and consumer behaviour have undergone in the last few years has brought some new challenges to retailers operating in competitive markets. Alongside the increase in competition came a more defiant customer- showcasing a different kind of behaviour than before- making more conscious and informed choices and being more demanding regarding the quality of the products and services available, thus becoming less loyal than one used to be. Therefore, the need to define new strategies and to set ambitious goals has grown, making retail companies more aware of the importance of analytics as an opportunity to add real value to both the business and the consumer, by providing a better understanding of them.

Hence in the past few years, loyalty programs have emerged as a rewards program offered by a company to customers who frequently make purchases, and which allow the collection of the customer's socio-demographic, behavioural and transactional data. This customer activity data will update the marketing teams' knowledge on customer behaviour and influence future decisions, by enabling the creation of more suitable and valuable offers to the consumer and providing better products and services.

By introducing various feature engineering techniques and machine learning algorithms we have tried to build a new Coupon Prediction Model. This model has the ability to deal with historical data and to provide near accurate predictions of the future coupons use of customers. In our project the problem in our hand is a probability prediction problem where the target variable is a binary class, and we would make the predictions by using various non-linear classification models of machine learning and evaluate the performance of the model through various evaluation metrics.

## 1.1 Problem Statement

An analytics company named XYZ regularly helps its merchants understand their data better by providing machine learning and analytics consulting. ABC is an established Brick & Mortar retailer that frequently conducts marketing campaigns for its diverse product range. As a merchant of XYZ, they have sought XYZ to assist them in their discount marketing process.

ABC's promotions are shared across various channels which includes email, notifications, etc. A number of these campaigns include coupon discounts that are

offered for a specific product/range of products. These campaigns include coupon discounts that are offered for a specific product/range of products.

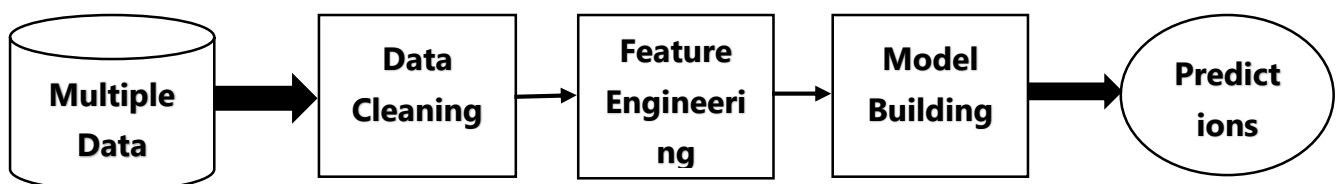
The retailer would like the ability to predict whether customers redeem the coupons received across channels, which will enable the retailer's marketing team to accurately design coupon construct, and develop more precise and targeted marketing strategies.

## 1.2 Findings and Expectations

The scope of this project is to build an adaptive machine learning model to identify the probability of obtaining a positive response to a promotional offer for a target group of customers; using the historical records of customers provided in different data sources. Throughout this process, the following issues will be addressed: What are the chances of stimulated customers making use of a promotional offer in the next marketing campaign, based on their different data provided about the customers and their interactions in previous marketing campaigns? What percentage of promotional coupons sent through various mode of communication like email, mobile apps etc. will effectively be redeemed by the customers? Does the implementation of our implemented machine learning model bring any kind of improvements to the ongoing processes i.e. by improving the accuracy of the existing model?

## 2. Overview of Methodology Used

For the development of the machine learning model to predict whether the customers will redeem the coupon or not we have followed the steps mentioned below in which each and every step is an important building block to build an effective machine learning classification model. Below mentioned is the overview of our machine learning pipeline which consists of various steps to build our model.



In the real world scenario, the data obtained by organisations for an analytics project usually comes from multiple data sources. Similarly, in this project we have data from multiple sources which we have integrated into one single data source keeping the train data as our base table and using different kind of merging techniques whilst

extracting the important features from each of the multiple data source provided. This has lead us to a more comprehensive single data source which will be used for analysis and model building. Moreover, prior to the merging process we have derived insights from each of these data sources using various techniques of EDA and Feature Engineering.

### **3. Problem Solving Approach**

This section describes all the necessary steps performed to build our machine learning model where all the necessary steps performed are explained in a detailed manner. We have started with exploring the data and deriving useful insights from all the multiple data sources available to us. After discovering the hidden patterns through various techniques of EDA and data visualization we have proceeded with the feature engineering part where new features are created using the existing ones. This project also requires data merging of different data sources into a single source which is done using various techniques of joining the data for instance merging two data-frames using left join. After obtaining a single comprehensive data source we have built our machine learning model. In the following sub-sections to come we have tried to explain every step of our machine learning pipeline along with the results and inferences obtained from each of those in a detailed manner.

#### **3.1 Exploratory Data Analysis**

Exploratory data analysis is a set of techniques applied on the data which helps us in finding useful insights from the data. EDA helps in discovering features that add to the overall picture that the data presents and provides us with a firm feature set that can be later used for building the machine learning model. In other words, it is a critical step performed before model building which enables us to discover hidden patterns and summarizes the main characteristics of the data.

In our project there are multiple data sources on which the data exploration is performed and below are some basic statistics obtained from the data along with the necessary visualizations. In the section follow data exploration is done using both statistical and visualization techniques.

### 3.1.1 Descriptive Statistics

In this section we have carried out the basic statistical analysis of the different data sources present in our project and presented the results in the form of a table.

#### 1.train.csv

##### train\_data

**Dimensions:** 78369 x 5

**Duplicates:** 0

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Valid	Missing
1	id [integer]	Mean (sd) : 64348 (37126.4) min < med < max: 1 < 64318 < 128595 IQR (CV) : 64317 (0.6)	78369 distinct values		78369 (100%)	0 (0%)
2	campaign_id [integer]	Mean (sd) : 14 (8) min < med < max: 1 < 13 < 30 IQR (CV) : 5 (0.6)	18 distinct values		78369 (100%)	0 (0%)
3	coupon_id [integer]	Mean (sd) : 566.4 (330) min < med < max: 1 < 597 < 1115 IQR (CV) : 577 (0.6)	866 distinct values		78369 (100%)	0 (0%)
4	customer_id [integer]	Mean (sd) : 787.5 (456.8) min < med < max: 1 < 781 < 1582 IQR (CV) : 791 (0.6)	1428 distinct values		78369 (100%)	0 (0%)
5	redemption_status [integer]	Min : 0 Mean : 0 Max : 1	0: 77640 ( 99.1%) 1: 729 ( 0.9%)		78369 (100%)	0 (0%)

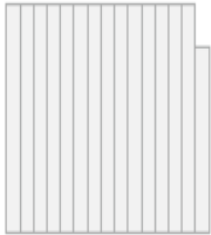
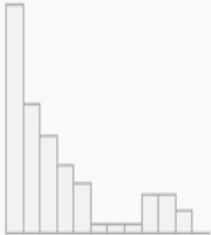
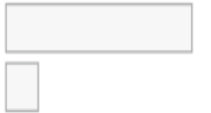



## 2.item\_data.csv

### item\_data

Dimensions: 74066 x 4

Duplicates: 0


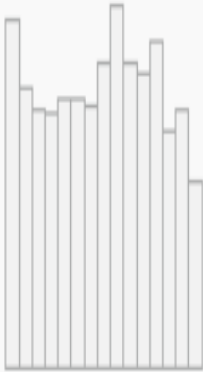
No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Valid	Missing
1	item_id [integer]	Mean (sd) : 37033.5 (21381.2) min < med < max: 1 < 37033.5 < 74066 IQR (CV) : 37032.5 (0.6)	74066 distinct values (Integer sequence)		74066 (100%)	0 (0%)
2	brand [integer]	Mean (sd) : 1485.6 (1537.4) min < med < max: 1 < 978 < 5528 IQR (CV) : 1735 (1)	5528 distinct values		74066 (100%)	0 (0%)
3	brand_type [character]	1. Established 2. Local	62842 ( 84.9%) 11224 ( 15.2%)		74066 (100%)	0 (0%)
4	category [character]	1. Grocery 2. Pharmaceutical 3. Natural Products 4. Dairy, Juices & Snacks 5. Skin & Hair Care 6. Meat 7. Packaged Meat 8. Prepared Food 9. Bakery 10. Seafood [ 9 others ]	32448 ( 43.8%) 24471 ( 33.0%) 2533 ( 3.4%) 2425 ( 3.3%) 2244 ( 3.0%) 2080 ( 2.8%) 1966 ( 2.7%) 1880 ( 2.5%) 1679 ( 2.3%) 728 ( 1.0%) 1612 ( 2.2%)		74066 (100%)	0 (0%)

### 3.coupon\_item\_mapping.csv

## coupon\_item\_mapping

Dimensions: 92663 x 2

Duplicates: 0

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Valid	Missing
1	coupon_id [integer]	Mean (sd) : 156 (283) min < med < max: 1 < 30 < 1116 IQR (CV) : 20 (1.8)	1116 distinct values		92663 (100%)	0 (0%)
2	item_id [integer]	Mean (sd) : 36508.6 (21131.3) min < med < max: 1 < 37955 < 74061 IQR (CV) : 35936 (0.6)	36289 distinct values		92663 (100%)	0 (0%)

## 4.customer\_transaction.csv

### customer\_transaction

Dimensions: 1324566 x 7

Duplicates: 2916





No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Valid	Missing
1	date [character]	1. 2012-09-03 2. 2012-10-03 3. 2012-11-15 4. 2012-09-13 5. 2012-10-11 6. 2012-10-19 7. 2012-10-25 8. 2012-11-23 9. 2013-01-24 10. 2012-12-07 [ 539 others ]	4753 ( 0.4%) 4703 ( 0.4%) 4372 ( 0.3%) 4024 ( 0.3%) 3957 ( 0.3%) 3956 ( 0.3%) 3952 ( 0.3%) 3945 ( 0.3%) 3931 ( 0.3%) 3927 ( 0.3%) 1283046 ( 96.9%)		1324566 (100%)	0 (0%)
2	customer_id [integer]	Mean (sd) : 804 (457.3) min < med < max: 1 < 801 < 1582 IQR (CV) : 780 (0.6)	1582 distinct values		1324566 (100%)	0 (0%)
3	item_id [integer]	Mean (sd) : 29519 (17908.1) min < med < max: 1 < 26597 < 74066 IQR (CV) : 27721.8 (0.6)	74063 distinct values		1324566 (100%)	0 (0%)
4	quantity [integer]	Mean (sd) : 130.7 (1311.5) min < med < max: 1 < 1 < 89638 IQR (CV) : 0 (10)	9252 distinct values		1324566 (100%)	0 (0%)
5	selling_price [numeric]	Mean (sd) : 114.6 (152.9) min < med < max: 0.4 < 78 < 17809.6 IQR (CV) : 75.2 (1.3)	4923 distinct values		1324566 (100%)	0 (0%)
6	other_discount [numeric]	Mean (sd) : -17.8 (37.9) min < med < max: -3120.3 < -1.8 < 0 IQR (CV) : 23.1 (-2.1)	1418 distinct values		1324566 (100%)	0 (0%)
7	coupon_discount [numeric]	Mean (sd) : -0.6 (7.1) min < med < max: -1992.2 < 0 < 0 IQR (CV) : 0 (-11.9)	232 distinct values		1324566 (100%)	0 (0%)

## 5.campaign\_data

### campaign\_data

Dimensions: 28 x 4

Duplicates: 0

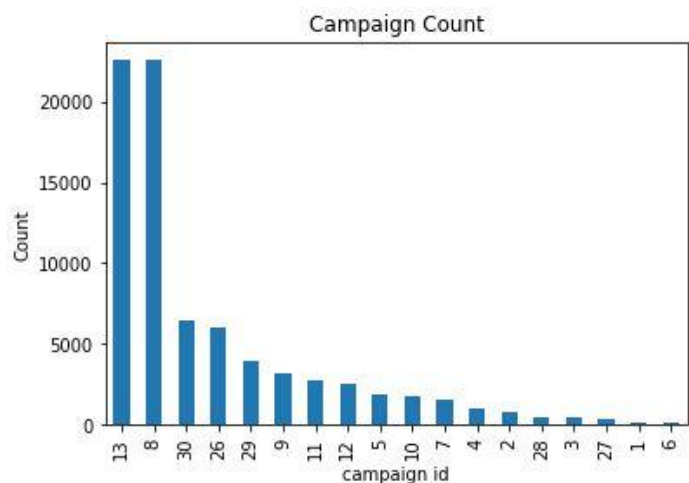
No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Valid	Missing
1	campaign_id [integer]	Mean (sd) : 15.6 (9.1) min < med < max: 1 < 16.5 < 30 IQR (CV) : 15.5 (0.6)	28 distinct values		28 (100%)	0 (0%)
2	campaign_type [character]	1. X 2. Y	6 (21.4%) 22 (78.6%)		28 (100%)	0 (0%)
3	start_date [character]	1. 16/09/13 2. 21/10/13 3. 22/04/13 4. 02/02/13 5. 07/01/13 6. 07/09/13 7. 08/04/13 8. 08/10/12 9. 08/10/13 10. 10/08/13 [ 15 others ]	2 ( 7.1%) 2 ( 7.1%) 2 ( 7.1%) 1 ( 3.6%) 1 ( 3.6%) 1 ( 3.6%) 1 ( 3.6%) 1 ( 3.6%) 1 ( 3.6%) 1 ( 3.6%) 15 (53.6%)		28 (100%)	0 (0%)
4	end_date [character]	1. 18/01/13 2. 18/10/13 3. 01/03/13 4. 04/01/13 5. 04/10/13 6. 05/04/13 7. 05/07/13 8. 07/06/13 9. 08/02/13 10. 08/03/13 [ 16 others ]	2 ( 7.1%) 2 ( 7.1%) 1 ( 3.6%) 1 ( 3.6%) 1 ( 3.6%) 1 ( 3.6%) 1 ( 3.6%) 1 ( 3.6%) 1 ( 3.6%) 1 ( 3.6%) 16 (57.1%)		28 (100%)	0 (0%)

### 3.1.2 EDA - Data Visualization Techniques

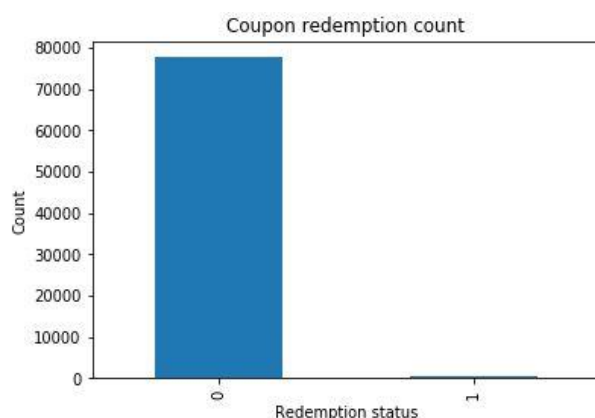
In this section we will perform univariate, bivariate and multivariate analysis of the features present in the dataset through various techniques of data visualization and derive insights from each of those.

#### Univariate Analysis & Findings

##### 1.train\_data.csv



The figure above shows that campaign id's 13,8 are the most frequently occurring campaigns while 1 and 6 are the least frequently occurring.

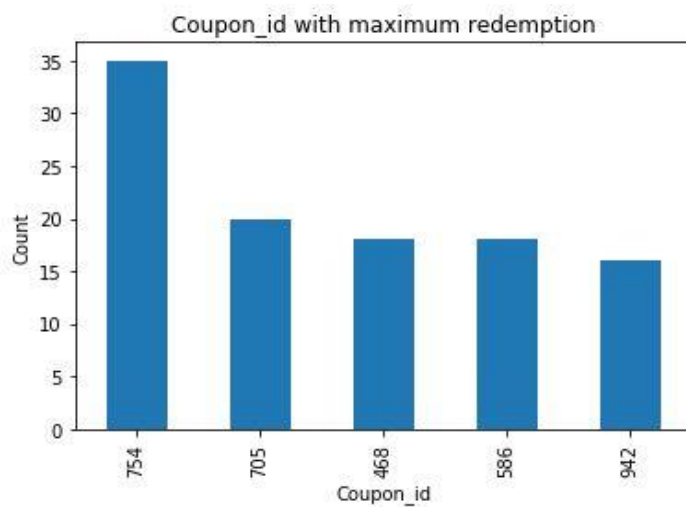


The figure above is the target variable redemption status where 0 means that the customer has not redeemed the coupon while 1 means the customer has redeemed the coupon. The target variable is imbalanced which shows that only 0.93% coupons are redeemed while 99.06% are not redeemed.

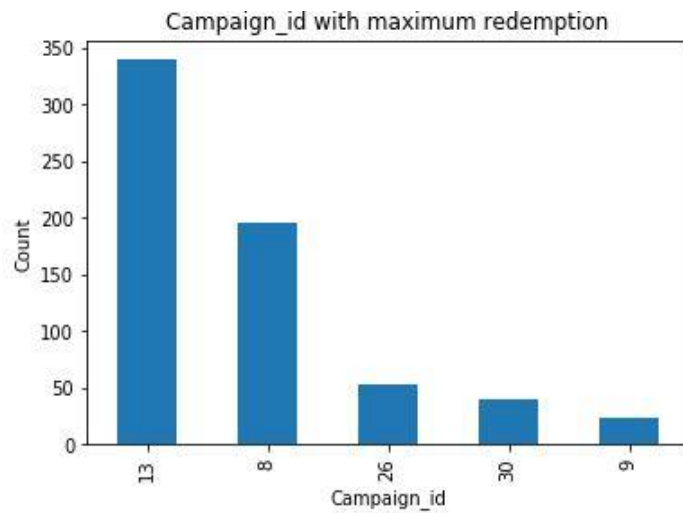
*The treatment of this imbalanced data is done using the SMOTE method at the later stages.*



*The figure above shows the top 5 customers and the count of each of them redeeming the coupon.*

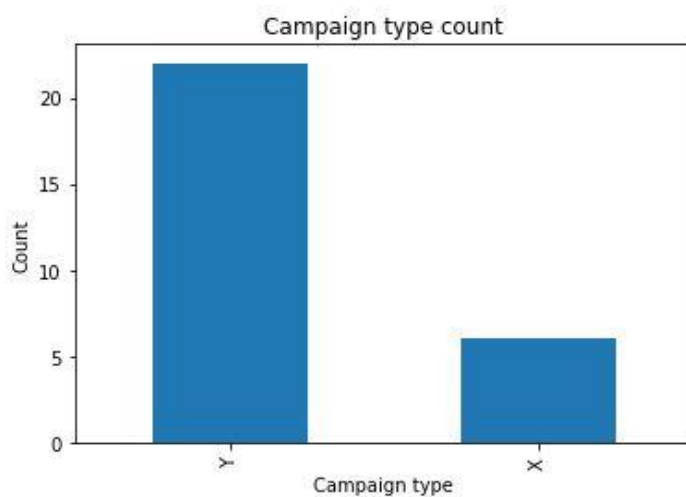


*The figure above shows the top 5 coupons/coupon\_id redeemed by the customers.*

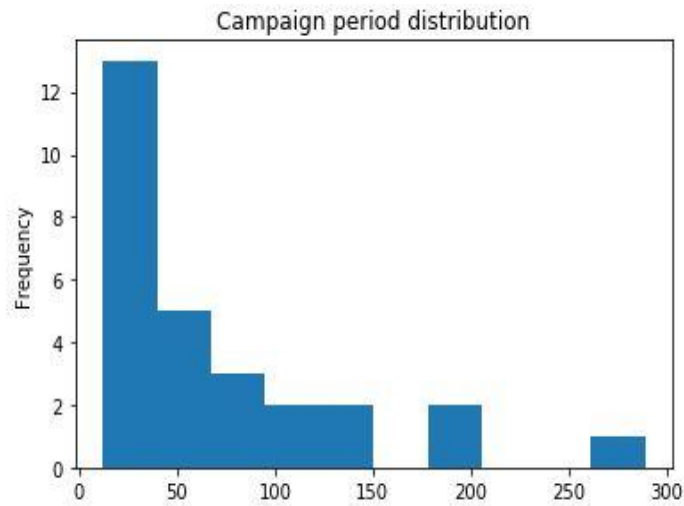


*The figure above shows the top 5 occurring campaigns with id 13 having count as 346.*

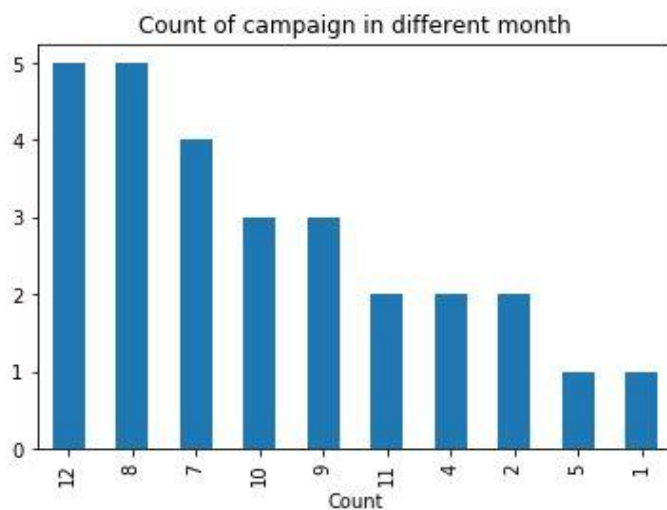
## 2.campaign\_data



*The figure above shows two different type of campaign X& Y where X is occurring 6 times and Y being more frequent occurring 22 times out of the total 28 campaigns.*



*The figure above shows the distribution if the campaign period in days which is right skewed or the data is not normal in this case.*

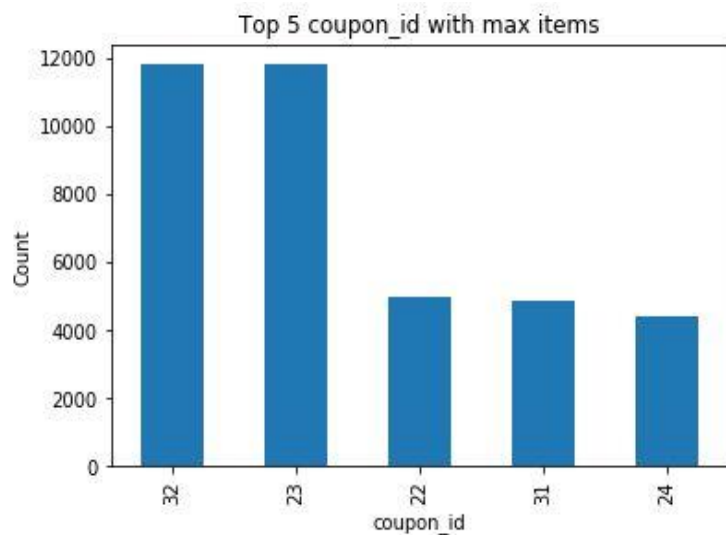


*The figure above shows the count of campaign occurring in a month with December & August months having the highest with count as 5 each.*

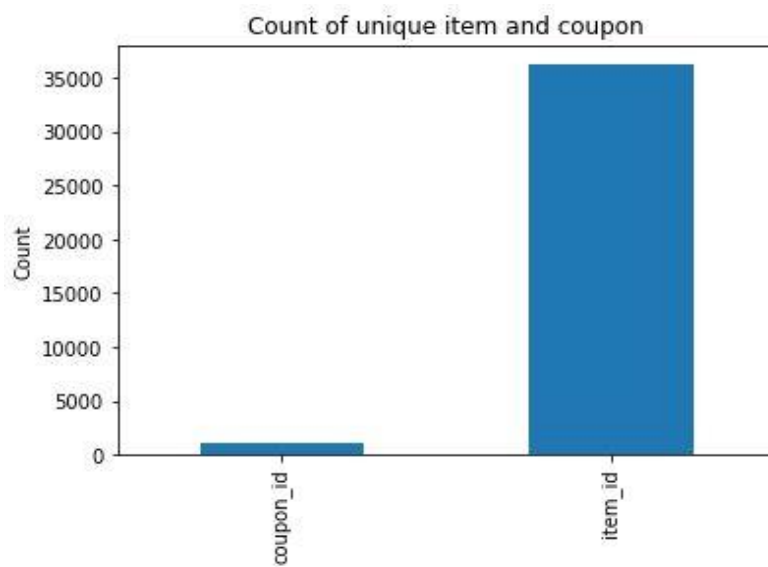
### **3.coupon\_item\_mapping.csv**

*The table contains mapping of coupon and items which are valid for discount under that coupon.*

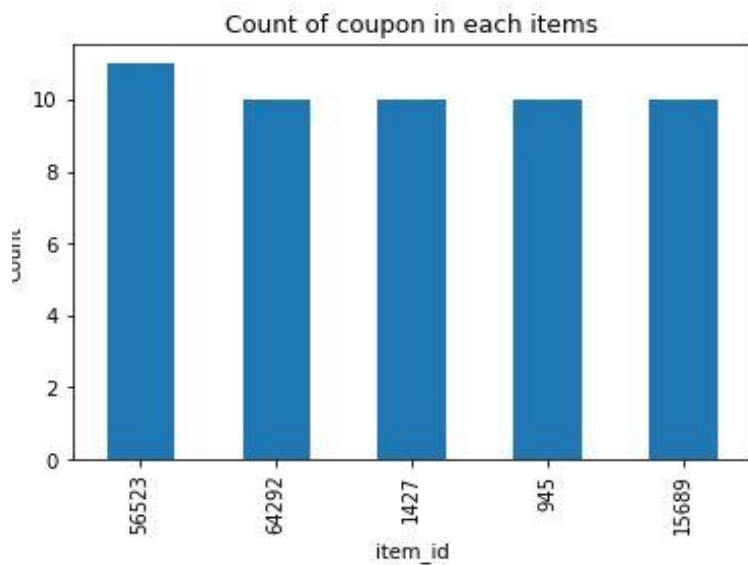




*The figure above shows top 5 coupon id's with coupon\_id 32 being most frequent with count of 11814 and coupon\_id 23 with count of 11813.*

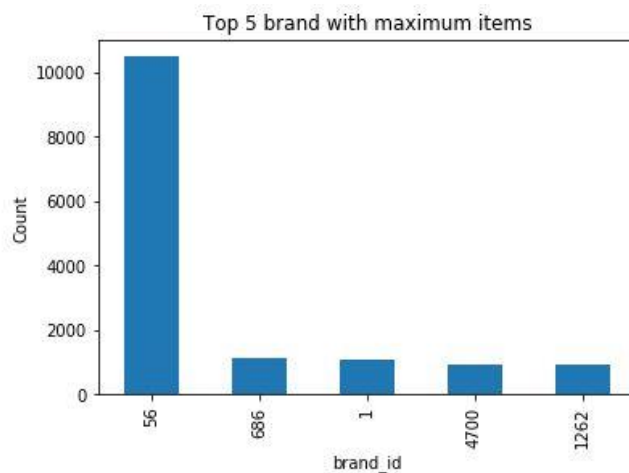


*The figure above shows count of unique items and coupon in the table with coupon\_id count as 1116 and item\_id count as 36289.*



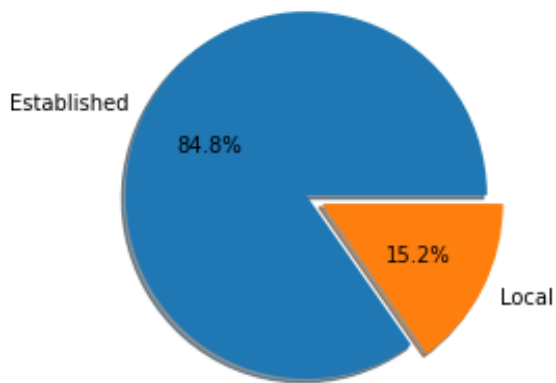
*The figure above shows the count of coupon in each items. It can be observed that item\_id 56523 is the highest.*

#### 4. item\_data.csv

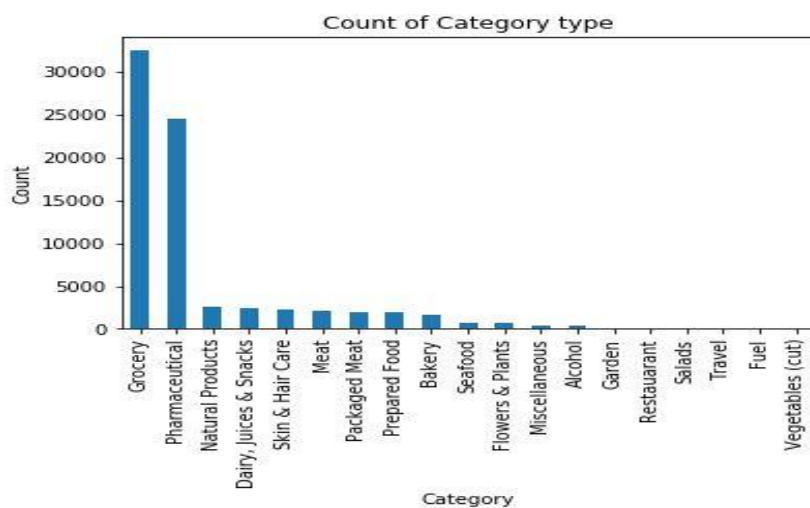


*The figure above shows the top 5 brands with maximum items with brand\_id 56 being the most frequent having count as 10480.*

Brand Type distribution



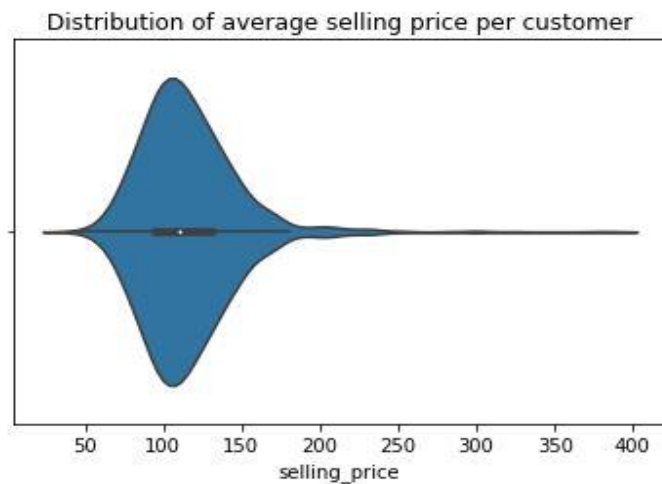
The figure above displays two types of brands established and local where 84.8% are established while 15.2% are local brands.



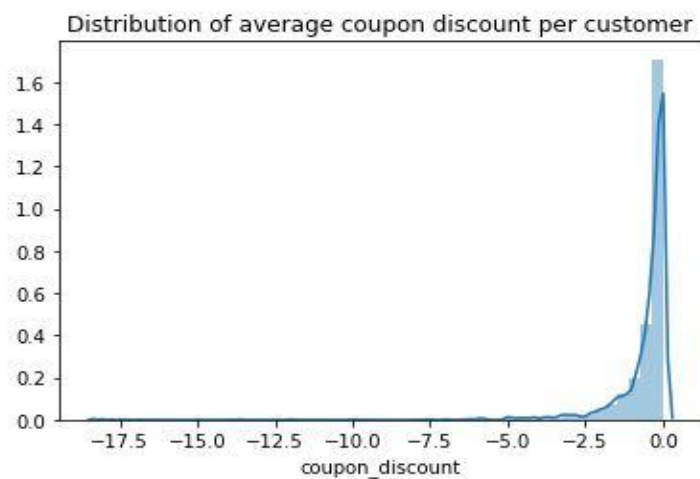
The figure above has 19 different categories with grocery being the most frequent having count as 32448.

## 5. customer\_transaction.csv

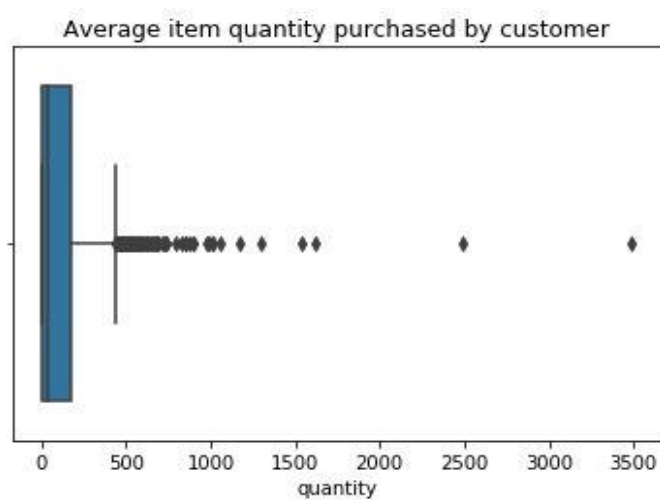
The customer transaction table below contains graphs of important features. It has transaction data of customers with total count of transactions approx. 13 lakhs for the duration of campaigns.



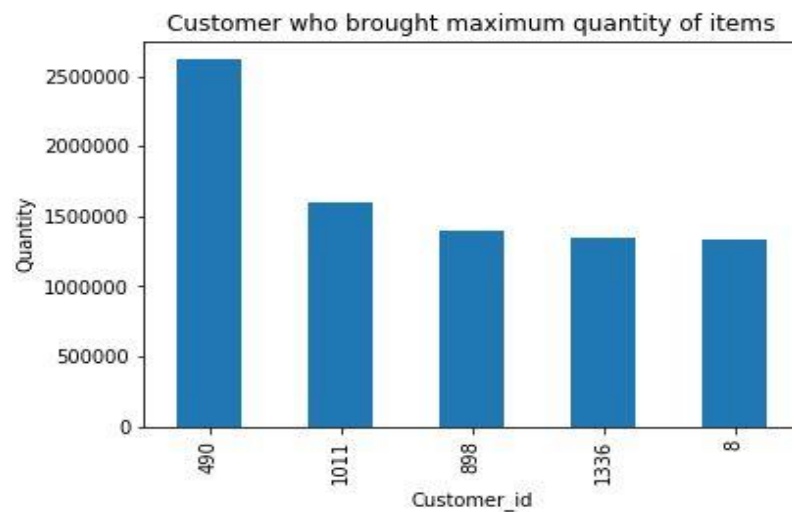
*The figure above shows the distribution of the average selling price per customer.*



*The figure above shows the distribution of the average coupon discount per customer.*

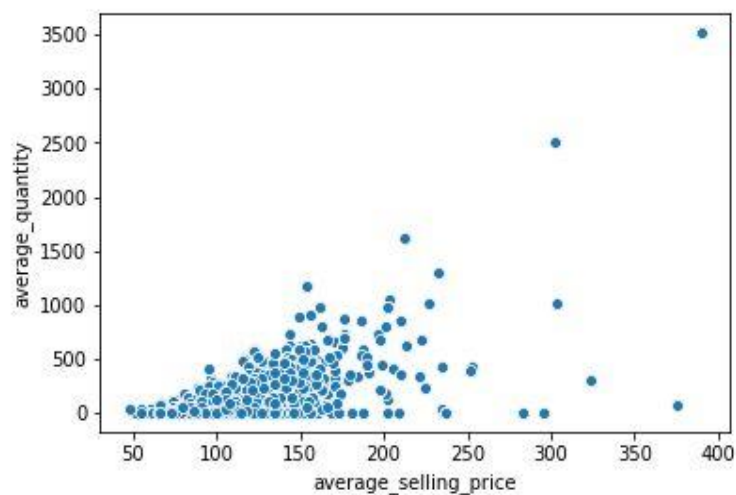


The figure above shows the boxplot displaying average item quantity purchased by a customer.

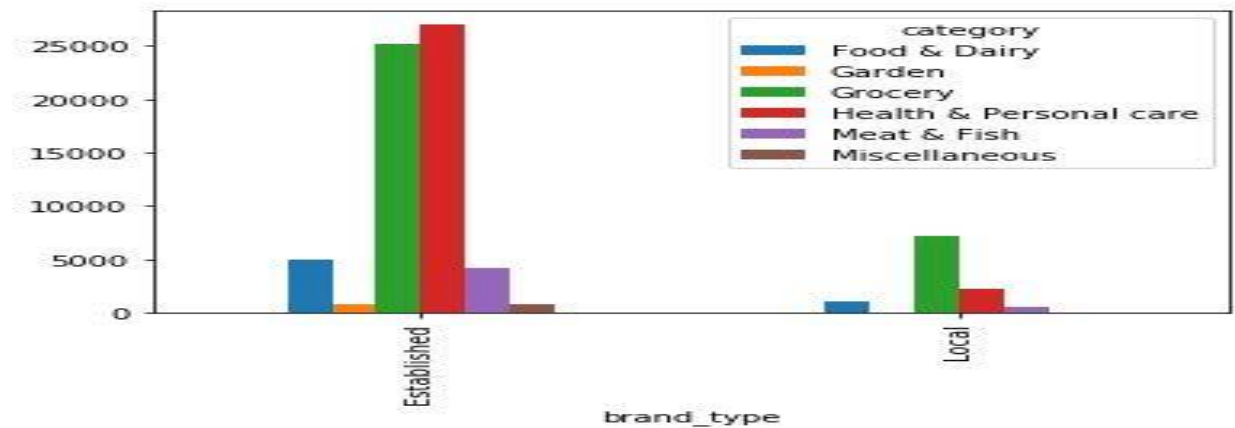


The figure above shows the customer/customer\_id who bought maximum quantity of items with customer\_id 490 being most frequent.

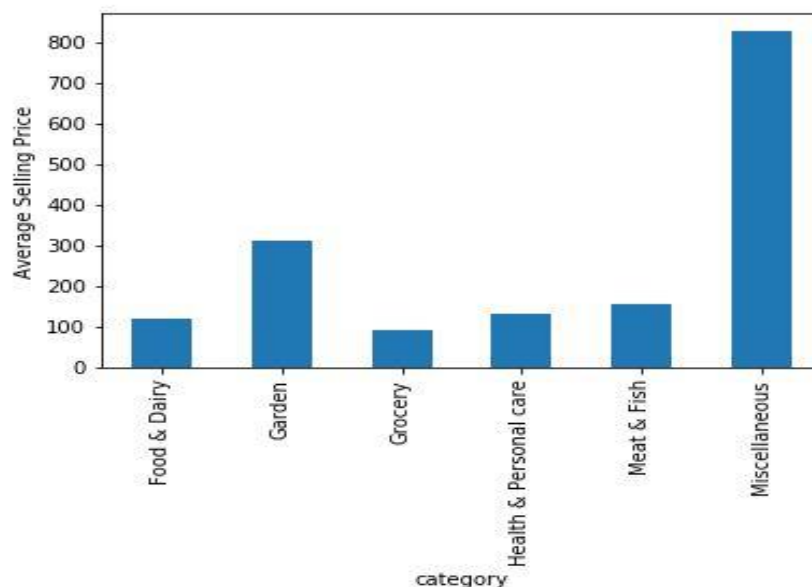
### **Bivariate Analysis**



The figure above shows the scatterplot of continuous variables average quantity vs average\_selling\_price having low degree of positive correlation.



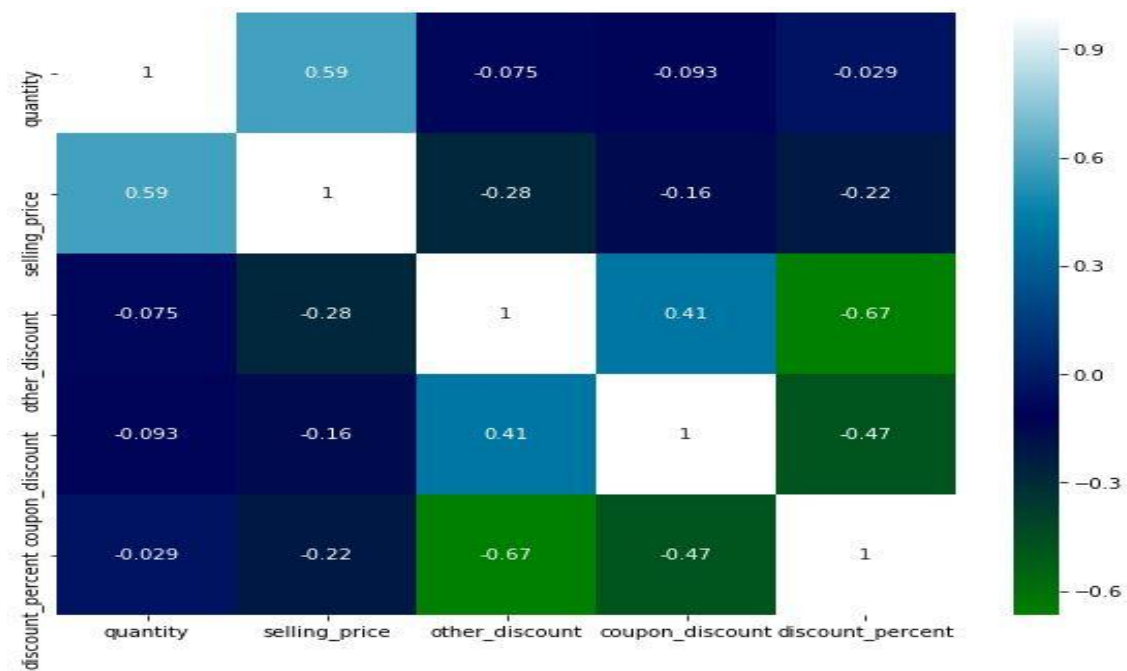
The figure above shows the types of items in each of the brands established and local with health and personal care items having maximum count in established and grocery in local brands.



The figure above shows average selling price for different kinds of items present under different categories in the items data.

### Multivariate Analysis

The figure below shows the heatmap of the customer transaction data with the correlation values of one feature with respect to the other.



After exploring the data and understanding the underlying stories that the data tells us through the help of various visualization graphs plotted above we then proceeded with the next step of our machine learning pipeline i.e. feature engineering.

### 3.2 Feature Engineering

Feature engineering is about creating new input features from your existing ones. These features can be used to improve the performance of machine learning algorithms. In our analysis we have implemented feature engineering techniques on few features in order to gain more insights from our data.

1. Using the campaign data we have created new features like campaign period, campaign year and campaign month. These three features are created from the existing features campaign start date and campaign end date which contains the start and end dates for a particular campaign.

campaign period = Difference between campaign start and end dates in days.

campaign year = Difference between campaign start and end dates in years.

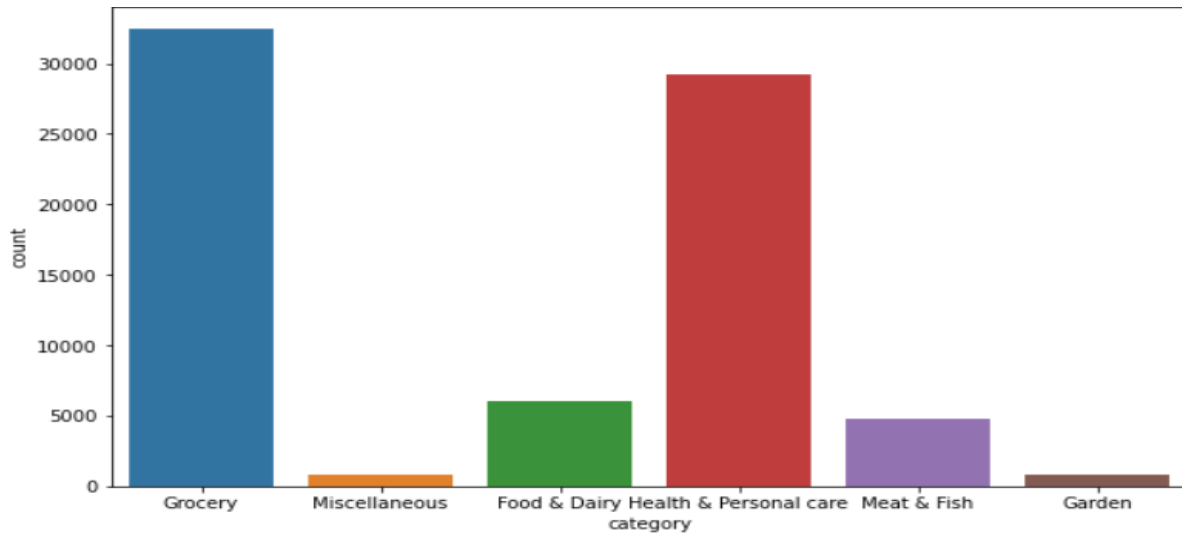
campaign months = Difference between campaign start and end dates in months.

2. Using the campaign data new feature named quarter has been created in which the years 2012 and 2013 are divided into quarters.

- January, February, and March (Q1)
- April, May, and June (Q2)

- July, August, and September (Q3)
- October, November, and December (Q4)

3. In retail industry the products are broken down into discrete groups of similar or related products known as product categories. The item data in our project has 19 different items available that are generally used for basic household needs. Using the same concept, we have clubbed similar items and grouped them under a particular category. The diagram below illustrates the implementation.



4. Using the customer transaction data we have created discount percent feature which is the percentage of total discount that the customer has received per transaction.

5. After merging coupon\_item\_mapping and item data, we have created 2 new features brand\_unique and coupon\_item\_count. The brand\_unique is the count of unique brand for each coupon and coupon\_item is the total number of items per coupon.

6. Using aggregated transaction data we have created features like customer buying frequency, total\_item\_customer\_purchased and count of item purchased for different category. The buying frequency is the number of times customer have visited the store. The total\_item\_customer\_purchased is the total items purchased by customer in the transaction data. We have created six features which describes the total number of items purchased per category.

7. Using aggregated transaction data we have created features like no\_of\_customer\_visits, total\_item\_customer\_purchased and count of item purchased for different category. The no\_of\_customer\_visits is the number of times customer have visited the store. The total\_item\_customer\_purchased is the total items purchased by



customer in the transaction data. We have created six features which describes the total number of items purchased by customer per category before the campaign starts.

8. Using aggregated transaction data we have created new feature `no_of_coupon_discount` which describes the number of coupon redeemed by customer before the start of a new campaign.

### 3.3 Merging Techniques Used

The data for this problem statement had multiple data sources which had to be logically merged based on various techniques of merging. This required exploring and understanding different data sources before proceeding with the merging part. The final single source of data obtained was then analysed and it was used for building our machine learning model. Before mentioning the steps performed for merging we have described the types of relationships and joins used to create our final single data source which will be used in the later stages for model building.

#### *Understanding Types of Relationships*

There are different types of table relationships possible:

Relationship	Description
one-to-one	Both tables can have only one record on either side of the relationship. Each primary key value relates to only one (or no) record in the related table
one-to-many	The parent table (or primary key table) contains only one record that relates to none, one, or many records in the child table
many-to-many	Each record in both tables can relate to any number of records (or no records) in the other table.

The following depicts the type of table relationships.

one-to-many relationship tables

Tables	Key
campaign_data and train data	campaign_id
item_data and coupon_item_mapping	item_id
item_data and customer_transaction_data	item_id

many-to-many relationship tables

Tables	Key
train and customer_transaction_data	customer_id
coupon_item_mapping and train data	coupon_id
coupon_item_mapping and customer_transaction_data	item_id

### Merging Technique Used

Separate merging process will be applied for one-to-many and many-to-many relationships.

- For one-to-many, simple merge of both tables will provide combined features
- for many-to-many, aggregation of columns such as mean, count, min, max etc need to be performed on the table that will be joined.

### Merging Steps

- Creating train\_campaign table with simple merge of train data and campaign data using campaign\_id
- Creating item\_coupon\_map table with simple merge of item data and coupon\_item\_mapping using item\_id
- Creating dummy columns for item\_coupon\_map categorical columns and then aggregating data with respect to coupon\_id. Columns are aggregated using different functions like sum, count and nunique
- Merging item\_coupon\_map table with train\_campaign on coupon\_id and creating new table train\_camp\_item\_coupon\_map
- Aggregating customer transaction data with respect to customer\_id using function like sum and count
- Creating final table by merging aggregated customer transaction data with train\_camp\_item\_coupon\_map table on customer\_id

### **3.4 Model Building Approach**

According to the problem definition we have to build a classification model as our target variable is a binary class categorical variable. After applying various techniques of EDA, Feature Engineering and merging multiple data sources into a single data source we would build our machine learning model and check and compare the performance of the different models by evaluating it through various evaluation metrics used in the classification model. Post the model building phase we would try to further improve the efficiency of our models through hyper parameter tuning.

## 4. Model Evaluation

Model evaluation aims to estimate the generalization accuracy of a model on future unseen data. In this section we have built different machine learning model and evaluated the performance of each of them through various evaluation metrics. Moving forward we will start by introducing the models and evaluation metrics that we have used for building our model along with the results obtained for each of them.

The different evaluation metrics to check the performance of our machine learning model are mentioned in the points below. The evaluation metrics discussed below are the ones keeping in mind that the problem we are trying to solve is a classification problem.

### 1. Cross Validation

Cross-validation is a technique that involves partitioning the original observation dataset into a training set, used to train the model, and an independent set used to evaluate the analysis.

The most common cross-validation technique is k-fold cross-validation, where the original dataset is partitioned into k equal size subsamples, called folds. The k is a user-specified number, usually with 5 or 10 as its preferred value. This is repeated k times, such that each time, one of the k subsets is used as the test set/validation set and the other k-1 subsets are put together to form a training set. The error estimation is averaged over all k trials to get the total effectiveness of our model.

### 2. Classification Accuracy

Accuracy is a common evaluation metric for classification problems. It's the number of correct predictions made as a ratio of all predictions made.

### 3. Confusion Matrix

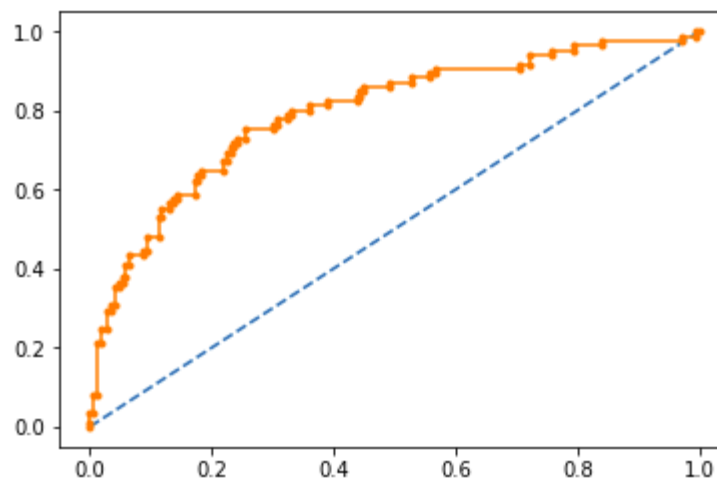
A confusion matrix provides a more detailed breakdown of correct and incorrect classifications for each class. It has various components which can be best understood from the diagram below.

		Assigned class		
		Positive	Negative	
Real class	Positive	TP	FN	Recall $\frac{TP}{TP+FN}$
	Negative	FP	TN	False positive rate $\frac{FP}{TN+FP}$
		Precision $\frac{TP}{TP+FP}$	Specificity $\frac{TN}{TN+FN}$	Accuracy $\frac{TP+TN}{TP+TN+FP+FN}$

#### 4.Area Under Curve AUC

Area under ROC Curve is a performance metric for measuring the ability of a binary classifier to discriminate between positive and negative classes. For example AUC is relatively close to 1 and greater than 0.5 in this case.

AUC - Test Set: 79.55%



#### 5. ROC Curve

ROC stands for receiver operating characteristic and the graph is plotted against TPR and FPR for various threshold values. As TPR increases FPR also increases. Also, ROC-AUC is just the area under the curve, the higher its numerical value the better the predictions.

$$\text{True Positive Rate (TPR)} = \text{RECALL} = \frac{TP}{TP+FN}$$

$$\text{False Positive Rate (FPR)} = 1 - \text{Specificity} = \frac{FP}{TN+FP}$$

## 6. F-Measure/ F1 Score

F-measure (also F-score) is a measure of a test's accuracy that considers both the precision and the recall of the test to compute the score. Precision is the number of correct positive results divided by the total predicted positive observations. Recall, on the other hand, is the number of correct positive results divided by the number of all relevant samples. Basically, it is the harmonic mean of precision and recall.

$$\frac{2}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

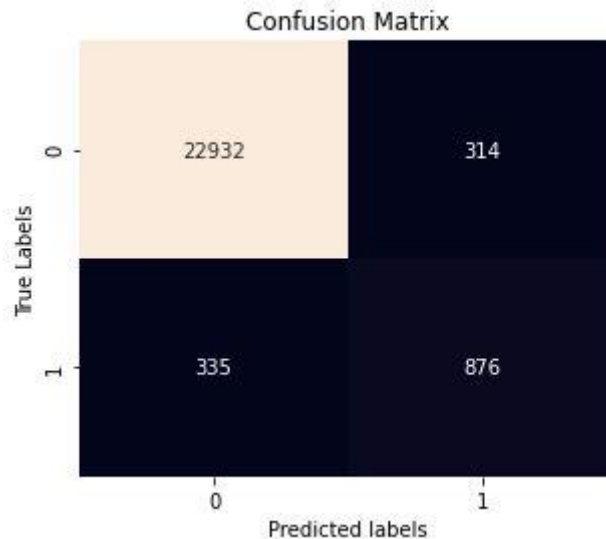
We have evaluated our model using aforementioned evaluation metrics and the results for different models are mentioned below.

### 1. Decision Tree

Decision tree learning is a supervised machine learning technique for inducing a decision tree from training data. A decision tree (also referred to as a classification tree or a reduction tree) is a predictive model which is a mapping from observations about an item to conclusions about its target value. In the tree structures, leaves represent classifications (also referred to as labels), non-leaf nodes are features, and branches represent conjunctions of features that lead to the classifications.

Below are the results obtained from the decision tree model.

```
Train Accuracy 0.99975466573206
Test Accuracy = 0.973463630044568
ROC_AUC = 0.8565242269644892
Precision = 0.7361344537815127
Recall = 0.7233691164327003      Sensitivity 0.7233691164327003
F1 Score = 0.7296959600166598    Specificity 0.9864922997504947
```

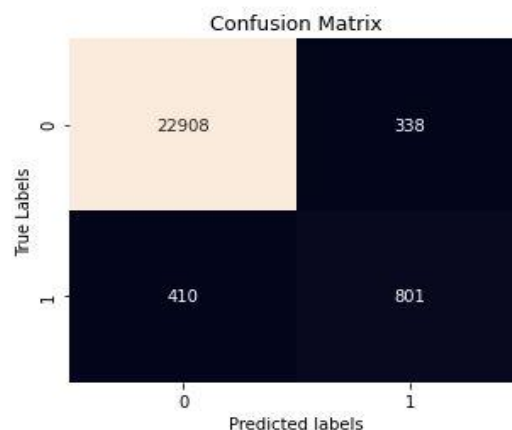


## 2.K-Nearest Neighbour

The k-nearest neighbours (KNN) algorithm is a simple, easy-to-implement supervised machine learning algorithm that can be used to solve both classification and regression problems. K-nearest neighbours (KNN) algorithm uses 'feature similarity' to predict the values of new data points which further means that the new data point will be assigned a value based on how closely it matches the points in the training set.

Below are the results obtained from KNN Classifier Model.

Train Accuracy 0.9817401209147464	
Test Accuracy = 0.9694157092039088	
ROC_AUC = 0.9620788581369282	
Precision = 0.7032484635645303	Sensitivity 0.6614368290668868
Recall = 0.6614368290668868	Specificity 0.9854598640626344
F1 Score = 0.6817021276595745	

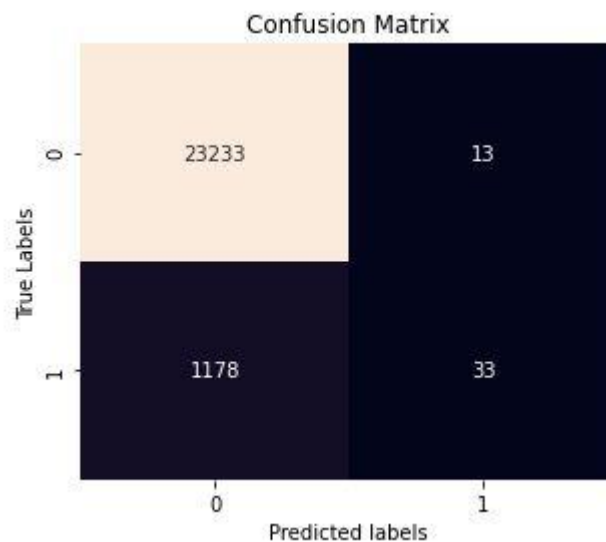


### 3.Support Vector Machine

Support Vector Machine or SVM's are supervised learning models with associated learning algorithms that analyse data used for classification and regression analysis. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible.

Below are the results obtained from KNN Classifier Model.

```
Train Accuracy 0.9540874441426443
Test Accuracy = 0.9513022856441918
ROC_AUC = 0.8011099003350016
Precision = 0.717391304347826
Recall = 0.027250206440957887
F1 Score = 0.05250596658711217
Sensitivity 0.027250206440957887
Specificity 0.999440764002409
```



### 4.Random Forest

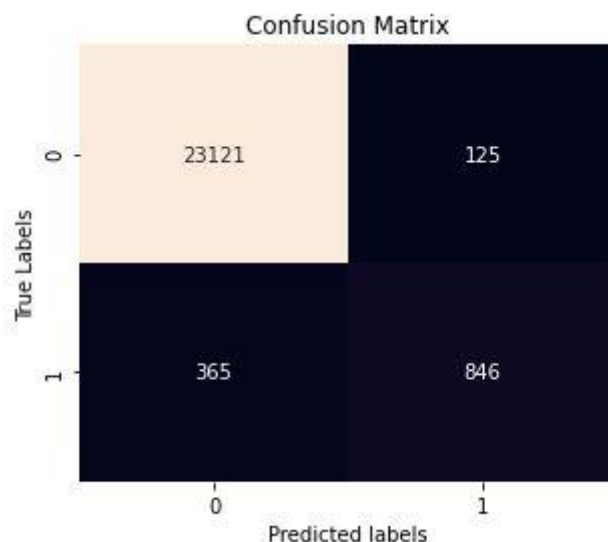
Random Forest is a classification algorithm consisting of many decisions trees. It uses bagging and feature randomness when building each individual tree to try to create an uncorrelated forest of trees whose prediction by committee is more accurate than that of any individual tree.

Below are the results obtained from Random Forest Model.



Train Accuracy 0.99975466573206  
 Test Accuracy = 0.9799648362432024  
 ROC\_AUC = 0.9824185587490506  
 Precision = 0.8712667353244078  
 Recall = 0.6985962014863749  
 F1 Score = 0.7754353803849678

Sensitivity 0.6985962014863749  
 Specificity 0.9946227307923944



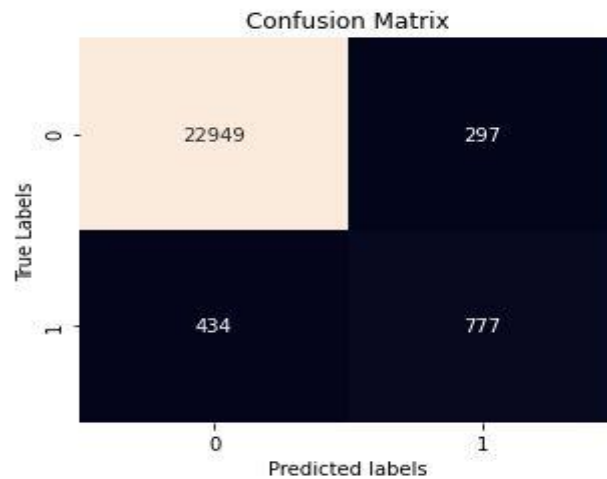
## 5. Bagging KNN

Bagging is a method of generating multiple versions of a predictor or a classifier and then using those to get an aggregated classifier. We have performed evaluation of Bagging K-nearest neighbour classifiers model. We have investigated varying softening methods of aggregation would yield better results than just sum and vote.

Below are the results obtained from Bagging KNN model.

Train Accuracy 0.9822658372031894  
 Test Accuracy = 0.9701108067220019  
 ROC\_AUC = 0.9742313266933575  
 Precision = 0.723463687150838  
 Recall = 0.6416184971098265  
 F1 Score = 0.6800875273522975

Sensitivity 0.6416184971098265  
 Specificity 0.987223608362729



## 6. AdaBoost

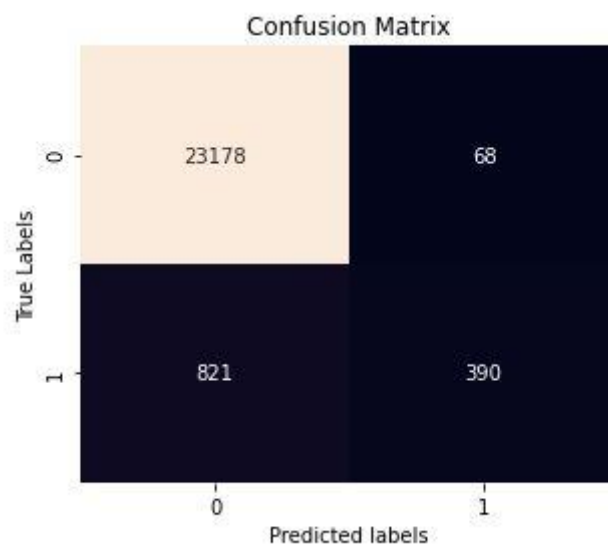
AdaBoost is an iterative ensemble method. AdaBoost classifier builds a strong classifier by combining multiple poorly performing classifiers so that you will get high accuracy strong classifier.

Below are the results obtained from Bagging AdaBoost model.

```

Train Accuracy 0.9661964426531149
Test Accuracy = 0.9636504886126671
ROC_AUC = 0.945920443910402
Precision = 0.851528384279476
Recall = 0.3220478943022296
F1 Score = 0.46734571599760344
Sensitivity 0.3220478943022296
Specificity 0.9970747655510626

```

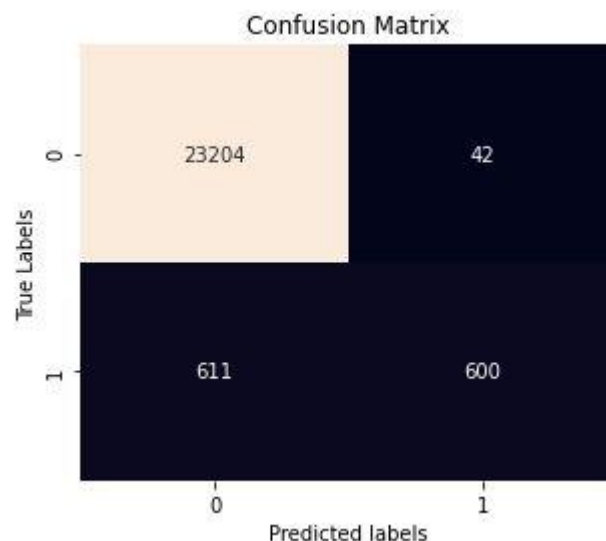


## 7.Gradient Boost

A Gradient Boosting Machine or GBM combines the predictions from multiple decision trees to generate the final predictions. Keep in mind that all the weak learners in a gradient boosting machine are decision trees. The key point here is that the nodes in every decision tree take a different subset of features for selecting the best split.

Below are the results obtained from Gradient Boost Model.

```
Train Accuracy 0.9760799088758433
Test Accuracy = 0.9733000776873697
ROC_AUC = 0.9699551765758445
Precision = 0.9345794392523364
Recall = 0.495458298926507
F1 Score = 0.6475984889368592
Sensitivity 0.495458298926507
Specificity 0.9981932375462446
```



**Note:** For building all our machine learning model we have used sklearn library and also used the SMOTE method for treating imbalanced data.

## 5.Model Tuning

Tuning is the process of maximizing a model's performance without overfitting or creating too high of a variance. In machine learning, this is accomplished by selecting appropriate hyperparameters. In machine learning, a hyperparameter is a parameter whose value is used to control the learning process. By contrast, the values of other

parameters (typically node weights) are derived via training. Hyperparameters can be classified as model hyperparameters, that cannot be inferred while fitting the machine to the training set because they refer to the model selection task, or algorithm hyperparameters, that in principle have no influence on the performance of the model but affect the speed and quality of the learning process. Different model training algorithms require different hyperparameters.

**Note:** As hyper parameter tuning requires huge computational-power we have performed tuning on the top selected models which have given us good accuracy scores according to the evaluation metrics.

### Model performance before HyperParameter Tuning

	Train Accuracy	Test Accuracy	ROC AUC	Precision	Recall	F1 score
<b>Decision Tree</b>	0.999755	0.973464	0.856524	0.736134	0.723369	0.729696
<b>KNN</b>	0.981740	0.969416	0.962079	0.703248	0.661437	0.681702
<b>SVM</b>	0.954087	0.951302	0.801110	0.717391	0.027250	0.052506
<b>Random Forest</b>	0.999755	0.979965	0.982419	0.871267	0.698596	0.775435
<b>KNN Bagging</b>	0.982266	0.970111	0.974231	0.723464	0.641618	0.680088
<b>AdaBoost</b>	0.966196	0.963650	0.945920	0.851528	0.322048	0.467346
<b>Gradient Boosting</b>	0.976080	0.973300	0.969955	0.934579	0.495458	0.647598

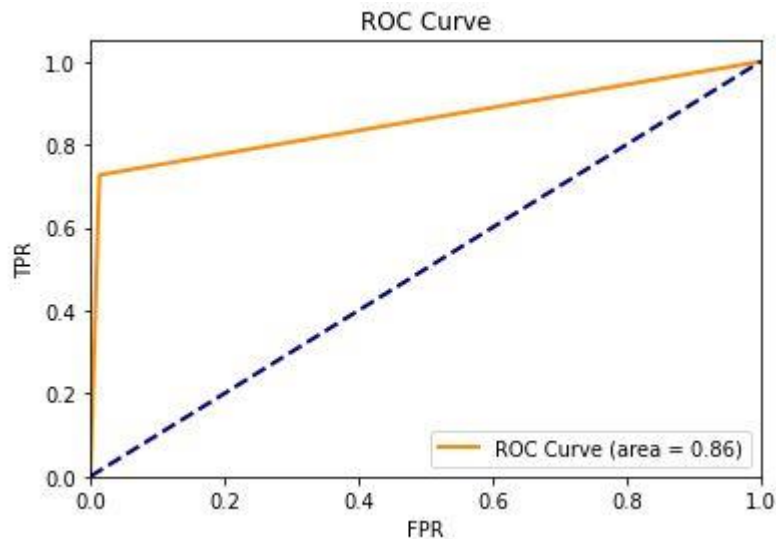
### Model performance after HyperParameter Tuning

	Train Accuracy	Test Accuracy	ROC AUC	Precision	Recall	F1 score
<b>Decision Tree</b>	0.978656	0.970847	0.921652	0.836486	0.511148	0.634546
<b>KNN</b>	0.999755	0.969743	0.977143	0.811096	0.507019	0.623984
<b>Random Forest</b>	0.999755	0.979801	0.982420	0.869969	0.696119	0.773394
<b>KNN Bagging</b>	0.982686	0.970315	0.976244	0.720655	0.654005	0.685714
<b>AdaBoost</b>	0.965969	0.963610	0.945821	0.855876	0.318745	0.464501
<b>Gradient Boosting</b>	0.964795	0.962015	0.952652	0.907514	0.259290	0.403340

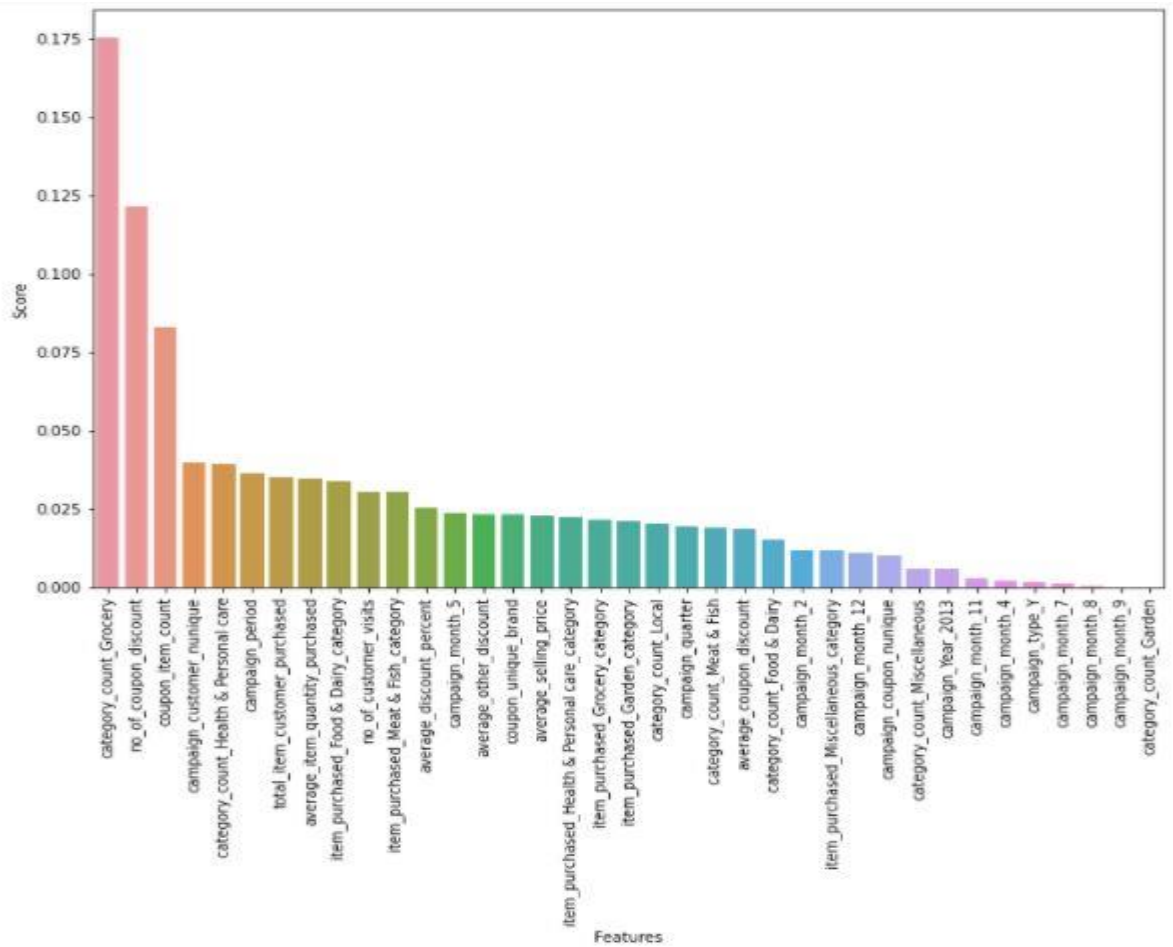
## 6.Data Visualizations

Visualizations helps us find out the hidden details of a complex dataset and makes interpretation of the machine learning model built easier. It helps us enhance the understanding of the business problem in hand. In this section we have used various techniques of data visualization to interpret the results obtained from our non-linear machine learning classification model and their results are summarized below.

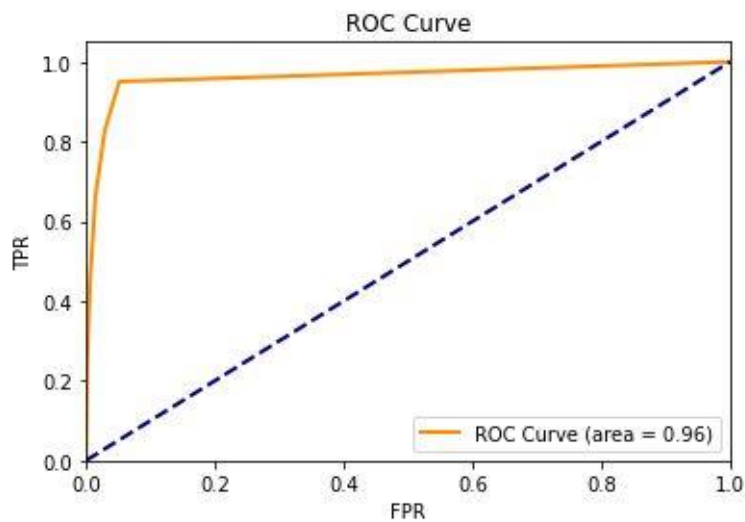
### 1. Decision Tree Model



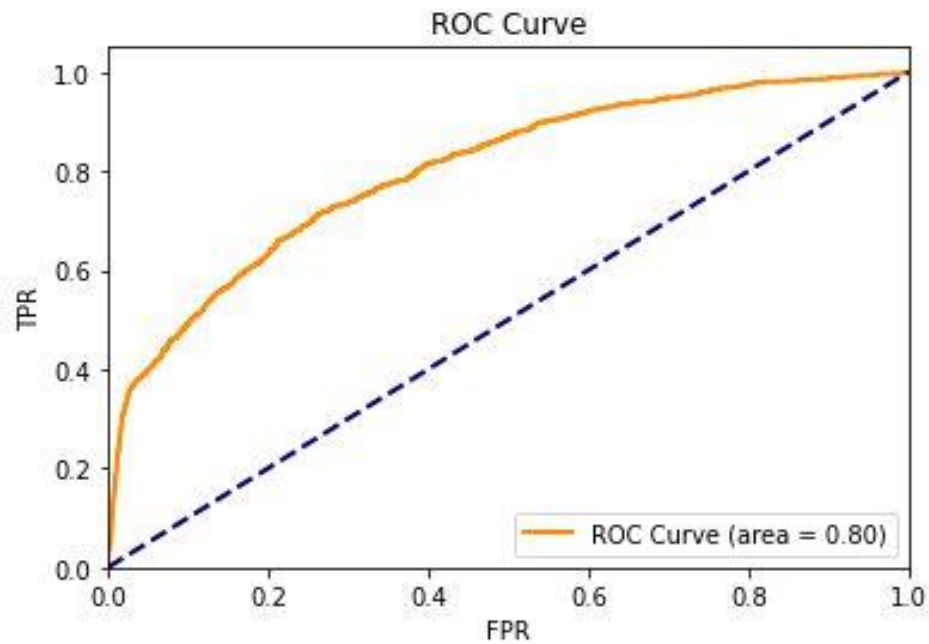
	precision	recall	f1-score	support
0	0.99	0.99	0.99	23246
1	0.74	0.72	0.73	1211
accuracy			0.97	24457
macro avg	0.86	0.85	0.86	24457
weighted avg	0.97	0.97	0.97	24457



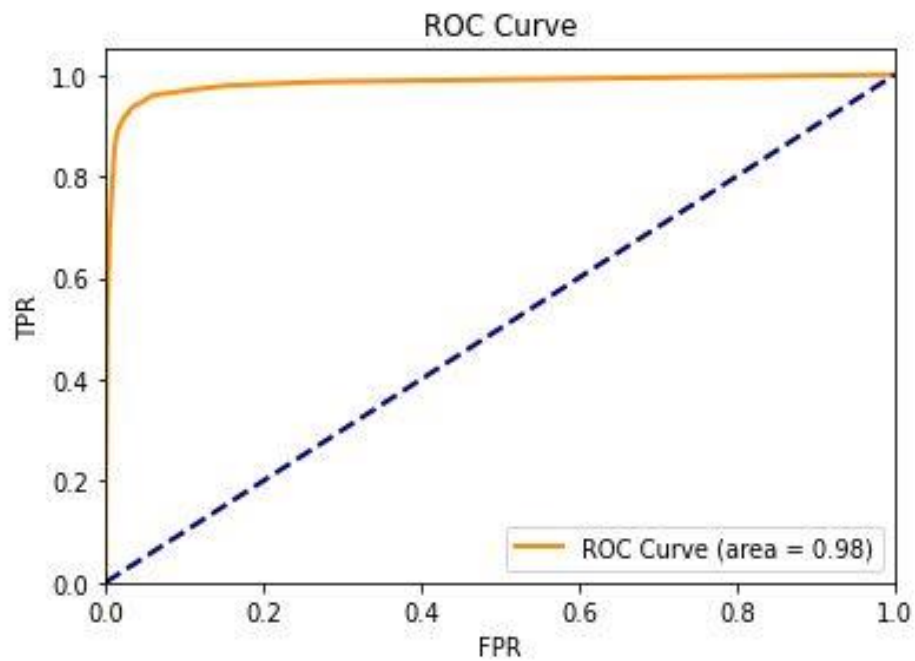
## 2. KNN Model



### 3. SVM Model

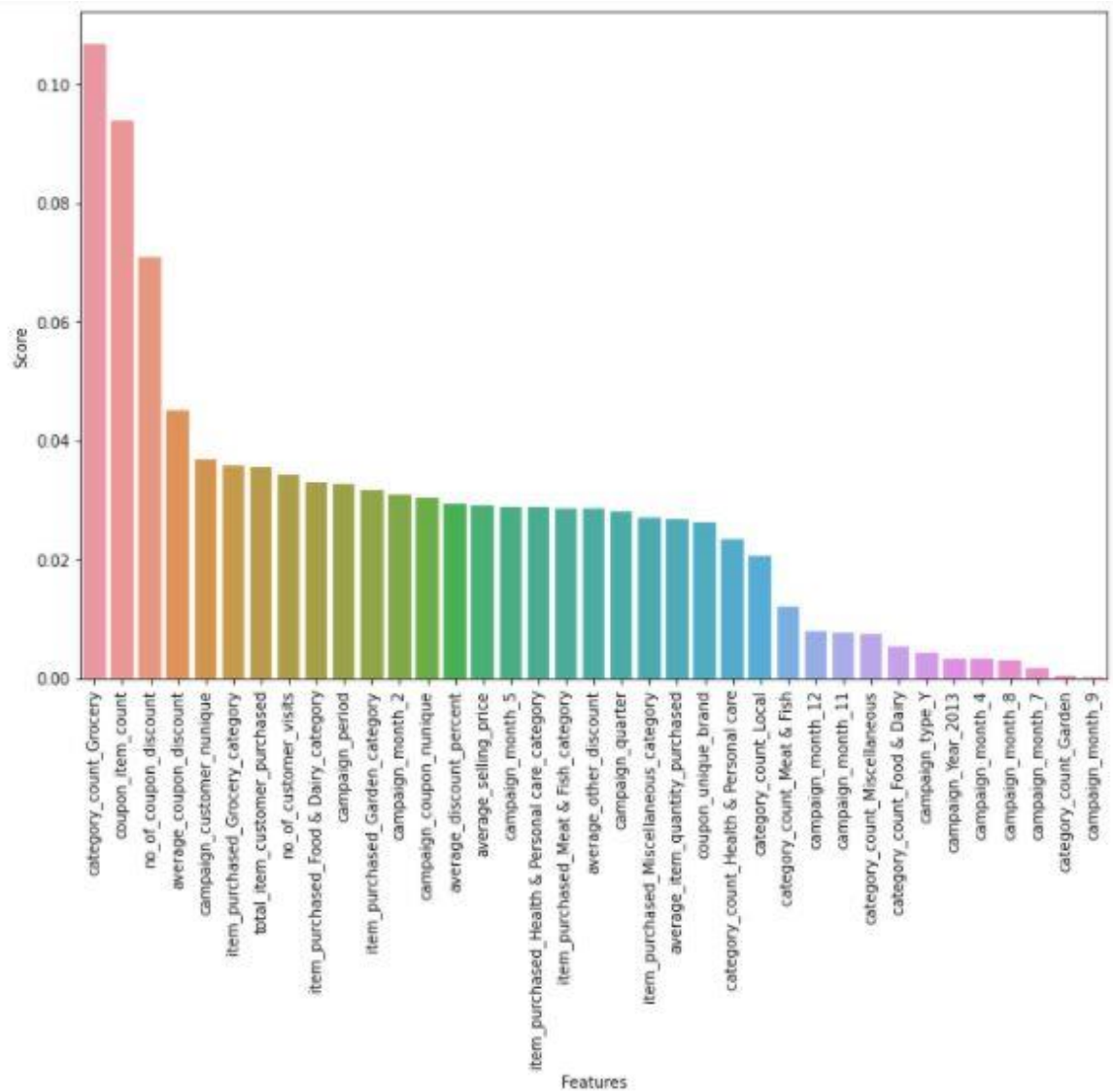


### 4. Random Forest Model



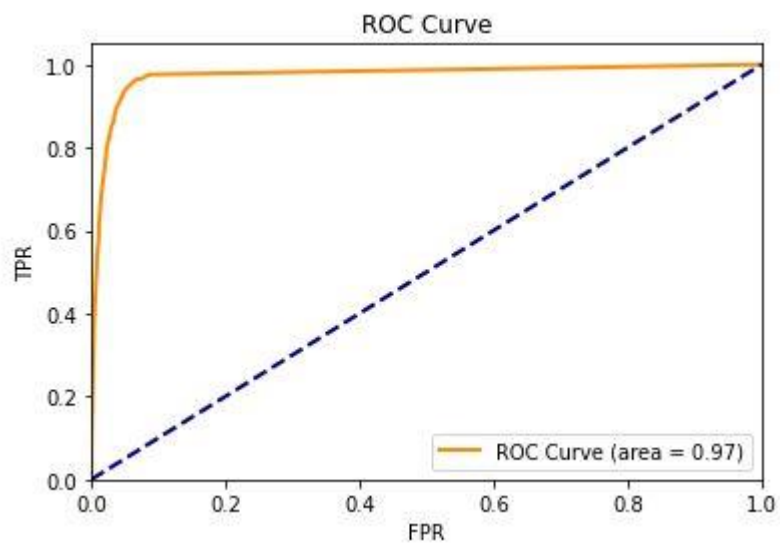


	precision	recall	f1-score	support
0	0.98	0.99	0.99	23246
1	0.87	0.70	0.78	1211
accuracy			0.98	24457
macro avg	0.93	0.85	0.88	24457
weighted avg	0.98	0.98	0.98	24457

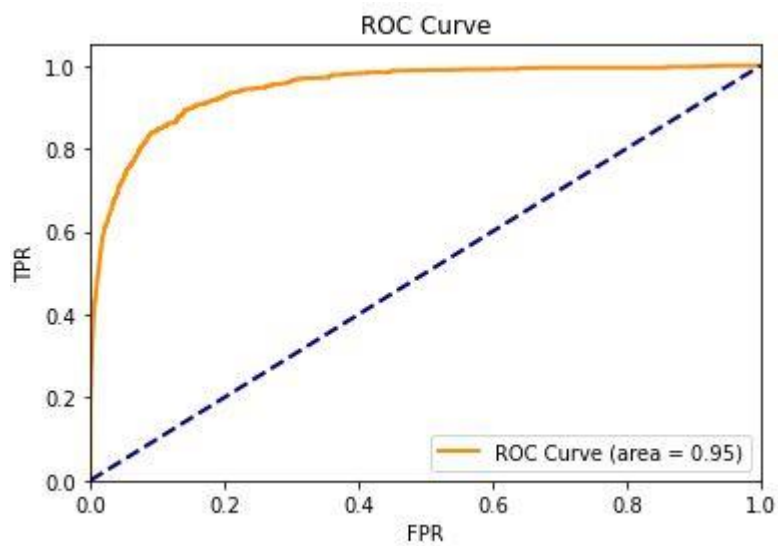


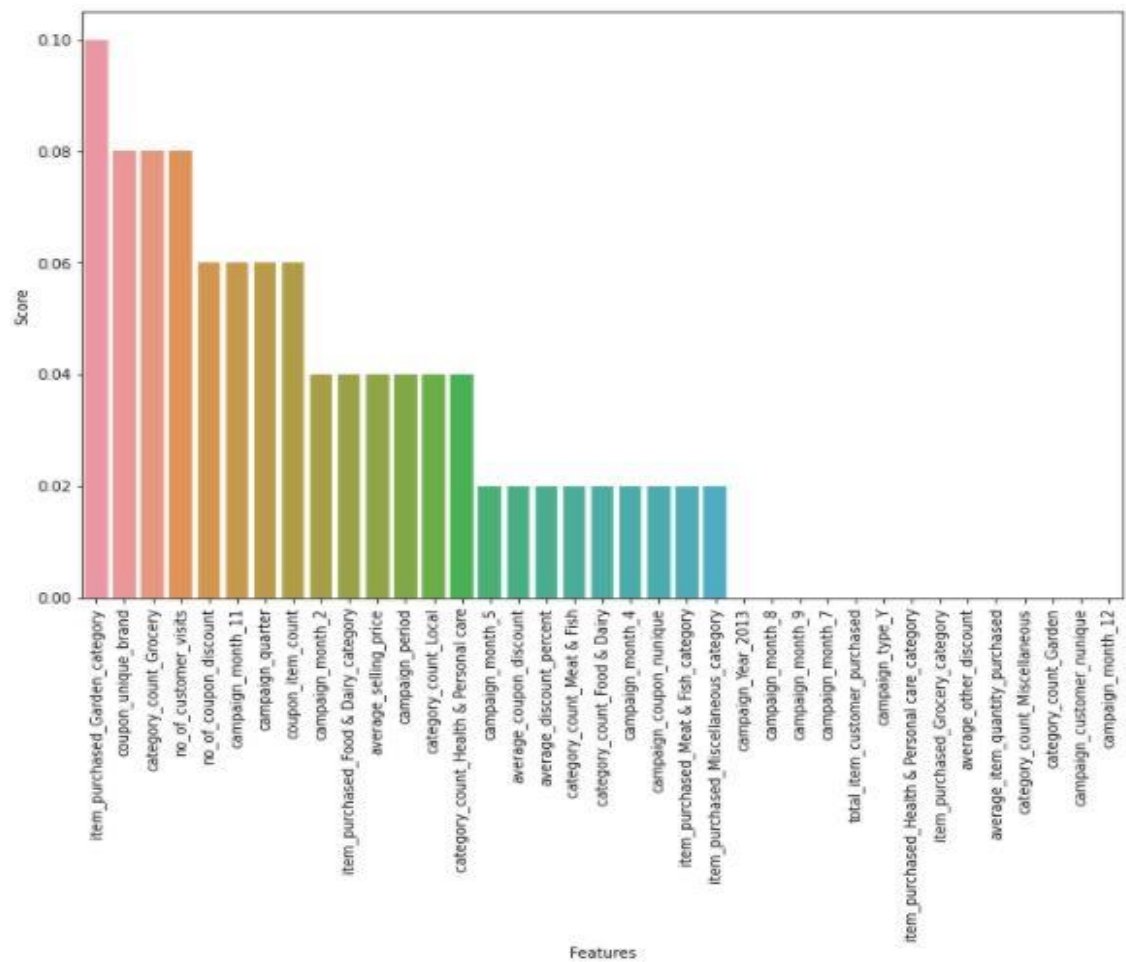


## 5. KNN Bagging Model

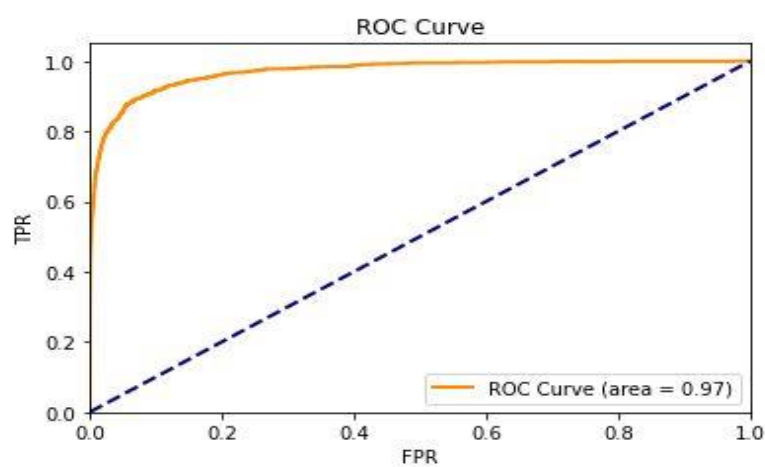


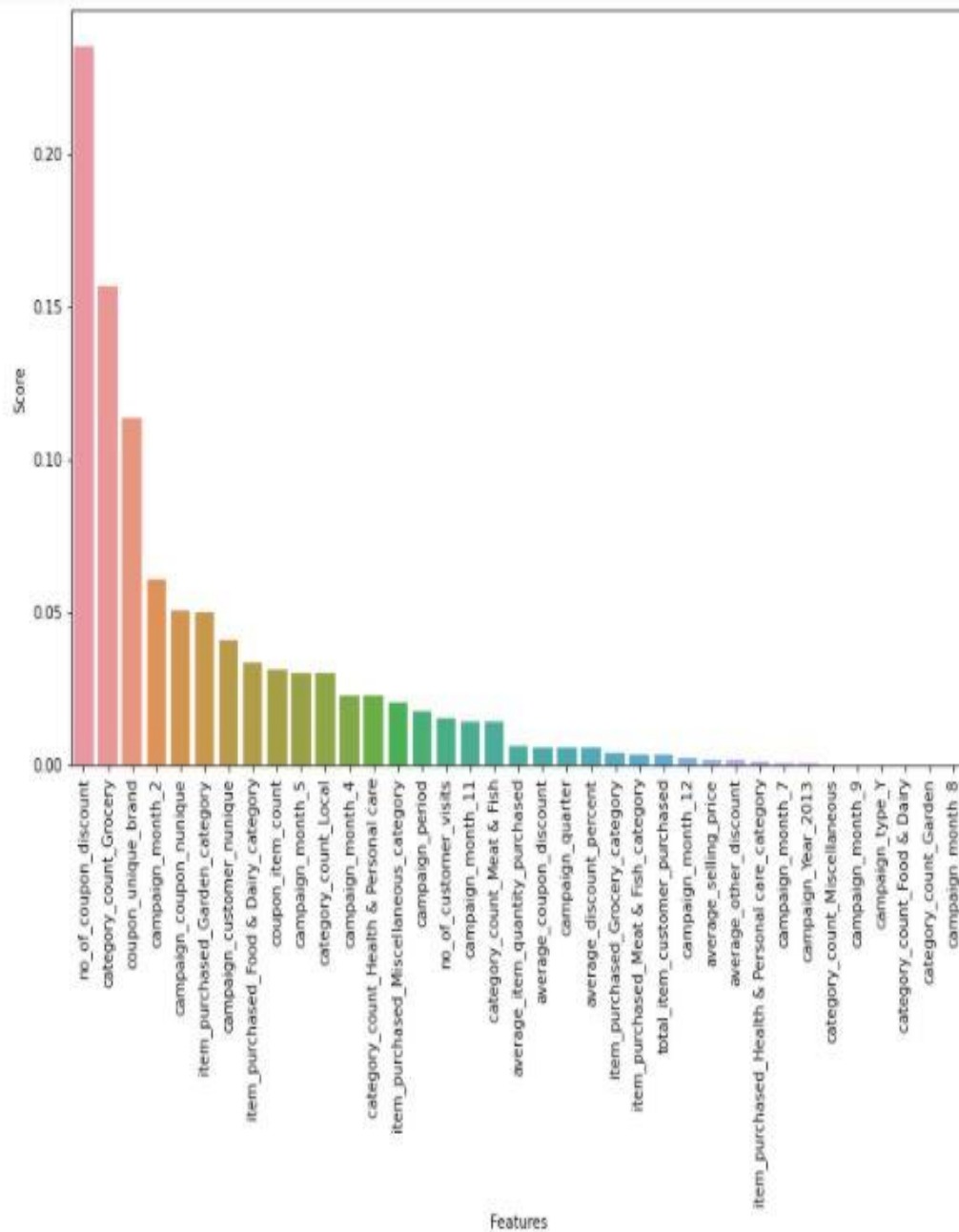
## 6. AdaBoost Model



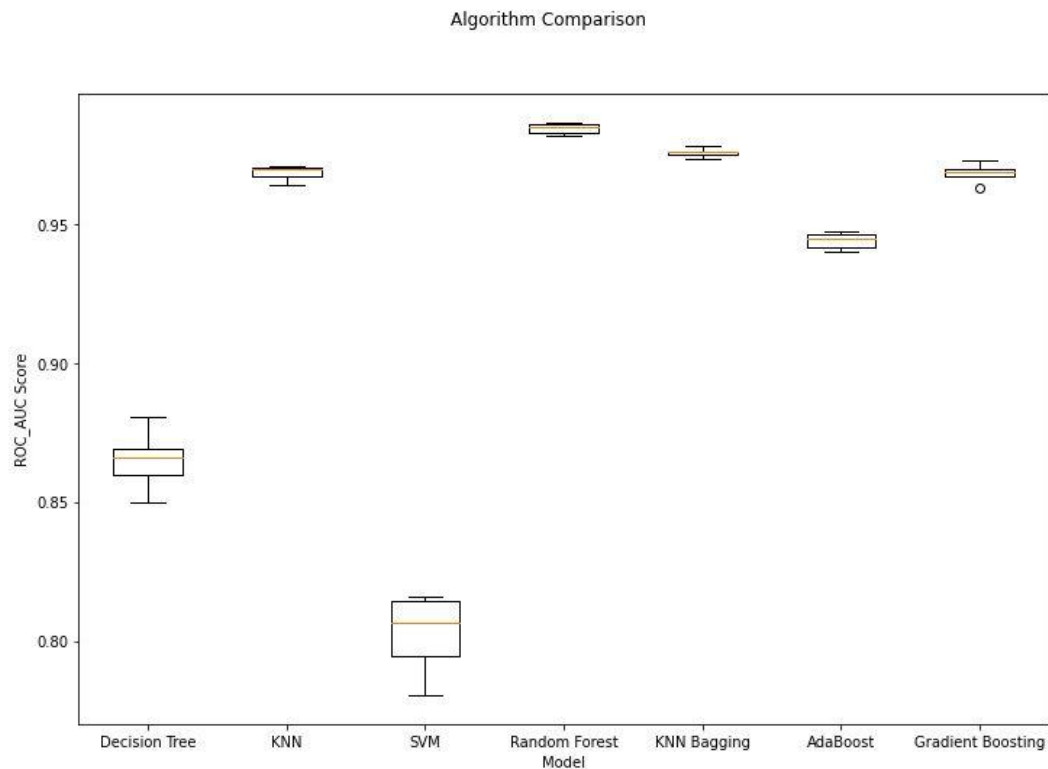


## 7. Gradient Boosting Model





From the results and inferences obtained after hyper parameter tuning and after building different machine learning models and comparing their results through evaluation metrics we have found that the performance for Random Forest model would be the best if we select it as our final model. Below is the boxplot comparing the ROC-AUC values of different machine learning models.



## 7.Limitations

Even though we have obtained decent accuracy for the model built and it can be considered that the business goals were achieved, we do have some limitations and future grounds which are discussed in the points below.

The model that we have built to predict whether the customer would redeem the coupon or not is not an adaptive machine learning model which basically means that it might not be able to accurately predict the outcome whether a new customer coming in the future would redeem the coupon or not. For the machine learning model to be an adaptive one it requires the application and knowledge of advanced concepts of deep learning along with some domain knowledge of the marketing/ retail industry.

Dealing with huge size of data would become difficult for this machine learning model and it may require the intervention of Big Data for tasks beginning from the data pre-processing to the model building part. Most organisations nowadays are moving towards Big-Data Technologies for enhancing the customer experience as big data frameworks have the ability to deal with datasets which is beyond the capability of other commonly used methods.

In addition, it might be relevant to test the inclusion of a channel of contact variable since, as mentioned before, the customers receive the coupons by different means of contact such as letter, newsletter and mobile app- and might respond differently to each of these as most customers nowadays are more easy to reach out using digital means of communications.

## **8.Closing Reflections**

The project we have worked on has given us the opportunity to learn, explore and tackle different kinds of challenges in each and every step of our machine learning model building phase for the business problem described in the aforementioned sections of this project. All our learnings, challenges, and the approach taken to solve them are described in the paragraph to follow.

One of the major reasons of selecting this problem statement was because of its importance in the retail industry. Organisations nowadays are carrying out promotional campaigns like in our case offering coupons to engage their customers and trying to make them buy more products which in result boost their sales. The data for this problem statement had multiple data sources which had to be logically merged based on various techniques of merging. This required the in-depth understanding of each of these 5 independent data sources before proceeding with the merging part. The final single source of data obtained was then analysed and it was used for building our machine learning model. This scenario of merging data from different independent data sources is something similar which happens in the industries nowadays and thus working on this project gave us a hands-on exposure or the glimpses of working on a live project which could be similar to this. This was the major learning outcome from our project.

Another important learning outcome was handling the imbalanced data. The target variable to be predicted was highly imbalanced and required necessary treatment. For resolving this issue, we used SMOTE techniques where the data was synthetically cooked for about 5%. This small change helped us in increasing the accuracy/efficiency of our model and hence we found no further need of increasing the percentage as it is not always desirable or in other words it is not considered a good practice to synthetically cook large percentage of data though the techniques of SMOTE varies from industry to industry and the problem in hand.

## 9. References

1. Introduction to Machine Learning Using Python - O'Reilly Media
2. <https://www.data-driven-investor.com/2019/07/11/machine-learning-applications-in-retail-6-real-world-examples-from-market-leaders/>
3. <https://en.wikipedia.org/>
4. <https://towardsdatascience.com/an-extensive-guide-to-exploratory-data-analysis-ddd99a03199e>
5. <https://www.kdnuggets.com/2017/11/interpreting-machine-learning-models-overview.html>
6. <https://machinelearningmastery.com/difference-between-a-parameter-and-a-hyperparameter/>
7. [https://en.wikipedia.org/wiki/Hyperparameter\\_\(machine\\_learning\)](https://en.wikipedia.org/wiki/Hyperparameter_(machine_learning))