MANIPAL INSTITUTE OF TECHNOLOGY

MANIPAL

(A constituent institution of MAHE, Manipal)

IDENTIFYING PROFITABLE CUSTOMERS THROUGH CUSTOMER SEGMENTATION FOR CROSS-SELLING CREDIT CARDS TO MOVIE CUSTOMERS

A Graduate Project Report submitted to Manipal Academy of Higher Education in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

in

Mechanical Engineering

by

ATUL VIRENDRA PODDAR

Under the guidance of

Guide Name : Prof. Vinayas

Designation: Assistant Professor

Department: Mechanical Engineering

Name of the institution: MIT

E mail id:

Guide Name: Harshit Pandey

Designation: Associate

Department: Data Science

Name of the Organisation: MathCo.

E mail id: harshit@themathcompany.com

August 2021

MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL
(A constituent institution of MAHE, Manipal)

Manipal
dd/mm/yyyy

CERTIFICATE

This is to certify that the project titled **IDENTIFYING PROFITABLE CUSTOMERS THROUGH CUSTOMER SEGMENTATION FOR CROSS-SELLING CREDIT CARDS TO MOVIE CUSTOMERS** is a record of the bonafide work done by **Atul Poddar (170909178)** submitted in partial fulfillment of the requirements for the award of the degree of **BACHELOR OF TECHNOLOGY** in **MECHANICAL ENGINEERING** of Manipal Institute of Technology, Manipal, Karnataka (A constituent college of MAHE, Manipal) during the year 2020 - 2021.

Professor Vinyas
Assistant Professor
Department of Mechanical and

Dr. Sathyashankara Sharma
Head of the Department
Department of Mechanical and

TheMathCompany
Building analytical capabilities for organizations

Awfis Space Solutions Pvt Ltd, 1st Floor, # 24, Sabari Complex,
Residency Road, Bengaluru, Karnataka India 560025
CIN - U74999KA2016PTC096027

21-May-2021

TO WHOM IT MAY CONCERN

This is to certify that **Atul Poddar** has successfully completed 4 (Four) months (From 18th Jan 2021 to 21st May 2021) internship programme at TheMathCompany Pvt Limited. During the period of the internship programme with us he was found punctual, hardworking, and inquisitive.

We wish you every success in life.

For TheMathCompany Pvt Ltd.

Ashish Thomas Sam
Head– People & Operations


TheMathCompany
Upskilling
— with —
Co.ach
By completing the Dumbledore Co.ach program
Atul Poddar
is officially recognized to be fully competent
in the analytics full stack and is certified a
FULL STACK ANALYTICS PROFESSIONAL
28-05-2021
Date
Sayandeb Banerjee
CEO & Co-founder, TheMathCompany

CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION

The conglomerate at hand is a leading shopping mall, communities, retail and leisure pioneer across the Middle East, Africa, and Asia. The mastercard business of the company is interested in capitalizing untapped acquisition potential within its movie customer base. The objective is to identify and acquire profitable customers for the mastercard business from the movie customer base through acquisition campaigns.

First, Industry and Client Understanding is done to get a clear idea of how the industry is and how the company is performing. Next, we break down the problem statement into what the status of the company is and what could be the future state and how to bridge the gap. Problem is broken down into four buckets of Business Opportunity, Analytical Problem, Analytical Outcome and Business Outcome. Next, we move on to understanding the customer level data. A factor map is generated to understand the various factors which can impact the central question of –

"Who can be Profitable?"

Subsequently, a data dictionary is generated to understand all the variables at hand in depth followed by Data pre-processing steps. Exploratory data analysis like univariate analysis, bivariate analysis is carried out to gain insights from the data. Hypothesis testing is carried to check if our hypothesis on the data holds true or not. From the previous steps, we have got the factors to consider for profitability. Split the data into train and test and then model the data for the factors. Then, test for the goodness of the model and tweak the model to get a higher accuracy.

2. IMPORTANCE OF PROPOSED WORK

The business believes that there is a huge potential of cinema customers for mastercards and there is an ability to cross-sell mastercards to those customers. To make use of the opportunity, we try to understand the behavior of customers who use mastercards for payments at cinemas and then identify who can be potential profitable customers. At the end of this analysis, we will have a set of characteristics or factors with which a customer can be deemed profitable and a framework to identify profitable customers to target for mastercards. The business impact is that appropriate acquisition campaigns can be run on segmented groups and would increase customer base and revenue.

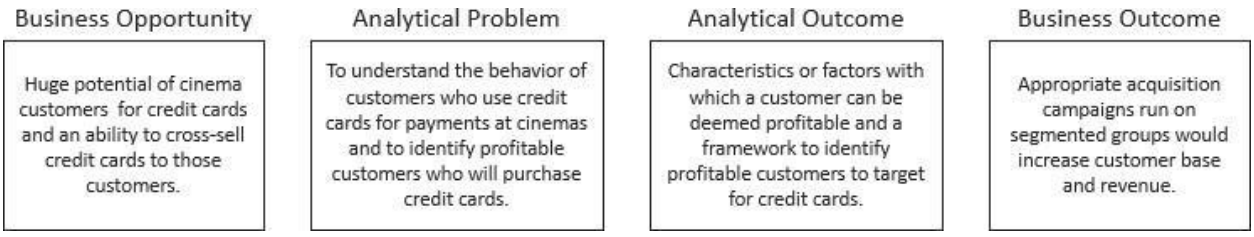

Figure 1.1: Problem Breakdown

We build a classification model to spot customers who are profitable for the mastercard companies from the movie industry. This would be accomplished by understanding the customer behavior and identifying repeated relationships to predict if they create profitable income to the mastercard companies.

The principal analysis is administered to know the attributes with which we will regard a consumer to be profitable for the mastercard company. This detailed analysis is conducted to verify with certainty and accuracy, if a customer is profitable or not and if the customer is profitable then how profitable can they be for the business.

3. LITERATURE REVIEW

In this chapter, few key concepts required to solve the problem are discussed.

3.1. Exploratory Data Analysis

Exploratory Data Analysis defines the crucial system of performing initial investigations on data to find out patterns, to spot anomalies, to check speculation and to test assumptions with the help of summary information and graphical representations. It combines the descriptive and diagnostic evaluation of a research. From Jebb et al. (2017), Exploratory statistics evaluation (EDA) includes numerous analysis additives to it. predominant ones to be univariate, bivariate, multivariate and correlation analysis. Univariate analysis also referred to as descriptive analysis allows us in locating the measures of valuable tendency (mean, median, mode). measures of dispersion (interquartile range, standard deviation), measures of asymmetry(skewness) and measures of flatness(kurtosis) of the records. Skewness is an important degree in our evaluation. If the distribution is positively skewed, it approaches there are better wide variety of larger values than smaller values (although the probabilities of them is probably small, they're higher than possibilities of smaller values). wonderful skew is gift due to advantageous outliers. Bivariate analysis reveals the connection between two variables. We conduct bivariate assessment for several permutations of continuous variables and specific variables. Count and percentage count are the metrics we build a two-way table on to analysis the relationships Class one variables constitute the rows and alternate variables represent the columns. Count of observations and percentage count are displayed for every aggregate of row and column classes.

3.2. Correlation Analysis and VIF

Correlation analysis is the method of studying relationships between quantitative variables or categorical variables. It is a measure of how things are related to each other.

Variance Inflation Factor (VIF) is a measure of multicollinearity among the independent variables in a multiple regression model. Let us assume X variables are standardized. VIF can be calculated over this standardized data. The design matrix after standardization is as follows,

$$\mathbf{X}^* = \begin{bmatrix} 1 & X_{11} & \dots & X_{1,p-1} \\ 1 & X_{21} & \dots & X_{2,p-1} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & X_{n1} & \dots & X_{n,p-1} \end{bmatrix}$$

Then,

$$\mathbf{X}^{*'}\mathbf{X}^* = \begin{bmatrix} n & \mathbf{0}' \\ \mathbf{0} & \mathbf{r}_{XX} \end{bmatrix},$$

where r_{xx} is the correlation matrix of X variables. Also,

$$\begin{aligned} \sigma^2\{\hat{\beta}\} &= \sigma^2(\mathbf{X}^{*'}\mathbf{X}^*)^{-1} \\ &= \sigma^2 \begin{bmatrix} \frac{1}{n} & \mathbf{0}' \\ \mathbf{0} & \mathbf{r}_{XX}^{-1}. \end{bmatrix} \end{aligned}$$

VIF_k for k = 1, 2,…,p1 is the *k*-th diagonal term of r^{-1}_{xx}. We only need to prove this for k = 1 because we can permute the rows and columns of r_{xx} to get the result for other k. Let's define:

$$\mathbf{X}_{(-1)} = \begin{bmatrix} X_{12} & \dots & X_{1,p-1} \\ X_{22} & \dots & X_{2,p-1} \\ \vdots & \vdots & \vdots \\ X_{n2} & \dots & X_{n,p-1} \end{bmatrix}, \mathbf{X}_1 = \begin{bmatrix} X_{11} \\ X_{21} \\ \vdots \\ X_{n1} \end{bmatrix}.$$

Note: both matrices are different from design matrices. Since we only care about the coefficients of X variables to establish a relationship with correlation, the 1-vector of a design matrix can be ignored in the calculation. Hence, by using Schur's complement from Zhang (2006),

$$\begin{aligned} r_{XX}^{-1}(1,1) &= (r_{11} - r_{1\mathbf{X}_{(-1)}} r_{\mathbf{X}_{(-1)}\mathbf{X}_{(-1)}}^{-1} r_{\mathbf{X}_{(-1)}1})^{-1} \\ &= (r_{11} - [r_{1\mathbf{X}_{(-1)}} r_{\mathbf{X}_{(-1)}\mathbf{X}_{(-1)}}^{-1}] r_{\mathbf{X}_{(-1)}\mathbf{X}_{(-1)}} [r_{\mathbf{X}_{(-1)}\mathbf{X}_{(-1)}}^{-1} r_{\mathbf{X}_{(-1)}1}])^{-1} \\ &= (1 - \beta'_{1\mathbf{X}_{(-1)}} \mathbf{X}'_{(-1)} \mathbf{X}_{(-1)} \beta_{1\mathbf{X}_{(-1)}})^{-1}, \end{aligned}$$

where $\beta_{1X(-1)}{}^{-1}$ is the regression coefficients of X1 on X2,. . . ,Xp1 except the intercept. In fact, the intercept should be the origin, since all X variables are standardized with mean zero. On the other hand,

$$\begin{aligned} R_1^2 = \frac{SSR}{SSTO} &= \frac{\beta'_{1\mathbf{X}_{(-1)}} \mathbf{X}'_{(-1)} \mathbf{X}_{(-1)} \beta_{1\mathbf{X}_{(-1)}}}{1} \\ &= \beta'_{1\mathbf{X}_{(-1)}} \mathbf{X}'_{(-1)} \mathbf{X}_{(-1)} \beta_{1\mathbf{X}_{(-1)}}. \end{aligned}$$

Therefore,

$$VIF_1 = r_{XX}^{-1}(1,1) = \frac{1}{1 - R_1^2}.$$

From the above derivation, we can find that, VIF is indirectly proportional R_1^2, R_1^2 is the R2 - value obtained by regressing the jth predictor on the remaining predictors. A VIF of 1 means that there is no correlation among the jth predictor and the remaining predictor variables, and hence the variance of b_j is not inflated at all. The general rule of thumb is that VIFs exceeding 4 warrant further investigation, while VIFs exceeding 10 are signs of serious multicollinearity requiring correction.

3.3. Stratification

From Sechidis et al. (2011), Stratified sampling is a sampling method that takes into ac- count the existence of disjoint groups within a population and produces samples where the proportion of these groups is maintained.

3.4. Evaluation Metrics

3.4.1. Confusion Matrix

True Positive (TP): When a predicted observation belongs to a class and it does belong to that class.

True Negative (TN): When a predicted observation does not belong to a class and it actually does not belong to that class.

False Positive (FP): When a predicted observation belongs to a class and it does not belong to that class.

False Negative (FN): When a predicted observation does not belong to a class and it does belong to that class.

Accuracy = $\frac{TP+TN}{TP+TN+FP+FN}$

Precision = $\frac{TP}{TP+FP}$

Recall = $\frac{TP}{TP+FN}$

F1- Score: The harmonic mean of the model's precision and recall. A better measure to use if you are seeking a balance between Precision and Recall. F1 is a quick way to tell whether the classifier is good at identifying members of a class, or if it is finding shortcuts (e.g., just identifying everything as a member of a large class).

$$F1 = \frac{2*Precision*Recall}{Precision+Recall} = \frac{2*TP}{2*TP+FP+FN}$$

3.5. AOC-ROC Curve

Receiver Operating Characteristics (ROC) is used for identifying problems with probability outputs. So, by changing the threshold, numbers will also change in the confusion matrix.

1. False Positive Rate (1-Specificity): Fraction of negative instances that are incorrectly classified as positive.
2. True Positive Rate (Sensitivity): Fraction of positive instances that are correctly predicted as positive.

From the above graph, the TPR increases at a higher rate but suddenly at a certain threshold, the TPR saturates. For every increase in TPR, we must pay the cost of an increase in FPR. At the initial stage, the TPR increase is higher than FPR. Select the threshold for which the TPR is high and FPR is low.

Area Under ROC Curve (AUC) is the measure of the ability of a classifier to distinguish between classes and is used as a summary of the ROC curve. The higher the AUC, the better the performance of the model at distinguishing between the positive and negative classes.

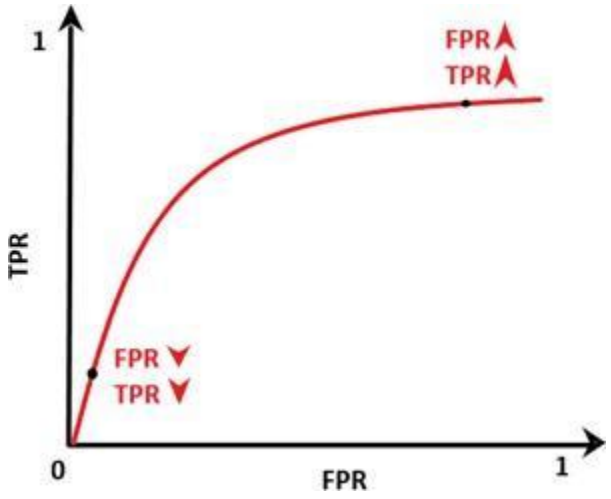

Figure 2.3: ROC Curve - Sample

3.6. Data Set

The dataset includes transactional records of clients visiting a film theatre and use the business enterprise's mastecard for buys. We understand through the dataset the prevailing clients between the two groups, their behavioral styles related to sales and revenue generated from the master card business. From these intersectional records, the model algorithm would be able to analyze the elements which a purchaser becomes valuable and thus consequently profitable, and apply that to the customers of the cinema business to target the profitable cluster.

The data is an aggregated dataset which consists of transactional data of 22488 unique consumers. It has nearly 53 data variables comprising of aggregate customer spends over weekends, total spends, movie tickets, genre of movie, type of cinema theatre, food and beverage spends and so on. The cinema industries are in the Middle- East, which makes standard currency of transaction as AED (United - Arab - Emirates - Dirhams). Revenue generated by the customers of the mastercard company are set as the target variable. The continuous variable is revenue. For our analysis the target variable needs to be converted to a binary variable which is achieved by setting a threshold determined by the mastercard company based acquisition cost of a customer and the intrinsic value that makes a customer profitable. This a metric decided by the business and based on this we set our target variable to a new column generated, profitability.

The data is imbalanced as the rate of profitability or even rate in the data is calculated to be 23.14%. The chance of incident of one class is greater than the opposite class. A sampling technique called SMOTE can be used to over-sample the data to make it more balanced. But, for the scope of this problem statement, we decided to maintain the given aspect ratio.

3.7. Software/ Tools Requirements

The main programming language used here is Python (version: 3.7.5), as it has numerous libraries for data science and machine learning. Also, there are various forums and reference documents available for python. The python libraries used for the project are NumPy (version: 1.18.1) as it supports high-level mathematical operations on large multi-dimensional arrays and

matrices. Next is Pandas (version: 1.0.4). This is our mostly used tool to manipulate the data tables and perform various pre-processing over the data. The machine learning library used is Scikit-Learn (version: 0.23.1). For all visualizations in python, Plotly (version: 4.10.0) was used. The IDE used for coding was Visual Studio Code with an integrated support for version control with Git. Jupyter was used for coding purposes and Excel was also used for minor data pre-processing and reporting purposes.

4. PROBLEM STATEMENT

X is a leading shopping mall, communities, retail and leisure pioneer across the Middle East, Africa, and Asia. The mastercard business of the company is interested in capitalizing untapped acquisition potential within its movie customer base. The objective is to identify and acquire profitable customers for the mastercard business from the movie customer base through acquisition campaigns.

Cross-selling is a strategy where a company introduces/suggests complementary or better products to customers according to the patterns exhibited in their past behavior. In our, case, Mr. C already goes to cinemas. based on his purchases and transactions at the cinema, we are suggesting him for a mastercard.

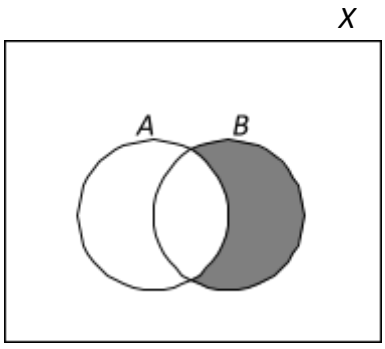

Figure 2.1: Target Customers: $B - A$

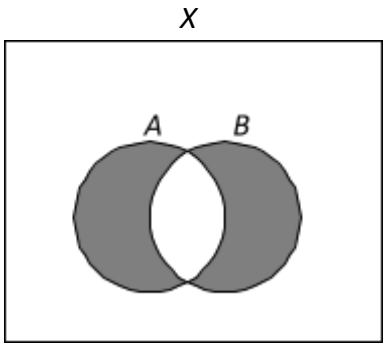

Figure 2.2: Customers of Interest: $A \cap B$

The figures depict the Company X with a Mastercard Business A and a Cinema Business B. In figure 2.2, the shaded portion $B - A$ is our target customers. In figure 2.1, the white portion $A \cap B$ will be our customers of interest. This set of customers will be used for our analysis to gain insights an obtain the profitable features.

5. OBJECTIVES

The objective of the problem statement is quite straight forward in its definition. It is to acquire “potential” customers from the movie customer base. “Potential” is an important term as we are looking for new customers who can increase the profits for the Mastercard business. The dataset we have is customer level data of the cinema industry. Customers using the corporation’s mastercard for transactions and payments when they visit cinemas, are the populace of interest. We analyze the behavior of this populace to determine the attributes with which a patron can be deemed worthwhile or profitable to target. The preferred consumer acquisition desires set by means of the master card enterprise is not accomplished. Even though, there may be a high customer-acquisition capacity in the pre-existing group of customers, it goes untapped because the mastercard enterprise is not able to determine who are likely the most profitable customers to target. The modern methods in this area are just traditional advertising and marketing methods. Higher return on investments can be achieved with data driven marketing campaigns and careful tailored and targeted customer acquisition campaigns. They cognizance on leveraging the present consumer base to make exceptional commercial enterprise selections. for this reason, through having this model in hand the employer can target the potentially profitable clients.

6. METHODOLOGY AND EXPERIMENTAL SETUP

6.1. System Analysis

The system is designed in such a way it is reusable for any type of classification problem. Modularization, parameterization, reusability, and scalability are the main motives for such a system design. All modules are designed with software engineering best practices.

6.1.1. System Requirement Analysis

- Python - v3.7.5
- Jupyter Notebook
- Git
- GitLab
- Visual Studio Code
- Excel
- Azure Cloud Storage 8. NumPy – v1.18.1
- Pandas – v1.0.4
- Scikit-Learn – v0.23.1
- Plotly – v4.10.0
- OS: 64-bit Windows 10 Pro

6.1.2. Setup Design

1. Data Quality Check
    - Data check
    - Percentage Missing data
    - Percentage Redundant data
    - Directory of variables of different data types

2. Data Cleaning
    - Punctuation Removal and extra characters
    - Null Value Formatting
    - Data type Conversion
    - Case Sensitivity Check
    - Duplicates Removal
    - Dropping columns with > = 90% missing values, redundant columns

3. Exploratory Data Analysis
    - Factor Mapping – Brainstorm possible factors of profitability & frame hypotheses
    - Univariate Analysis – Check data distribution, Imputation methods, Outlier Identification
    - Data Imputation
    - Bivariate Analysis – Decile variables, Relationship against Profitability, Test the hypothesis
    - Multivariate Analysis
    - Correlation Analysis

4. Feature Engineering
    - One hot encoding of Categorical Variables
    - Feature generation - New feature generation with existing features
    - Feature Selection - Variance Inflation factoring to select the final features
    - Outlier Treatment - Outliers identified with IQR are treated with Quantile Capping Technique

5. Modelling
    - Split Train & Test

- Build base models with no hyperparameters
- Hyperparameter Tuning
- Model Rebuild
- Prediction
- Performance Evaluation

6.1.3. Flow Diagram of the System

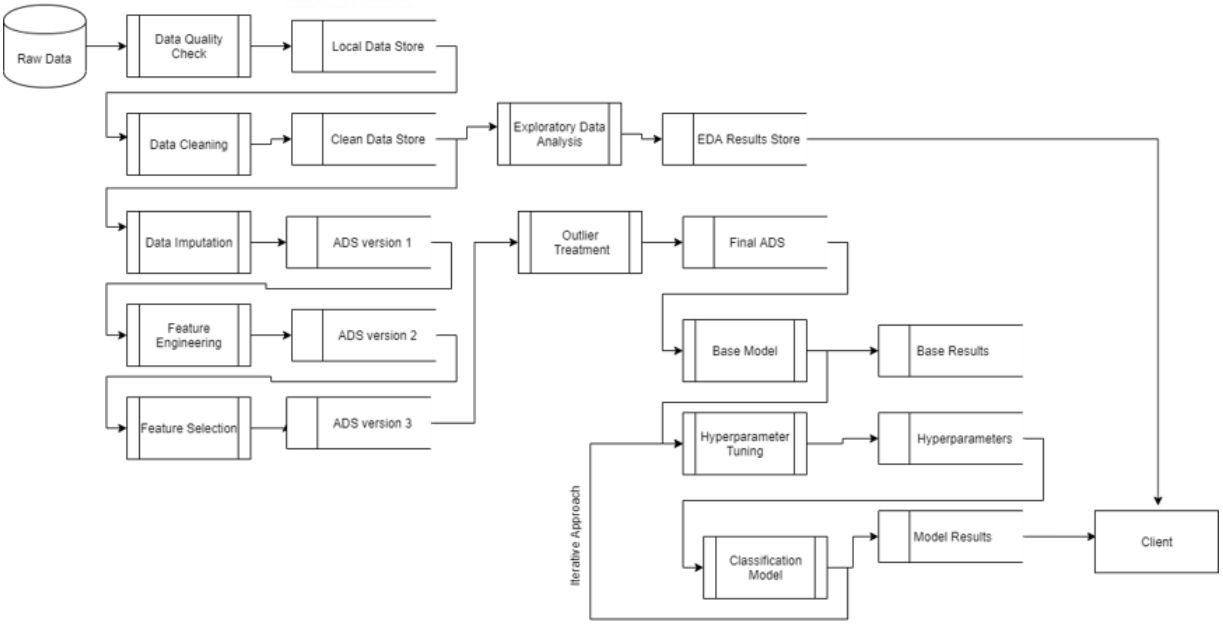

Figure 3.1: Data Flow Diagram

6.2. Implementation

Factor mapping was conducted to recognize the capability of elements that could have an impact on profitability of a cinema enterprise customer and a hypothesis was generated for the same. This is further validated in our exploratory data analysis.

Central Question			What are the factors with which you deem a customer to be PROFITABLE?			
S.no	Factors	Sub Category	Data Elements			Hypothesis
			Demographic	Transaction	Exogenous sources	
1	Age		✓			Gen X people are more profitable .
2	Offers Availed			✓		Customers who use CC just because they have offers are profitable.
3	# of transactions			✓		Customer making more transactions can be are profitable.
4	Resident Location		✓			Customers who have malls very close to them, visit it often and can be profitable.
5	Day of visit			✓		People visiting on weekends are profitable.
6	Cancellation			✓		People who don't apply for cancellation are more profitable
7	Frequency of visits			✓		People who visit cinemas more often are profitable
8	Quality of screens they visit	Screen Quality		✓		People visiting premier screens are more likely to be classified as people with good spending capacity and hence, are more profitable.
		Type of Seats		✓		Customers who choose more luxurious seats are more profitable.
9	Genre of movies			✓		People who watch all genres of movies are more profitable.
10	Snacks and refreshments	Buy refreshments		✓		People purchasing snacks and refreshments on their visits to cinemas are profitable as compared to people who do not.
		Total amount spent		✓		People spending more money on snacks and refreshments are profitable.

Figure 4.1: Factor Mapping and Hypothesis – Sample

6.2.1. Data Cleaning

A preliminary data check is performed to verify the cleanliness of the data at hand which incorporates new statistics, discrepancy in data in different columns, type of records, missing-data percentage, percentage of redundant records, listing datatypes of different variables from numerical, date-time and categorical. These data checks were summarized to create a data-dictionary for future reference.

Fundamental pre-processing of data is conducted like punctuation removal, special characters removal, formatting null values and garbage values to standard null (NaN) values. To ensure data columns to be in the right format for use data type conversion is also conducted. Case sensitivity test is performed to ensure case categories to not be considered as one different input. Any duplicate record observed were removed.

```
Missing Value Per Column
def missing_values_per_column(df):
    """
    Look for missing values(np.nan) in all the columns.

    Parameters
    ----------
    df : Required dataframe

    Returns
    -------
        Returns a dataframe with the missing value count and missing percent.
    """
    missingTable = pd.DataFrame()
    missingTable['MissingCount'] = df.isnull(
    ).sum().sort_values(ascending=False)
    missingTable['MissingPercent'] = 100 * df.isnull().sum() / len(df)

    return missingTable
```

Figure 4.2: Data Quality Check - Sample Code

Summary							
	No. of records	No. of Variables	Primary Key	Data Availability Issues	Frequency of Data Load/Update	From	To
	65,790	53	Customer_ID	NO	Yearly	2018	2019

Details							
Sno	Variable Name	Variable Type	Data Type	Variable Description	Sample Values	# of duplicates	# of Null Observations
1	Customer_ID	Nominal	int64	Identification Number	1,2,3,...22488	44905	0
2	Booking_Amount	Continuous	float64	Purchase Amount	60, 45	59992	0
3	Redemption_Amount	Continuous	object	Redeemed amount on the purchase	29.4, NA, 20	54659	5486
4	No_of_Tickets	Discrete	object	Number of Tickets Purchased	2, 50, NA	43982	14
5	Latest_Transaction	DateTime	object	Customer's last transaction date (DD-MM-YYYY)	08-12-2018, 28-11-2019,NA	64772	6
6	Food_Bev_Spend	Continuous	object	Amount spend on food and beverage	9, 18, 144, NA	65604	0
7	weekend_spend	Continuous	object	Amount spend during weekends	#VALUE!, 65.1,	50000	20
8	Movies_Watched	Discrete	object	Number of Movies watched	1,2,5,8, NA	60000	40
9	cinema_experience	Nominal	object	Type of cinema experience	regular, premium	65843	40
10	Movie_Duration_Avg	Continuous	object	Average Movie Duration in Hrs	2.5, 3.0833333333	60549	0

Figure 4.3: Data Dictionary-Sample

6.2.2. Exploratory Data Analysis

We start off with univariate analysis for our exploratory data analysis. To comprehend the distribution of data points, a descriptive analysis is carried out in which we identify outliers and methods suitable for imputation of missing values.

On analysis, we came to conclusion that most of the data was positively skewed which made us understand that majority of the consumers range from the lower deciles of the data. 40% of the variables had outliers and 7% of that data were outliers. Thus, further along the line while we conduct our modelling, we must ensure that algorithms used tends to handle outliers well.

Data imputation component is executed to make sure we have the correct data for our diagnostic analysis. Based on the percentage of missing values, data imputation methods were chosen.

Missing Values %	Numerical	Categorical
Less than equal to 60%	Columns Dropped	Columns Dropped
Less than 1%	Imputation with Median	Imputation with Mode
Between 1% - 40%	KNN Imputation	NaN
Flag Columns	NaN	Imputation with similar columns

Table 4.1: Imputation Techniques for Different Types of Variables

KNN Imputation

```
def knn_imputation(df):
    """
    Replace NaN with KNN model.

    Parameters
    ----------
    df: Dataframe containing the cols

    Returns
    -------
    Dataframe where the missing values have been imputed.
    """
    scaler = MinMaxScaler()
    df_x = pd.DataFrame(scaler.fit_transform(df), columns=df.columns)

    imputer = KNNImputer(n_neighbors=5)
    df_y = pd.DataFrame(imputer.fit_transform(df_x), columns=df_x.columns)

    num_data = scaler.inverse_transform(df_y)
    num_data = pd.DataFrame(num_data, columns=df_y.columns)

    return num_data
```

Figure 4.4: KNN Imputation - Sample Code

Once the imputation is performed. To validate the hypothesis generated with respect to the factor mapping, we carry our bivariate and multivariate analysis. The graphs generated are done so by binning which is segregating deciles of numerical variables or percentiles as ranges and they are mapped against Profitability. Likewise for categorical variables, the same analysis is done with

respect to the different categories at hand for a variable and that is mapped against profitability. The threshold value of ~ 23% is used in our analysis to compare who are profitable.

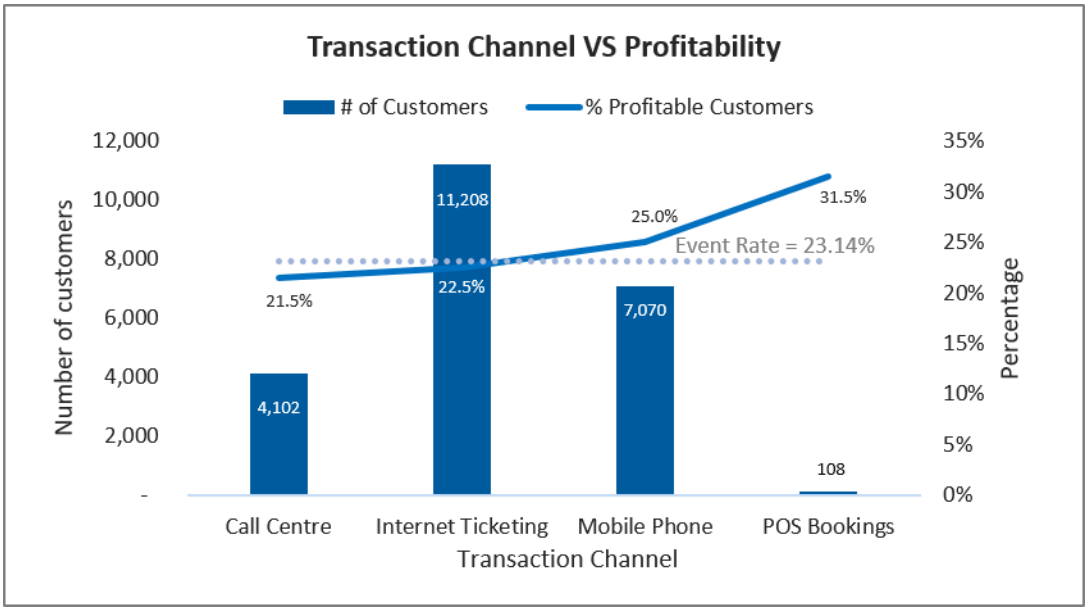

Figure 4.5: Multivariate Analysis - Transactional Channel

On examination of the purchasing channel customers use, it is observed that a substantial segment of people uses Internet Ticketing to purchase tickets. Only a minuscule amount of people books their tickets directly at the Point of Sale (POS). The noteworthy part is that this group of people are the most profitable as compared to other groups who purchase their tickets at other channels. After diagnosis, we conclude that the probability of them making impulsive purchases likely buying premium tickets is higher at POS, as they buy whatever is available at the moment which increases the likelihood of them being profitable.

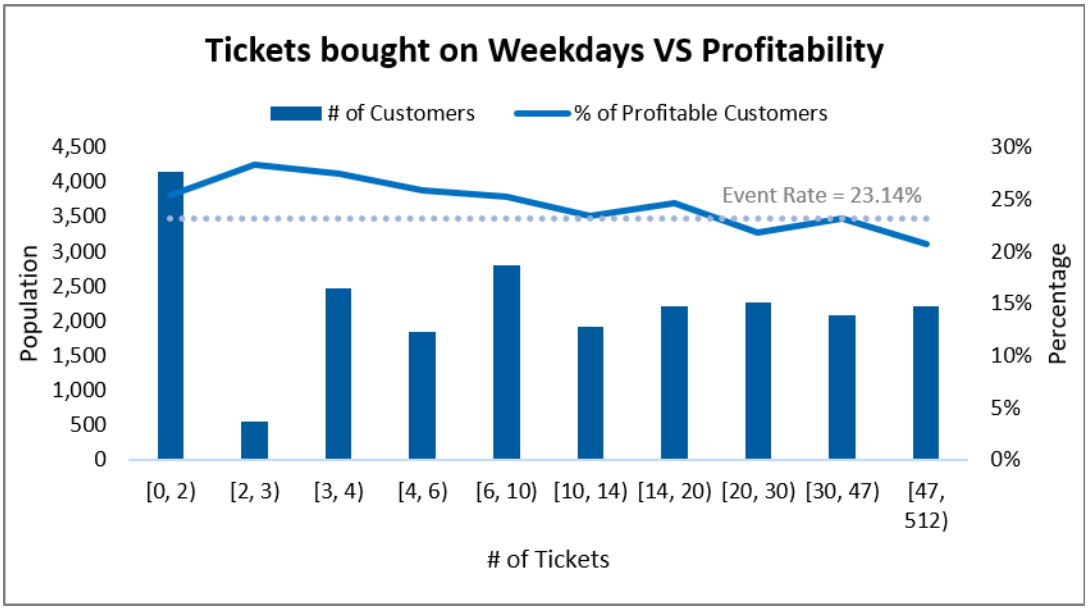

Figure 4.6: Multivariate Analysis - Weekdays Tickets

We generated new features to help our analysis. For ex.: From overall tickets and tickets on weekends, we generated Tickets bought on weekdays. We observe that people buying more tickets during the weekdays are less profitable. On diagnosis, this could be the result of them availing offers given away during the weekdays to keep a steady volume of sales.

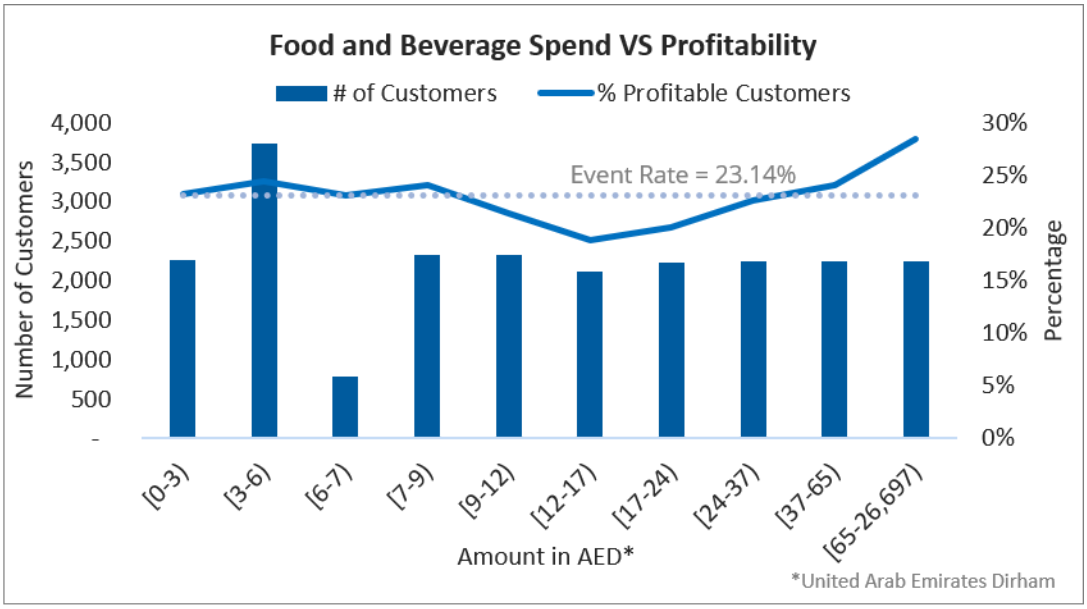

Figure 4.7: Multivariate Analysis - Beverage and Food Spend

From the above graph, we can observe an uptrend in profitability for consumers shilling more that 12 AED on beverages and foods at movie theaters. But the people who are the most profitable are the ones who spend more than 37 AED over a year on food and beverages.

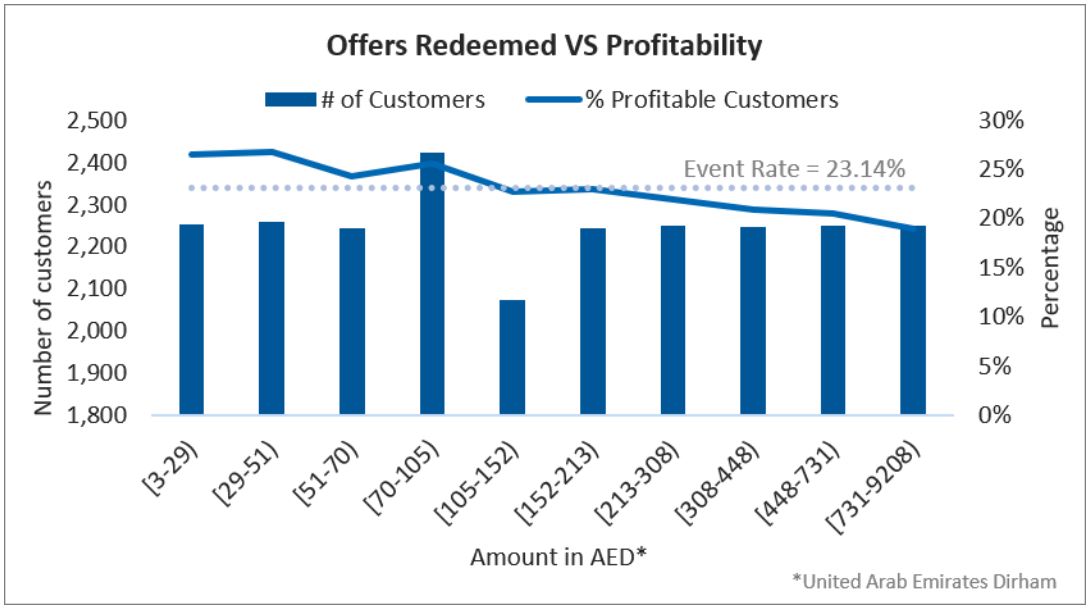

Figure 4.8: Multivariate Analysis - Offers Availed

In our factor mapping we created a hypothesis stating that people availing for more offers are unprofitable. In the above graph, we can see a clear downtrend in profitability as the consumer avails more offers, thus increasing the total money saved through offers in a year. Consumers availing greater than 105 AED a year on their bookings are unprofitable to the company.

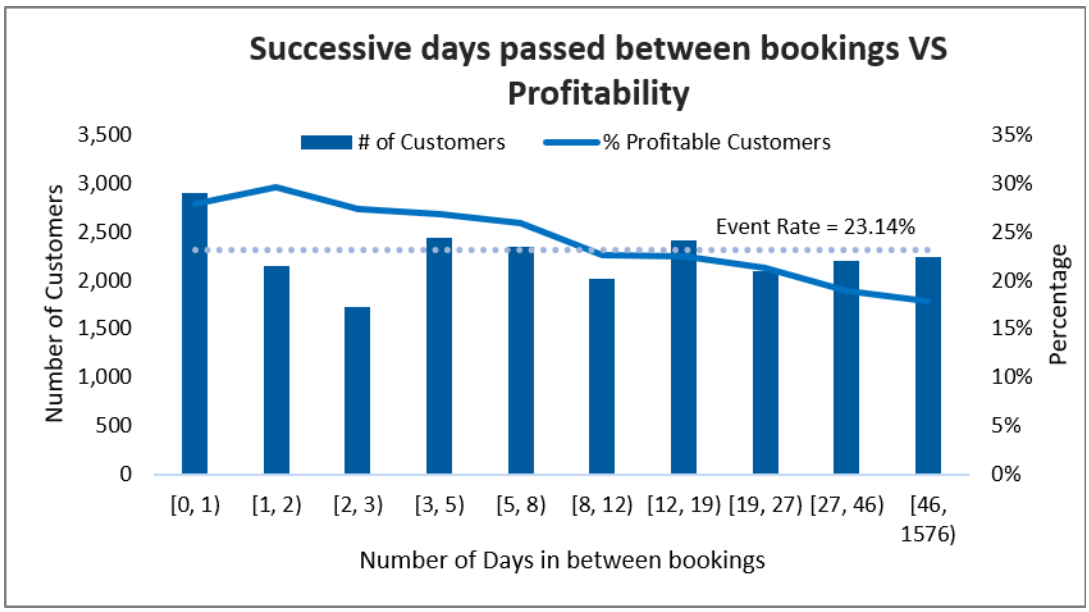

Figure 4.9: Multivariate Analysis - Days between bookings

From the above graph, we understand that customer making a booking within a week of their previous booking are further profitable to the business than those taking more than week. To conclude our exploratory data analysis, we gained useful insights and finding which can help us make more informed business decisions.

6.2.3. Feature Engineering and Feature Selection

6.2.3.1. Feature Engineering - Primary

After our primary evaluation is completed, we flow to prepare for predictive analysis. Certain feature engineering like one hot encoding of categorical variables must be carried out before the data is fed into the model. We keep the most occurring categories of a categorical variables as many categories are present and bundle the remaining categories into others. This guarantees we only have highly populated categories.

6.2.3.2. Feature Selection

As we move into modelling phase, we need to comprehend the impact of variables on each other and which variables directly affect profitability of a consumer. From, Paul and Kumar (2006) even with multicollinearity many classification models can do predictions but, to understand the impact of independent variables on the target variable, multicollinearity can create problems. Katrutsa and Strijov (2017) states various multicollinearity problems can be solved using quadratic programming techniques, but we stick with Variance Inflation Factor or VIF to remove multicollinear variables from the dataset to reduce unnecessary complexity.

Correlation is a measure of relationship between two variables while Variance Inflation Factor (VIF) is an overall index of strength between all variables at hand. From Curto and Pinto (2011), states that VIF values should be lower than 10 and values over 10 signify multicollinearity. We used the correlation matrix generated and the findings from our exploratory data analysis to conduct various iterations of VIF. After multiple iterations, we end up with 19 features from the initial list of 53 data variables. A couple of new features related to ratios were also generated during exploratory data analysis. Following is a list of features, that directly affect the profitability of a customer.

Number of Tickets bought on Weekends, Booking Amount, Offers Redeemed, Average Movie Duration, Average Ticket Cost, Transaction Channel (Internet Ticketing), Transaction Channel (Mobile Phone), Watched an action movie or not, Watched a Hollywood movie or not, Amount spent on preferred cinema location, Amount spent on preferred film rating, Amount spent on preferred cinema experience, Amount spent on Food and Beverages, Number of Unique Movies Watched, Number of Visits in Last 90 days, Average time taken to make a booking, Number of Tickets bought on Weekdays, Average Spend per Visit, Customer Tenure

```
VIF

def vif_operation(df, cols):

    if cols:
        df.drop(cols,axis=1,inplace=True)

    vif_data = pd.DataFrame()
    vif_data["Features"] = df.columns

    #calculating VIF for each feature
    vif_data["VIF"] = [VIF(df.values, i)
                           for i in range(len(df.columns))]
    vif_data = vif_data.sort_values(by='VIF', ascending=False,
                                            ignore_index=True)

    cols_tuple =tuple(cols)
    d[cols_tuple] = vif_data

    return vif_data
```

Figure 4.10: Variance Inflation Factor - Sample Code

6.2.3.3. Feature Engineering – Secondary

After feature selection, secondary feature engineering must be completed to ensure removal of outliers so that the data can be scaled for modelling. From Raju et al. (2020), scaling techniques like min-max scaler, standard scaler, and robust scaler were used to normalize the data. We used robust scaler which is a capping technique till the 99th percentile, as it was giving better results than other scaling methods. Furthermore, in our EDA, the 99th percentile to the 100th percentile range of our continuous data showed huge variation signaling that this is where our outliers presided. Thus, outliers were removed using the above-mentioned quantile-based technique. Now, when a standard scaling was performed, the results were better than the regular robust scaler. Hence, treating the outlier first and the scaling gave better results for us.

6.2.4. Modelling

6.2.4.1. Train and Test Split

Bearing in mind, after our numerous iterations pertaining to finding the critical features for modelling, we split the data into train and test. Two ratios of 70 - 30 and 80 -20 were

attempted for splitting the dataset. From Farias et al. (2020) and Verstraeten and Van den Poel (2006) we should split the train and test data while conserving the similarity which in turn leads to robust testing. Thus, we use the stratify parameter and specify it while splitting our dataset. Stratifying works extremely well and is helpful for imbalanced datasets which is the case here. After multiple iterations, we concluded on splitting our dataset in an 80--20 ratio with stratify metrics in place.

6.2.4.2. Classification Algorithms

We used Supervised Machine learning technique for prediction as our data has a target variable. From Kumari and Srivastava (2017) there are innumerable amounts of classifiers present ; base, combined and ensemble techniques. We experiment with different classifiers in this project and check their consequent result. Base classifiers used are as follows: Logistic regression, naïve-bayes and k-nearest neighbor. Classifiers coming under the ensemble techniques like random forest were also implemented.

Originally, a base model is developed and trained which goes into a prediction component for testing. The model then enters the hyperparameter tuning phase where the right hyperparameters for our model is generated. We us Random Search CV to obtain the optimal hyperparameters for our model but we also manually tune the model sometimes to get the best results. Multiple iterations of modelling are executed with numerous different hyperparameters.

```
y = data['Profitable']
X = data.drop(['Profitable'], axis=1)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20,
                                                    random_state=42, stratify=y)
# train model
model = RandomForestClassifier(n_estimators=800, min_samples_split=15,
                               min_samples_leaf=30, max_samples=1000,
                               max_features='log2', max_depth=20,
                               criterion='entropy', bootstrap=True
                               class_weight='balanced_subsample')

model.fit(X_train, y_train)
# generate precision-recall table
y_test_probs = model.predict_proba(X_test)[:,1]
# Containers for true positive / false positive rates
precision_scores = []
recall_scores = []
f1_scores = []
# Define probability thresholds to use, between 0 and 1
# probability_thresholds = np.linspace(0, 1, num=20)
probability_thresholds = np.arange(0, 1, 0.01)

# Find true positive / false positive rate for each threshold
for p in probability_thresholds:

    y_test_preds = [0 if prob<=p else 1 for prob in y_test_probs]

    precision = precision_score(y_test, y_test_preds)
    recall = recall_score(y_test, y_test_preds)
    f1 = f1_score(y_test, y_test_preds)

    precision_scores.append(precision)
    recall_scores.append(recall)
    f1_scores.append(f1)

df = pd.DataFrame({'Prob_Threshold':probability_thresholds, 'Precision':precision_scores,
                                        'Recall':recall_scores, 'F1_Score':f1_scores})

train_preds = np.where(model.predict_proba(X_train)[:,1] >= 0.45, 1, 0)
test_preds = np.where(model.predict_proba(X_test)[:,1] >= 0.45, 1, 0)

# generate classification report

classification_report(y_train, train_preds, y_test, test_preds)

# plot confusion matrix

plot_confusion_mat(y_test, test_preds)
```

Figure 4.11: Prediction Module - Sample Code

6.2.4.3. Evaluation Metrics

For our evaluation, the standard evaluation metrics are not accurate as they are not right for imbalanced datasets. For our problem statement, the distribution of majority-minority class is 77 : 23, and consequently categorizing majority class data points would give us 77 % accuracy which is a decent score but, our model would not have gained any learning. We focus

on three key metrics namely Recall, AUC score, and Precision from Wasikowski and Chen (2010). Capture rate or recall indicates the portion of customers who are correctly labeled as profitable from all the possible profitable customers.

Hyper Parameter Tuning

```
y = data['Profitable']
X = data.drop(['Profitable'] , axis=1)

X_train,X_test,y_train,y_test=train_test_split(X, y, test_size = 0.2, random_state = 42, st
ratify = y)

Model =  RandomForestClassifier()

param_grid = [
            {
                  'criterion': ["gini", "entropy"],
                  'bootstrap': [True, False],
                  'max_depth': [5,10,15,20,25,30,None],
                  'max_features': ['auto', 'log2'],
                  'min_samples_leaf': np.arange(10,100,10),
                  'min_samples_split': np.arange(5, 50, 5),
                  'n_estimators': [100,200, 400, 600, 800, 1000, 1200],
                  'class_weight'  : ["balanced", "balanced_subsample", None],
                  'max_samples' : np.arange(100, 2000, 100)
            }
]

hp = RandomizedSearchCV(estimator = Model, param_distributions = param_grid, n_iter = 100,
cv = 3, verbose=True, random_state=42, n_jobs = -1, scoring='roc_auc')


best_hp= hp.fit(X_train, y_train)

best_hp.best_params_
```

Figure 4.12: Hyperparameter Tuning - Sample Code

AUC score quantifies how well the model can distinguish between profitable and non-profitable customers. With low recall, the model incorrectly classifies profitable customers as non-profitable and when precision is low, the opposite happens where the model classifies non profitable customers as profitable. There is a trade-off between these two which we must examine. Taking the business requirements under consideration, we chose recall as out primary metric and sacrificed on precision. "The cost of lower recall is way higher than the cost of lower precision". Conversion rate or precision is the portion of those customers who are truly profitable out of all the predicted profitable customers. To make prediction we get a threshold

using Precision-Recall Curve (PRC) and F1-score. At the highest value of F1 score we obtain our threshold.

7. RESULT ANALYSIS DISCUSSIONS

From the EDA conducted we recognize the actual significance of the variables at hand and get an in depth understanding of how important the various variables at hand are indeed. Using the data at hand we conclude that there are some important features which directly affect profitability. On generation of new features, we get great insights and findings, especially for the ones generated using ratios of weekdays to overall spend and spend of food and beverages to total spend. Generating such features helped us gain some quality insights. Removing features with high multicollinearity is important for us to not make our model overlearn so we remove such features to ensure that.

The 4 classification model results have been discussed below. The classification models have undergone an iterative method of training in which the model builds, evaluates, compares, and rebuilds. To show the progress of model training procedure we show some of the multiple iterations conducted. The final model is picked on the basis of their performance with the test set.

	Train					Test					
Iteration	Accuracy	Precision	Recall	AUC Score	F1 Score	Accuracy	Precision	Recall	AUC Score	F1 Score	Threshold
1	75	55	0.3	0.5	0.7	53	28	61	56	39	24
2	33	25	91	53	40	32	25	91	52	39	19
3	31	25	93	52	40	31	25	93	52.6	39.9	18
4	33	25	91	53	40	32	25	90	52	39	19
5	49	28	68	56	40	48	27	67	55	39	23

Figure 5.1: Model Results - Train and Test – Logistic-Regression

First iteration recall values, Train: 0.3, Test: 61. We can observe a lot of disparity between test and train recall. AUC is 0.5 for the train set which denotes absolutely zero learning on the models' part, and it basically takes a 50 / 50 chance in distinguishing between the 0 and 1 class. In the 2nd iteration the recall is high: 91%, and a precision of 25 % is observed with a slight increase in AUC score. In the third iteration, we have a lower AUC score. The 4th iteration is similar to the second iteration with a high recall and low precision value. In the 5th iteration, we noticed a fine balance between precision and recall. Using manual hyperparameter tuning we tried to improve the AUC score but there was no substantial change. Overfitting was a common problem in other iterations. Thus, we selected the 5th iteration to be the satisfactory result for logistic regression model after several iterations. Hyper-parameters for this model are, Max-iter = 5000, C = 0.885 penalty = L2, solver = sag / saga. (Figure: 5.1)

	Train					Test					
Iteration	Accuracy	Precision	Recall	AUC Score	F1 Score	Accuracy	Precision	Recall	AUC Score	F1 Score	Threshold
1	48	27	68	55	39	47	27	68	54	38	5
2	50	27	66	55	39	49	27	65	54	38	4
3	45	26	72	54	39	43	25	70	53	38	10
4	56	29	55	56	38	55	29	56	55	38	1
5	50	28	66	56	39	49	27	66	55	38	45

Figure 5.2: Model Results - Train and Test – Naïve-Bayes

Gaussian and Complement Naïve Bayes were used for this modelling phase. The first 2 iterations are Gaussian and the remainder are complementing. The best Naïve Bayes AUC score was 55 % with maximum recall: 66 %. So, the fifth iteration of the model was selected as the best Naïve Bayes model. (Figure: 5.2)

	Train					Test					
Iteration	Accuracy	Precision	Recall	AUC Score	F1 Score	Accuracy	Precision	Recall	AUC Score	F1 Score	Threshold
1	51	33	100	67	50	39	25	76	52	38	19
2	48	30	83	60	44	43	26	75	54	39	20
3	74	48	71	73	57	60	28	41	53	33	20
4	37	27	92	56	42	34	25	88	52	40	16
5	43	28	87	58	43	39	26	81	53	39	19

Figure 5.3: Model Results - Train and Test – K-NN

KNN generaed a good batch of results after unsatisfactory results from the previous model deployed with a good recall: 81 % but had a poor AUC score of 53 % which signified that the model was inadequate at distinguishing between non profitable and profitable customers. But after tuning the hyperparameters we could generate good results with distance-metric = 'manhattan', n-neighbors = 36. (Figure: 5.3)

	Train					Test					
Iteration	Accuracy	Precision	Recall	AUC Score	F1 Score	Accuracy	Precision	Recall	AUC Score	F1 Score	Threshold
1	98	92	1	98	96	47	30	74	56	40	21
2	50	32	96	65	48	40	27	82	54	40	20
3	64	39	87	72	54	52	29	67	57	40	23
4	68	42	82	73	56	56	30	61	58	40	24
5	47	29	79	58	42	47	28	77	57.8	41	43

Figure 5.4: Model Results - Train and Test – Random-Forest

Random Forest as expected earlier in our discussion, had to perform well. After numerous iterations with different hyperparameters and manual tuning, a Recall of 77 % and an AUC score of 58 % was attained. For our best RF model, the hyperparameters are as follows: n-estimators= 801, class-weight = 'balanced-subsample', max-samples =1001,

criterion='entropy', max-depth = 20, max-features = 'log2', min-samples- leaf=30, min-samples-split=15. (Figure: 5.4)

After numerous consecutive iterations of the 4 models, a champion-challenger model methodology was employed for the 4 different models after rigorous iterative modelling processes to select the best model out of all. After exclusion of models which do not fit our solution, we zero in on our challenger and champion model.

Model	Accuracy	Precision	Recall	AUC Score	F1 Score	Threshold
Logistic Regression	48	27	67	55	39	23
Naïve Bayes	49	27	66	55	38	45
K Nearest Neighbors	39	26	81	53	39	19
Random Forest	47	28	77	57.8	41	43

Figure 5.5: Test Performance Results of All Models

From, figure 5.5, elimination of LogisticRegression and NaïveBayes was obvious because of poor recall scores compared to the other models. From Kaviani and Dhotre (2017), NaïveBayes is well suited for text classification problems. The assumption it makes during analysis that there is full independency amongst features at hand is not useful while solving real problem statements. Also, the presence of inter-dependent features in our model sets up NB for failure.

Model	AUC score	Recall
K-NN	53.1	81.1
Random-Forest	57.8	77.2

Table 5.1: Performance Results of KNN and Random Forest

Recall and AUC score are the 2 main parameters we examine for models left which are Random Forest and KNN. From table 5.1, AUC scores of Random Forest and KNN are respectively as follows: 57.8%. and 54% with KNN having slightly higher recall value than Random Forest.

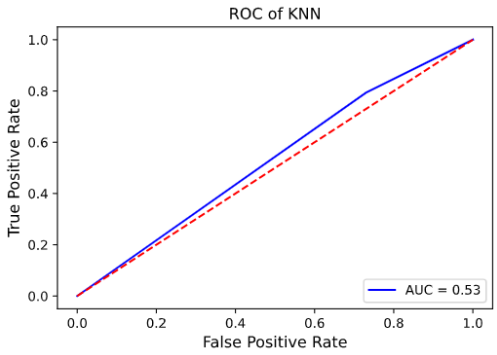

Figure 5.6: KNN - ROC Curve

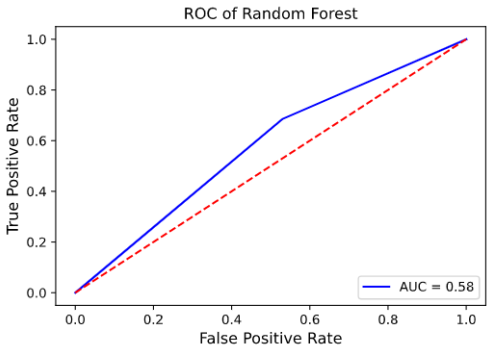

Figure 5.7: RF - ROC Curve

On further analysis, we reason that as we used KNN imputer for imputation the numbers might be inflated for KNN model and it also has a lower AUC score. (Figure: 5.6).

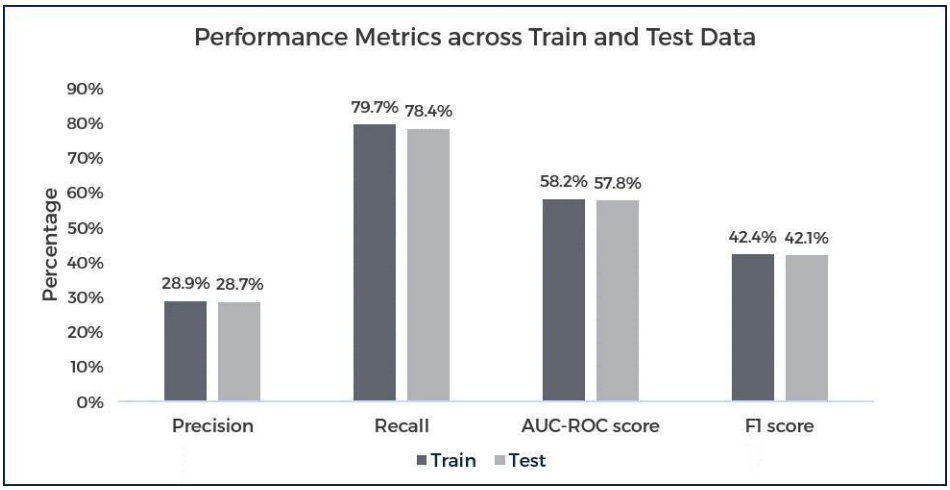

Figure 5.8: Train-Test Results of RF

From figure 5.7, not much variation is observed in the train and test score which signifies no overfitting in the model. Thus, we pick Random Forest as our champion model.

8. CONCLUSION AND SCOPE FOR FUTURE WORK

From our preliminary analysis we gained the understanding of how some of our features affected profitability of a customer and on further engineering new features from the pre-existing one, we gained quality insights. To ensure that the model does not overlearn we also concluded on keeping low multicollinearity by removing variables curving towards that aspect. Rather than highly pre-processing the data at hand which would result in the loss of original data points, we ensure that we use models that can handle multicollinearity versus the classifiers like Naïve bayes, KNN and Logistic Regression that are sensitive to multicollinearity. Ensemble classifiers like Random-Forest are quite robust to multicollinearity and outliers which works for our dataset and in this feature selection can also be avoided. Superior results were achieved using ensemble classifiers over base classifiers. 78 % of profitable customers in the target segment were captured by our model. Within this engineered framework, identifying profitable customers for mastercard companies is easier and then we are enabling to further target them with tailored and customized acquisition campaigns leading to heavy savings in primary marketing expenses and thus, getting high return on investment through strategic targeted marketing.

There is a huge further scope for this model with additional fine tuning with generating more features and obtaining new features associated to seasonality and cancellation. For the scope of this problem statement only recall evaluation metrics like AUC and recall are used but furthermore reliable and robust metrics like concordance and gains table should be utilized to assess the model to gain tremendous business impact and insights. The venture can be further advanced and scaled up by conducting a propensity analysis which does way more than just classifying if a consumer is profitable or not. A model can be developed to score consumers' propensity. Tailored marketing campaigns can be deployed according to a customers calculated propensity to purchase. An automated recommendation engine can be developed for the marketing team which proposes suitable acquisition strategies custom built and tailored uniquely tailored by analysis the attributes and propensity of the specific customer.

9. REFERENCES

1. Curto, J. D. and Pinto, J. C. (2011). “The corrected vif (cvif).” Journal of Applied Statistics, 38(7), 1499–1507.
2. Farias, F. C., Ludermir, T. B., and Filho, C. J. A. B. (2020). “Similarity based stratified splitting: an approach to train better classifiers.” CoRR, abs/2010.06099.
3. Jebb, A. T., Parrigon, S., and Woo, S. E. (2017). “Exploratory data analysis as a foun- dation of inductive research.” Human Resource Management Review, 27(2), 265–276.
4. Katrutsa, A. and Strijov, V. (2017). “Comprehensive study of feature selection methods to solve multicollinearity problem according to evaluation criteria.” Expert Systems with Applications, 76, 1–11.
5. Kaviani, P. and Dhotre, S. (2017). “Short survey on naive bayes algorithm.” Interna- tional Journal of Advance Research in Computer Science and Management, 04(1).
6. Kumari, R. and Srivastava, S. K. (2017). “Machine learning: A review on binary clas- sification.” International Journal of Computer Applications, 160(7).
7. Paul and Kumar, R. (2006). “Multicollinearity: Causes, effects and remedies.
8. Raju, V. N. G., Lakshmi, K. P., Jain, V. M., Kalidindi, A., and Padma, V. (2020). “Study the influence of normalization/transformation process on the accuracy of supervised classification.” 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT). 729–735.
9. Sechidis, K., Tsoumakas, G., and Vlahavas, I. (2011). “On the stratification of multi- label data.” Machine Learning and Knowledge Discovery in Databases, D. Gunopulos,
10. T. Hofmann, D. Malerba, and M. Vazirgiannis, eds., Berlin, Heidelberg. Springer Berlin Heidelberg, 145–158.
11. Verstraeten, G. and Van den Poel, D. (2006). “Using predicted outcome stratified sam- pling to reduce the variability in predictive performance of a one-shot train-and-test split for individual customer predictions. 214–224.
12. Wasikowski, M. and Chen, X.-w. (2010). “Combating the small sample class imbalance problem using feature selection.” IEEE Transactions on Knowledge and Data Engineer- ing, 22(10), 1388–1400.
13. Zhang, F. (2006). The Schur complement and its applications, Vol. 4. Springer Science & Business Media.