Customer Personality Analysis

Team 13 Belle Tan, Bennur Kaya, Marcus Pradel, Masoumeh Shekari, and Vishal Chauhan

University of Mannheim

Abstract. Customer Personality Analysis is a detailed analysis of a company's ideal customers. It helps a business to better understand its customers and enables them to modify products according to the specific needs, behaviours and concerns of various types of customers.

1 Data

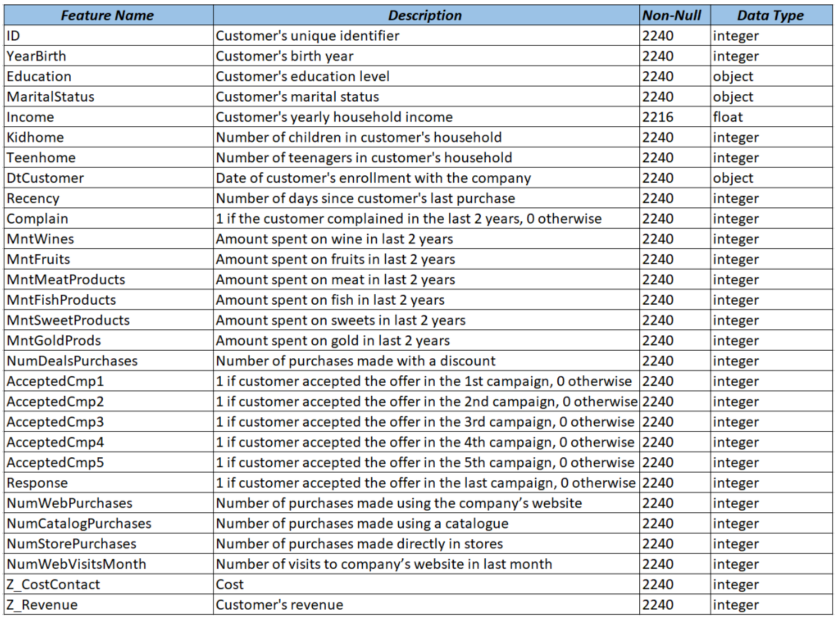
1.1 Introduction

The relationship between businesses and customers is constantly evolving. In this era, which is entirely driven by technology, there are numerous channels through which customers can be reached. Personalised customer services aim to cater the exact needs and wants of the customer. Delivering the best customer experience is critical for businesses to remain profitable, especially in the highly competitive markets.The knowledge and insights which are gained by using data mining techniques enable businesses to serve customers according to their preferences through targeting their actual needs. On top of that, customer personality analysis helps businesses to modify their product based on their target customers from varied types of customer segments. Models built by using customer behavior data for future marketing actions can save time and reduce company's marketing expenses.

This analysis is done to better understand customers and offer ideas to modify marketing campaigns and products to the specific needs, behaviours, and concerns of different types of customers. Moreover, another purpose of this analysis is to learn the behavior of customers who benefited from past campaigns, and to build various models based on that. This will enable the company to take actions that will increase the success of future campaigns and reach potential customers.

1.2 Data Understanding

In this project, the Customer Personality Analysis data set which is provided by Dr. Omar Romero-Hernandez will be used [1]. The data set contains socio-demographic features of about 2240 customers who were contacted in one file. There are 29 attributes and they indicate the personal information of customers, product information, promotion information and where the product was purchased.



Feature Name	Description	Non-Null	Data Type
ID	Customer's unique identifier	2240	integer
YearBirth	Customer's birth year	2240	integer
Education	Customer's education level	2240	object
MaritalStatus	Customer's marital status	2240	object
Income	Customer's yearly household income	2216	float
Kidhome	Number of children in customer's household	2240	integer
Teenhome	Number of teenagers in customer's household	2240	integer
DtCustomer	Date of customer's enrollment with the company	2240	object
Recency	Number of days since customer's last purchase	2240	integer
Complain	1 if the customer complained in the last 2 years, 0 otherwise	2240	integer
MntWines	Amount spent on wine in last 2 years	2240	integer
MntFruits	Amount spent on fruits in last 2 years	2240	integer
MntMeatProducts	Amount spent on meat in last 2 years	2240	integer
MntFishProducts	Amount spent on fish in last 2 years	2240	integer
MntSweetProducts	Amount spent on sweets in last 2 years	2240	integer
MntGoldProds	Amount spent on gold in last 2 years	2240	integer
NumDealsPurchases	Number of purchases made with a discount	2240	integer
AcceptedCmp1	1 if customer accepted the offer in the 1st campaign, 0 otherwise	2240	integer
AcceptedCmp2	1 if customer accepted the offer in the 2nd campaign, 0 otherwise	2240	integer
AcceptedCmp3	1 if customer accepted the offer in the 3rd campaign, 0 otherwise	2240	integer
AcceptedCmp4	1 if customer accepted the offer in the 4th campaign, 0 otherwise	2240	integer
AcceptedCmp5	1 if customer accepted the offer in the 5th campaign, 0 otherwise	2240	integer
Response	1 if customer accepted the offer in the last campaign, 0 otherwise	2240	integer
NumWebPurchases	Number of purchases made using the company's website	2240	integer
NumCatalogPurchases	Number of purchases made using a catalogue	2240	integer
NumStorePurchases	Number of purchases made directly in stores	2240	integer
NumWebVisitsMonth	Number of visits to company's website in last month	2240	integer
Z_CostContact	Cost	2240	integer
Z_Revenue	Customer's revenue	2240	integer

Table 1. Description of Features

As it can be seen from Table 1, there are 2240 observations in the data set. That indicates, *Income* variable has 2216 observations and 24 missing values. On the other hand, *Dt_Customers* is in object class. It should be converted to DateTime format. Furthermore, there exists categorical features such as *Education* and *Marital_Status* Variables. Later, they will be converted to numeric class.

Looking at the summary of the data, it is seen that the *Z_CostContact* and *Z_Revenue* variables have fixed values. Therefore, these are excluded from the data set. Moreover, the oldest registration date is 2014-12-06 while the oldest registration date is 2012-01-08. Furthermore, *Marital_Status* has 7 different class as Married, Divorced etc. Likewise, *Education* has 5 different group as Graduation, PhD, Master, 2n Cycle and Basic. Also, the mean value of the *Income* variable (52247.25) is very close to its median value (51381.50). Therefore, mean imputation for missing data is done for *Income* attribute. After solving the missing value problem, it is confirmed that there is no duplicate in the data set. As a final step, the class of the *DtCustomer* variable is converted from object to DateTime.

1.3 Data Preprossessing and Exploratory Data Analysis

Data Preprossessing

The analysis has undergone data preprocessing to ensure and improve the performance by modifying the existing variables and creating new variables. First, the *Customer_Date* is created which indicates for how long customers have been registered. That means, *Customer_Date* is a numeric value where 0 represents the day is today. Also, *Kids* variable representing the total number of children in the house was created by using the *Kids_Home* and *Teen_Home* variables. Then, *Expenses* variable is created which indicates the total amount spends in last 2 years. Besides, *Accepted* variable representing the total number of accepted campaigns is created. Moreover, *Purchase* variable representing the number of purchases made using channels is created. Also, *Age* variable which indicates the age of the customer is created by using YearBirth attribute. Afterwards, *Family_size* variable representing the members in the house is created by using the *Marital_Status* and *Kids* variables. Also, since *Marital_Status* has seven different groups. Absurd, Alone and Yolo are changed to Single class. On the other hand, since 2nd level Professional is counted as Master's Program, 2n Cycle group in *Education* variable is converted to Master. Then, the names of the columns have been changed to easily identify the features. Lastly, *ID*, *Year_Birth*, *Kid_home*, *Dt_Customer* and *Teen_home* variables are excluded since the variables created by using them will be used instead in the next steps.

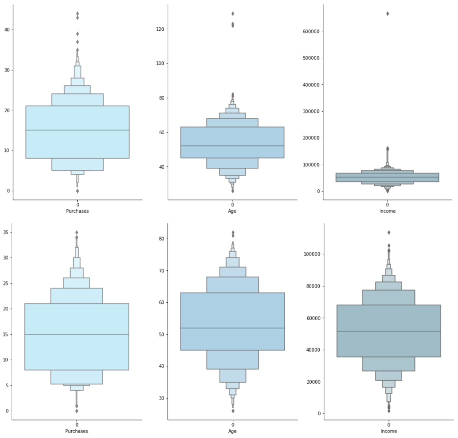


Fig. 1. Distributions before/after Outlier Elimination

In the next step, the distributions of the variables were examined. Particularly *Purchases*, *Age* and *Income* seem to have outlier problems. Assuming the data is Normally distributed, outliers are removed by standard deviation technique. 99.7% of the data falls within three standard deviations of the mean and it helps to eliminate most of the outliers. As in the box plots in Figure 1, most of the outliers were eliminated. After the outlier elimination, the number of observations decreased from 2240 to 2227. Also, the distribution of the *Income variable*, which is right skewed before the outlier eliminations, approaches normality.

Exploratory Data Analysis

The Exploratory Data Analysis (EDA) is conducted to discover hidden patterns, relationships and allow for a better understanding of the data set. First, when looking at correlation between variables, there are various high correlation values. For example, it is seen that *Income* has strong relationship with *Wines*, *Meat*, *Expenses*, *Purchases* with a correlation value higher than 0.7. This might indicate that customers with a higher income tend to consume more wine and meat and are more likely to spend more on purchases. In addition, logically, it can be thought that product consumption increases as the number of individuals in the family increases. However, there are negative correlation values between *Family_Size*, all the product groups (Wines, Fruits, Meat, Fish, Sweets, Gold), and *Income*. Furthermore, the most common age range in the data is 40-60 and the most common age is 46. Also, graduation which represents university graduation is by far the most common group in the *Education* variable. The least common group is the basic group, which represents the basic education level. When the distribution of *Income* by *Education* is examined, the incomes of customers with basic education level are significantly lower, although there are generally similar income levels between other groups. Also, the proportion of people who are married or have a partner is %64.4, while the proportion of people who are divorced, widow or single is %35.6 in the data set. By looking at the distribution of income by marital status, single people have the widest income range, while widows have the narrowest income range. Finally, *Complain* variable will not be included in the data set as only 20 out of 2227 customers complained in the last 2 years.

In the last stage of the EDA, the relationship between the *Response* attribute, which will be used in the classification models, with other attributes is examined. As seen in figure 2, when the income level increases, the number of customers that accepted the offer in the last campaign increases as well. Furthermore, as the number of days after a customer's enrollment increases, the number of people benefiting from the last campaign increases. New members tended to be suspicious about the last campaign. In addition, although the number of customers whose *Marital_Status* is either married or together in the data set is almost twice as many the total of other groups (single, widow and divorced), the number of customers that accepted the offer in the last campaign is higher

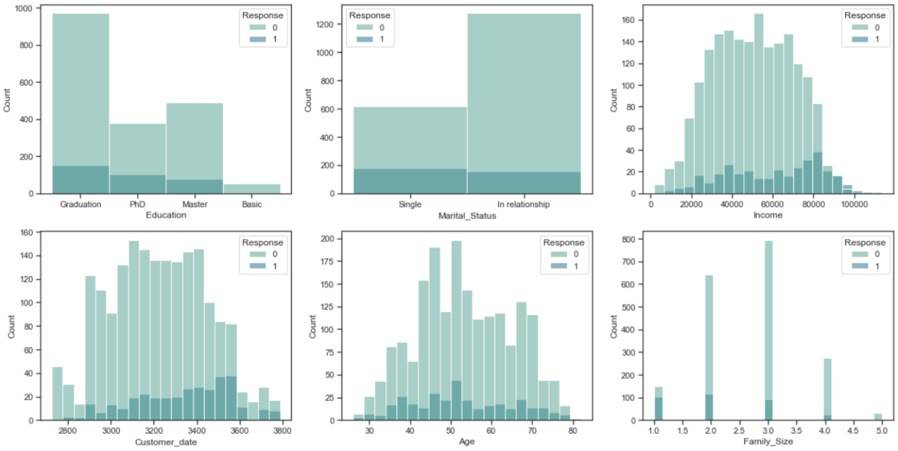


Fig. 2. Distributions by *Response*

for the group which have no partner in their life. Furthermore, as the number of people in the family increases, the number of people benefiting from the campaign decreases. Lastly, contrary to other product groups, it is observed that the number of people benefiting from the campaign increased as the amount spent on gold increases in the last 2 years.

Encoding is applied to the categorical variables *Education* and *Marital_Status* so that the data with converted categorical values can be provided to the models to give and improve the predictions.

1.4 Data Mining

Machine Learning Approaches

Clustering Analysis is used to identify groups of customers who have similar shopping behaviour. It helps to explore the data to identify intrinsic patterns. K-Means Clustering and DBSCAN methods is implemented in the analysis. It is aimed to improve classification models' performance using the clustering attribute. After Clustering method, Classification models are built to classify if a customer accepted the offer in the last campaign (Response column) by using data from the last 5 campaigns. These models can provide new insights about potential customers for the future marketing campaigns. K-nearest neighbours, Support Vector Machines, Naive Bayes and Random Forest are the classification methods which are implemented.

Clustering

In clustering, K-Means and DBSCAN are implemented. In order to find the optimal number of clusters in K-Means Clustering, the Elbow method is used. Based on the method, the optimal number of K is 5, where Within-Cluster Sum of Square (ESS) shapes the elbow. As another approach, Principal Component Analysis (PCA) method is used to reduce the dimensionality of the large data set by transforming a large set of variables into a smaller one while still containing most of the information. After using the PCA method with keeping 75% of the variations, the number of clusters reduced to 4. In the classification part, the model with the highest performance is aimed by using the both cluster groups. In DBSCAN clustering, due to high dimensionality of the data set, the number of noise points remained very high. After applying the PCA method, the quality of clustering did not improve significantly. Hence, the clustering outcomes of K-Means method has been used in further classification algorithms.

Classification

As the first classification method, K-nearest neighbours (KNN) is used. The aim is to predict the correct class of the test data by calculating the distance between the test data and all the training points where K-value indicates the count of the nearest neighbours. Firstly, standardisation is made to manipulate the data of all variables except the target variable. Next, all independent data features are collected into the X data-frame while the target variable is collected into the y data-frame. Following that, the data sets are split to train and test data sets. Since the *Response* variable is imbalanced with 1893 customers who did not accept the offer and 334 customers who accepted it in the last campaign, ”stratify” parameter is used in train-test data splitting. This parameter makes a split such that the proportion of values in the sample produced will be the same as the proportion of values provided to parameter stratify. After splitting the data, two-thirds of the data is used for training and the remaining for testing purposes. SMOTE is then applied to balance the train data set. The classifier model is built by initialising K between 1 and 15. The best model score is captured at K = 2. To be able to improve the model performance, clustering attribute is added to data set and the KNN analysis process is repeated. As the final KNN model, PCA is implemented before clustering to test if PCA is able to improve the model.

As the second classification method, Support Vector Machines (SVM) was used. For this model, the preprocessed data was split into training and testing sets with the response column as the labels. Due to high imbalance in the labels, the training data was then balanced. For classification, initially out of the box SVM model was trained, where C is 1, gamma is set to scale and kernel used is radial basis function (RBF), which performed well. After testing preliminary settings, in order to optimise the model, different values of C, gamma, and additional kernels along with RBF, such as linear, polynomial, and sigmoid were tried. Using the accuracy score while trying out different parameters, it was found that the model could be optimised further by setting C as 100 and gamma as 0.001, however RBF kernel was still found to be the best kernel for this data. Clustering techniques such as K-means were also leveraged to improve model performance. Using clusters as an additional feature further improved the performance and resulted in the optimized SVM with Kmeans clustering where C was 5, gamma was scale and kernel was RBF, to be the best model to predict both labels. To visualise and understand how the model arrived at its predictions, a decision surface was plotted. Since there were 30 features including clusters, it was impossible to visualise it as it was. To overcome that, the number of features were reduced to two using PCA.

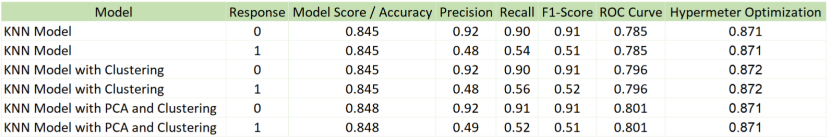
As the third classification method, Naive Bayes is applied. To begin with, the default parameters of Bernoulli Naive Bayes is applied in the classification model. The model performed well overall with an average weighted F1-score of 0.84. After the preliminary testing of Naive Bayes, hyper-parameter tuning is performed to find the optimal parameters to this classification model, primarily the parameter "alpha". It is found that this Naive Bayes model can be optimised with alpha set as 10.0. However, after applying the optimal parameters, the performance of the classification model did not improve. Next, the clustering attribute was added to the data set to improve the classification model. After doing so, it can be observed that the overall performance of the model had dropped with the average weighted F1-Score decreasing from 0.84 to 0.74.

As the final classification method, Random Forests are applied. The Random Forest method is characterised by its short training and building time as well as its effectiveness with large data sets. The data set preprocessed in the previous steps is being utilised, which is also divided into training and testing set. Although random forests are also applicable to unbalanced data, for comparability the training data set is balanced. For the creation of the random forests, the method Scikit-learn is used. After creating a first decision tree and random forest, a baseline for the default values is achieved. The next step is to improve the model using hyper-parameter tuning. To find the ideal parameters, GridSearchCV can be use to approximate the best possible model even more precisely. Over several cycles more and more precise values can be obtained. This was achieved with fine adjustments of the parameters. But even more parameter values can be tested with the help of RandomizedSearchCV. This function implements randomly generated values that are within the predefined frame for the parameters. This allows many variations to be gone through quickly. The importance of the individual attributes can now also be calculated. This calculated feature importance is relevant for the creation of the model and provides context for the interpretation of the data as well. For the model creation the aim is to determine whether fewer attributes could lead to a better result. The XG-

Boost method can also be used for the creation of random forests with the help of parallel tree boosting. It stands for “Extreme Gradient Boosting”. Here, the optimal parameters for the creation are determined like with the two previous functions, but with the help of boosted tree algorithm.

1.5 Results and Evaluation

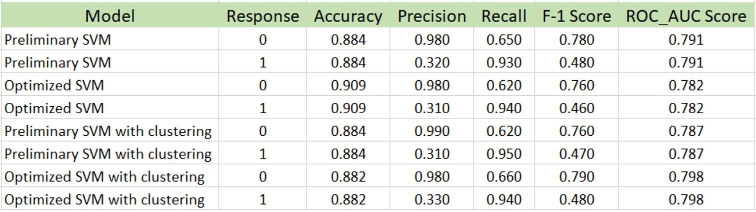
For the KNN method, the best model result is captured as 0.845 with k equals to 2 when clustering attribute and PCA are not added into model. That means, 84 out 100 customers are correctly predicted with a ratio of 0.84. Also, to be able to have better insights about the performance of the model, precision, recall and f1-score results are also checked. As it can be seen in Table 2, these metrics gives lower values for the customers who benefit from the last campaign. After adding clustering attribute to data set, the performance metrics didn't improve. After applying PCA, the performance metrics slightly improved. The model performance after hyper-parameter optimisation are similar for all the models. The area under ROC Curve is the highest when PCA and Clustering are applied to the data set.



Model	Response	Model Score / Accuracy	Precision	Recall	F1-Score	ROC Curve	Hypermeter Optimization
KNN Model	0	0.845	0.92	0.90	0.91	0.785	0.871
KNN Model	1	0.845	0.48	0.54	0.51	0.785	0.871
KNN Model with Clustering	0	0.845	0.92	0.90	0.91	0.796	0.872
KNN Model with Clustering	1	0.845	0.48	0.56	0.52	0.796	0.872
KNN Model with PCA and Clustering	0	0.848	0.92	0.91	0.91	0.801	0.871
KNN Model with PCA and Clustering	1	0.848	0.49	0.52	0.51	0.801	0.871

Table 2. KNN Performance Metrics

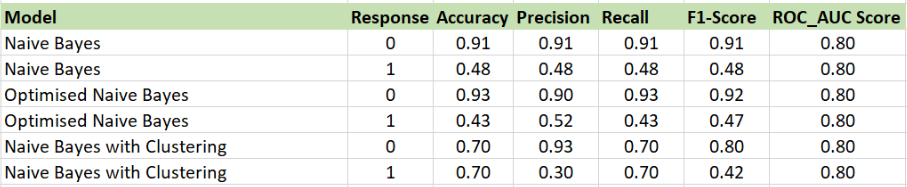
For SVM classification method, since predicting 'Responded' label is the aim of our analysis, optimized SVM model with Kmeans clusters as additional feature where C = 5, gamma = 'scale', and kernel = 'RBF' performed best as can be seen in the table below. This model also performs better than other SVM models to predict 'Did not respond' label as well which can be seen in the table below.



Model	Response	Accuracy	Precision	Recall	F-1 Score	ROC_AUC Score
Preliminary SVM	0	0.884	0.980	0.650	0.780	0.791
Preliminary SVM	1	0.884	0.320	0.930	0.480	0.791
Optimized SVM	0	0.909	0.980	0.620	0.760	0.782
Optimized SVM	1	0.909	0.310	0.940	0.460	0.782
Preliminary SVM with clustering	0	0.884	0.990	0.620	0.760	0.787
Preliminary SVM with clustering	1	0.884	0.310	0.950	0.470	0.787
Optimized SVM with clustering	0	0.882	0.980	0.660	0.790	0.798
Optimized SVM with clustering	1	0.882	0.330	0.940	0.480	0.798

Table 3. SVM Performance Metrics

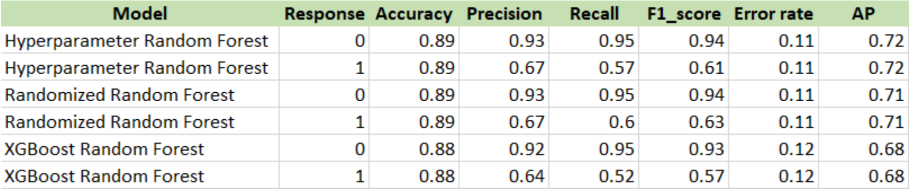
For the Naive Bayes method, both the preliminary testing of the model and the optimised model with hyper-parameter tuning yielded the best results. It did well in classifying the customers' response to the campaign overall with an average weighted F1-score of 0.84. However, the results of the model's performance in identifying customers that responded to the campaign is less than ideal with an F1-score of 0.48. All these can be seen in the table below.



Model	Response	Accuracy	Precision	Recall	F1-Score	ROC_AUC Score
Naive Bayes	0	0.91	0.91	0.91	0.91	0.80
Naive Bayes	1	0.48	0.48	0.48	0.48	0.80
Optimised Naive Bayes	0	0.93	0.90	0.93	0.92	0.80
Optimised Naive Bayes	1	0.43	0.52	0.43	0.47	0.80
Naive Bayes with Clustering	0	0.70	0.93	0.70	0.80	0.80
Naive Bayes with Clustering	1	0.70	0.30	0.70	0.42	0.80

Table 4. Naive Bayes Performance Metrics

For the random forests method, the result for the hyper-parameter tuned, the randomised and the XGBoost models are pretty close, with only slight performance differences. All models have a accuracy between 0.88 to 0.89 with a precision between 0.92 to 0.93. The best model for the correct prediction of Response=1 is the randomised model with a f1-Score of 0.63, which is not the anticipated result.In a real life application, this model would have missed 27 customers in the test data of 446 customers. Customer are too often categorised as negative. The parameters found by the randomised function, which deliver the best results, are: {'n_estimators': 1600, 'min_samples_leaf': 1, 'max_depth': 60} While hyper-parameter and randomised where really close the XGBoost model came in last with it performing therefore slightly worse. But as the end result, all three models are pretty close to each other and would need more data for a correct categorisation of the positive customers. For the feature importance, it becomes obvious that the attributes Recency and Family_Size are the most relevant for the customers' decision. However, the random forest model requires



Model	Response	Accuracy	Precision	Recall	F1_score	Error rate	AP
Hyperparameter Random Forest	0	0.89	0.93	0.95	0.94	0.11	0.72
Hyperparameter Random Forest	1	0.89	0.67	0.57	0.61	0.11	0.72
Randomized Random Forest	0	0.89	0.93	0.95	0.94	0.11	0.71
Randomized Random Forest	1	0.89	0.67	0.6	0.63	0.11	0.71
XGBoost Random Forest	0	0.88	0.92	0.95	0.93	0.12	0.68
XGBoost Random Forest	1	0.88	0.64	0.52	0.57	0.12	0.68

Table 5. Random Forests Performance Metrics

21 of the 29 attributes for a culminated importance of 95%. The random forest can therefore not be broken down to only a handful of attributes.

1.6 Conclusion

This analysis was done to better understand customers and offer ideas to modify marketing campaigns and products to the specific needs, behaviours, and concerns of different types of customers. Data preprocessing was applied to ensure and improve the performance for the model. Furthermore, in Exploratory Data Analysis, new variables are also created while making changes to the existing variables. Clustering Analysis is used to identify groups of customers who have similar shopping behaviour and it is aimed to develop classification models using the clustering attribute. Classification models as KNN, SVM, Naive Bayes and Random Forest were built to provide new insights about potential customers for the future campaigns. After experimenting with different models, the different performance metrics such as accuracy, precision, recall, F1 score and ROC Curve were checked and compared. It was found that Random Forest has the best performance in identifying customers who responded to the last campaign. The analysis can be further improved by collecting more data regarding the customers who accepted the last campaign as this would solve the imbalanced data set problem. Moreover, different data sets such as the product preferences of customers in their baskets might improve the analysis and it open doors to use different classification techniques such as association analysis.

1.7 References

References

1. Romero-Hernandez, D., 2022. Customer Personality Analysis. [online] Kaggle.com. Available at: https://www.kaggle.com/datasets/imakash3011/customer-personality-analysis [Accessed 23 May 2022].