KOLEJ UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF COMPUTING AND INFORMATION TECHNOLOGY

Assignment

BMCS2114 MACHINE LEARNING

2020/2021

Student's name/ ID Number : Yeoh Yi Hang / 2008892

Student's name/ ID Number : Ooi Yen Chun / 2008884

Student's name/ ID Number : Tan Yan Xue / 2008889

Programme : Bachelor of Computer Science in Data Science

Tutorial Group : RDS2G2

Date of Submission to Tutor : 18 April 2021

Project Title 1

Introduction 1

Business Problem 1

Analytics Goal 1

Project Plan 2
- Part A 2
- Part B 3

Task Allocation 4

Methodology 4
- Data Understanding 4
- Part A 5
  - Check Data Quality 5
  - Simple Data Analysis 5
  - Data Preprocessing 5
  - Identify Suitable Classifiers 5
  - Identify Suitable Scalers 6
  - Evaluate performance of classifier 6
- Part B 9
  - Data Transformation 9
  - Evaluate the performance of shortlisted classifiers 9
  - Grid Search for Parameter Tuning 9

Result 10

Project Title

Marketing Campaign Response Prediction

Introduction

The dataset chosen is related to the direct marketing campaign of a Malaysia hypermarket - **Giant**. The marketing campaigns were based on customer's buying patterns and also their willingness to participate when confronted. For a more intuitive analytic, part of the customer personal details such as household income and their response towards past marketing campaigns are also included as factors. Often, more catalogs and flyers have to be printed and sent out to every member for every marketing campaign as the company is unable to predict if the customer will actually visit the store ('yes') or not ('no').

The dataset consists of **2240** observations and **29** inputs (features).

Business Problem

There has been an increase in business expenses for Giant ever since the infamous pandemic, Covid-19, hit the global market and they would like to know what actions to take to reduce their expenses. After a series of serious investigations they found out that the root cause is that the excessive spending in marketing campaigns usually does not guarantee an acceptable return. This is mainly caused by lack of response from the customers especially those registered as members who are given higher priority to receive new marketing campaigns. Knowing that they can boost the efficiency of marketing of a marketing campaign by increasing response rate so that the company can spend less money on printing marketing materials. In addition, Giant also holds a better chance to persuade customers into buying their new offerings such as new microwaves or televisions to further increase their revenues. As a result, Giant hypermarket would like to identify existing members that have higher chances to respond to their next marketing campaign and focus marketing efforts on such potential customers.

Analytics Goal

A classification approach to predict which clients are more likely to respond to an offer for a product or a service. In the dataset, 0 in "response" means the specific client will not respond to the advertisement, else the client will respond if "respond" labelled as 1.

Project Plan

Part A

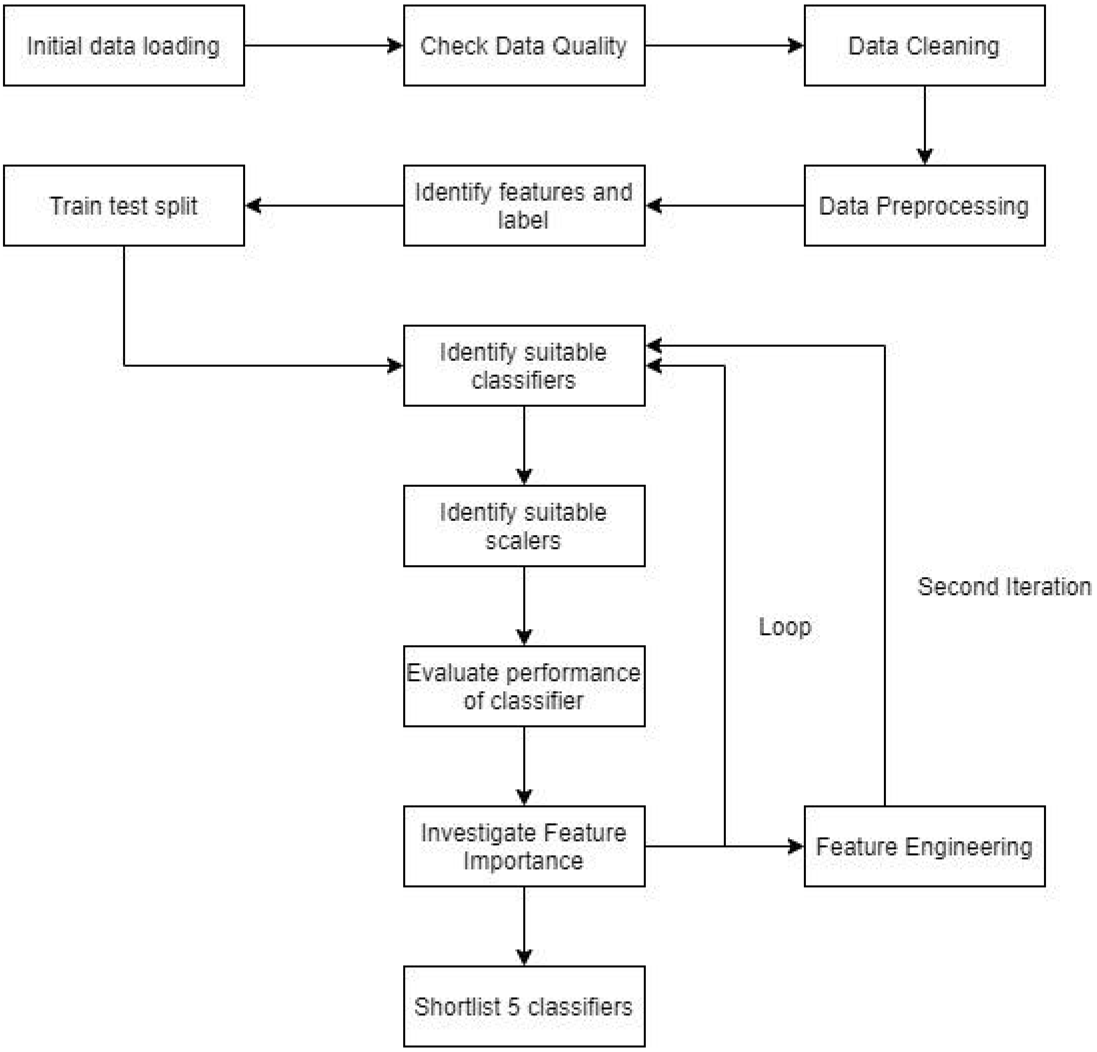


Figure 1.1: Project flow chart for Part A

Part B

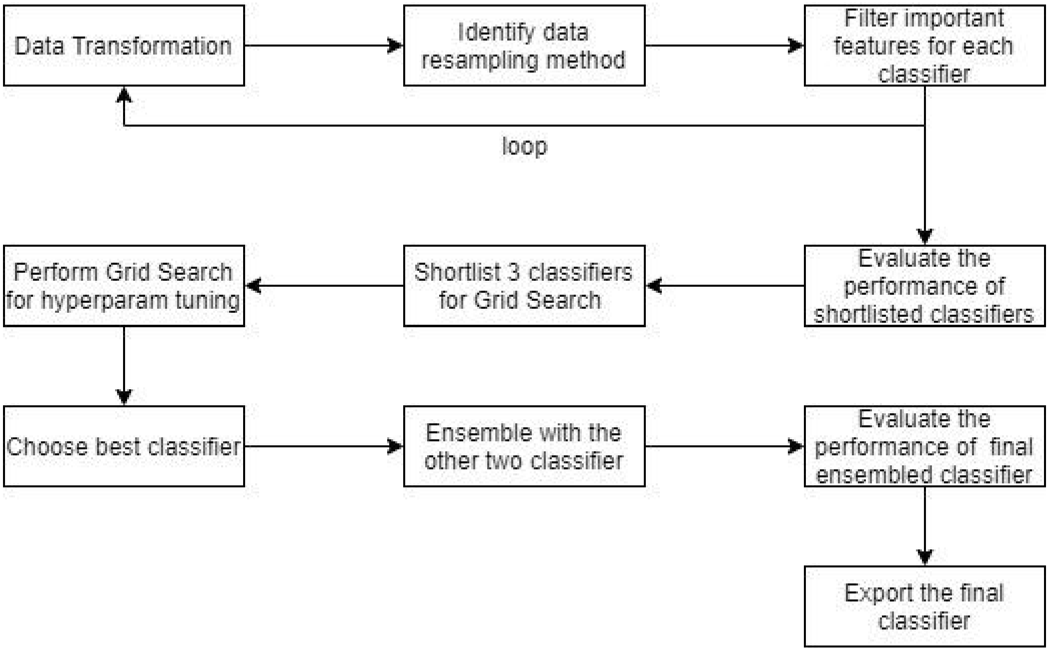

Data Transformation
Identify data resampling method
Filter important features for each classifier
loop
Evaluate the performance of shortlisted classifiers
Shortlist 3 classifiers for Grid Search
Perform Grid Search for hyperparam tuning
Choose best classifier
Ensemble with the other two classifier
Evaluate the performance of final ensembled classifier
Export the final classifier

Figure 1.2: Project flow chart for Part B

Task Allocation

Member	Tasks
Yeoh Yi Hang	Data Preparation, Transformation and Setup looping for pipeline
Ooi Yen Chun	Setup model evaluation, Grid Search and Ensemble classifier
Tan Yan Xue	Visualize model evaluation performance matrix, Sort and Filter model performance matrix

Methodology

Data Understanding

Upon understanding the requirements from the client, a set of customers' data is requested and retrieved from the database administration department. Particularly the data retrieved is related to the information of the customer that possesses membership under Giant hypermarket. In order to protect the privacy of the customer, we have only managed to receive a sampled dataset from the big data warehouse of the company. For the data analysis to perform at its maximum performance, the data team at Giant aggregated a diversity of data that came from a different database from different branches across various states in Malaysia. Hence, we can analyze the data of customers from different areas and states from just a single file distributed by Giant. As for the documentation for the dataset itself, the team also provided us with a booklet as a data dictionary to help us to understand the meaning of each attribute and attribute value in business terms. For example, the column "DtCustomer" shows date of customer's enrolment with the company while the column "Kidhome " shows number of small children in customer's household. We are to combine our knowledge from business understanding and the info provided by the booklet to perform the rest of the data mining steps. If we are confused during these steps, we were to go back to business understanding as the transformation from business to data understanding is not linear but cyclic.

Part A

Check Data Quality

Upon checking for potential missing values inside the dataset, we can see that the column “Income” does contain "NULL" values which indicate that we have to impute those missing values. For testing purposes, we will use the “ffill” method to impute, which propagates the last valid observation forward to the next valid backfill. After that, we check the number of rows and columns inside the dataset. A total of 2240 rows of data is retrieved along with 28 columns of features plus 1 label for customer response. Since the goal of our data mining is to identify the characteristics of potential customers that would respond to future marketing campaigns, we decided to not include all the sensitive information of the customer such as the customer id and the revenue gained from the customer. This ensures the data privacy of the customer is protected throughout the process of data mining.

Simple Data Analysis

Furthermore, we also did some descriptive analysis for the dataset to show us the mean, mode and median. In this case, mean is able to show us if there is any extraordinary value inside the data set that might affect us in the process of data mining. The mode provides us an insight on what is the most common value for that particular column while the median tells us the approximate middle value of one column. With the help of boxplot, we can visualize the outlier and verify the result after correction.

Data Preprocessing

In order to make computers understand the dataset, we are to convert all the strings to integers. While data preprocessing can be related to many sophisticated methods such as One-hot encoding and binning, only data encoding applies to this data set. Data encoding is executed to convert the String value of Yes and No value to 1 and 0 respectively is done to make the data more suitable for training purposes.

“Marital Status” columns represent the marital status of customers. Due to its categorical nature, we decided to encode it and spread it into multiple columns. Each column only contains 1 and 0. Besides that, Label Encoder is deployed to encode the Education column as it is ordinal in nature.

Identify Suitable Classifiers

After the pre-processing phase of the data is completed, we will move on to the next step, which is to identify amongst many classifiers we have selected, which can produce the most accurate results.

The results of data understanding show that we are going to perform classification from our dataset. This means that the data should be linearly separable. Due to the time limitation for this project, we decided to go with the modeling technique that consumes less time while providing an acceptable accuracy.

In total we have chosen 20 classifiers and the process is to reduce them to 5. To aid us in evaluating each of the individual classifiers, we have created a list of models in which we can automatically loop into a function which helps calculate the various metrics score of the classifiers.

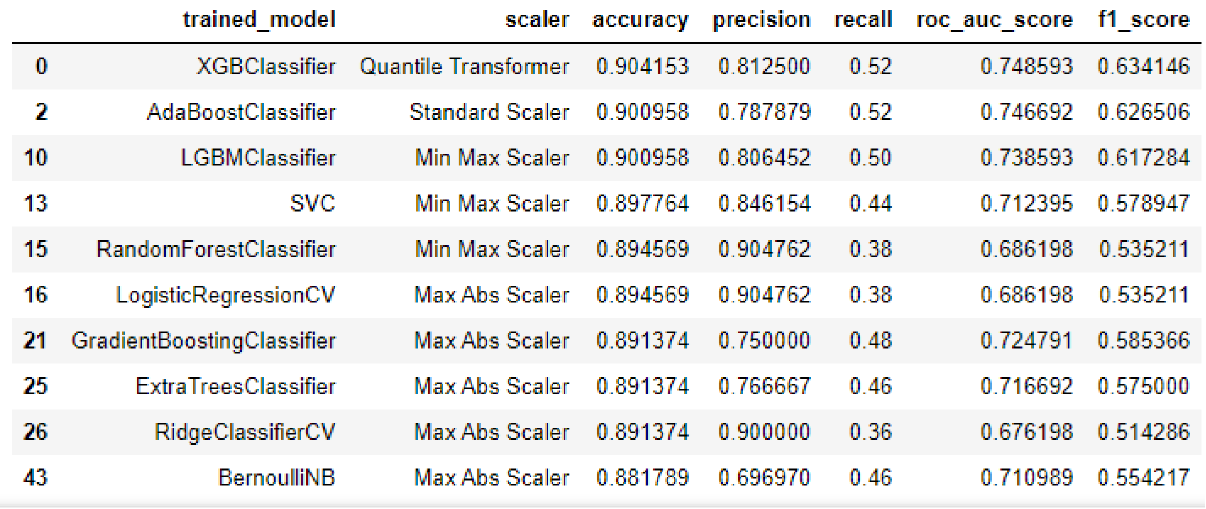
Identify Suitable Scalers

Normalization involves rescaling of attributes value to have a mean of 0 and a standard deviation of 1 (unit variance). This process is performed using scikit learn to ensure the objective function of every machine learning algorithm involves lower biases on training and feature selection process. The classifiers each will also be looped and tested with 5 different scalers which are as follows, ‘Standard Scaler’, ‘Min Max Scaler’, ‘Max Abs Scaler’, ‘Robust Scaler’ and ‘Quantile Transformer’.

Evaluate performance of classifier

The whole dataset is splitted into a training set and testing set. The training set will be further split into two sets by the KFold so that cross-validation can be performed for an accuracy score.

In the end we will generate a series of graphs ranking each of the classifiers on their accuracy, precision, recall, f1-score, specificity,g-mean, mean square error, r2, log loss and roc-auc score. A much detailed excel file containing all of the accuracy, recall, precision and f1-score from the combinations of the classifiers and scalers is also produced for a much easier view of all of the results. In the end, we have selected 5 models which are the XGB Classifier, LGBM Classifier, SVC, Random Forest Classifier and Logistic Regression CV. The AdaBoost Classifier was omitted from the final 5 as its result was similar to the XGB classifier. The emphasis on accuracy and recall was due to the fact, we wish to reduce the rate of which we misidentified potential customers as unresponsive and as such costing the hypermarket in potential earnings.



	trained_model	scaler	accuracy	precision	recall	roc_auc_score	f1_score
0	XGBClassifier	Quantile Transformer	0.904153	0.812500	0.52	0.748593	0.634146
2	AdaBoostClassifier	Standard Scaler	0.900958	0.787879	0.52	0.746692	0.626506
10	LGBMClassifier	Min Max Scaler	0.900958	0.806452	0.50	0.738593	0.617284
13	SVC	Min Max Scaler	0.897764	0.846154	0.44	0.712395	0.578947
15	RandomForestClassifier	Min Max Scaler	0.894569	0.904762	0.38	0.686198	0.535211
16	LogisticRegressionCV	Max Abs Scaler	0.894569	0.904762	0.38	0.686198	0.535211
21	GradientBoostingClassifier	Max Abs Scaler	0.891374	0.750000	0.48	0.724791	0.585366
25	ExtraTreesClassifier	Max Abs Scaler	0.891374	0.766667	0.46	0.716692	0.575000
26	RidgeClassifierCV	Max Abs Scaler	0.891374	0.900000	0.36	0.676198	0.514286
43	BernoulliNB	Max Abs Scaler	0.881789	0.696970	0.46	0.710989	0.554217

Table 2.1 - Top 10 Best Performing Models Grouped By Classifier Name Based On Accuracy, Recall and ROC-AUC Score.

To determine the best features of each of the classifiers, we have also created another loop which will utilize YellowBrick library's FeatureImportances function to generate a chart containing each of the model's Top 8 features, which we will record down later for uses during the ensembling phase of our final selected 3 models.

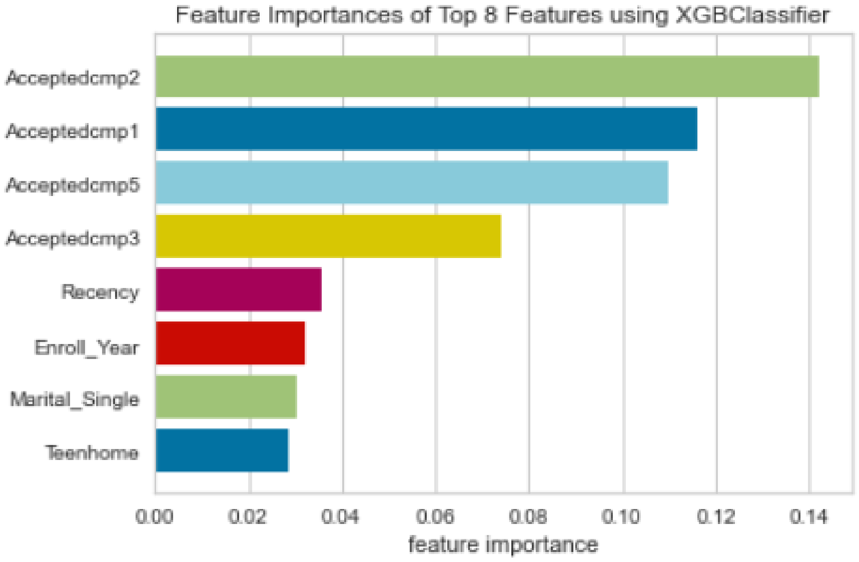


Figure 2.1 - Feature Importances of Top 8 Features using XGB Classifier

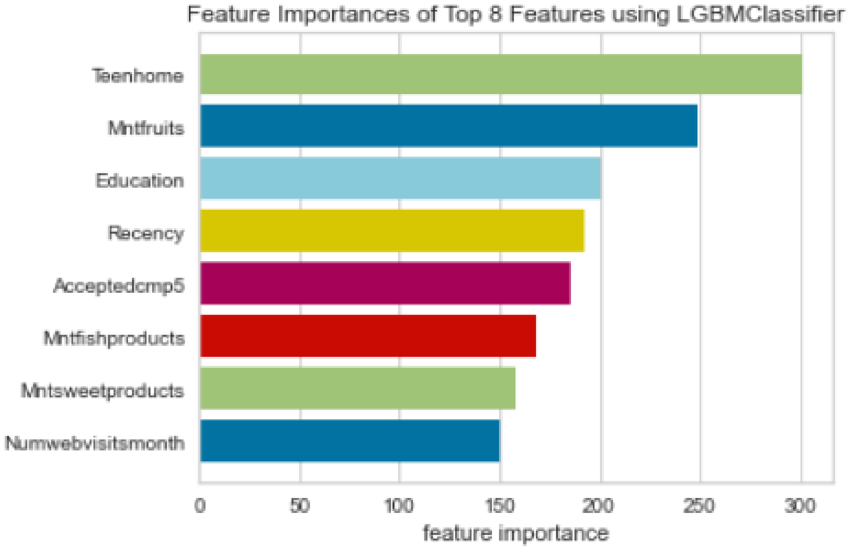


Figure 2.2 - Feature Importances of Top 8 Features using LGBM Classifier

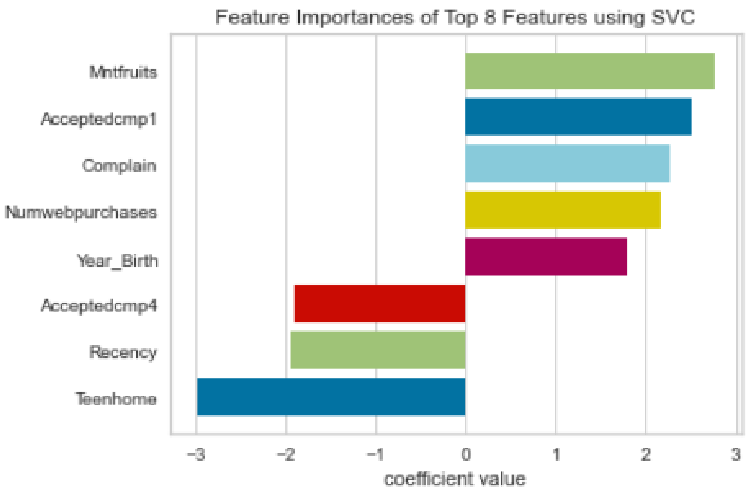


Figure 2.3 - Feature Importances of Top 8 Features using SVC

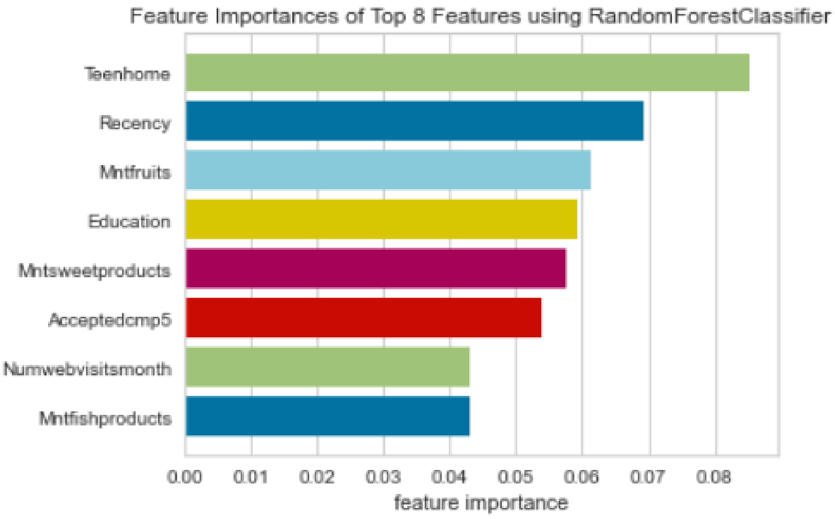


Figure 2.4 - Feature Importances of Top 8 Features using Random Forest Classifier

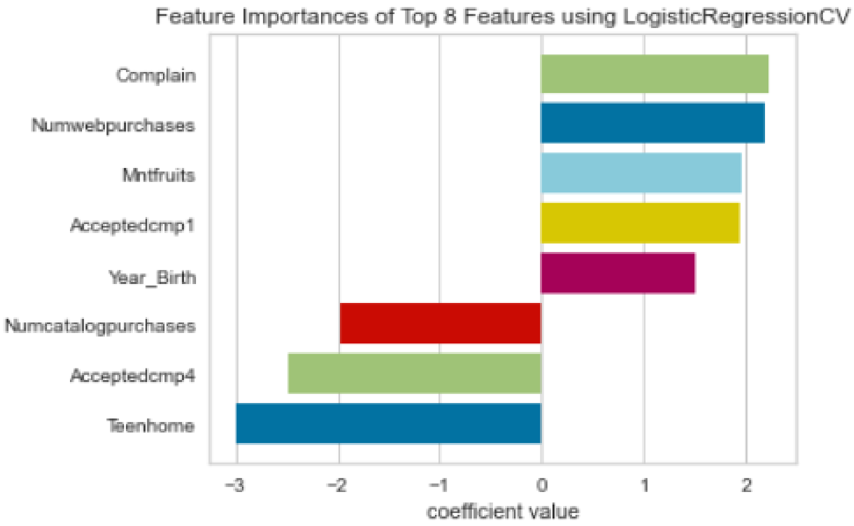


Figure 2.5 - Feature Importances of Top 8 Features using Logistic Regression CV

Part B

Data Transformation

We have chosen a few data transformation techniques to be part of the hyperparameter tuning procedure. For example, the Birth year of the customer can be further transformed into age, giving us a new perspective on the dataset. Each data transformation technique is to be looped to give different combinations so that we can deduct which data transformation technique works the best for the given dataset.

The column processing method can be detailed as followed. The enrolment date will be processed into either days only or day, month and year value. Then, the null value of the income will be treated by using KNN imputation, forward fill, backward fill, mean, median or directly drop the column. Next, if the income column still stays in the data, it will either not be processed, binned, normalized, or binned and normalized.

In order to speed up the training process and produce a better classifier, we decided to pick only 8 essential features for each classifier based on the results obtained from Part A - Feature Importance Ranking chart. Each data is also looped with different scalers so that we can see which scaler suits the most to a given combination.

Due to the imbalance label inside the dataset, we decided to resample the training dataset with a different sampling method instead of stratifying the data. With this method, we are able to ensure that the model has no imbalanced classifiers and all the individuals within the dataset are represented properly.

Evaluate the performance of shortlisted classifiers

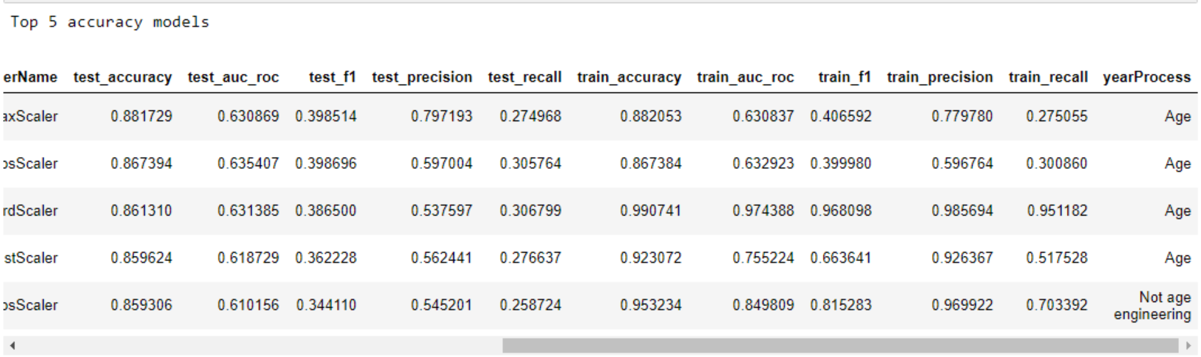
As for the result, we find out that some of the modeling techniques do not perform very well with our dataset. It turns out that some modeling techniques might require more features to be accurate. Therefore, we decided to just nitpick some modeling techniques that produce acceptable results and perform Grid Search CV for their respective parameters for optimization purposes.

The dataset will be evaluated through these few criterias and the final results will be exported into an excel file for further analysis, the program will also display the top 5 best performing models for each metrics score as results in the terminal. Criterias involved are, how the income, date-time and year columns in the dataset are processed, the sampling method utilized for each iterations and also the final results are categorized into classifier and scaler just like before in part A. We will evaluate the efficiency of these classifier-to-scaler pairs through a final train-test metrics score results, which will be produced after running it through Stratified K-Fold and retrieving the average results. The total results will be stored into an excel file, similarly to part A's metric score evaluation phase. To further improve the performance speed of our models, we introduced the nystrom method to process and perform subsampling on our data within the kernel for our automated hyperparameter tuning via data transformation phase. This is achieved by implementing the nystrom library into our modelling process. We will also be introducing the similar nystroem method again when running our RandomizedSearchCV.

Automated Hyperparameter Tuning via Data Transformation

Once again before the randomized search CV is ran, the data will go through the similar process status mentioned above including null value forward fill, date column transformation, income column transformation, training data oversampling, data scaling using Quantile Transformer, Nystroem data transformation and lastly model parameter search. The search included the models of SVC, LGBM Classifier, Random Forest Classifier, XGB Classifier and Logistic Regression. The results of the search were recorded in a DataFrame. Then, the DataFrame will be applied into a Voting Function to create a model and

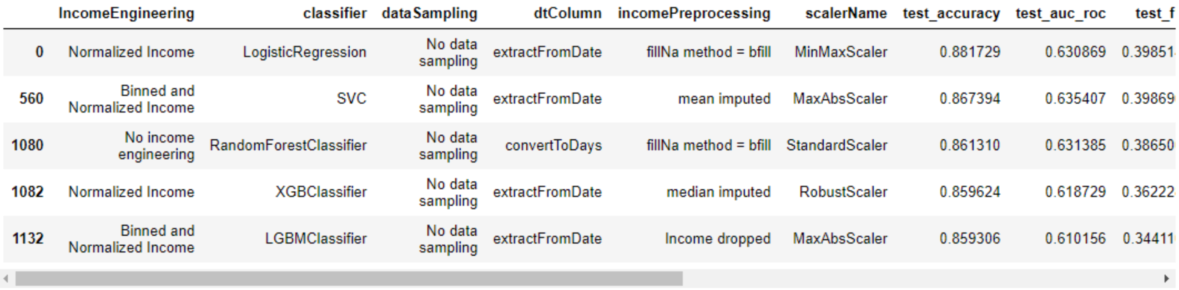
The results of the automated hyperparameter tuning has shown that the top 3 models to return the highest accuracy are the Logistic Regression, SVC and Random Forest Classifier. We will focus on the Logistic Regression to obtain our optimal data transformation methods. The results show that our Income data must be Normalized and the pre-processing method chosen for the Income data is to fill the null values by bfill method. No data sampling is required and the date time required will be extracted from the original dataset's date column. The scaler utilized is the MinMaxScaler and we will process the Year_Birth column from the original dataset and transform it into age as it will represent the data more accurately. Hence this is how our parameters will be tuned for later when generating the final model.



Top 5 accuracy models

erName	test_accuracy	test_auc_roc	test_f1	test_precision	test_recall	train_accuracy	train_auc_roc	train_f1	train_precision	train_recall	yearProcess
axScaler	0.881729	0.630869	0.398514	0.797193	0.274968	0.882053	0.630837	0.406592	0.779780	0.275055	Age
ɔsScaler	0.867394	0.635407	0.398696	0.597004	0.305764	0.867384	0.632923	0.399980	0.596764	0.300860	Age
rdScaler	0.861310	0.631385	0.386500	0.537597	0.306799	0.990741	0.974388	0.968098	0.985694	0.951182	Age
stScaler	0.859624	0.618729	0.362228	0.562441	0.276637	0.923072	0.755224	0.663641	0.926367	0.517528	Age
ɔsScaler	0.859306	0.610156	0.344110	0.545201	0.258724	0.953234	0.849809	0.815283	0.969922	0.703392	Not age engineering

Table 3.1.1 - Automated Hyperparameter Tuning via Data Transformation Results (Pt 1)



	IncomeEngineering	classifier	dataSampling	dtColumn	incomePreprocessing	scalerName	test_accuracy	test_auc_roc	test_f
0	Normalized Income	LogisticRegression	No data sampling	extractFromDate	fillNa method = bfill	MinMaxScaler	0.881729	0.630869	0.39851
560	Binned and Normalized Income	SVC	No data sampling	extractFromDate	mean imputed	MaxAbsScaler	0.867394	0.635407	0.39869
1080	No income engineering	RandomForestClassifier	No data sampling	convertToDays	fillNa method = bfill	StandardScaler	0.861310	0.631385	0.38650
1082	Normalized Income	XGBClassifier	No data sampling	extractFromDate	median imputed	RobustScaler	0.859624	0.618729	0.36222
1132	Binned and Normalized Income	LGBMClassifier	No data sampling	extractFromDate	Income dropped	MaxAbsScaler	0.859306	0.610156	0.34411

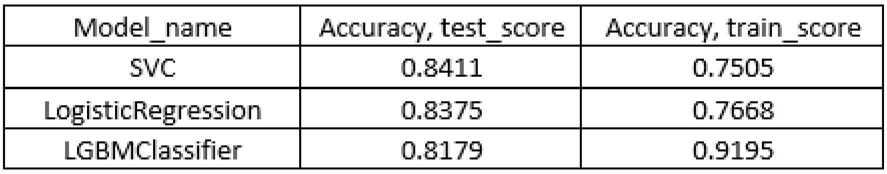
Table 3.1.2 - Automated Hyperparameter Tuning via Data Transformation Results (Pt 2)

Grid Search for Parameter Tuning

After selecting out the best models and best parameters for our final models through our hyperparameter tuning, we will also be running the same classifiers and scalers through a RandomSearch CV once again. This time we will narrow down the processing of the data based on the results received from our previous function and transform the data based on that. The transformed data will then be fitted into nystroem and then passed into the Random Search Cross Validation function. The final results will be passed back for us to review via dataframes and excel files and through those results, we will pick our final 3 models used for ensembling.

To evaluate our model as mentioned above, we have chosen accuracy as our main evaluation metric as it's the most common and the easiest to understand. A high accuracy score meant that the model is accurate in predicting both instances as a whole, hence it will allow the hypermarket to have more confidence in the results of our model. The use of stratified and unstratified data in testing and training our model can also be said to have contributed enormously in helping us determine the true accuracy of our models.

The accuracy above showed that SVC performed the best in terms of test score, 84% accuracy even though it has the worst of training scores at the accuracy of 75%. In contrast, LGBM Classifiers perform the best among the three in terms of training score at accuracy of 92% but the worst when it comes to testing coming in at 81%. In the meantime, Logistic Regression was slightly underperformed when compared against SVC in terms of test score at 83% for accuracy but 77% when it comes to accuracy during training score. This shows that SVC was good in terms of generalization and it is the most suitable for our data.



Model_name	Accuracy, test_score	Accuracy, train_score
SVC	0.8411	0.7505
LogisticRegression	0.8375	0.7668
LGBMClassifier	0.8179	0.9195

Table 3.1.3 - Accuracy Score Comparison between Top 3 Performers Sorted by Test Score

Result

To prevent the model from overfitting, a confusion matrix is created within the model and is used to evaluate if the model is an overfitted model.This table is useful as it can tell us how the model actually performs in term of True Positive, True Negative, False Positive and False Negative. Based on the true negative and true positive we will be able to know how many cases were predicted correctly, for both responded yes and responded no. If a model can produce a good result in classification but it performs badly when evaluated using a confusion matrix, that particular model cannot be counted as a good model. Hence, this provides us a more detailed insight that we could not get from classification accuracy.

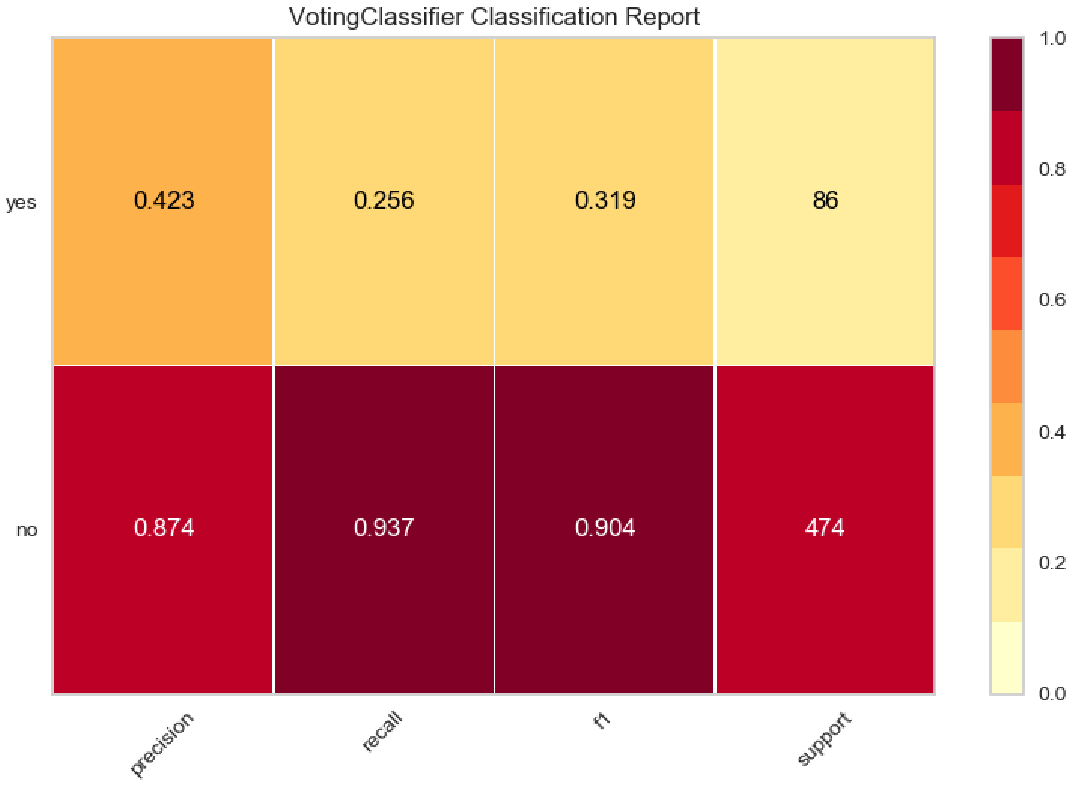


Figure 4.1 Voting Classifier Classification Report

By referring to the classification report produced by our final ensemble model, we can come to a firm conclusion that results of our model are within logical and acceptable range. Due to the extensive checking and selection of our features, we are confident to say that the features used in our model training has contributed significantly to the accuracy of our model. The high accuracy in predicting the likelihood of customers responding will be able to aid the hypermarket executives in planning and selecting the correct customer base to perform their marketing campaigns. Furthermore, the high recall in our model also ensures that the model has a much lower chance in mislabeling potential customers as not responsive, hence mitigating the cost of losing potential profit for the hypermarket.

Therefore, to achieve our business goal, we need to obtain a model with both a high recall value and also a high precision score. With high precision, we can ensure that the company can minimize the number of responded no customers to receive excess marketing campaign flyers and catalog with high recall, the company will not miss out on potential customers that might respond positively to future marketing campaigns and reward the right people to ensure customers loyalty in the long run.

Another metric that was utilized during the model evaluation is the Receiver Operating Characteristic Curve (ROC) and Area Under the Curve (AUC) score that is useful in determining whether a model is useful in binary classification problems. Optimally, achieving a ROC AUC score of 1.0 that means perfect separation between two class output is the best.

Train roc_auc_Score	0.6847067039106145
Test roc_auc_Score	0.5803895594151702

Figure 4.1 Train and Test ROC AUC Score

Assessment Rubric

Name(s): Programme: Group: Date:

Project Part A – Shortlist promising models (40%)

No	Item	Criteria			Final Marks
		Poor	Accomplished	Good	
1	Problem statement (10)	No or very little discussion on existing problem and the project The proposed project already exists, or with very minor change. No discussion or very little of introduction given to the related system or technology. 0-4	Little discussion on existing problem and introduction of proposed project. Minor ideas are modified from existing system(s). Introduction to the related system is given, but no evaluation provided. 5-7	Good discussion and evaluation of existing problem and the proposed project. Ideas modified from existing system, with some creative ideas are added. Good discussion and evaluation of the related system. 8-10	
2	Programming (20)	The end product fails with many logic errors, many actions lacked exception handling. Solutions are over-simplified. Programming skill needs improvement. Evaluation steps of different models are not automated. 0-7	Major parts are logical, but some steps to complete a specific job may be tedious or unnecessarily complicated. Program algorithm demonstrates acceptable level of complexity. The student is qualified to be a programmer Some evaluation steps are automated. 8-15	Correct and logical flow, exceptions are handled well. Demonstrates appropriate or high level of complex algorithms and programming skills. Almost all evaluation steps are automated. 16-20	
3	degree of completion (10)	Too much still remain to be done. Basic requirements are not fulfilled. The end product produces enormous errors, faults or incorrect results. Limited performance metrics are used. 0-4	All required features present in the interface within the required scope, but some are simplified. Or one or two features are missing. The system is able to run with minor errors. More than 5 performance metrics are shown. 5-7	All required features present in the interface within or beyond the required scope. No bugs apparent during demonstration. More than 8 performance metrics are shown. 8-10	
				Sum of Score	

No	Item	Criteria			Final Marks
		Poor	Accomplished	Good	
1	Model Optimization (10)	The model is not optimized. Default setting is used without any adjustments. 0-4	The model is optimized based on performance metrics. Different parameters are regularized to optimize the model. Ensemble classifier is not attempted. 5-7	The model is optimized based on performance metrics. Different parameters are regularized to optimize the model Ensemble classifier is evaluated. 8-10	
2	system implementation (10)	The end product is produced with different system design or approach, which is not related to the initial proposal. 0-4	The end product conforms to most of the system design, but some are different from the specification. 5-7	The end product fully conforms to the proposed system design. 8-10	
3	Results (Performance measurement) (10)	Analytical methods were missing or inappropriately aligned with data and research design. Results were confusing. 0-4	The analytical methods were identified. Results were presented. All were related to the research question and design. Sufficient metric or measurement is applied. 5-7	Analytical methods and results presentation were sufficient, specific, clear, structured and appropriate based on the research questions and research design. Extra metric or measurement is applied. 8-10	
4	Organization (5)	The structure of the paper was weak. Transition was weak and difficult to understand. 0-2	A workable structure was presented for presenting ideas. Transition was smooth and clear. 3-4	Structure was intuitive and sufficiently inclusive of important information of the research. Transition from one to another was smooth and organized. 5	
				Sum of Score	

Project Part B – Fine-tune the system (35%)

Presentation (25%)

No	Item	Criteria			Final Marks
		Poor	Accomplished	Good	
1	Output (10)	Inadequate information/outputs needed are generated. Most of the information/outputs generated are less accurate. Results visualization is overly cluttered or the design seems inappropriate for problem area. Lack of information that are useful for the user 0-4	Adequate information/outputs needed are generated. The information/output generated are accurate but some with errors. Pleasant looking, clean, well-organized results visualization The information displayed are useful for the user, but some details are omitted. 5-7	All the necessary information/outputs are generated. All or most of the information/outputs generated are accurate. Minor errors can be ignored. The results are visually pleasing and appealing. Great use of colors, fonts, graphics and layout. The information displayed are useful to the users and complete with necessary details. 8-10	
2	Presentation (10)	Presentation was unclear. Results were presented without justifications and reasons. Results were not supported with ML concepts and theories. 0-4	Presentation is well organized for the most part, but more clarity with transitions is needed. Answers to the research question and system performances were supported with sufficient ML concepts and theories. 5-7	Presentation was concise and straight to the points. Discussions of the results were presented and illustrated in easily interpretable graphs or charts. The research question and system performance were answered and identified.\ 8-10	
3	Question and answer (5)	The student is unclear about the work produced, sometimes not even knowing where to find the source code. 0-1	The student knows the code whereabouts, but sometimes may not be clear why the work was done in such a way. 2-3	The student is clear about every piece of the work done. 4-5	
				Sum of Score	