Prediction of Marketing Campaign

Name:	**Shraddha Agarwal**
Registration No./Roll No.:	19294
Institute/University Name:	IISER Bhopal
Program/Stream:	DSE
Problem Release date:	February 02, 2022
Date of Submission:	April 24, 2022

1 Introduction

The objective is to develop supervised machine learning frameworks to predict the success or failure of the recent marketing campaign based on customers profile, their preferences and responses to the previous campaigns.
The training data consists of 2016 samples that have 25 features based on customer's profile, product preferences, channel performance and response to the previous campaigns. The training class label indicates the success (denoted as '1') and failure (denoted as '0').

2 Methods

I began the project with the cleaning of the dataset and fixing the null values present in the 'income' feature by imputing them with the median values in place of mean values to avoid skewing of data due to outliers. Feature engineering has been employed by transforming some of the features and creating new features from the existing features. Since two features contain categorical values, I implemented one hot encoder to transform them into the numerical values.
Since there are large number of features in the data I have used the decision tree as the feature selection method to select the most relevant features for modelling.
Step 1: The one hot encoded dataset had been fit into Decision Tree Classifier and represented in the text format by using export tree from the tree module of sklearn library.
Step 2: The tree had been plotted with nodes containing relevant features and edges using plot tree from the tree module of sklearn library.
Six features that are on the top nodes of the decision tree were considered for building the models.

The training data split into validation set with stratified class labels. A pipeline had been created using 8 models that are Logistic Regression, Decision Tree, Random Forest, Multinomial Naïve Bayes, Support Vector Classifier, AdaBoost Classifier, k-Nearest Neighbor Classifier and Artificial Neural Networks along with parameter tuning techniques like Grid Search and Randomized Search. The mentioned models had been implemented using the scikit-learn library of Python.
https://github.com/shraddhaagarwal10/Prediction-of-Marketing-Campaign

3 Evaluation Criteria

The pipeline prints the confusion matrix and classification report of the predicted class labels of validation set for each model. A Confusion matrix is an N x N matrix used for evaluating the performance of a classification model, where N is the number of target classes. The matrix compares the actual target values with those predicted by the machine learning model.

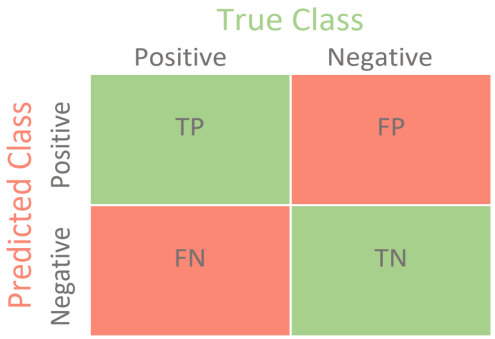


Figure 1: Confusion Matrix

- True positive — actual = 1, predicted = 1
- False positive — actual = 0, predicted = 1
- False negative — actual = 1, predicted = 0
- True negative — actual = 0, predicted = 0

A classification report is a performance evaluation metric in machine learning. It is used to show the precision, recall, f1-score, and support of the trained classification model.

Accuracy is fraction predicted correctly.
$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$

Precision is fraction of predicted positives that are actually positive.
$Precision = \frac{TP}{TP+FP}$

Recall is fraction of positives predicted correctly.
$Recall = \frac{TP}{TP+FN}$

The f1 score is the harmonic mean of recall and precision, with a higher score as a better model.
$f1 - score = \frac{2 \cdot precision \cdot recall}{precision+recall}$

The reported averages include macro average (averaging the unweighted mean per label), weighted average (averaging the support-weighted mean per label). Micro average (averaging the total true positives, false negatives and false positives) is only shown for multi-label or multi-class with a subset of classes and it corresponds to the accuracy otherwise and would be the same for all metrics. [1]
This is the reason why I have considered the performance of every classification model on the basis of the accuracy score as this is the case of binary classification.

4 Analysis of Results

The macro averaged precision, recall and f1-scorefor 8 models (i.e., LR: Logistic Regression, DT: Decision Tree Classifier, RF: Random Forest Classifier, MNB: Multinomial Naïve Bayes Classifier, SVM: Support Vector Machine, AB: AdaBoost Classifier, KNN: k-Nearest Neighbor, ANN: Artificial Neural Network) with grid search and random search parameter tuning technique has been shown in 1. The performance measures are described by the classification reports.

Normally, accuracy and recall are the parameters for evaluating a model's performance in marketing domain[1]. The confusion matrix contains information about actual and predicted classifications which is obtained from the execution of the algorithm. The confusion matrix for implemented classifiers is shown in Table 2.

[1] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report

Table 1: Performance Metrics Of Different Classifiers Using All Terms

Tuning Technique	Classifier	Precision	Recall	F-1 score	Accuracy
Grid	LR	0.69	0.80	0.71	0.81
	DT	0.88	0.65	0.70	0.89
	RF	0.81	0.71	0.74	0.89
	MNB	0.57	0.63	0.53	0.62
	SVM	0.61	0.63	0.62	0.79
	AB	0.84	0.71	0.76	0.90
	KNN	0.78	0.60	0.63	0.87
	ANN	0.86	0.62	0.66	0.88
Random	LR	0.69	0.80	0.71	0.81
	DT	0.85	0.67	0.71	0.89
	RF	0.83	0.70	0.74	0.89
	MNB	0.57	0.63	0.53	0.62
	SVM	0.62	0.66	0.63	0.79
	AB	0.84	0.71	0.76	0.90
	KNN	0.78	0.60	0.63	0.87
	ANN	0.43	0.50	0.46	0.85

Table 2: Confusion Matrices of all the 8 classification models Using All Terms

LR

	Predicted Class	
	Success	Failure
Success	280	64
Failure	13	47

MNB

	Predicted Class	
	Success	Failure
Success	213	131
Failure	22	38

SVM

	Predicted Class	
	Success	Failure
Success	296	48
Failure	36	24

AB

	Predicted Class	
	Success	Failure
Success	334	10
Failure	34	26

KNN

	Predicted Class	
	Success	Failure
Success	338	6
Failure	47	13

ANN

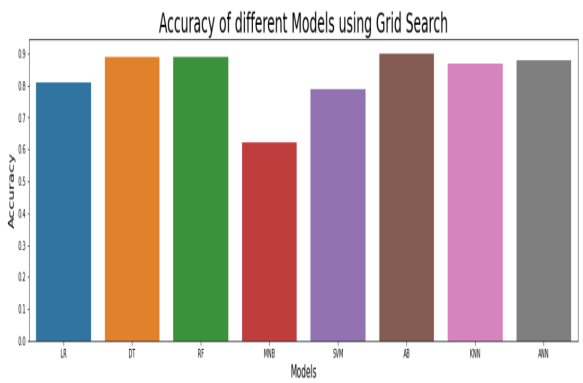
	Predicted Class	
	Success	Failure
Success	341	3
Failure	45	15

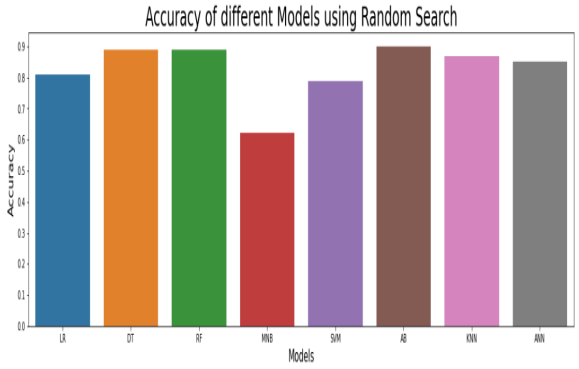
DT

	Predicted Class	
	Success	Failure
Success	341	3
Failure	41	19

RF

	Predicted Class	
	Success	Failure
Success	334	10
Failure	33	27





5 Discussions and Conclusion

I have chosen eight mostly used algorithms for targeting a pool of customers for marketing campaigns. Among all the classifiers, experimental results show that the accuracy is highest for the AdaBoost Classifier with both Grid Search and Random Search. This is the best fit model on the validation set and hence it is considered as the high performing model with high performing parameter tuning as Random Search to predict the class labels of test set. Random Search is taken into the consideration because it found better models in most cases and required less computational time. This is because grid search allocate too many trials to the exploration of dimensions that do not matter and suffer from poor coverage in dimensions that are important.[2] However, the worst classification was performed by Multinomial Naïve Bayes as MNB achieves lowest accuracy and recall. Additionally, MNB achieves the highest error rate with a large number of false positive and false negative cases when compared to other classifiers.

We can conclude that the company can plan effective marketing of its products by selecting the target customers. For selecting the right customers they can use AdaBoost Classification algorithm which fits the data set correctly. This technique would help the marketing department to identify the respondents so that they would be basically targeted for specific campaigning activity. Also, it prevents wasteful expenditure of sending promotion offers to the non-respondents.

This work could be further enhanced by developing a new algorithm which would classify the data with high accuracy and low error rate. None of the classifiers achieved all the parameters to the satisfactory level. So, a hybrid algorithm can be developed, which achieves higher accuracy and recall.

References

[1] Tapan Kumar Das. A customer classification prediction model based on machine learning techniques. In *2015 International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)*, pages 321–326. IEEE, 2015.

[2] James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2), 2012.