Predicting Customer Responses to Marketing Offers

Meenakshy Manju

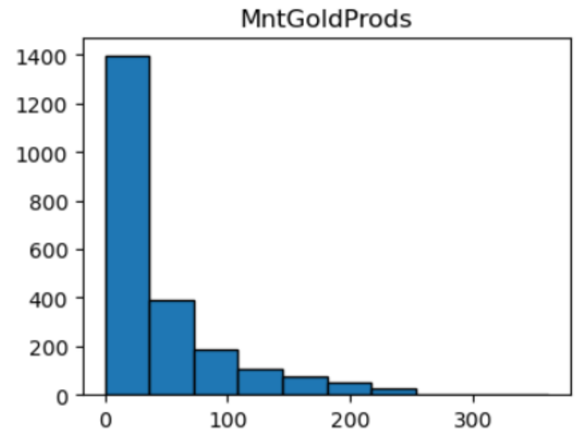
1. Introduction

Customer Personality Analysis is a detailed analysis of a company's ideal customers. It helps a business to better understand its customers and makes it easier for them to modify products according to the specific needs, behaviors and concerns of different types of customers. The aim of this project is to predict whether a customer will accept a marketing campaign offer based on their demographic and behavioral data. This is a binary classification problem, where the target variable is the customer's response to the offer (1 for acceptance, 0 for rejection). By identifying which customers are more likely to accept an offer, companies can target marketing efforts more effectively, optimize resource allocation and improve overall campaign success rates. The dataset used for this analysis is the **Marketing Campaign Dataset** from Kaggle. It contains records of customer demographics, purchasing behavior, and past interactions with marketing campaigns.

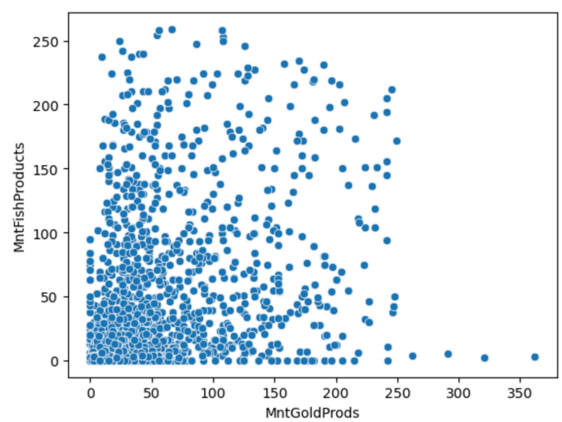
2. Exploratory Data Analysis (EDA)

The dataset has 2,240 rows and 29 columns. Each row represents a unique customer. The data includes customer demographics, purchasing behavior, and previous marketing campaign responses. Most columns are self-explanatory but the least obvious columns are 'Z_CostContact' and 'Z_Revenue'. These are constant across all rows and provide no useful information for modeling, so they were removed. The 'Income' column has 24 missing values and were handled through imputation (median). The 'Response' column is the target variable where 1 means customer accepted the offer and 0 means customer did not accept the offer. It is imbalanced with class 0 having 85% and class 1 having 15%.

Several numeric features exhibited right-skewed distributions, such as 'MntWines','MntFruits','MntMeatProducts','MntFishProducts','MntSweetProducts','NumDealsPurchases','NumWebPurchases','NumCatalogPurchases' and 'MntGoldProds', hence log transformation was used.



Additionally, correlation analysis showed that some features were correlated with each other.



3. Data Preparation and Preprocessing

To prepare the dataset for modeling, several data engineering steps were performed. 'Id', 'Z_CostContact' and 'Z_Revenue' had constant values so they were removed. New features were engineered such as:

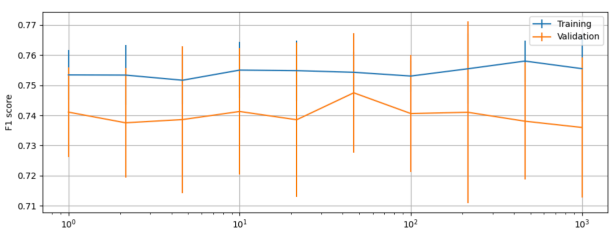
- 'Age' was calculated by subtracting 'Year_Birth' from the current year.
- 'Customer_Month' was derived from the 'Dt_Customer' column to represent the month of customer enrollment.
- 'Children' was created by summing the 'Kidhome' and 'Teenhome' columns to reflect the total number of children in the household.

The data was split into 80% training, 10% validation and 10% test set. The dataset did not contain raw text features, so TF-IDF was not necessary. The month was extracted from ‘Dt_Customer’ and converted it into numerical format to capture customer tenure so no cyclic transformation was required. Numerical columns were standardized using StandardScaler while categorical variables like ‘Education’, ‘Marital_Status’ were encoded using one-hot encoding. Ordinal encoding was not used as no ordinal relationships were defined among the categories. The target column (0 or 1) was already binary and did not require any further transformation.

4. Model Training and Results

4.1 Logistic Regression

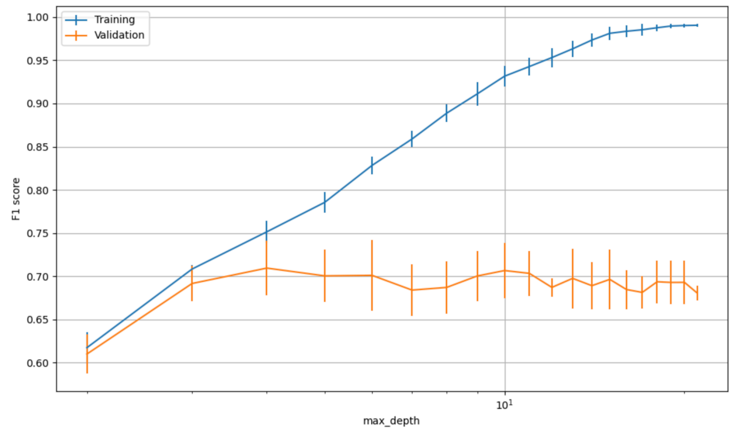
For the Logistic Regression model, a range of regularization strengths was tested by taking the logarithmic scale of C values from 10^0 to 10^3. After performing grid search cross-validation, the optimal value of C was found to be 46.41.



Training set has the high F1-score whereas validation set has low F1-score. Also the error bars are long which shows there is high variance.

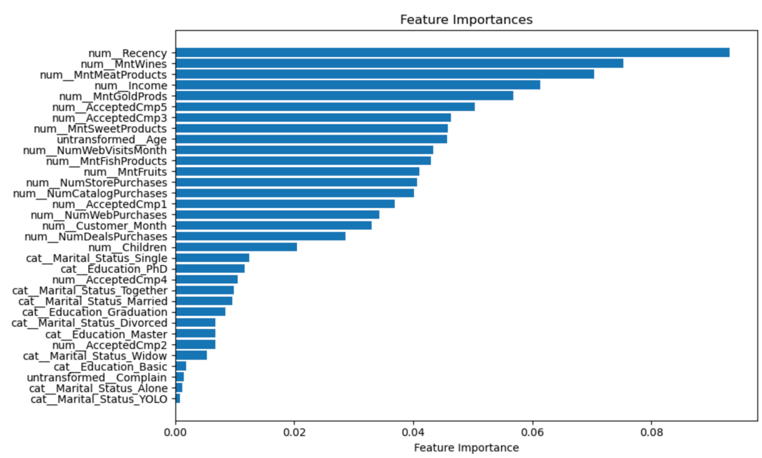
4.2 Decision Tree

For the Decision Tree model, max_depth was used as the hyper parameter to control the complexity of the tree and prevent overfitting. After evaluating multiple values, the best-performing configuration was 4. The training set has high F1 score than validation set which indicates overfitting.



4.3 Random Forest Classifier

For the Random Forest Classifier, two key hyper parameters were tuned using GridSearchCV, max_depth and n_estimators.The best parameters were max_depth = 16 and n_estimators = 100.



From the above plot, ’num_Recency’,’num_MntWines’ and ’num_MntMeatProducts’ are the most important features.

4.4 Gradient Boosting Classifier

For the Gradient Boosting Classifier, default hyperparameters were used without performing grid search. The model achieved an accuracy of 0.88 and f1-score, 0.35.

4.5 Support Vector Classifier

The Support Vector Classifier was tuned using grid search with C, gamma and kernel. The best combination of hyper parameters obtained through grid search was C = 100, gamma: 0.001, kernel: 'rbf'.

4.6 Neural Network

For the Neural Network model, five different architectures were tested, each varying in the number of hidden layers, neurons, activation functions, and dropout usage. These were labeled as scenario_1 to scenario_5. Among all scenarios, scenario_2 performed best, achieving the lowest test loss of 0.2897 and test accuracy of 0.8929. Scenario 2 architecture consists of an input layer, dense layer with 10 neurons, ReLU activation, dropout layer with rate 0.3 and output layer with 1 neuron, sigmoid activation. Early stopping was used during training to prevent overfitting.

Model	Accuracy	Precision	Recall	F1 Score
Logistic Regression	0.89	0.83	0.32	0.46
Decision Tree	0.86	0.5	0.19	0.27
Random Forest Classifier	0.89	0.83	0.32	0.46
Gradient Boosting Classifier	0.88	0.77	0.22	0.35
Support Vector Classifiers	**0.88**	**0.66**	**0.38**	**0.48**
Neural Networks	0.88	0.72	0.22	0.35

While Logistic Regression and Random Forest had the highest accuracy (0.89), their recall was low, indicating poor performance in detecting positive cases. Support Vector Classifier had the highest F1 score (0.48), balancing both precision and recall reasonably well. The Neural Network model achieved comparable performance, with moderate precision and low recall.

5. Conclusion

- Based on overall performance metrics, the Support Vector Classifier (SVC) emerged as the best-performing model. It achieved a strong balance between precision (0.66) and recall (0.38), resulting in the highest F1 Score of 0.48.

- Logistic Regression was the simplest model with the fewest parameters and easiest interpretation. However, its recall was the lowest (0.32).

- Neural Network had decent accuracy (0.88) and moderate precision but added complexity with little gain over simpler models.

- Random Forest and Gradient Boosting provided good accuracy but were more complex and did not outperform SVC in terms of F1 score.

6. Future Scopes

- Class Imbalance Handling: Recall values are consistently low across all models, indicating possible class imbalance. Use SMOTE with an imblearn pipeline to avoid data leakage, and compare results before and after balancing.

- Feature Engineering: Explore new interaction features (e.g., combining purchase behavior and age) and investigate clustering-based segmentation as a new feature input.