CUSTOMER PERSONALITY ANALYSIS

Doğasel Günel
Computer Engineering
Kadir Has University
Istanbul, Türkiye
dogasel.gunel@stu.khas.edu.tr

Onur Sarıaltun
Management Information Systems
Kadir Has University
Istanbul, Türkiye
onur.sarialtun@stu.khas.edu.tr

Süleyman Aygün
Management Information Systems
Kadir Has University
Istanbul, Türkiye
suleyman.aygun@stu.khas.edu.tr

ABSTRACT

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.

Customer personality analysis helps a business to modify its product based on its target customers from different types of customer segments. For example, instead of spending money to market a new product to every customer in the company's database, a company can analyze which customer segment is most likely to buy the product and then market the product only on that particular segment.

I.INTRODUCTION

This technical paper provides a complete study of a marketing campaign dataset, focusing on Customer Personality study using supervised classification algorithms. The study, thoroughly documented in a Jupyter Notebook, aims to improve marketing strategies by revealing intricate patterns in consumer data. The collection contains precise information about 2240 clients, including demographics and purchasing patterns.

This paper clearly describes the methodology used, including data preprocessing and normalization, analytical processes, and significant results from the categorization study. Businesses that use focused and successful customer segmentation may better match their strategies with consumer demands and habits, resulting in increased marketing efficacy and customer satisfaction.

II. FIGURES & TABLES

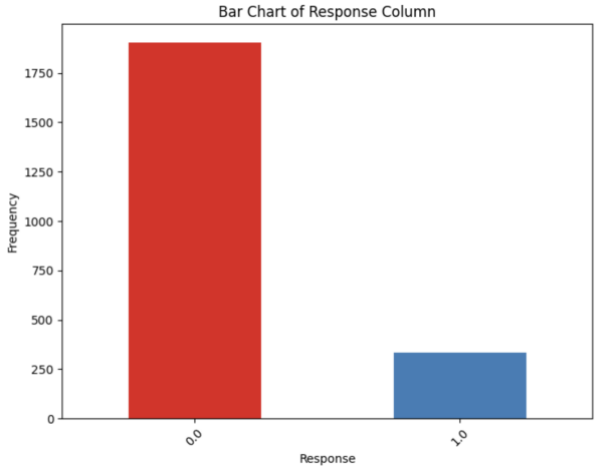

Figure 1: Imbalance label column "Response"

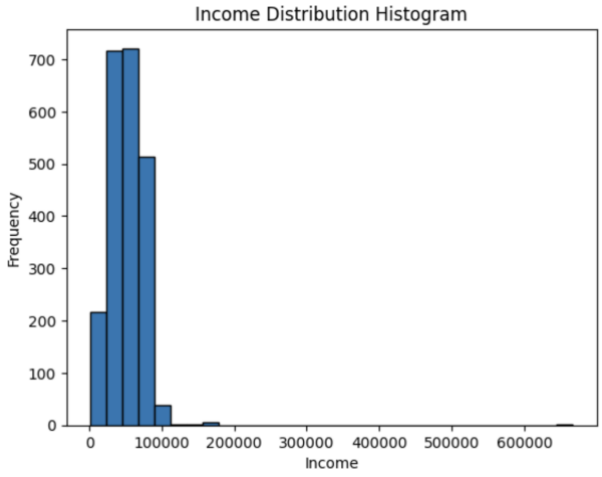

Figure 2: Histogram of "Income" column

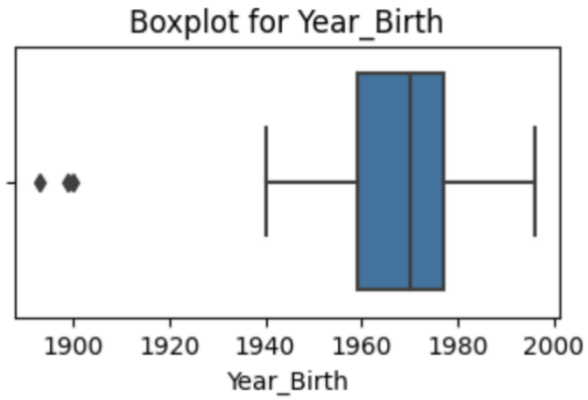

Figure 3: Boxplot of "Year_Birth" column

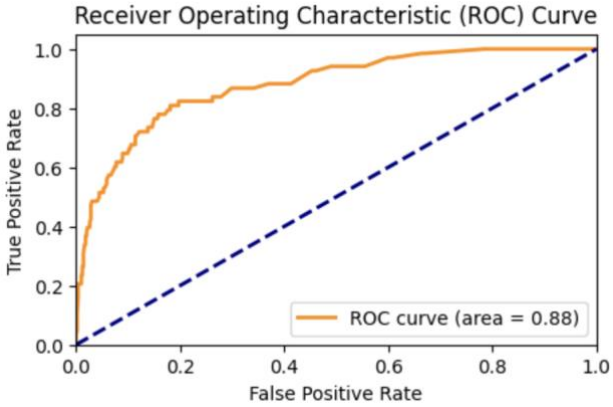

Figure 4: ROC Curve of Random Forest Classifier

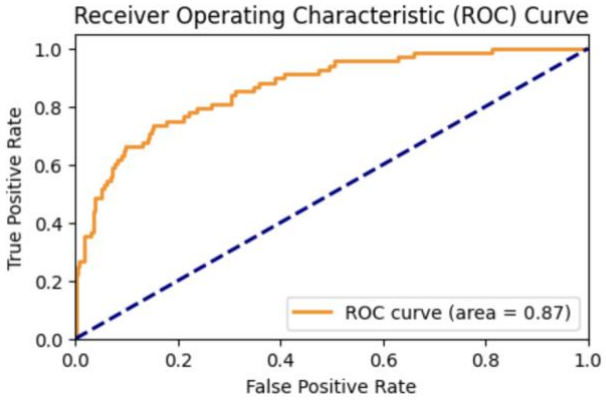

Figure 5: ROC Curve of Logistic Regression Classifier

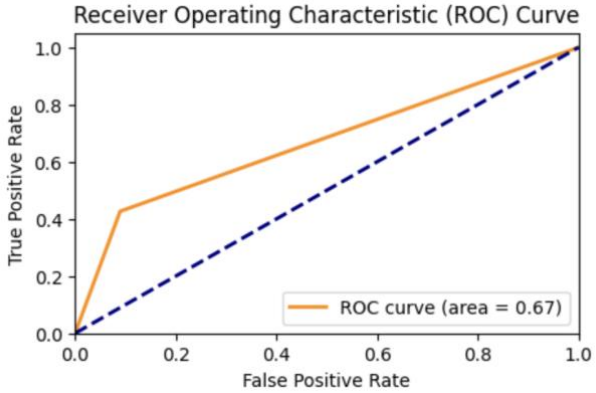

Figure 6: ROC Curve of Decision Tree Classifier

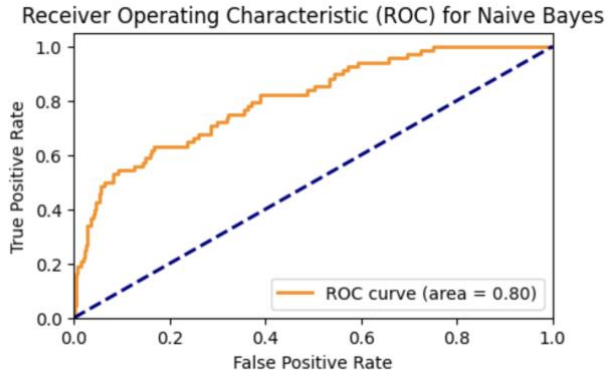

Figure 7: ROC Curve of Naïve Bayes Classifier

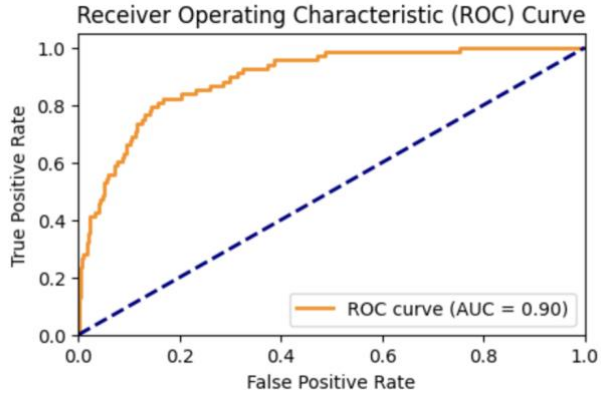

Figure 8: ROC Curve of Gradient Boosting Classifier

Method	ROC Curve	Accuracy	F1-Score	Precision	Recall
Random Forest Classifier	0.88	0.91	0.51	0.71	0.40
Logistic Regression	0.87	0.90	0.44	0.71	0.32
Decision Tree Classifier	0.67	0.85	0.41	0.40	0.43
Naive Bayes Classifier	0.80	0.85	0.47	0.41	0.54
Gradient Boosting	0.90	0.90	0.47	0.71	0.35

Table 1: Overall scores of all maching learning methods

III. DESCRIPTIVE ANALYSIS

In this section, a comprehensive descriptive analysis of the dataset was conducted. This analysis is crucial in forming a foundational understanding of the data, which is essential for effective preparation for subsequent mining processes.

A. Library Importation:

Necessary libraries were imported in the initial phase. This step equips the analysis environment with essential tools, such as Pandas for data manipulation and Matplotlib for visualization.

B. Data Loading:

The dataset was loaded using Pandas, a robust data manipulation library. This action is fundamental for accessing and manipulating the data in an organized and efficient manner.

C. Data Type Inspection:

An examination of the data types of each column in the dataset was performed. Identifying data types is critical for understanding the nature of the data (numerical, categorical, etc.) and guiding appropriate preprocessing techniques.

D. Histogram Analysis:

Histograms were utilized to gain insights into the distribution of the data. These visualizations are instrumental in revealing the frequency distribution of numerical data, providing clarity on data skewness or symmetry.

E. Label Distribution Analysis:

As it can be seen in the Figure 1, the distribution of the 'Label' column was closely analyzed. This analysis is important in supervised learning tasks to understand the balance or imbalance in the target variable.

F. Null Value Identification:

A search for all null values within the dataset was conducted. Detecting missing values is crucial in data cleaning, as it impacts the quality and integrity of the data.

G. Strategy for Handling Null Values:

Upon identifying the columns with null values, their distribution was examined using histograms. This examination informed the decision-making process regarding the most suitable methods for handling these missing values.

H. Outlier Detection with Boxplots:

Boxplots were employed for each column to detect outliers. There is an example of boxplot for outlier detection in Figure 3. These plots are effective for displaying the distribution of the data and highlighting data points that significantly deviate from the norm.

I. Outlier Quantification:

The number of outliers in each column was quantified. This step is crucial, as outliers can have a significant impact on data analysis and modeling. By identifying and quantifying these outliers, an assessment of their impact was made, guiding decisions on potential outlier treatment or removal.

IV.PREPROCESSING

The preprocessing phase of the "Customer Personality Analysis" project involved several crucial steps to enhance the quality and effectiveness of the dataset for subsequent machine learning methods.

A. Imputation of Null Values:

As it can be seen in the Figure 2, column “Income” has a right skewed distribution. According to this information, null values within the "Income" column were addressed by filling them with the mean of the column. This strategy was implemented due to the singular presence of missing values within the "customer personality analysis" dataset.

B. Handling Imbalance and Outliers:

Given the imbalanced nature of the dataset and the existence of outliers in the label column, an outlier removal technique utilizing the Interquartile Range (IQR) method was applied. Notably, the label column was exempt from this process.

C. Value Consolidation:

Instances where multiple columns conveyed identical meanings were identified. Subsequently, values with analogous meanings were consolidated into a single representative value, streamlining the dataset.

D. Conversion of Non-Integer Columns:

Building upon insights from descriptive analysis, non-integer columns were transformed into integer types. This conversion was particularly relevant, considering the predominantly integer composition of the dataset, aiming to optimize the efficacy of machine learning methods.

E. Aggregation of Similar Integer Columns:

Integer-type columns sharing similar meanings were amalgamated by adding their values together. The resultant consolidated column was retained, while the redundant columns were removed from the dataset.

F. Transformation of Birth Year to Age:

To facilitate a more concise and understandable representation, the "customer birth year" column was transformed into an "age" column, contributing to improved dataset interpretability.

G. Removal of Redundant Columns:

Columns where all rows exhibited uniform responses were identified and subsequently eliminated. These columns were deemed extraneous for the application of machine learning methods.

H. Exclusion of Customers Aged Over 90:

Customers aged over 90 were excluded from the dataset based on the assumption that this demographic is less likely to participate in the campaign, aligning with campaign targeting considerations.

I. Min-Max Normalization of Integerized Columns:

Following the transformation of all columns in the dataset into integers, normalization was performed using the min-max method. This method ensured a consistent scale across all columns, mitigating disparities arising from the initial diversities in data range.

J. Visualization of Normalized and Preprocessed Columns:

Subsequently, a comprehensive visualization was executed through histogram graphs, encompassing all columns that underwent both normalization and preprocessing. This graphical representation provided insights into the distribution characteristics of the data, aiding in the assessment of the effectiveness of the normalization process.

These normalization procedures collectively contributed to the establishment of a standardized and scaled dataset, fostering a more conducive environment for the subsequent application of machine learning algorithms. The visualization aspect further facilitated a qualitative understanding of the transformed data distribution, offering valuable perspectives for the ensuing stages of the project.

These preprocessing steps collectively aimed to refine the dataset, address data integrity issues, and optimize its compatibility with subsequent machine learning algorithms. The resulting dataset is poised for more robust and accurate model training and evaluation.

VI. MACHINE LEARNING

Five different machine learning techniques were used in the process of creating an efficient prediction model for our dataset: Random Forest Classifier, Logistic Regression, Decision Tree, Naive Bayes, and Gradient Boosting. Each method was applied to categorize and forecast the target variable based on the features included in the dataset.

A. Random Forest Classifier

The Random Forest Classifier [1] is an ensemble learning algorithm that combines the predictive power of multiple decision trees. During training, a specified number of decision trees are constructed, each utilizing a subset of the features and a random subset of the training data. The final classification is determined by aggregating the predictions of all individual trees, often through a majority voting mechanism. This ensemble approach enhances the model's robustness and generalization capabilities by mitigating overfitting tendencies commonly associated with individual decision trees. Random Forests are effective in capturing complex relationships within the data, making them suitable for a variety of classification tasks.

B. Logistic Regression Classifier

Logistic Regression [2] is a linear model designed for binary classification tasks. It models the probability of a sample belonging to a particular class using the logistic function. The logistic function, also known as the sigmoid function, transforms the linear combination of input features into a range between 0 and 1. This transformed value is interpreted as the probability of the sample belonging to the positive class. Logistic Regression optimizes the model parameters to maximize the likelihood of observing the given set of outcomes. Despite its simplicity, Logistic Regression is powerful and interpretable, making it a widely used algorithm in various domains.

C. Decision Tree Classifier

Decision Trees [3] are hierarchical structures that recursively partition the dataset based on the features, ultimately leading

to a decision or prediction. At each node of the tree, a decision is made based on a specific feature, and the dataset is split into subsets. The process continues until a predefined stopping criterion is met, such as a maximum tree depth or a minimum number of samples in a leaf node. Decision Trees are advantageous for their interpretability and ability to capture nonlinear relationships within the data. However, they are prone to overfitting, which can be addressed through techniques like pruning.

D. Naïve Bayes Classifier

Naive Bayes [4] is a probabilistic algorithm grounded in Bayes' theorem. It assumes that the features are conditionally independent given the class label, which simplifies the computation of probabilities. The model calculates the likelihood of observing a set of features given each class and combines it with prior probabilities to determine the most probable class for a given sample. Despite its "naive" assumption of independence, Naive Bayes often performs well, particularly in text classification and other high-dimensional datasets.

E. Gradient Boosting Classifier

Gradient Boosting [5] is an ensemble learning technique that builds a strong predictive model by combining the outputs of weak learners, usually decision trees. Unlike Random Forests, Gradient Boosting constructs trees sequentially, with each subsequent tree correcting the errors of the previous ones. The model minimizes a loss function by adjusting the weights of individual trees during training. Gradient Boosting is powerful in capturing complex relationships, handling missing data, and providing high predictive accuracy. However, it requires careful tuning to prevent overfitting.

VII. DISCUSSION

The results of the machine learning analysis, as presented in the evaluation metrics for each classifier, reveal varying degrees of performance across the different methods. Key metrics considered include the ROC Curve, Accuracy, F1-Score, Precision, and Recall.

The Random Forest Classifier demonstrated robust performance with a high ROC Curve score of 0.88 (Figure 4), indicating its ability to balance true positive rates and false positive rates effectively. This classifier achieved the highest accuracy of 91%, an F1-Score of 0.51, a precision of 0.71, and a recall of 0.40. The ensemble learning approach of Random Forests, aggregating predictions from multiple decision trees, contributes to its ability to capture intricate patterns in the dataset.

Logistic Regression, while achieving a commendable ROC Curve score of 0.87 (Figure 5), displayed slightly lower accuracy at 90%. The F1-Score, precision, and recall values were 0.44, 0.71, and 0.32, respectively. Logistic Regression's linear modeling approach is effective, especially in scenarios where interpretability is crucial, and it performed competitively in this context.

The Decision Tree Classifier, with a ROC Curve score of 0.67 (Figure 6), demonstrated an accuracy of 85%, an F1-Score of 0.41, precision of 0.40, and recall of 0.43. Decision Trees are known for their interpretability, but the model's performance fell slightly behind in terms of accuracy and F1-Score compared to other methods.

The Naive Bayes Classifier, with a ROC Curve score of 0.80 (Figure 7), achieved an accuracy of 85%. Notably, it showed a balanced performance in precision (0.41) and recall (0.54), resulting in an F1-Score of 0.47. Naive Bayes, based on Bayes' theorem, is suitable for probabilistic classification and demonstrated effectiveness in this analysis.

Gradient Boosting emerged as a strong performer with the highest ROC Curve score of 0.90 (Figure 8). It achieved an accuracy of 90%, a precision of 0.71, recall of 0.35, and an F1-Score of 0.47. Gradient Boosting's ensemble learning approach, sequentially building trees to correct errors, proved effective in capturing complex relationships within the data.

Choice of Classifiers:
The selection of Random Forest Classifier and Gradient Boosting over the other options was driven by their superior performance across multiple metrics (Table 1). Random Forest, through its ensemble learning approach, excelled in achieving a balanced accuracy, precision, and recall. It demonstrated resilience against overfitting, making it suitable for robust and generalized predictions.

Gradient Boosting, with its sequential tree-building mechanism, achieved the highest ROC Curve score and demonstrated strong predictive power. While it showed a lower recall compared to Random Forest, its overall performance, particularly in accuracy and precision, supported its suitability for this specific classification task.

In conclusion, the decision to choose Random Forest Classifier and Gradient Boosting was based on their superior performance in capturing the intricacies of the dataset, achieving high accuracy, and demonstrating a balanced trade-off between precision and recall. These classifiers are well-suited for the task of categorizing and predicting the target variable in the marketing campaign dataset, providing valuable insights for informed decision-making in marketing strategies.

VIII. CONCLUSION

In conclusion, the complete analysis and processing of the dataset in this study have contributed to a better knowledge and implementation of data mining techniques. The process began with an in-depth study of the dataset, which revealed crucial qualities and trends. This research set the groundwork for the next rounds of data preparation and model selection.

A variety of strategies were used throughout the preprocessing step. Outliers were carefully detected and eliminated from the data to assure its correctness and quality. Columns containing null values were dealt correctly, either by filling them in or by eliminating them as necessary. Redundant columns were removed to simplify the dataset and improve the analysis' efficiency. This rigorous data cleansing and preparation was critical to the accuracy of the following predictive modeling.

Normalization procedures were then performed to the data to ensure that the dataset's value range was consistent and appropriate for modeling. Following this, the dataset was divided into training and testing sets. This split is a basic machine learning approach that allows model performance to be evaluated on previously unseen data.

The analysis used five distinct classifiers: Random Forest, Logistic Regression, Decision Tree, Naive Bayes, and Gradient Boosting. Each of these models was chosen because of their shown efficacy in categorization tasks.

Their performance was extensively assessed using many criteria, including recall, F-score, accuracy, precision, and ROC curve. These measurements offered a thorough insight of each model's capacity to forecast properly.

After a thorough analysis of their results, the Random Forest Classifier and Gradient Boosting were determined to be the best appropriate models when above measurements are being analyzed.

REFERENCES

[1] *Machine Learning Random Forest Algorithm - JavatPoint*. (n.d.). www.javatpoint.com. https://www.javatpoint.com/machine-learning-random-forest-algorithm

[2] Nettleton, D. (2014). Data modeling. In *Elsevier eBooks* (pp. 137–157). https://doi.org/10.1016/b978-0-12-416602-8.00009-1

[3] *1.10. Decision Trees*. (n.d.). Scikit-learn. https://scikit-learn.org/stable/modules/tree.html

[4] Ray, S. (2023, December 1). *Naive Bayes Classifier explained: Applications and practice problems of Naive Bayes Classifier*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/

[5] Saini, A. (2024, January 10). *Gradient Boosting Algorithm: A complete guide for beginners*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/09/gradient-boosting-algorithm-a-complete-guide-for-beginners/