**Machine Learning using Python**

**Project Report**

**Customer Segmentation With Machine Learning**

| **College Name :** | **Government College of Engineering Bodinayakanur** |
| --- | --- |
| **Team Name :** | **Cluster Minds** |

| **Team Members:** | 1. **21CS134 -** 2. **22CSL09 -** 3. **21CS132-** 4. **21CS117-** | **Periasamy M**  **Veerasurya A**  **Patturaj K**  **Jayaprakash N** |
| --- | --- | --- |

1. **ABSTRACT**

Our project demonstrates a machine learning approach for customer segmentation. By analyzing customer data such as age, income, and spending habits, various models categorize customers into distinct segments. This allows businesses to tailor their marketing strategies more effectively. The pipeline includes data cleaning, feature scaling, and training several models to find the best fit. Predictions for new customers help in targeted marketing and resource allocation. The results support improved customer engagement and sales growth.

1. **MODEL**

**Overview**

This report summarizes the performance of various classification models applied to the customer segmentation dataset. The primary goal is to classify customers into different segments based on features such as Age, Annual Income, and Spending Score. We used several models, evaluated their performance using metrics like accuracy, precision, recall, and F1-score, and incorporated techniques to handle class imbalance and optimize hyperparameters.

**Models and Techniques Used**

Logistic Regression

Decision Tree Classifier

Random Forest Classifier

Support Vector Classifier (SVC)

K-Nearest Neighbors (KNN)

Gradient Boosting Classifier

1. **ALGORITHM IMPLEMENTATION**

**Classification Algorithm**

A classification algorithm is a type of supervised learning method used to categorize data into predefined classes or labels based on its features. The overall goal is to train a model that can predict the class label of unseen data with high accuracy. Here’s a detailed explanation of the classification algorithm's theoretical implementation:

**a. Data Collection :** Gather a dataset containing features (input variables) and corresponding class labels (output variables). The dataset should be representative of the problem you aim to solve.

**b. Data Cleaning :** Handle missing values and outliers. Missing values can be filled using techniques such as mean imputation or forward fill, or removed altogether. Outliers should be examined to determine if they are errors or legitimate extreme values.

**c. Feature Selection :** Identify which features (columns) are relevant to the classification task. This can involve domain knowledge, correlation analysis, or automated methods to reduce dimensionality and improve model performance.

**d. Data Encoding :** Convert categorical variables into numeric formats that algorithms can process. Techniques such as Label Encoding (assigning a unique integer to each category) or One-Hot Encoding (creating binary columns for each category) are commonly used.

**e. Data Splitting :** Divide the dataset into training and test sets. The training set is used to train the model, while the test set is reserved for evaluating the model’s performance. Common splits include 80/20 or 70/30 ratios.

**f.** **Model Training:** Fit the model to training data by learning patterns that separate different classes. Common algorithms include Logistic Regression, Decision Trees, and Support Vector Machines.

**g**. **Model Evaluation:** Assess performance using metrics like accuracy, precision, recall, and F1-score on a test set.

**h.Prediction:** Use the trained model to classify new, unseen data based on learned patterns.

1. **PREDICTION COMPARISON REPORT**

The model comparison reveals varied performance:

* **Logistic Regression** achieved 30.77% accuracy with balanced performance across classes.
* **Decision Tree** performed slightly better at 33.33%, excelling with Segment\_A ,Segment\_B but some difficulty in analyzing the Segment\_C.
* **Random Forest**, **SVC**, **KNN**, and **Gradient Boosting** all showed low accuracy (around 25%) and struggled with specific segments, indicating potential issues with data handling or model parameters.

1. **FINAL PREDICTION**

The **'Gradient Boosting'** model is the most promising with an accuracy of 44.44%. It effectively identifies Segment\_A, showing its strong capability to handle specific classifications. Its relatively higher accuracy compared to other models indicates that it can capture important patterns in the data. The model’s interpretability is also a significant advantage, making it easier to understand and explain decisions. With further refinement, such as tuning hyperparameters and addressing any overfitting, the Decision Tree has the potential to deliver even better performance and generalization across all segments, making it a strong candidate for practical use.

1. **CONCLUSION**

The project demonstrates that the **'Gradient Boosting'** model, with its 44.44% accuracy, is the most effective for classifying customer segments. Its interpretability and relative performance make it suitable for practical applications, such as targeted marketing strategies and customer segmentation in businesses. By refining the model and addressing limitations, it can enhance decision-making processes and improve customer engagement. Implementing this model in a real-world setting can help companies tailor their offerings, optimize marketing campaigns, and better understand customer behavior, ultimately leading to increased customer satisfaction and business growth

**Retail Marketing**: Retail store could identify high-value customers using the algorithm and it helps to create personalized marketing campaigns, such as special discounts or product recommendations, tailored to each segment.