CSE422 Lab Project Report

Project Name: Customer Category Classifier

Group No: 06, Section: 26, Summer 2025	
ID	**Name**
22299088	Promit Debnath
23101445	Sifatul Karim

Table of Contents

Introduction:

In today's competitive business world, understanding customers and their behavior is essential for organizations to design better marketing strategies, improve customer experience, and increase profitability.
This project focuses on Customer Segmentation, the process of dividing customers into distinct groups based on their characteristics and spending behavior. The dataset contains features such as age, gender, marital status, profession, family size, education, and spending scores etc. The aim is to evaluate multiple machine learning models, compare their performance, and explore clustering for unsupervised insights. The project applies a combination of supervised machine learning models (such as Decision Tree, Logistic Regression and Neural Networks) and unsupervised learning techniques (K-Means clustering).
Ultimately, the project seeks to answer: Which model can most effectively classify customer segments, and what underlying factors influence these predictions?

Dataset description:

1. How many features?

– There are 10 input features and a target. Input features: ID, Gender, Ever_Married, Age, Graduated, Profession, Work_Experience, Spending_Score, Family_Size, Var_1 and Segmentation (target)

2. Classification or regression problem? Why do you think so?

– This project is a classification problem because the target variable, segmentation, consists of discrete categorical outcomes such as A, B, C and D. In machine learning, regression problems are used when the target variable is continuous and numerical, whereas classification problems are used when the target variable represents categories or classes. Since the goal here is to predict which category a customer will fall into, it clearly falls under classification.

3. How many data points?

– There are 8068 datapoints in this dataset.

4. What kind of features are in your dataset? (Quantitative / Categorical)

– The dataset has both quantitative (numerical) and qualitative features.

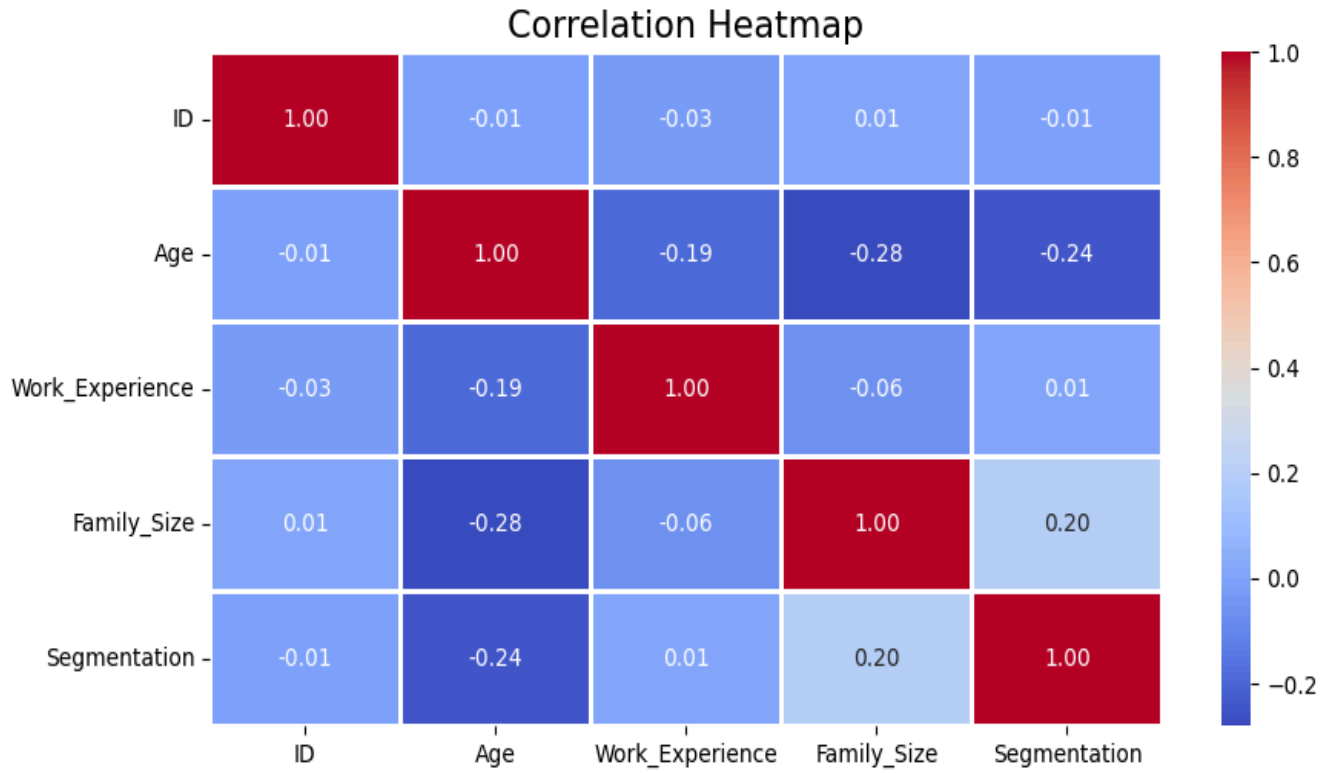
Quantitative: Age, Work_Experience, Family_size

Categorical: Gender, Ever_Married, Graduated, Profession, Spending_Score, Var_1

5. Do you need to encode the categorical variables, why or why not?

– Yes, categorical features need encoding because ML models require numerical input. Without encoding, the model cannot interpret text-based categories. The choice of encoding method depends on the type of categorical variable.

6. Correlation of all the features (input and output features) (apply heatmap using the seaborn library)



7. What do you understand after the correlation test?

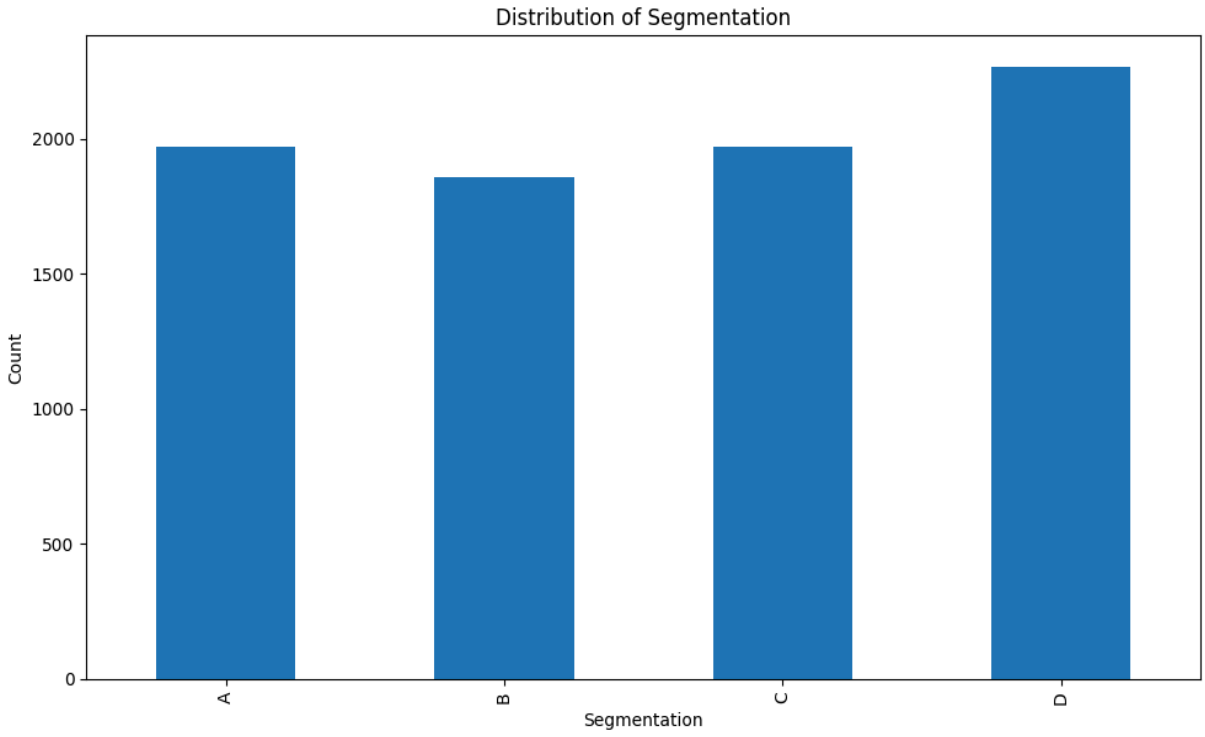
– The correlation heatmap shows that Age has a negative correlation with segmentation, meaning older customers are less likely to fall into certain segments. Family_Size has a positive correlation, so it can be said that larger families tend to belong to specific segments. Work_experience shows almost no correlation with segmentation, making it a weaker feature. ID has no meaningful correlation and should be dropped since it does not contribute to classification.

8. Imbalanced Dataset:

- For the output feature, do all unique classes have an equal number of instances or not?

– For the target variable, the classes do not have an equal number of instances which means this is an imbalanced dataset. Segment D is the class that has the most number of instances, while Segment B has the least number of instances.

- Represent using a bar chart of N classes (N=number of classes you have in your dataset).



Dataset pre-processing:

1. Null / Missing Values:

- Fault: The dataset contained missing values, which can lead to errors in model training and clustering.
- Solution: applied imputation, replacing missing entries with the mean or most frequent values. This ensured the dataset had no null values.

2. Categorical Values

- Fault: ML requires numerical data, but the dataset had categorical features.
- Solution: used label encoding to convert categorical variables into numeric values. This made the dataset compatible with clustering.

3. Feature Scaling

- Fault: The dataset had features on different scales (e.g., age vs. income), which could bias the models.
- Solution: applied Standardization, bringing all features to a similar scale. This ensured all features contribute equally to the model.

Dataset splitting:

Using the entire dataset for training causes overfitting and gives unreliable evaluation.

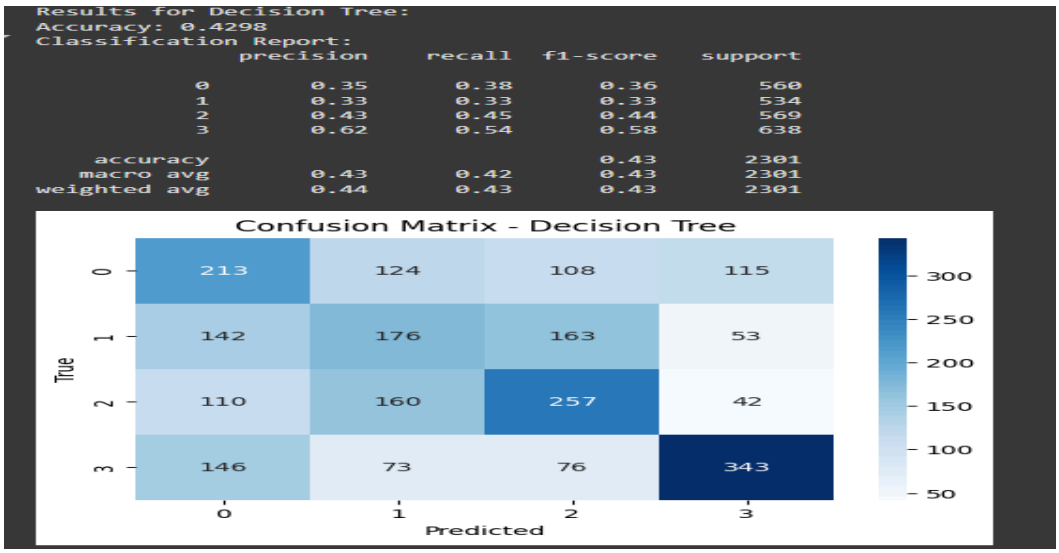
The dataset was split into 70% for the training set, test set 30% for the final evaluation and stratified on the target.

Model training & testing (Supervised):

Used models: Decision tree, Logistic regression and Neural Network

- Decision tree:
  Decision Tree model was trained, which splits data based on feature thresholds to form a tree-like structure. This model is easy to interpret and handles both categorical and numerical data. It can automatically capture non-linear relationships between features and the target variable.

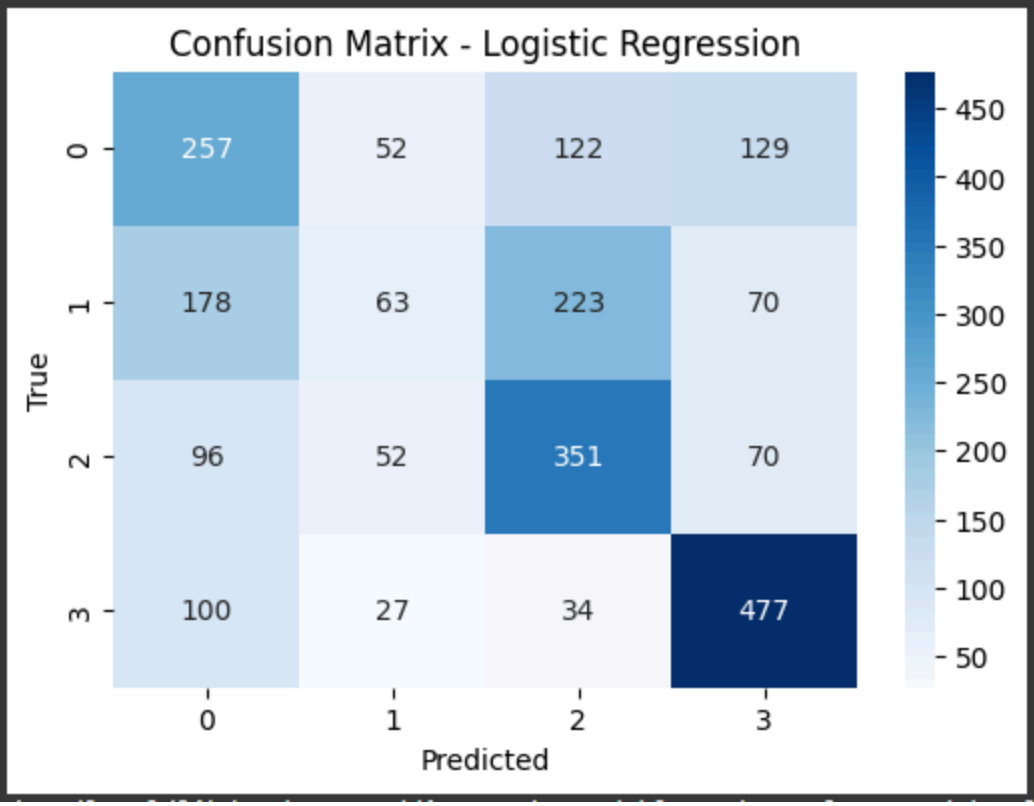
```
Results for Decision Tree:
Accuracy: 0.4298
Classification Report:
              precision    recall  f1-score   support

           0       0.35      0.38      0.36       560
           1       0.33      0.33      0.33       534
           2       0.43      0.45      0.44       569
           3       0.62      0.54      0.58       638

    accuracy                           0.43      2301
   macro avg       0.43      0.42      0.43      2301
weighted avg       0.44      0.43      0.43      2301
```



- Logistic regression:

  Target variable Segmentation has 4 classes (A, B, C, D), making this a multi-class classification problem. Logistic regression can handle this through one-vs-rest or multinomial approaches. This model provides coefficients that show the relationship between features and the target variable, making it easier to understand what drives customer segmentation.

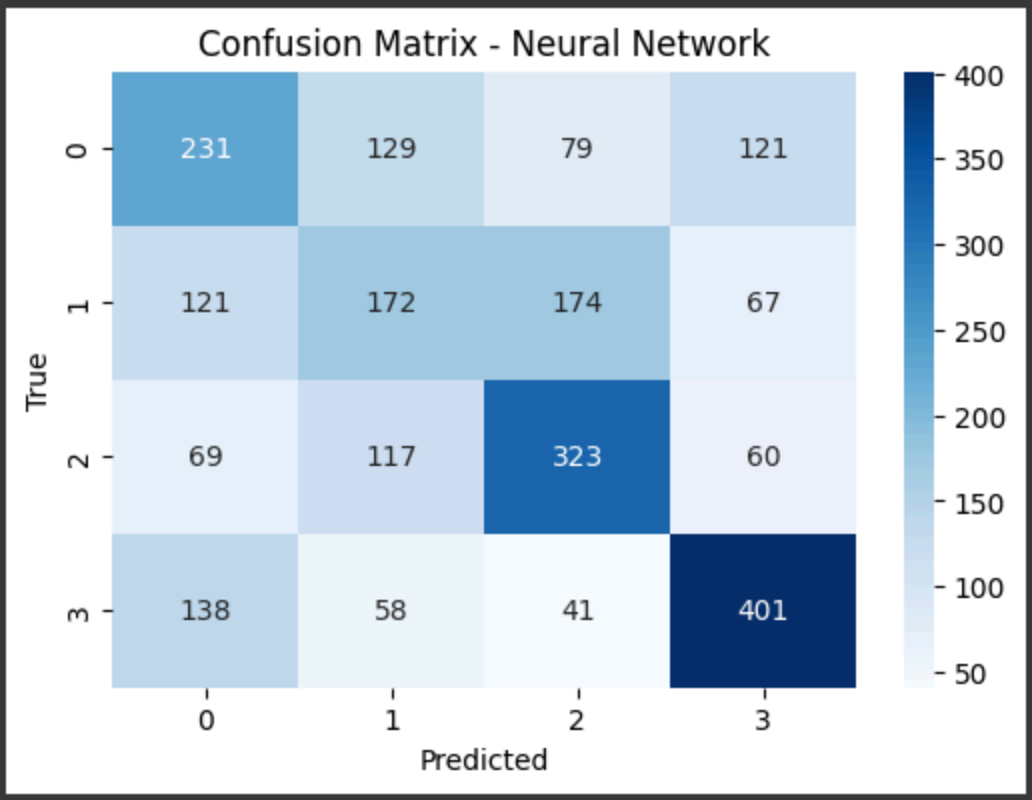
```
Results for Logistic Regression:
Accuracy: 0.4989
Classification Report:
              precision    recall  f1-score   support

           0       0.41      0.46      0.43       560
           1       0.32      0.12      0.17       534
           2       0.48      0.62      0.54       569
           3       0.64      0.75      0.69       638

    accuracy                           0.50      2301
   macro avg       0.46      0.49      0.46      2301
weighted avg       0.47      0.50      0.47      2301
```



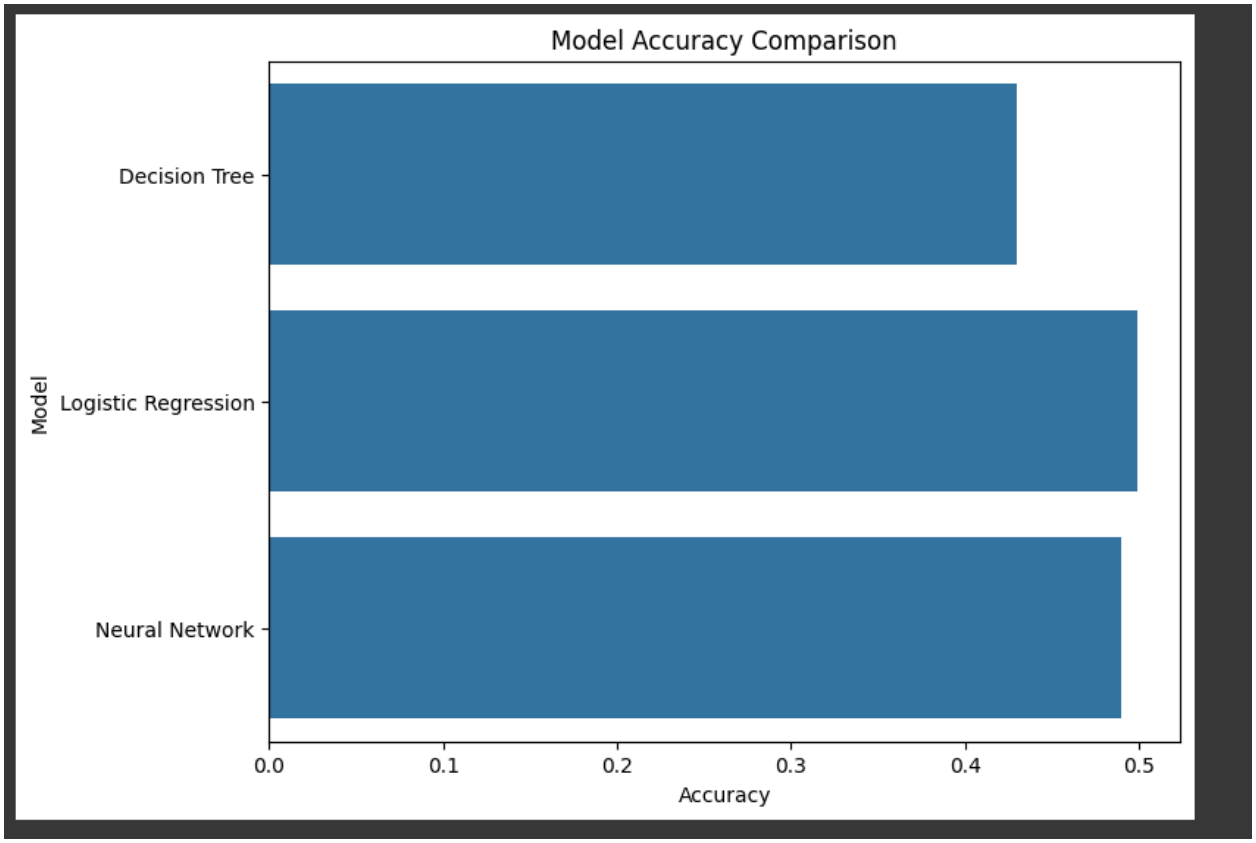
- Neural Network :

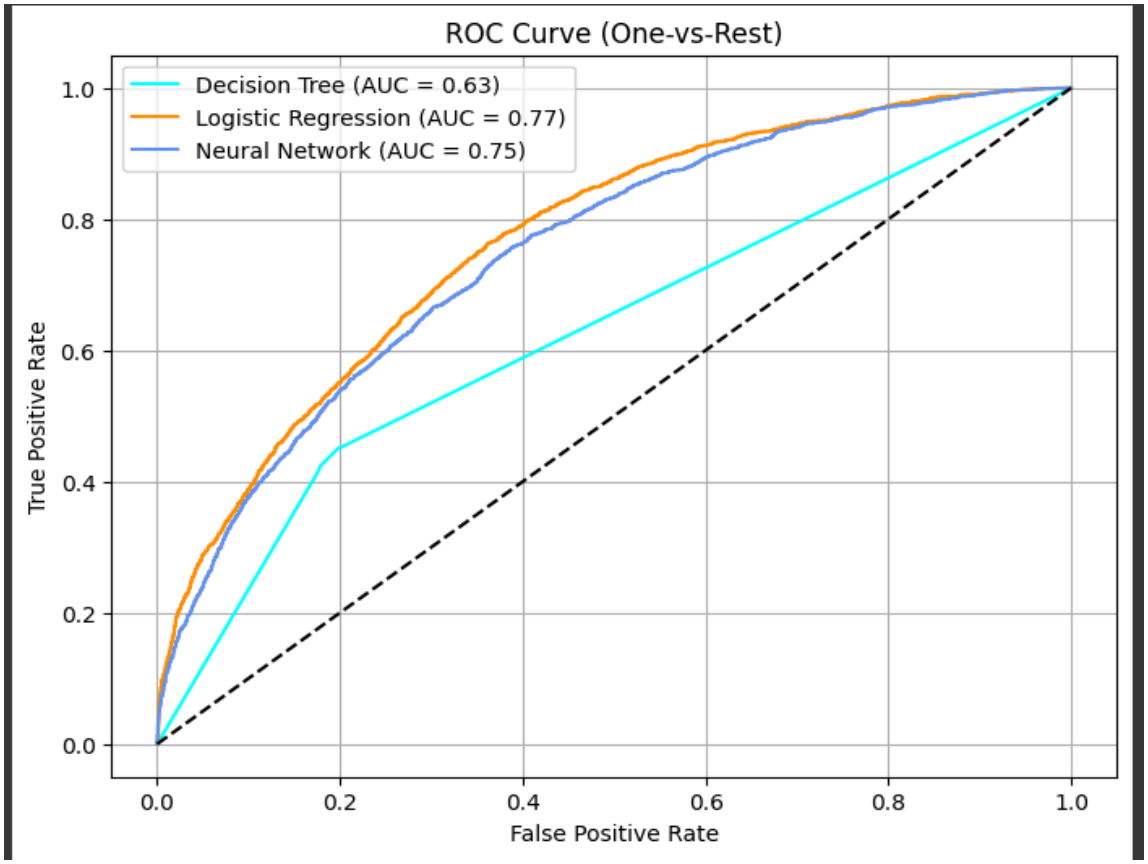
  This model automatically discovers relevant patterns and feature interactions without manual engineering. For complex problems like customer segmentation, neural networks often achieve better predictive performance.

```
Results for Neural Network:
Accuracy: 0.4898
Classification Report:
              precision    recall  f1-score   support

           0       0.41      0.41      0.41       560
           1       0.36      0.32      0.34       534
           2       0.52      0.57      0.54       569
           3       0.62      0.63      0.62       638

    accuracy                           0.49      2301
   macro avg       0.48      0.48      0.48      2301
weighted avg       0.49      0.49      0.49      2301
```



Model selection/Comparison analysis:

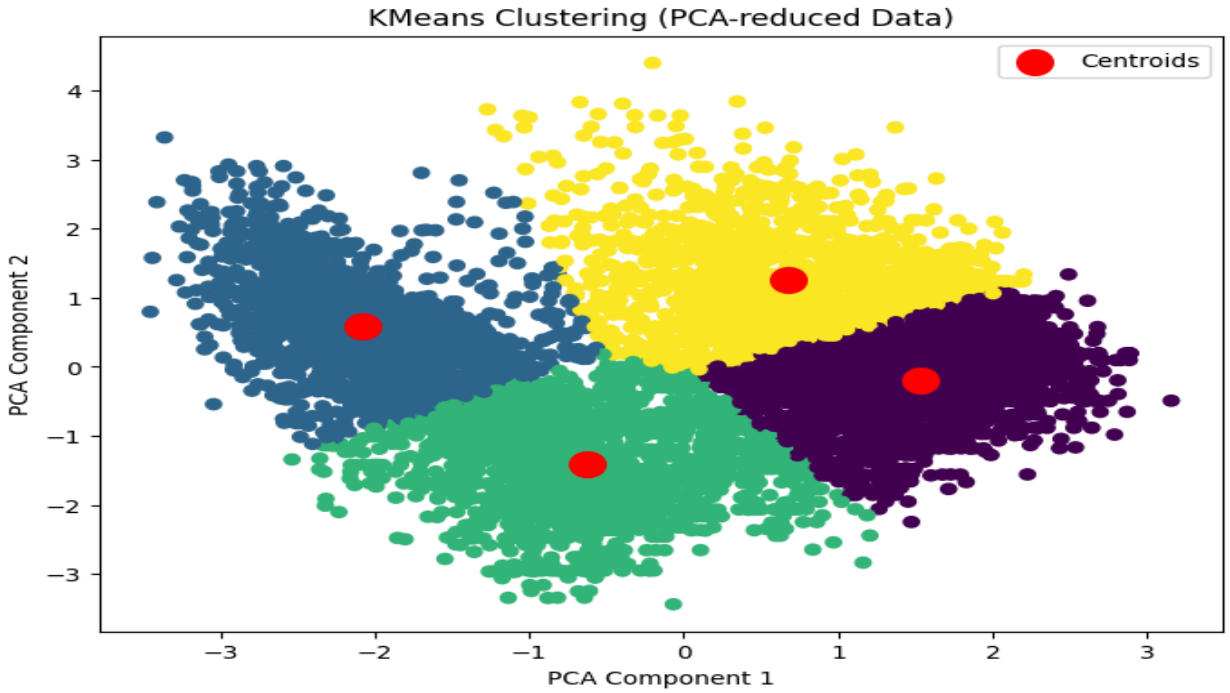




Based on the observation, Logistic Regression has the best accuracy for this customer segmentation. This model performs best in overall correctness of predictions. Decision Tree likely has the worst performance.

The Neural Network's slightly lower accuracy suggests the problem might not have enough complex patterns to justify its complexity.

Unsupervised: KMeans:



Conclusion:

The objective of this project was to classify customers into one of four segments (A, B, C, D) using their demographic and behavioral data. The results show that the Logistic Regression model achieved the highest accuracy at 50%, closely followed by the Neural Network at 49%. The Decision Tree model had the lowest performance with an accuracy of 43%. Clearly, Logistic Regression outperformed both Decision Tree and Neural Network, demonstrating that for the customer segmentation dataset, a simpler linear model captured the patterns most effectively. The Neural Network was likely overly complex for this problem, while the Decision Tree probably overfit the training data.

Simpler models can outperform complex ones when the data has clear linear decision boundaries between classes. The result suggests that customer segments can be effectively distinguished using linear relationships between the features.