**PREDICTIVE ANALYSIS PROJECT REPORT**

**Implementation of ML Algorithms**

**Using**

**R Programming**

Submitted by

**Hardik**

Registration No.: -12109777

Roll No.- 14

Course Code: - **INT234**

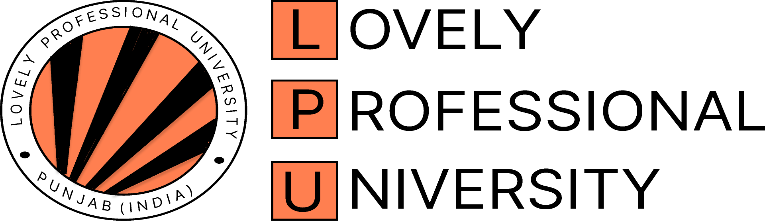
Under the Guidance of

**Prof. Tanima Thakur (28373)**

**Discipline of CSE/IT**

**School of Computer Science and Engineering(P-132)**

**Lovely Professional University, Phagwara**



**Student Declaration**

# I, Hardik, Registration Number 12109777, a Bachelor of Technology (B. Tech) student in the Computer Science and Engineering/Information Technology discipline at Lovely Professional University, Punjab, hereby formally declare that the work presented in this project report is my original work and has not been submitted previously at this or any other institution for academic qualification or certification. I affirm that all data, results, and analyses contained in this report are genuine and based on my own independent research and effort.

**Name of student: Hardik**

**Registration Number: 12109777**

**Dated: 10/11/2024**

# ACKNOWLEDGMENT

I would like to express my sincere gratitude to my professor, **Tanima Thakur**, for their invaluable support, guidance, and encouragement throughout the course of this project. Their expertise and constructive feedback played a significant role in enhancing the quality and depth of my work.

I also wish to extend my heartfelt thanks to **Lovely Professional University** for providing me with the necessary resources and environment to pursue this project. The opportunity to work under the guidance of such esteemed faculty has been a remarkable learning experience.

Furthermore, I am deeply grateful to my friends for their constant support and insightful feedback. Their suggestions were essential in helping me refine my ideas and approach.

Thank you all for your unwavering support and encouragement.

**Name of student: Hardik**

**Registration Number: 121009777**

## Dated: 10/11/2024

1. **📌Introduction**

The purpose of this project is to perform customer segmentation analysis using a dataset containing demographic and behavioural information. Customer segmentation helps companies understand the unique needs of different customer groups, which can optimize marketing strategies, improve customer retention, and increase revenue by providing more personalized experiences.

**2. Objective/Scope of Analysis:**    
  
This project aims to categorize customers based on their purchasing behaviours and demographics, enabling the identification of high-spending segments. By categorizing spending scores into "Low," "Medium," and "High," we can draw actionable insights to improve customer engagement. Our analysis will employ machine learning algorithms to predict spending behaviour, evaluate their performance, and choose the most effective model.

**3. 🗄️Dataset**

The dataset includes the following features:

1. **CustomerID**: Unique identifier for each customer.
2. **Genre**: Customer gender.
3. **Age**: Age of the customer, used to observe spending trends across age groups.
4. **Annual Income**: Customer's annual income, which serves as an indicator of purchasing power.
5. **Spending Score**: Based on customer behavior and purchasing patterns, categorized into "Low," "Medium," and "High." This is the target variable for predictive analysis.

The dataset offers a diverse set of demographic and behavioral characteristics that provide a robust foundation for customer segmentation.

**4.** **🧠 Algorithms Applied**

The following machine learning algorithms were used to predict and categorize customer spending behavior:

1. **K-Nearest Neighbors (KNN)**
   * **Description**: A distance-based algorithm that assigns each test observation to the most common category among its k-nearest neighbors. This approach is effective in datasets with well-defined separations between categories.
   * **Implementation**: With k=5k = 5k=5, "Age" and "Annual Income" were used as features. KNN assigns spending categories based on the majority label among the five closest neighbors.
   * **Results**: KNN achieved moderate accuracy and precision but may struggle with overlapping or densely packed clusters.
2. **Naive Bayes**
   * **Description**: A probabilistic classifier based on Bayes' theorem that assumes feature independence, making it suitable for categorical variables and well-separated classes.
   * **Implementation**: Using "Spending Score" as a function of "Age" and "Annual Income," Naive Bayes calculates the probability of each class and assigns the highest-probability label to each observation.
   * **Results**: Naive Bayes showed solid recall and precision, though feature independence may limit its applicability in real-world settings.
3. **Decision Tree**
   * **Description**: A tree-like model that splits data into branches based on feature values, ultimately leading to a decision or category, making it interpretable and insightful for identifying key features.
   * **Implementation**: The Decision Tree splits data using "Age" and "Annual Income" until pure nodes or a stopping criterion is reached.
   * **Results**: High accuracy and interpretability, though the model may overfit with limited data or numerous categories.
4. **Logistic Regression**
   * **Description**: Predicts binary outcomes, which is useful in binary classification problems or when the target variable can be binarized.
   * **Implementation**: Binarized "Spending Score" into "High" vs. "Low/Medium" categories. The logistic function maps predictions to a probability score, classifying scores above 0.5 as "High" spending.
   * **Results**: Strong performance on binarized spending predictions, though less effective for multi-category segmentation.
5. **K-Means Clustering**
   * **Description**: An unsupervised algorithm for segmenting data into k clusters based on feature similarity.
   * **Implementation**: Setting k=3k = 3k=3, K-Means clustered customers by similarity in "Age" and "Annual Income," roughly mapping clusters to "Low," "Medium," and "High" spending categories.
   * **Results**: While not intended for supervised prediction, K-Means revealed natural clusters within customer demographics.

**5.** **📊 Performance Comparison**

Each model was evaluated using metrics like accuracy, precision, recall, F1 score, and error rate to determine the best-performing algorithm for customer segmentation.

A screenshot of a graph

Description automatically generated

**5.** **Algorithms Used**

**K-Nearest Neighbours (K-NN):**

A screenshot of a computer

Description automatically generated

This R code implements a K-Nearest Neighbors (KNN) model to predict customer spending behavior based on demographic features like age and annual income. It trains the model using a training dataset and makes predictions on a test dataset. The predicted labels are then adjusted to match the factor levels of the actual labels. The performance of the KNN model is evaluated using various metrics such as accuracy, precision, recall, F1 score, and error rate. These metrics are reshaped for visualization, and a bar plot is created to display the evaluation results, helping to assess the model's performance.

**Naive Bayes:**

A screenshot of a computer

Description automatically generated

This R code implements a Naive Bayes model to predict customer spending behavior based on age and annual income. The model is trained using the training dataset and makes predictions on the test dataset. The predicted labels are compared to the actual labels, and the performance of the model is evaluated using metrics like accuracy, precision, recall, F1 score, and error rate. These evaluation metrics are reshaped for easier visualization and then plotted in a bar chart to assess the effectiveness of the Naive Bayes model.

D**ecision Tree:**

**A screenshot of a computer

Description automatically generated**

This code implements a Decision Tree model for classifying customer spending behavior based on the features "Age" and "AnnualIncome". It first trains the model using the rpart function with a classification method, then makes predictions on the test data. The calculate\_metrics function is used to evaluate the model's performance by calculating various metrics like accuracy, precision, recall, F1 score, and error rate. The results are then reshaped into a long format using the gather function, and the evaluation metrics are visualized in a bar plot using ggplot2 to assess the model's effectiveness in predicting the spending behavior categories.

**Logistic Regression:**

**A screenshot of a computer program

Description automatically generated**

This code implements a Decision Tree model for classifying customer spending behavior based on the features "Age" and "AnnualIncome". It first trains the model using the rpart function with a classification method, then makes predictions on the test data. The calculate\_metrics function is used to evaluate the model's performance by calculating various metrics like accuracy, precision, recall, F1 score, and error rate. The results are then reshaped into a long format using the gather function, and the evaluation metrics are visualized in a bar plot using ggplot2 to assess the model's effectiveness in predicting the spending behavior categories.

**K-Means Clustering:**

**A screenshot of a computer screen

Description automatically generated**

This code performs K-Means clustering on customer data using the features "Age" and "AnnualIncome" to group customers into three clusters. The number of clusters is set to 3, with each customer assigned to one of the clusters. The cluster assignments are mapped to spending categories ("Low", "Medium", "High") for evaluation purposes, and the calculate\_metrics function is used to assess the clustering results by comparing the clusters with the actual spending scores. The performance is evaluated using metrics such as accuracy, precision, recall, and F1 score, and the results are visualized in a bar plot to present the effectiveness of the clustering model.

**Result:**

**A screenshot of a computer

Description automatically generated**

This code combines the evaluation metrics from multiple models (KNN, Naive Bayes, Decision Tree, Logistic Regression, and K-Means) into a single data frame, then reshapes the data for plotting. The ggplot function is used to create a bar plot that compares the performance of each model across various evaluation metrics (such as accuracy, precision, recall, F1 score, etc.). The plot is faceted by the type of metric, allowing for a clear visual comparison of how each model performs in different aspects of evaluation. The result is a comprehensive visualization that helps to assess and compare the strengths of each model in the analysis.

**7.** **🔍 Conclusion**

Each algorithm demonstrated unique strengths:

* **Decision Tree and Naive Bayes** models excelled in interpretability and provided good predictive accuracy.
* **KNN and Logistic Regression** offered alternative insights for binary and multi-class spending predictions.
* **K-Means Clustering** revealed natural groupings, adding value through unsupervised segmentation.

For practical applications in customer segmentation, the **Decision Tree model** may be most advantageous due to its high accuracy and straightforward interpretability.

**8. GitHub Link**

[**Hardik-Girdhar/Machine-Learning-R-Algorithms**](https://github.com/Hardik-Girdhar/Machine-Learning-R-Algorithms)

**9.** **LINKEDIN Post**

[hardik-girdhar\_datascience-machinelearning-customeranalytics-activity](https://www.linkedin.com/posts/hardik-girdhar_datascience-machinelearning-customeranalytics-activity-7261427095732379648-MRSQ?utm_source=share&utm_medium=member_desktop)

