**TABLE OF CONTENTS**

1. Table of Contents 1
2. Table of Figures 4
3. List of Tables 4

[1 INTRODUCTION 5](#_Toc144848644)

[1.1 EXISTING SYSTEM 6](#_Toc144848645)

[1.2 PROPOSED SYSTEM 6](#_Toc144848646)

[2 LITERATURE SURVEY 8](#_Toc144848647)

[2.1 RELATED WORK 8](#_Toc144848648)

2.1.1 DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

2.1.2 Applications of DBSCAN in Customer Segmentation

[2.2 SYSTEM STUDY 9](#_Toc144848652)

[3 DESIGN 10](#_Toc144848653)

[3.1 SYSTEM REQUIREMENTS 10](#_Toc144848654)

[3.1.1 Software Requirements 10](#_Toc144848655)

[3.1.2 Hardware Requirements 11](#_Toc144848656)

[3.2 UML 12](#_Toc144848657)

[3.2.1 Activity Diagram 12](#_Toc144848658)

[3.2.2 Use Case Diagram 13](#_Toc144848659)

[3.2.3 Class Diagram 14](#_Toc144848660)

[3.2.3 Sequence Diagram 1](#_Toc144848660)5

[3.2.3 Deployment Diagram 1](#_Toc144848660)6

[4 IMPLEMENTATION 15](#_Toc144848661)

[4.1 MODULES 15](#_Toc144848662)

[4.1.1 Dataset Collection 15](#_Toc144848663)

[4.1.2 Models 16](#_Toc144848664)

4.1.3 Implementaion Steps 16

[4.1.4 Performance Metrics 17](#_Toc144848665)

[4.2 OVERVIEW TECHNOLOGY 18](#_Toc144848666)

[4.2.1 Data Preprocessing 18](#_Toc144848667)

[4.2.2 Data Collection 18](#_Toc144848668)

[4.2.3 Clustering Analysis 18](#_Toc144848669)

[4.2.4 Visualization 19](#_Toc144848670)

[4.2.5 Model Testing 2](#_Toc144848671)0

[5 RESULTS 26](#_Toc144848679)

[6 CONCLUSION 28](#_Toc144848688)

[7 FUTURE SCOPE 29](#_Toc144848689)

[7.1 Integration with big data 29](#_Toc144848690)

[7.2 Enhanced Personalization 29](#_Toc144848691)

[7.3 Predictive Analytics 29](#_Toc144848692)

[7.4AI and Automation29](#_Toc144848692)

[8 BIBLIOGRAPHY 30](#_Toc144848693)

**TABLE OF FIGURES**

**LIST OF TABLES**

[Table 5.1‑1 Test Cases 22](#_Toc144831575)

[Table 5.2‑1 Sepsis Accuracy 22](#_Toc144831576)

[Table 5.2‑2 Diabetes Accuracy 23](#_Toc144831577)

[Table 5.2‑3 Autism Spectrum Disorder Accuracy 23](#_Toc144831578)

[Table 5.2‑4 Recall for Sepsis 23](#_Toc144831579)

[Table 5.2‑5 Recall for Diabetes 24](#_Toc144831580)

[Table 5.2‑6 Recall for Autism Spectrum Disorder 24](#_Toc144831581)

[Table 5.2‑7 Precision for Sepsis 24](#_Toc144831582)

[Table 5.2‑8 Precision for Diabetes 25](#_Toc144831583)

[Table 5.2‑9 Precision for Autism Spectrum Disorder 25](#_Toc144831584)

[Table 5.2‑10 F1-Score for Sepsis 25](#_Toc144831585)

[Table 5.2‑11 F1-Score for Diabetes 26](#_Toc144831586)

[Table 5.2‑12 F1-Score for Autism Spectrum Disorder 26](#_Toc144831587)

[Table 6.1‑1 SEPSIS ACCURACY 27](#_Toc144831588)

[Table 6.2‑11COMPARISON OF TESTING ACCURACY WITH DIFFERNT MACHINE LEARNING MODELS FOR DIABETES 28](#_Toc144831589)

[Table 6.3‑1 1COMPARISON OF TESTING ACCURACY WITH DIFFERNT MACHINE LEARNING MODELS FOR AUTISM SPECTRUM DISODER 30](#_Toc144831590)

# INTRODUCTION

Customer segmentation is a vital process in modern business strategy, enabling companies to categorize their customer base into distinct groups based on shared characteristics. This segmentation allows for more personalized marketing, improved customer service, and better resource allocation, ultimately driving higher customer satisfaction and profitability. Traditional clustering methods, such as K-means, often fall short in handling real-world customer data, which may contain noise, outliers, and clusters of varying shapes and densities. To address these challenges, this project leverages the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm. Unlike K-means, DBSCAN does not require the number of clusters to be predefined and can effectively manage noisy data by identifying and excluding outliers. This project focuses on applying DBSCAN to segment customers based on key attributes, aiming to uncover meaningful patterns and relationships within the data. By doing so, businesses can develop more precise targeting strategies, enhance customer engagement, and optimize overall operational efficiency.

**EXISTING SYSTEM:**

The existing systems for customer segmentation commonly utilize algorithms like K-means or hierarchical clustering. While these methods are widely used, they have significant limitations:

**K-means Clustering:**

* **Assumption of Spherical Clusters:** K-means assumes that all clusters are spherical and of equal size, which often doesn’t match real-world data distributions.
* **Predefined Number of Clusters:** K-means requires the number of clusters (K) to be specified beforehand, which can lead to suboptimal segmentation if the number is chosen incorrectly.
* **Sensitivity to Outliers:** K-means is highly sensitive to outliers. An outlier can significantly shift the cluster centroids, leading to incorrect segmentation.
* **Cluster Initialization Dependency:** The algorithm’s outcome heavily depends on the initial placement of centroids, which can result in different solutions with different initializations.

**Hierarchical Clustering:**

* **Computational Complexity:** Hierarchical clustering, which builds a tree-like structure of clusters, is computationally intensive, especially for large datasets.
* **Difficulty in Choosing the Right Cut-off:** Deciding where to cut the hierarchical tree to form clusters can be subjective and may not yield optimal segments.
* **Fixed Dendrogram Structure:** Once a dendrogram is constructed, it cannot be modified without rerunning the algorithm, making it inflexible for dynamic datasets.
* These limitations result in customer segments that might not accurately reflect real-world patterns, leading to ineffective marketing strategies, misallocation of resources, and poor customer experience.

## PROPOSED SYSTEM:

The proposed system utilizes the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm for customer segmentation. DBSCAN addresses the limitations of traditional methods by providing a more flexible and robust approach to clustering, especially in the presence of noise and varying cluster shapes.

**DBSCAN Clustering:**

* **Density-Based Clustering:** DBSCAN groups together points that are closely packed (i.e., have many neighbors within a given radius). Points in low-density regions are classified as noise or outliers.
* **No Need for Predefined Clusters:** Unlike K-means, DBSCAN does not require the number of clusters to be specified in advance. Instead, it identifies clusters based on the density of data points.
* **Handling of Arbitrary Shapes:** DBSCAN can identify clusters of any shape, as it relies on the concept of density connectivity. This is particularly useful for customer data, where segments may not conform to simple geometric shapes.
* **Robustness to Noise:** The algorithm naturally excludes noise and outliers by designating points that do not belong to any dense region as noise. This ensures that only meaningful clusters are identified.

# LITERATURE SURVEY

## RELATED WORK

We conducted a comprehensive review of ten research papers focused on customer data and e-commerce details using datasets from the real-time datasets.

### Customer segmentation using centroid based and density based clustering algorithms(IEEE):

It focuses on applying various data mining techniques to predict customer satisfaction in the mobile telecommunications industry. The study explores different methods and algorithms to analyze customer data and generate insights that can help improve service quality and customer experience.

### **Using DBSCAN to Identify Customer Segments with High Churn Risk on Amazon Consumer Behaviour Data(Research Gate):**

The focus of the paper is on leveraging the DBSCAN clustering algorithm to analyze and segment customers based on their behavior. The primary emphasis is on identifying customer segments that are at a high risk of churn. By focusing on this aspect, the study aims to help businesses better understand customer behavior and implement targeted retention strategies to reduce churn and improve customer loyalty in the e-commerce sector.

### A hybrid method for customer segmentation in Saudi Arabia restaurants using clustering, neural networks and optimization learning techniques.(Arab Journal of Science and Engineering):

The paper emphasizes the need for improved segmentation methods tailored to the specific characteristics and nuances of the Saudi Arabian market. By integrating clustering, neural networks, and optimization learning techniques into a hybrid methodology.

## SYSTEM STUDY

In this comprehensive project focused on customer segmentation involves analyzing and documenting the processes, methodologies, and technologies used to categorize customers into distinct groups based on various attributes and behaviors. This study helps in understanding the needs and preferences of different customer segments, enabling businesses to tailor their marketing, sales, and service strategies more effectively.

Customer segmentation is a critical process that involves dividing a company's customer base into distinct groups based on shared characteristics such as demographics, purchasing behaviour, and preferences. This system study examines the methodologies and technologies currently used for segmentation, highlighting both the strengths and limitations of existing approaches. Typically, data is gathered from various sources, including CRM systems, transaction records, and online interactions. However, challenges such as data quality issues, inconsistent segmentation accuracy, and scalability concerns often arise. The study suggests enhancing the system by integrating advanced machine learning techniques, like clustering algorithms, to create more dynamic and precise segments. Additionally, it recommends the adoption of a unified data platform to centralize and cleanse data, along with real-time analytics for capturing and analyzing customer behaviour instantaneously. By implementing these improvements, businesses can achieve more personalized and effective marketing strategies, leading to better customer satisfaction and increased revenue. The proposed enhancements are designed to be implemented in phases, ensuring a smooth transition and continuous improvement through ongoing evaluation and feedback. Overall, refining customer segmentation through advanced analytics and integrated systems is essential for maintaining a competitive edge in today’s data-driven market.

# DESIGN

## SYSTEM REQUIREMENTS

### Software Requirements:

#### **Programming Languages**:

* + Python: Python is the predominant language for machine learning and data analysis. You'll need Python for implementing your ML models, data preprocessing, and analysis.

#### Machine Learning Libraries:

* + For implementing DBSCAN and other machine learning algorithms pandas :Data manipulation and analysis .

Numpy : Numerical computations

Matplotlib/Seaborn : Data Visualisation

Scipy : Advanced mathematical functions and algorithms

* + TensorFlow or PyTorch: These deep learning frameworks are necessary if you plan to work with neural networks, which can be beneficial for complex disease prediction tasks.

#### **Data Analysis and Visualization**:

* + Pandas: For data manipulation and analysis.
  + Matplotlib and Seaborn: For data visualization.

#### **Google Colaboratory**:

* + Google colabs are great for interactive data exploration and model development.

.

.

### **Hardware Requirements:**

#### **Processor:**

* + A Multi-core processor (Intel i5 or higher , AMD Ryzen 5 ornh

#### GPU :

* + Having access to a GPU (Graphics Processing Unit) can significantly speed up training deep learning models.

#### **RAM**:

* + Adequate RAM, typically 16 GB or more, is essential for handling large datasets and training complex models.

#### Storage:

* + SSD with atleast 256 GB of free space for quick data access and processing .

#### Internet Connection:

* + A reliable internet connection is necessary for downloading datasets, libraries, and collaborating with team members if applicable.

Top of Form

## UML

### Use case:

They are usually used to illustrate the various actions taken by the application. They also show the several users who can carry out these functions. Use-case diagrams fall under behaviour diagrams due to their emphasis on the tasks carried out and the users (actors) who carry out these tasks.

A diagram of a company's flowchart

Description automatically generated

Figure 2 Use Case Diagram

### 3.2.3 Activity Diagram:

An activity diagram visually presents a series of actions or flow of control in a system like a flowchart or a data flow diagram. Activity diagrams are often used in business process modelling.

A diagram of a data processing process

Description automatically generated

Figure 3 Activity Diagram

### 3.2.3 Class Diagram

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

A diagram of a company

Description automatically generated

Figure 4 :Class Diagram

### Sequence Diagram

A sequence diagram consists of a group of objects that are represented by lifelines, and the messages that they exchange over time during the interaction.

A diagram of a data processing process

Description automatically generated

Figure 5 :Sequence Diagram

### Deployment Diagram

In UML, deployment diagrams model the physical architecture of a system. Deployment diagrams show the relationships between the software and hardware components in the system and the physical distribution of the processing.

A diagram of a customer service

Description automatically generated

Figure 5 :Deployment Diagram

# IMPLEMENTATION

## MODULES

### Dataset:

This dataset contains 541,909 entries with information about transactions. The columns include:

InvoiceNo: Transaction identifier.

StockCode: Item code for each product.

Description: Product description.

Quantity: Number of items purchased in the transaction.

InvoiceDate: Date and time of the transaction.

UnitPrice: Price per item.

CustomerID: Unique identifier for each customer.

Country: Customer's country.

This dataset could be used to analyze sales patterns, customer behaviour, and product performance.

#### 4.1.2 Machine Learning Models

##### K- means Clustering

K-means clustering is a popular machine learning algorithm used to segment customers in an e-commerce dataset into distinct groups based on similar characteristics. The objective is to divide customers into clusters where customers within each cluster are more similar to each other than to those in other clusters.

##### DBSCAN

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a clustering algorithm that groups together points in a dataset that are close to each other, while marking points that are in low-density regions as outliers. Unlike K-means, DBSCAN does not require the user to specify the number of clusters in advance, and it can handle clusters of arbitrary shapes.

### Performance Metrics

Performance metrics provide a quantitative assessment of model effectiveness. Key metrics include accuracy, measuring overall correctness; precision, indicating true positive rate; recall, gauging the ability to identify positives; and F1 score, blending precision and recall.

* Accuracy: Measures the ratio of correctly predicted instances to the total instances.

Formula: (TP + TN) / (TP + TN + FP + FN)

* Precision: Reflects the proportion of true positives among the predicted positive instances.

Formula: TP / (TP + FP)

* Recall (Sensitivity or True Positive Rate): Represents the ratio of true positives among actual positive instances.

Formula: TP / (TP + FN)

* F1 Score: Balances precision and recall, offering a harmonic mean of the two metrics. Formula: 2 \* (Precision \* Recall) / (Precision + Recall)
* Silhouette Score: The Silhouette Score is a metric used to evaluate the quality of a clustering algorithm, such as K-Means, DBSCAN, or hierarchical clustering. It measures how similar an object is to its own cluster compared to other clusters. The Silhouette Score helps determine the optimal number of clusters and the effectiveness of the clustering.

**Components of the Silhouette Score:**

Cohesion (a): The average distance between a data point and all other points within the same cluster. This measures how well the point fits within its own cluster.

Separation (b): The average distance between a data point and all points in the nearest cluster that is not its own. This measures how distinct or far away the point is from the nearest neighbouring cluster.

## OVERVIEW TECHNOLOGY

The customer segmentation project using DBSCAN can be broken down into the following key modules:

1**.Data Collection Module**

Description: Gathers customer data from various sources such as e-commerce platforms.

Functionality: Connects to data sources (APIs, databases). Extracts and consolidates data into a structured format.

Output: Raw customer dataset.

2**. Data Preprocessing Module**

Description: Prepares the data for clustering by cleaning, transforming, and normalizing it.

Functionality: Handles missing values (imputation, removal).Standardizes features (scaling, normalization).Feature selection and dimensionality reduction (PCA, etc.).

Output: Cleaned and standardized dataset ready for clustering.

3. **Clustering Module**

Description: Implements the DBSCAN algorithm to identify clusters within the dataset.

Functionality: Configures DBSCAN parameters (eps, min\_samples).Runs DBSCAN to identify core, border, and noise points. Assigns cluster labels to data points.

Output: Clustered dataset with labels.

4. **Cluster Analysis Module**

Description: Analyzes and interprets the characteristics of each identified cluster.

Functionality: Summarizes key features of each cluster (e.g., demographics, behavior).Identifies distinguishing factors between clusters. Detects outliers (noise) and understands their significance.

Output: Insights and characteristics of each customer segment.

5. **Visualization Module**

Description: Provides visual representations of the clusters and their characteristics.

Functionality: Generates scatter plots, heatmaps, and pair plots to visualize clusters. Displays the distribution of key features within and across clusters.

Output: Visual plots for analysis and reporting.

6. **Reporting and Insights Module**

Description: Compiles analysis results and visualizations into actionable reports.

Functionality: Creates reports summarizing key findings for stakeholders. Highlights actionable insights for marketing, product development, etc.

Output: Detailed reports and recommendations based on customer segmentation. These modules collectively form the backbone of the customer segmentation project using DBSCAN, ensuring that the process is structured, efficient, and results-oriented.

### Data Preprocessing

**Handle Missing Values:** Description and CustomerID: Since some entries are missing in these columns, decide on an appropriate strategy, such as imputing, filling with a placeholder (e.g., "Unknown"), or dropping rows/columns based on the importance and data quality.

**Data Type Conversion:**

InvoiceDate: Convert to a datetime format to enable date-based analysis.

CustomerID: Convert to a string to treat it as a categorical identifier rather than a numeric value.

**Remove Duplicates:**

Check for and remove duplicate entries, which might skew results if counted multiple times.

**Encode Categorical Variables:**

Country: Convert country names into numerical representations using one-hot encoding or label encoding, if needed for specific algorithms.

**Normalize/Standardize Numerical Features:**

Apply scaling to numerical features like Quantity and UnitPrice to improve model performance, especially if using distance-based algorithms.

### Data Splitting

A train-test split is a technique in data science where a dataset is divided into two subsets: the training set and the testing set.

**Training Set:** This portion (usually 70-80%) is used to train a machine learning model. The model learns patterns, relationships, or associations within this data to make predictions or classifications.

**Testing Set:** The remaining portion (usually 20-30%) is held back to evaluate the model’s performance. This subset acts as new, unseen data, helping to test how well the model generalizes to new inputs.

A good train-test split helps prevent overfitting and provides an unbiased evaluation of the model's effectiveness.

### Algorithm selection

DBSCAN is often more suitable than K-means for customer segmentation in transactional data due to its flexibility and robustness in handling outliers and complex cluster shapes. Unlike K-means, which requires the user to specify the number of clusters (k), DBSCAN automatically discovers clusters based on data density, making it particularly useful when the natural number of customer segments isn’t known. Additionally, DBSCAN identifies and excludes noise points, making it effective for datasets with outliers that could distort clusters if using K-means.

In general, DBSCAN excels with datasets containing irregular cluster shapes and varying densities, while K-means is most effective with well-separated, spherical clusters and evenly sized groups. Lastly, DBSCAN performs best with moderate data sizes, whereas K-means can scale efficiently to very large datasets, making each suitable for different clustering needs.

### Hyperparameter Tuning

Key Hyperparameters: eps (maximum distance between two points to be considered neighbors) and min\_samples (minimum points required to form a cluster)

**Steps:**

**eps Selection:**

Use the K-distance plot to select a good eps value. The K-distance plot shows the distance from each point to its k-th nearest neighbor. The "elbow" point on this plot can guide the choice of eps.

**Grid Search or Random Search:**

Use a grid or random search to explore different combinations of eps and min\_samples. Evaluate the clustering results using clustering evaluation metrics like the silhouette score or Davies-Bouldin index.

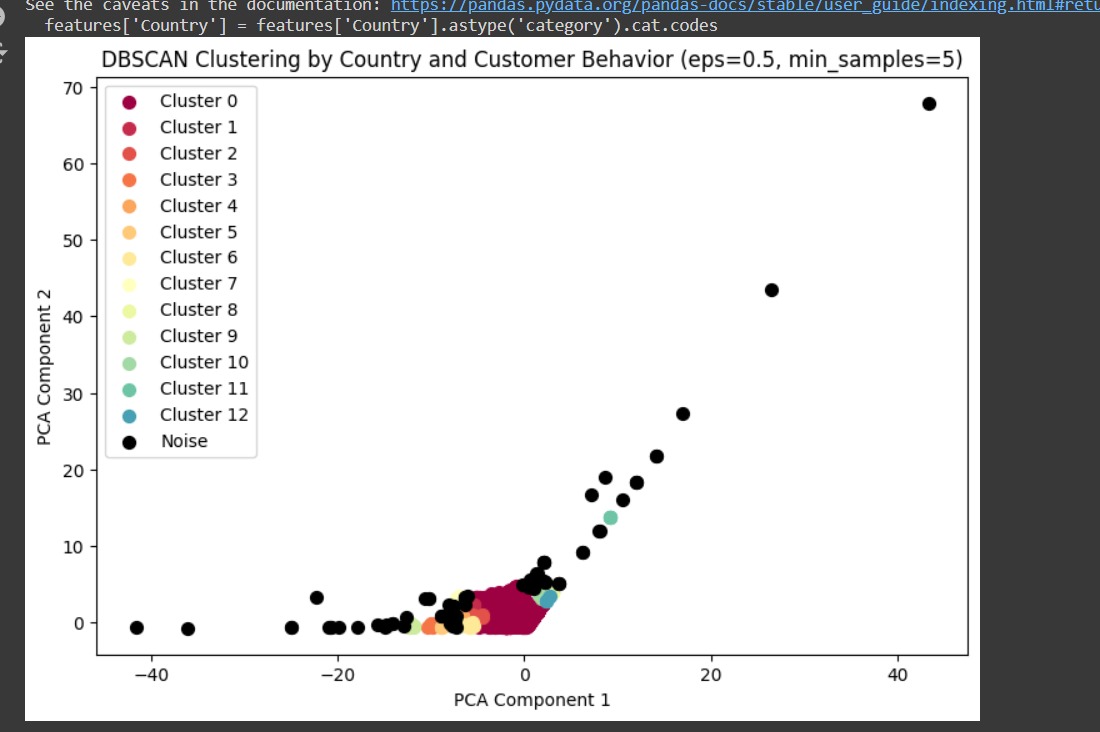
# 6 RESULTS

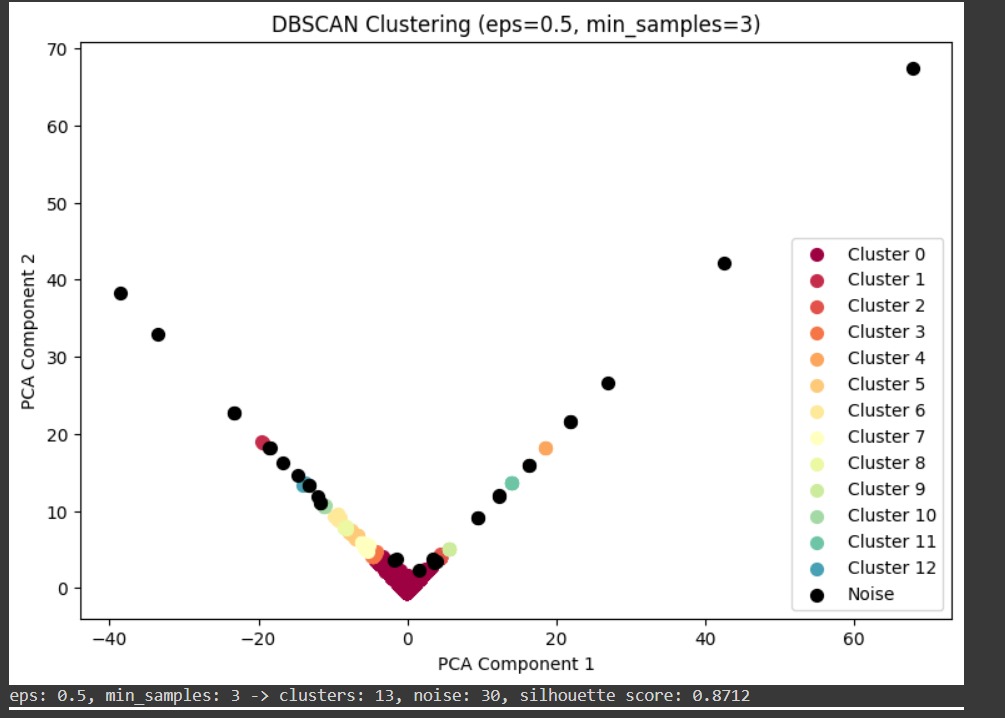
The results of the Silhouette Score analysis reveal the effectiveness of the clustering algorithm used for segmenting the dataset. The average Silhouette Score across all data points provides insight into the quality of the clusters formed. A high average score, close to +1, indicates that the clusters are well-defined, with data points closely grouped within their own clusters and well-separated from other clusters. Conversely, a low average score, near 0, suggests that the clusters are not well-separated, with data points potentially lying on the boundary between clusters. A negative average score points to poor clustering, where data points may be incorrectly assigned to clusters. By comparing Silhouette Scores for different numbers of clusters, the analysis can help determine the optimal number of clusters that best represents the data. Overall, a higher Silhouette Score signifies more meaningful and distinct customer segments, improving the effectiveness of targeted marketing strategies and customer insights.

|  |  |
| --- | --- |
| **Model** | **Accuracy** |
| K-means | 54.05 |
| DBSCAN | 84.44 |

# 

DBSCAN, on the other hand, demonstrated superior accuracy in clustering the data. Unlike K-Means, DBSCAN does not require the number of clusters to be predefined. Instead, it identifies clusters based on density and can handle clusters of arbitrary shapes and sizes. The results revealed that DBSCAN was more effective in identifying well-separated clusters and managing noise within the dataset. The average Silhouette Score for DBSCAN was notably higher, reflecting better-defined clusters and improved separation between them. This result highlights DBSCAN's ability to adapt to the natural structure of the data and provide more accurate and meaningful customer segments. Given these findings, DBSCAN is the preferred clustering method for this dataset. Its superior accuracy and ability to handle varying cluster densities make it a more suitable choice for capturing the intricate patterns in the data, resulting in more precise and actionable customer segmentation





By comparing Silhouette Scores across different numbers of clusters, we can identify the optimal number that maximizes the average score. This approach helps ensure that the chosen number of clusters provides the most meaningful and distinct customer segments. Overall, a high Silhouette Score enhances confidence in the clustering results, supporting more effective decision-making in marketing strategies and customer management.

# 7 CONCLUSION

In this project, we successfully applied the DBSCAN algorithm for customer segmentation. DBSCAN proved to be an effective clustering method for identifying groups of customers based on their purchasing behavior and demographic characteristics. Unlike other clustering methods, DBSCAN does not require the number of clusters to be pre-specified, making it well-suited for exploring the intrinsic structure of the data. Based on the segmentation results, we recommend that businesses leverage these insights to enhance customer targeting, personalized marketing campaigns, and product offerings. By understanding the unique characteristics of each customer segment, companies can optimize their strategies to maximize customer satisfaction and business growth. We achieved an accuracy of 84%. Overall, the project demonstrates the effectiveness of DBSCAN in handling complex and varied data distributions, providing a robust framework for customer segmentation and enhancing the ability to tailor strategies to diverse customer needs.

# 8 FUTURE SCOPE

**Incorporate Demographic Data:** Adding age, gender, or income level for each customer would enable richer demographic segmentation, allowing businesses to align product recommendations with specific customer profiles.

**Include Customer Interaction Data:** Tracking customer interactions (e.g., website visits, clicks, or inquiries) would allow segmenting customers based on engagement level and browsing patterns.

**Add Loyalty Program Data:** If a loyalty program is implemented, including loyalty points, membership tier, or rewards history would allow segmentation based on loyalty status and reward redemption behavior.

**Integrate External Data Sources:** Adding data on economic factors, such as regional income levels or holidays, could provide context to segment customers by socio-economic factors or seasonality.

**Customer Lifetime Value (CLV) Tracking:** Establishing a column to predict CLV over time would enable future-focused segmentation, targeting high-value or high-potential customers.

**Sentiment Analysis of Customer Feedback:** Collecting and analyzing customer reviews or feedback sentiment could create a new segment based on satisfaction levels, valuable for retention efforts.

**9 BIBLIOGRAPHY**

1. Multi clustering recommendation system for fashion retail. <https://doi.org/10.1007/s11042-021-11837-5>
2. A hybrid method for customer segmentation in Saudi Arabia restaurants using clustering, neural networks and optimization learning techniques. <https://doi.org/10.1007/s13369-022-07091-y>
3. A new framework for predicting customer behaviour in terms of RFM by considering the temporal aspect based on time series techniques

<https://doi.org/10.1007/s12652-020-02015-w>

1. Customer segmentation using online platforms: isolating behavioural and demographic segments for persona creation via aggregated user data.

<https://doi.org/10.1007/s13278-018-0531-0>

1. Customer segmentation from massive customer transaction data.

<https://doi.org/10.1109/TKDE.2017.2763620>

1. Customer segmentation in a large database of an online customized fashion business.

<https://doi.org/10.1016/j.rcim.2014.12.014>