Customer Ecommerce Data Segmentation

Nikhil Jagadeesh Sriram - 16352573

Radha Krishna Siram - 16356525

Krishna Teja Nuni - 16356387

Rumana Taj Shaik - 16357409

Introduction

- Online shops are facing fierce competition because of the e-commerce sector's explosive expansion. Understanding and addressing customer satisfaction levels is critical for firms looking to acquire a competitive advantage. E-commerce businesses may make wise decisions to promote client loyalty and improve services by determining the elements that impact consumer happiness.
- This project's objective is to analyze customer data and ascertain the degree of happiness among e-commerce consumers by applying unsupervised learning techniques, including DBSCAN, hierarchical clustering, and k-means clustering. It finds hidden patterns in the data and groups clients based on similarities by using these clustering techniques. Because of its user-friendly interface, which makes it simpler to visualize and explain the findings, k-means clustering is particularly pleasant to use.

Related Work

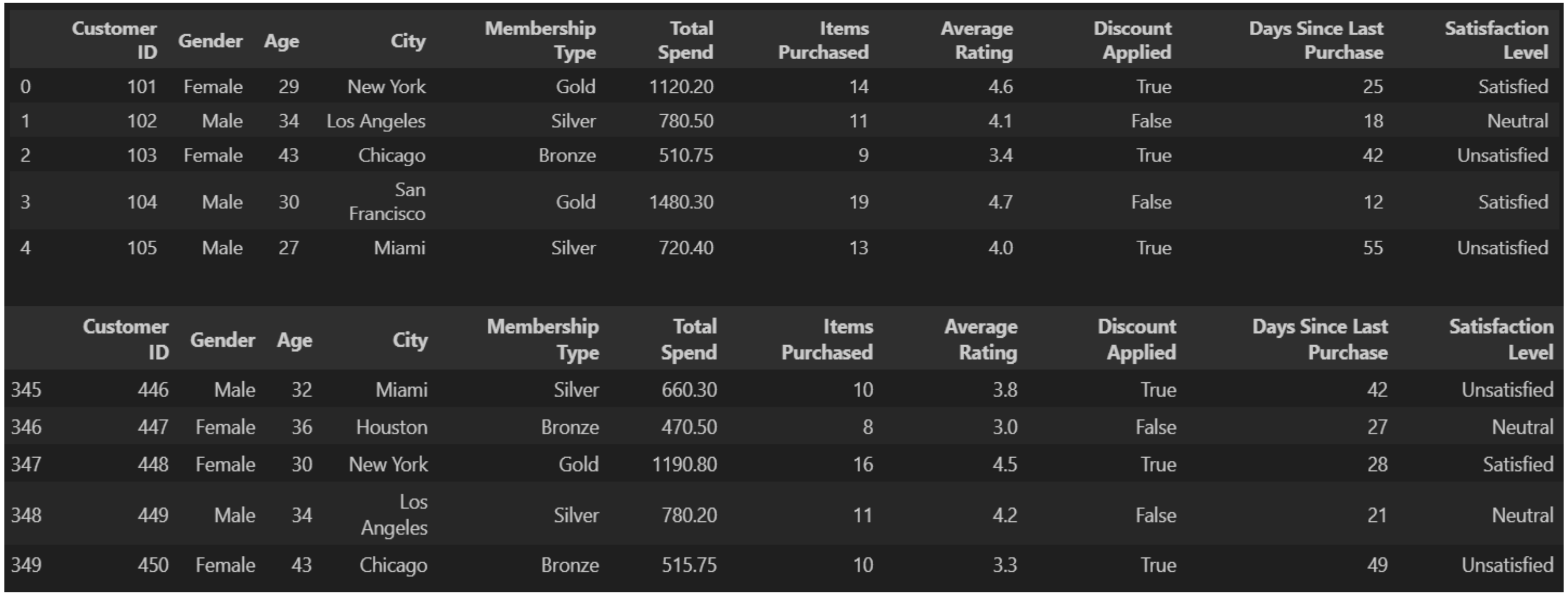
- Customer data segmentation for satisfaction in the e-commerce space has been studied in the past using a variety of techniques and approaches. Previous research has investigated the use of machine learning methods to predict customer happiness, including decision trees and neural networks. These supervised learning techniques, however, depend on labelled data, which isn't always accessible or easily used in practical situations. Consequently, unsupervised learning methods like clustering algorithms have become more and more well-liked for objectively assessing consumer happiness.
- In the realm of customer satisfaction analysis, DBSCAN, hierarchical clustering, and k-means clustering have drawn a lot of interest lately. The goal of the density-based clustering method DBSCAN is to locate dense areas in the data and categorize outliers differently. In contrast, either agglomerative or divisive methods can be used in hierarchical clustering to produce a hierarchy of clusters. Finally, by minimizing the sum of squares inside each cluster, k-means clustering divides the data into k unique clusters. Through an examination and comparison of different clustering approaches, then which method is most suited for our e-commerce dataset is decided.

Methodology Overview

- This section will detail the methodical process used to apply unsupervised learning techniques to the topic of e-commerce consumer happiness. Preprocessing the client data in the first place entailed resolving missing values and eliminating any unnecessary or redundant properties. After that, label encoding was used to make sure that every variable/ features data was on the same/ machine understandable format, allowing for equitable understanding across various characteristics.
- The pre-processed data was then subjected to the three unsupervised learning techniques: k-means clustering, hierarchical clustering, and DBSCAN. To maximize performance, each algorithm's hyperparameters were carefully chosen. The quality and efficiency of the clustering were assessed by utilizing suitable evaluation measures, such as output accuracy performance and silhouette scores, to analyze the outcomes produced by each approach.
- For every clustering algorithm, classification reports were made to help with interpretation and comprehension of the findings. A more thorough examination of consumer behavior and satisfaction levels was made possible by these report assistance in spotting clusters and outliers.

Data Set

- The model is to be trained on 350 distinct data entries with 11 columns: Customer ID, Gender, Age, City, Membership Type, Total Spend, Items Purchased, Average Rating, Discount Applied, Days Since Last Purchase, and Satisfaction Level. In order to manage missing values, encode categorical variables, and get the dataset ready for training, it is preprocessed and cleaned.



	Customer ID	Gender	Age	City	Membership Type	Total Spend	Items Purchased	Average Rating	Discount Applied	Days Since Last Purchase	Satisfaction Level
0	101	Female	29	New York	Gold	1120.20	14	4.6	True	25	Satisfied
1	102	Male	34	Los Angeles	Silver	780.50	11	4.1	False	18	Neutral
2	103	Female	43	Chicago	Bronze	510.75	9	3.4	True	42	Unsatisfied
3	104	Male	30	San Francisco	Gold	1480.30	19	4.7	False	12	Satisfied
4	105	Male	27	Miami	Silver	720.40	13	4.0	True	55	Unsatisfied

	Customer ID	Gender	Age	City	Membership Type	Total Spend	Items Purchased	Average Rating	Discount Applied	Days Since Last Purchase	Satisfaction Level
345	446	Male	32	Miami	Silver	660.30	10	3.8	True	42	Unsatisfied
346	447	Female	36	Houston	Bronze	470.50	8	3.0	False	27	Neutral
347	448	Female	30	New York	Gold	1190.80	16	4.5	True	28	Satisfied
348	449	Male	34	Los Angeles	Silver	780.20	11	4.2	False	21	Neutral
349	450	Female	43	Chicago	Bronze	515.75	10	3.3	True	49	Unsatisfied

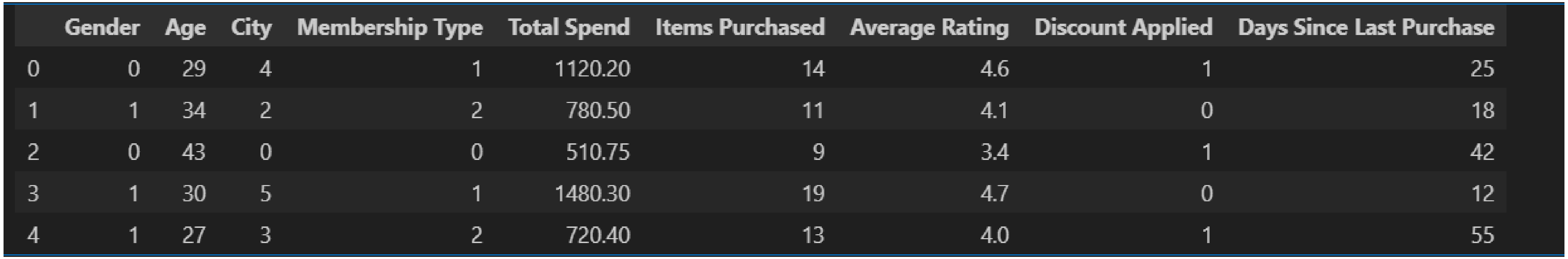
Label Encoding

- Label encoding converts categorical text data into a model-understandable numerical format. This step is essential for preparing data for machine learning algorithms which require numerical input.
- **Since in our data few columns has categorical data we have converted that to numerical data using label Encoder.**
  - **Gender**: Transforms categories (e.g., Male, Female) into numbers (0, 1).
  - **City**: Converts city names into unique numerical identifiers.
  - **Membership Type**: Assigns a unique number to each type of membership.
  - **Discount Applied**: Numerically encodes the presence or type of discount.
  - **Satisfaction Level**: Converts satisfaction categories (e.g., Satisfied, Neutral, Unsatisfied) into numerical codes.

Label Encoding

Preview of Transformed Data

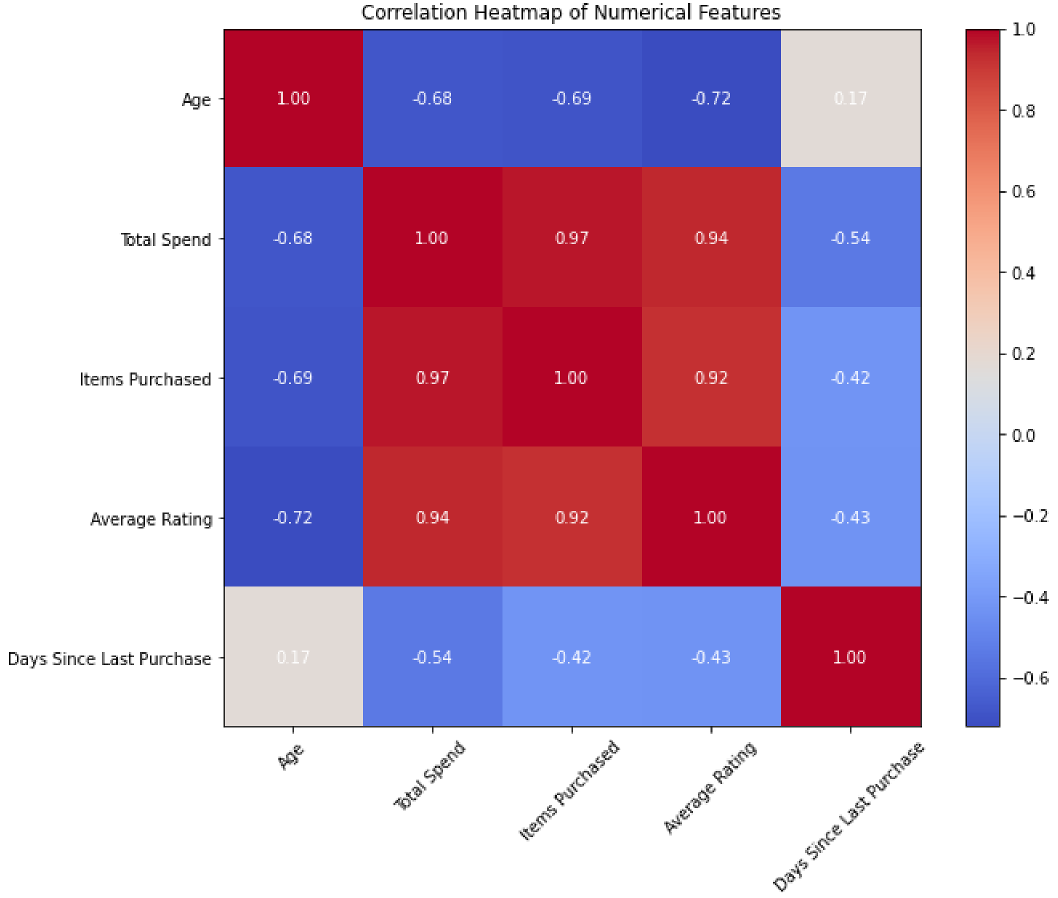
- Display of the first few rows of the `features` Data Frame to illustrate the encoded format.



	Gender	Age	City	Membership Type	Total Spend	Items Purchased	Average Rating	Discount Applied	Days Since Last Purchase
0	0	29	4	1	1120.20	14	4.6	1	25
1	1	34	2	2	780.50	11	4.1	0	18
2	0	43	0	0	510.75	9	3.4	1	42
3	1	30	5	1	1480.30	19	4.7	0	12
4	1	27	3	2	720.40	13	4.0	1	55

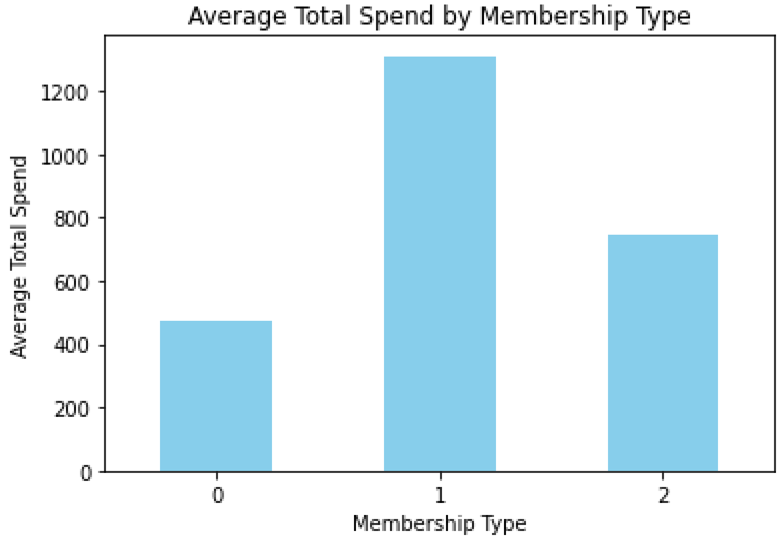
Correlations

- Red indicates positive correlation , blue indicates negative correlation , and yellow/light colors indicate little to no correlation.
- There's a positive correlation between age and total spending/items purchased, suggesting older customers might spend more per purchase.
- There's a negative correlation between days since last purchase and total spending/items purchased, indicating customers who haven't purchased recently tend to spend less.



Membership Type vs Total Spend

- Membership Type 1 is which is "Gold" tends to spend more
- Membership Type 2 which is Silver next to Type 1
- Last is bronze is which stands last



Model Selection

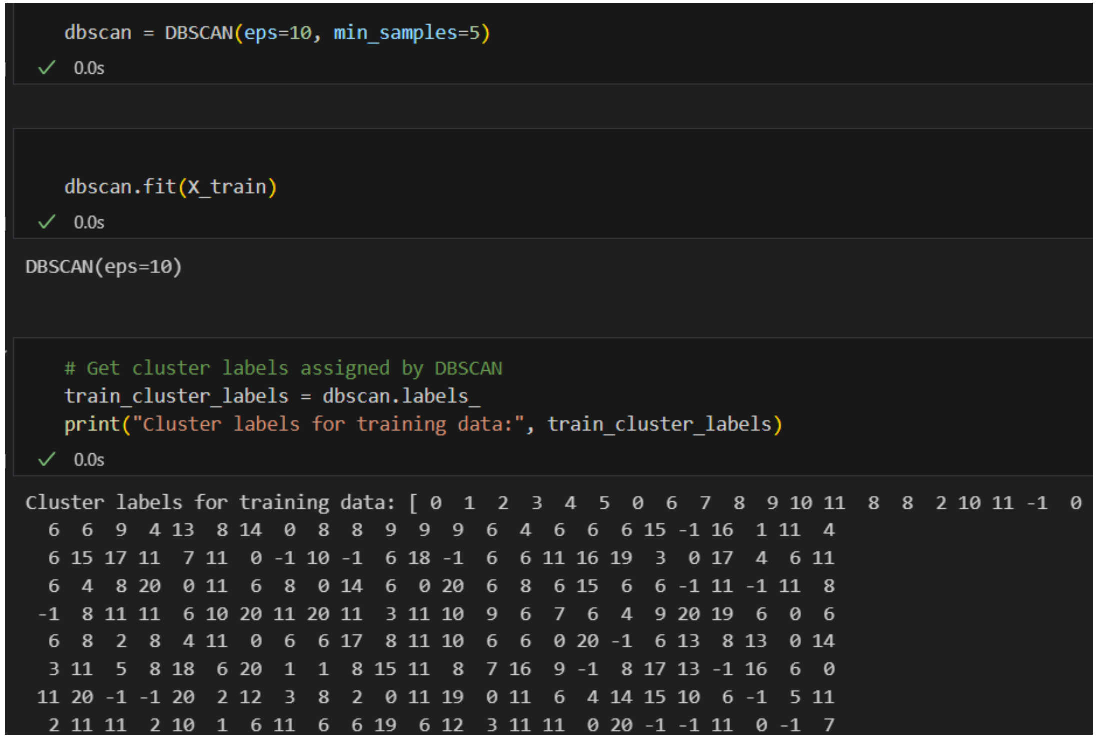
Three clustering techniques are used for this project:

- Applications with Noise: A Density-Based Spatial Clustering Approach (DBSCAN)
- Agglomerative clustering, or hierarchical clustering
- K-Means clustering

After applying each algorithm to the training set, measures including the silhouette score, accuracy, precision, recall, and F1 score are used to assess how well the algorithm performed. The algorithm that performs the best is chosen for more examination.

DBSCAN

We used DBSCAN to split our data into clusters, setting parameters for distance and minimum samples. After training, we assigned labels to our training data and predicted clusters for our test set. A silhouette score of 0.092 indicates clustering is very weak and clusters are poorly separated.



```
dbscan = DBSCAN(eps=10, min_samples=5)
```

✓ 0.0s

```
dbscan.fit(X_train)
```

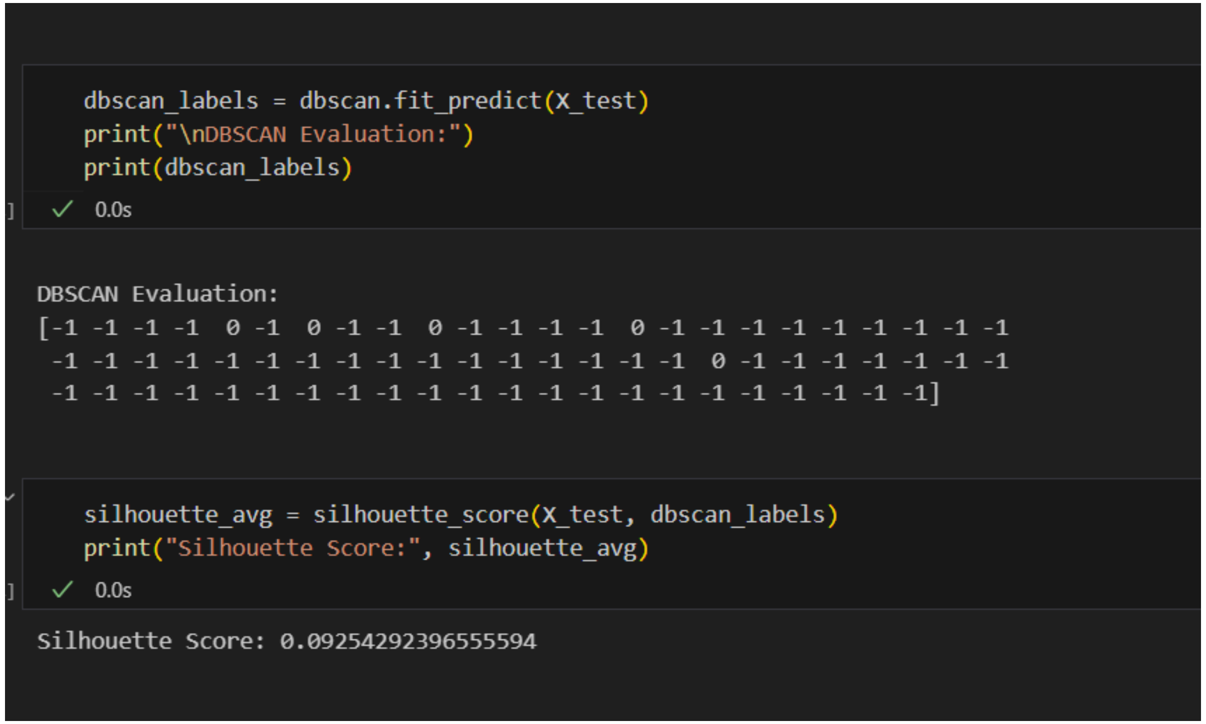
✓ 0.0s

```
DBSCAN(eps=10)
```

```
# Get cluster labels assigned by DBSCAN
train_cluster_labels = dbscan.labels_
print("Cluster labels for training data:", train_cluster_labels)
```

✓ 0.0s

```
Cluster labels for training data: [ 0  1  2  3  4  5  0  6  7  8  9 10 11  8  8  2 10 11 -1  0
  6  6  9  4 13  8 14  0  8  8  9  9  9  6  4  6  6  6 15 -1 16  1 11  4
  6 15 17 11  7 11  0 -1 10 -1  6 18 -1  6  6 11 16 19  3  0 17  4  6 11
  6  4  8 20  0 11  6  8  0 14  6  0 20  6  8  6 15  6  6 -1 11 -1 11  8
 -1  8 11 11  6 10 20 11 20 11  3 11 10  9  6  7  6  4  9 20 19  6  0  6
  6  8  2  8  4 11  0  6  6 17  8 11 10  6  6  0 20 -1  6 13  8 13  0 14
  3 11  5  8 18  6 20  1  1  8 15 11  8  7 16  9 -1  8 17 13 -1 16  6  0
 11 20 -1 -1 20  2 12  3  8  2  0 11 19  0 11  6  4 14 15 10  6 -1  5 11
  2 11 11  2 10  1  6 11  6  6 19  6 12  3 11 11  0 20 -1 -1 11  0 -1  7
```



```
dbscan_labels = dbscan.fit_predict(X_test)
print("\nDBSCAN Evaluation:")
print(dbscan_labels)
```

✓ 0.0s

```
DBSCAN Evaluation:
[-1 -1 -1 -1  0 -1  0 -1 -1  0 -1 -1 -1 -1  0 -1 -1 -1 -1 -1 -1 -1 -1 -1
 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1  0 -1 -1 -1 -1 -1 -1 -1
 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1]
```

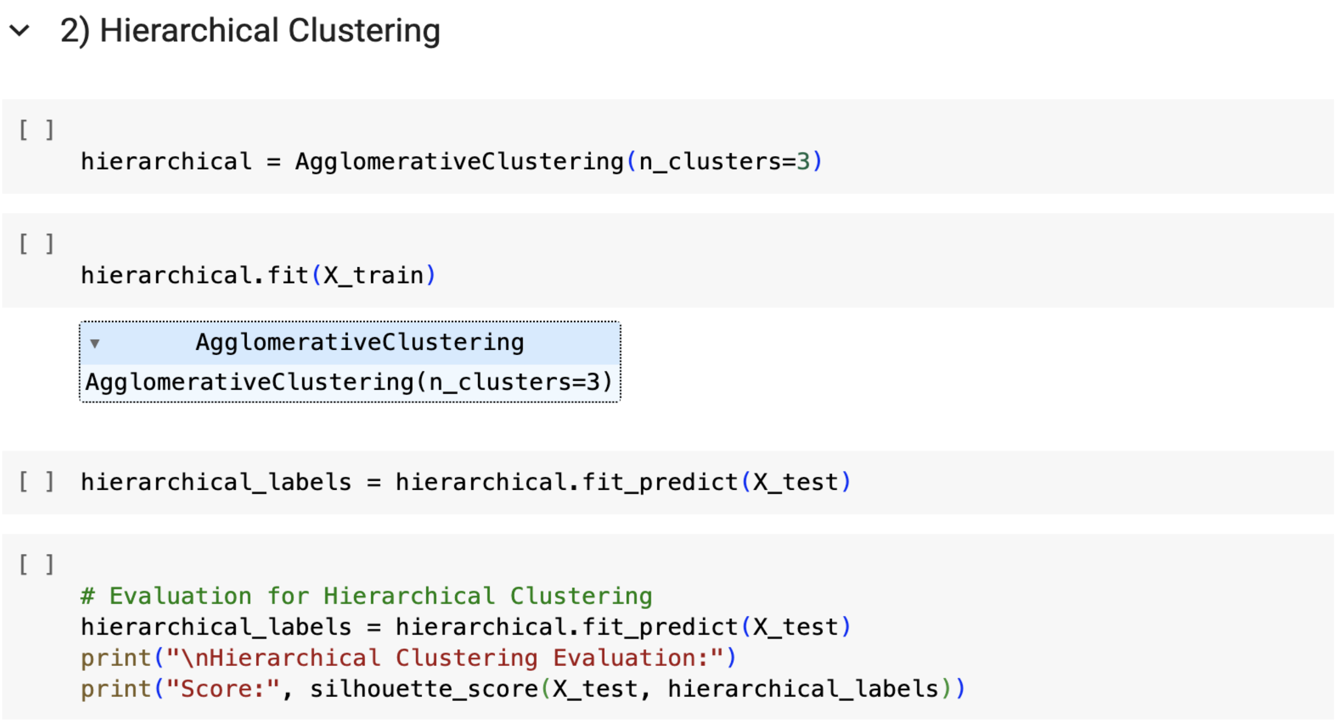
```
silhouette_avg = silhouette_score(X_test, dbscan_labels)
print("Silhouette Score:", silhouette_avg)
```

✓ 0.0s

```
Silhouette Score: 0.092542923965555594
```

Hierarchical Clustering

The output displays the results of evaluating a hierarchical clustering model with three clusters on a test dataset. The silhouette score of 0.7389 indicates the clusters are well separated.



2) Hierarchical Clustering

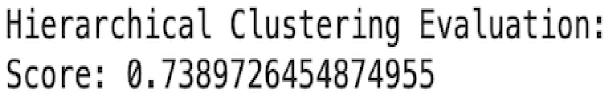
```
[ ]
    hierarchical = AgglomerativeClustering(n_clusters=3)
```

```
[ ]
    hierarchical.fit(X_train)

    ▾       AgglomerativeClustering
    AgglomerativeClustering(n_clusters=3)
```

```
[ ] hierarchical_labels = hierarchical.fit_predict(X_test)
```

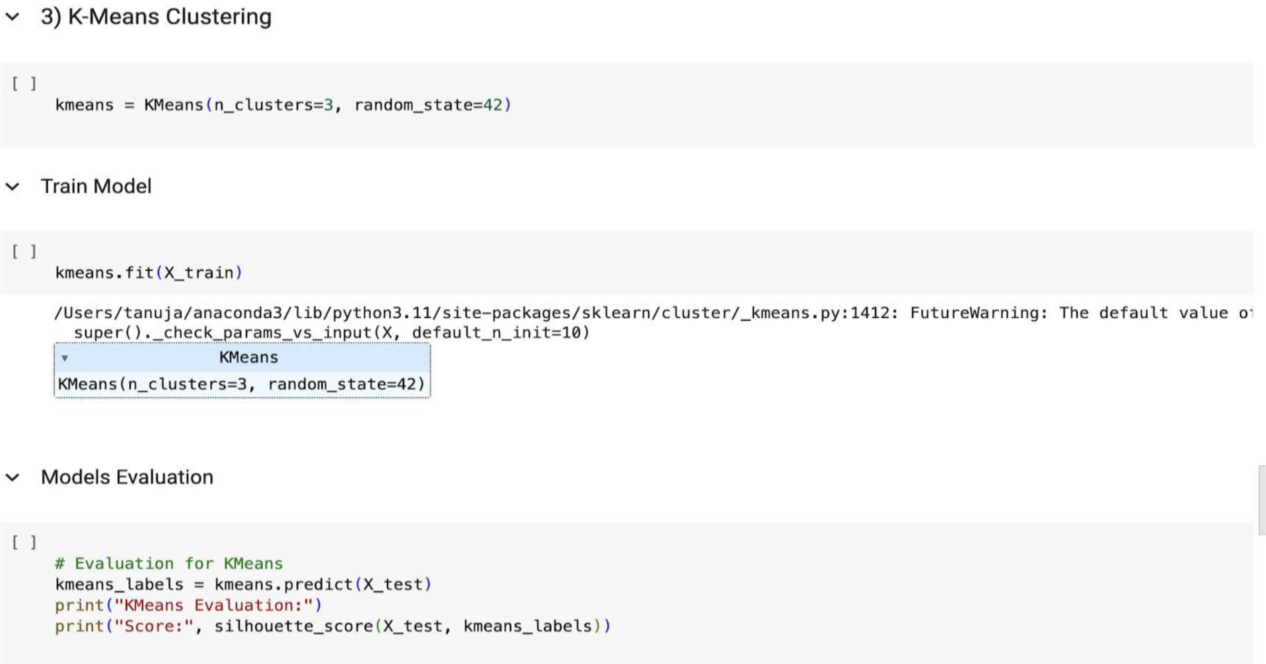
```
[ ]
    # Evaluation for Hierarchical Clustering
    hierarchical_labels = hierarchical.fit_predict(X_test)
    print("\nHierarchical Clustering Evaluation:")
    print("Score:", silhouette_score(X_test, hierarchical_labels))
```



```
Hierarchical Clustering Evaluation:
Score: 0.7389726454874955
```

K-means Clustering

The main objective of the K-Means algorithm is to minimize the sum of distances between the points and their respective cluster centroid. The K-means clustering evaluation shows that the algorithm was configured with three clusters and produced a silhouette score of 0.7389, indicating decent separation between the clusters.



3) K-Means Clustering

```
[ ]
    kmeans = KMeans(n_clusters=3, random_state=42)
```

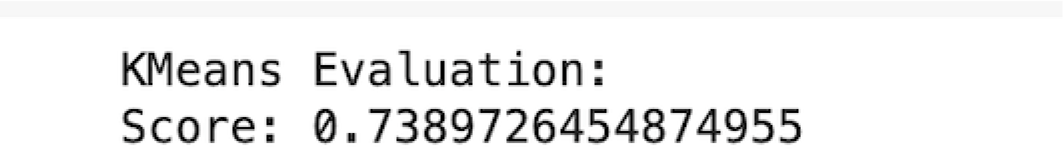
Train Model

```
[ ]
    kmeans.fit(X_train)

    /Users/tanuja/anaconda3/lib/python3.11/site-packages/sklearn/cluster/_kmeans.py:1412: FutureWarning: The default value o
      super()._check_params_vs_input(X, default_n_init=10)
                    KMeans
    KMeans(n_clusters=3, random_state=42)
```

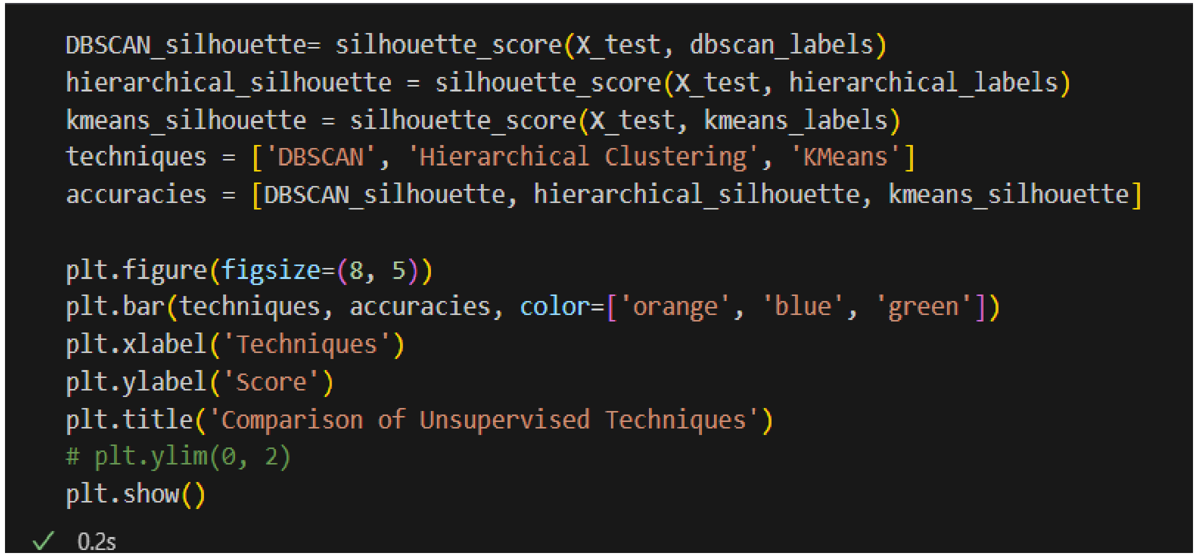
Models Evaluation

```
[ ]
    # Evaluation for KMeans
    kmeans_labels = kmeans.predict(X_test)
    print("KMeans Evaluation:")
    print("Score:", silhouette_score(X_test, kmeans_labels))
```



```
KMeans Evaluation:
Score: 0.7389726454874955
```

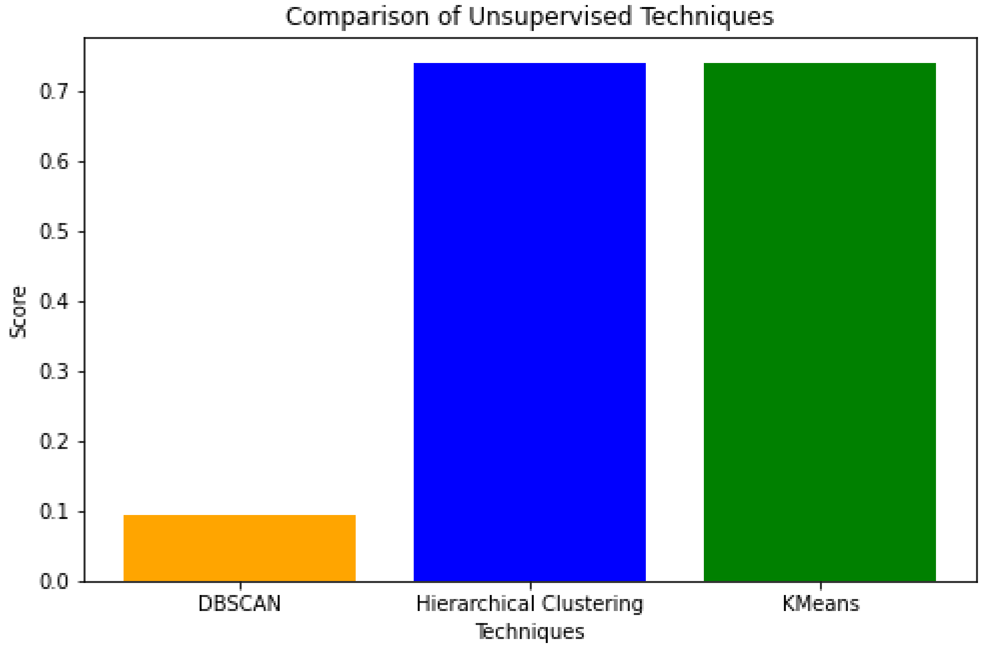
Comparision of Clustering Methods



```
DBSCAN_silhouette= silhouette_score(X_test, dbscan_labels)
hierarchical_silhouette = silhouette_score(X_test, hierarchical_labels)
kmeans_silhouette = silhouette_score(X_test, kmeans_labels)
techniques = ['DBSCAN', 'Hierarchical Clustering', 'KMeans']
accuracies = [DBSCAN_silhouette, hierarchical_silhouette, kmeans_silhouette]

plt.figure(figsize=(8, 5))
plt.bar(techniques, accuracies, color=['orange', 'blue', 'green'])
plt.xlabel('Techniques')
plt.ylabel('Score')
plt.title('Comparison of Unsupervised Techniques')
# plt.ylim(0, 2)
plt.show()
```

✓ 0.2s



Conclusion

- In summary, this study effectively used k-means clustering, hierarchical clustering, and DBSCAN unsupervised learning techniques to analyze and assess e-commerce consumers' satisfaction levels. Customer data was clustered to reveal hidden patterns and structures that offered important insights into the behavior and preferences of the target audience.
- Because of its simplicity and effectiveness, k-means clustering, and hierarchal clustering is a good method for analyzing consumer happiness in e-commerce, according to the findings of the comparison of the three clustering algorithms. Still, more investigation is needed to determine the possibilities of other unsupervised learning strategies and how well they work with various e-commerce datasets. Furthermore, investigating ensemble approaches and implementing more sophisticated data pretreatment procedures may improve the clustering analysis's resilience and accuracy.