CustomerDataFinalProject.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

Google Account
KYLE MOLNAR
kmm704@scarletmail.rutgers.edu

+ Code + Text

```
# Import necessary libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.cluster import KMeans, DBSCAN, AgglomerativeClustering
from sklearn.decomposition import PCA
from sklearn.preprocessing import StandardScaler
```

```
[23] # Load the dataset
file_path = '/content/customer_segmentation_data.csv'
df = pd.read_csv(file_path)

# Display the first few rows of the DataFrame
print(df.head())
```

```
   id  age  gender  income  spending_score  membership_years  \
0   1   38  Female   99342              90                 3
1   2   21  Female   78852              60                 2
2   3   60  Female  126573              30                 2
3   4   40   Other   47099              74                 9
4   5   65  Female  140621              21                 3

   purchase_frequency preferred_category  last_purchase_amount
0                  24          Groceries                113.53
1                  42             Sports                 41.93
2                  28           Clothing                424.36
3                   5      Home & Garden                991.93
4                  25        Electronics                347.08
```

```
[24] # Check for missing values
print(df.isnull().sum())

# Handle missing values if any
df = df.dropna()  # Or use other methods to handle missing data

# Convert categorical variables to numeric if needed
df['gender'] = df['gender'].astype('category').cat.codes
df['preferred_category'] = df['preferred_category'].astype('category').cat.codes

print(df.head())
```

CustomerDataFinalProject.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

Google Account
KYLE MOLNAR
kmm704@scarletmail.rutgers.edu

+ Code + Text

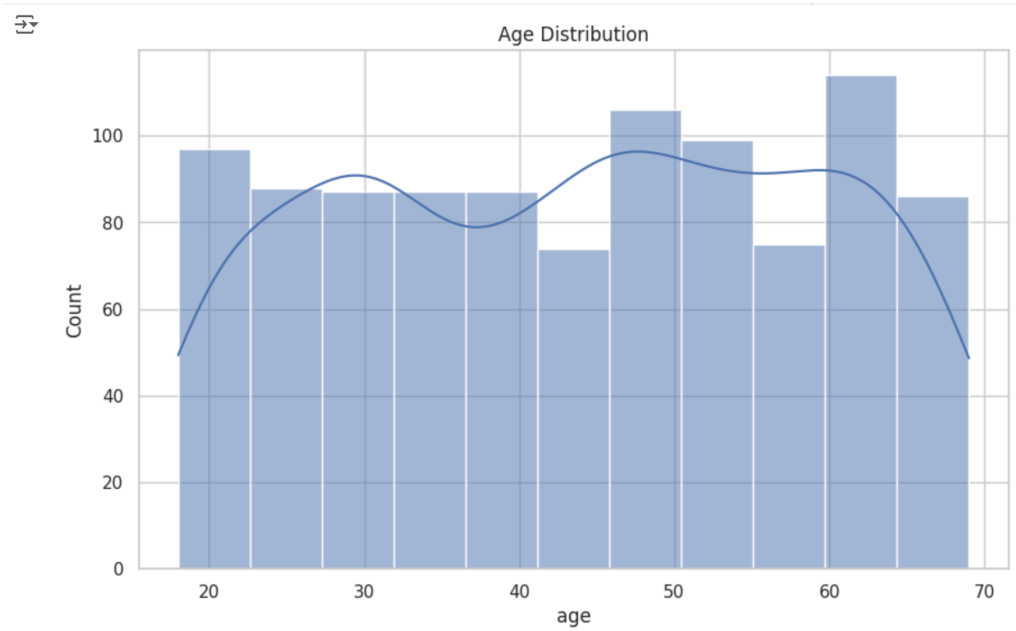
[24]
```
id                    0
age                   0
gender                0
income                0
spending_score        0
membership_years      0
purchase_frequency    0
preferred_category    0
last_purchase_amount  0
dtype: int64
```

	id	age	gender	income	spending_score	membership_years	purchase_frequency	preferred_category	last_purchase_amount
0	1	38	0	99342	90	3	24	2	113.53
1	2	21	0	78852	60	2	42	4	41.93
2	3	60	0	126573	30	2	28	0	424.36
3	4	40	2	47099	74	9	5	3	991.93
4	5	65	0	140621	21	3	25	1	347.08

[25]
```
# Age distribution
plt.figure(figsize=(10, 6))
sns.histplot(df['age'], kde=True)
plt.title('Age Distribution')
plt.show()

# Income by gender
plt.figure(figsize=(10, 6))
sns.boxplot(x='gender', y='income', data=df)
plt.title('Income by Gender')
plt.show()

# Spending score by age and preferred category
plt.figure(figsize=(10, 6))
sns.scatterplot(x='age', y='spending_score', hue='preferred_category', data=df)
plt.title('Spending Score by Age and Preferred Category')
plt.show()
```


Age Distribution
Count
age
0
20
40
60
80
100
20
30
40
50
60
70

CustomerDataFinalProject.ipynb

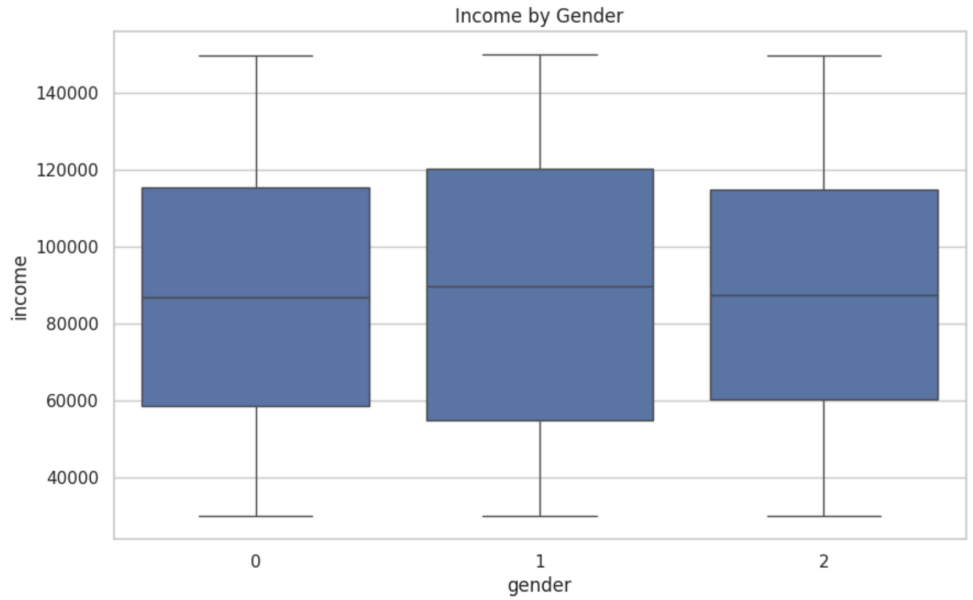
File Edit View Insert Runtime Tools Help All changes saved

Comment Share K

+ Code + Text

Google Account
KYLE MOLNAR
kmm704@scarletmail.rutgers.edu

[25]



CustomerDataFinalProject.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

Google Account
KYLE MOLNAR
kmm704@scarletmail.rutgers.edu

+ Code + Text

[25]


```
# Standardize the data
scaler = StandardScaler()
df_scaled = scaler.fit_transform(df[['age', 'income', 'spending_score', 'membership_years', 'purchase_frequency', 'last_purchase_amount']])

# Apply K-Means clustering with explicit n_init
kmeans = KMeans(n_clusters=4, n_init=10, random_state=42)
df['cluster_kmeans'] = kmeans.fit_predict(df_scaled)

# Visualize the clusters
plt.figure(figsize=(10, 6))
sns.scatterplot(x='age', y='spending_score', hue='cluster_kmeans', data=df, palette='viridis')
plt.title('Customer Segments (K-Means)')
plt.show()
```

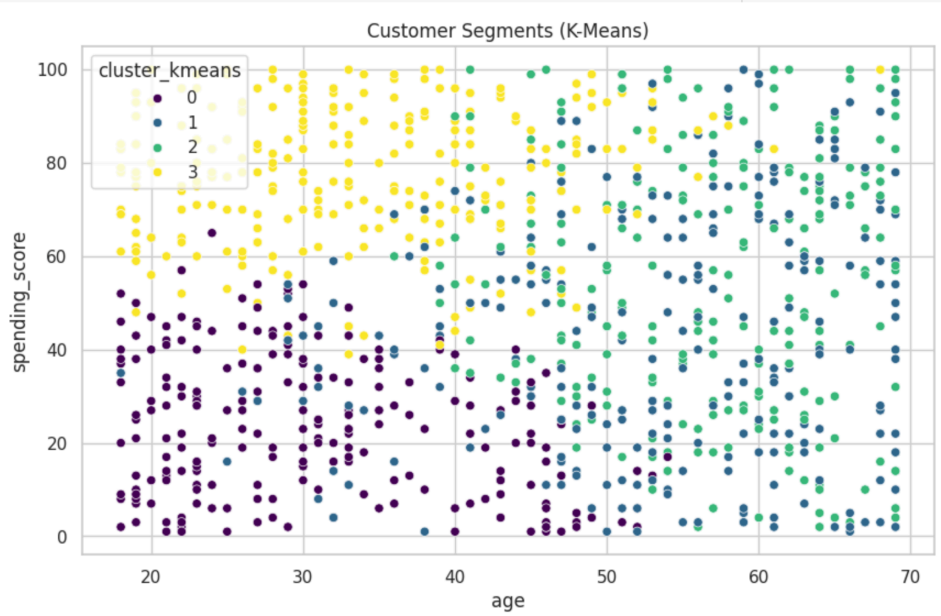
CustomerDataFinalProject.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share K

+ Code + Text

Google Account
KYLE MOLNAR
kmm704@scarletmail.rutgers.edu



[27]
```
# Apply DBSCAN clustering
dbscan = DBSCAN(eps=0.5, min_samples=5)
df['cluster_dbscan'] = dbscan.fit_predict(df_scaled)

# Visualize the clusters
plt.figure(figsize=(10, 6))
sns.scatterplot(x='age', y='spending_score', hue='cluster_dbscan', data=df, palette='viridis')
plt.title('Customer Segments (DBSCAN)')
plt.show()
```

CustomerDataFinalProject.ipynb

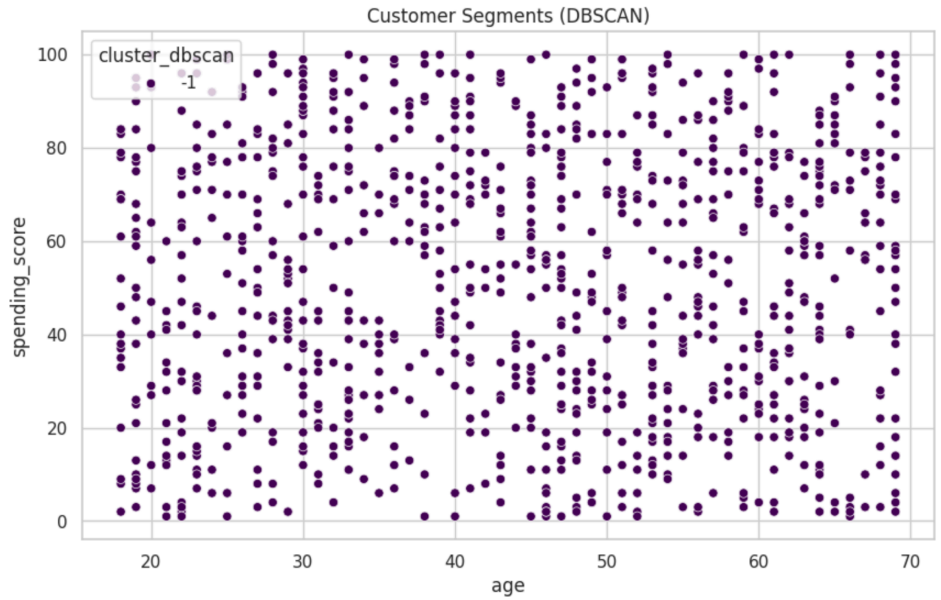
File Edit View Insert Runtime Tools Help All changes saved

Comment Share

Google Account
KYLE MOLNAR
kmm704@scarletmail.rutgers.edu

+ Code + Text

[27]



[28]
```
# Apply Agglomerative Clustering
agg = AgglomerativeClustering(n_clusters=4)
df['cluster_agg'] = agg.fit_predict(df_scaled)

# Visualize the clusters
plt.figure(figsize=(10, 6))
sns.scatterplot(x='age', y='spending_score', hue='cluster_agg', data=df, palette='viridis')
plt.title('Customer Segments (Agglomerative Clustering)')
plt.show()
```

CustomerDataFinalProject.ipynb

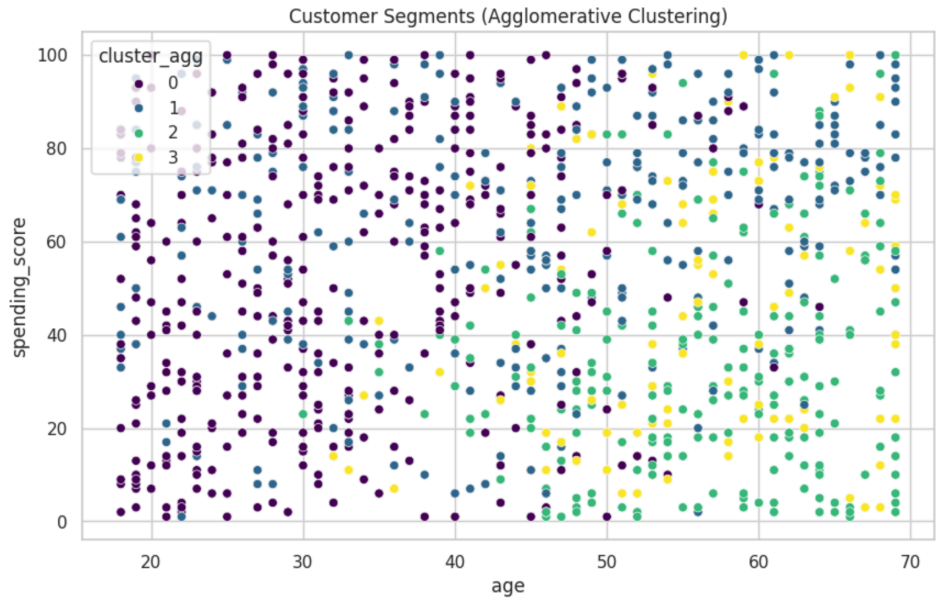
File Edit View Insert Runtime Tools Help All changes saved

Comment Share

Google Account
KYLE MOLNAR
kmm704@scarletmail.rutgers.edu

+ Code + Text

[28]



[29]
```
# Apply PCA
pca = PCA(n_components=2)
df_pca = pca.fit_transform(df_scaled)
df['pca_one'] = df_pca[:, 0]
df['pca_two'] = df_pca[:, 1]

# Visualize PCA components
plt.figure(figsize=(10, 6))
sns.scatterplot(x='pca_one', y='pca_two', hue='cluster_kmeans', data=df, palette='viridis')
plt.title('PCA of Customer Segments (K-Means)')
plt.show()
```

CustomerDataFinalProject.ipynb

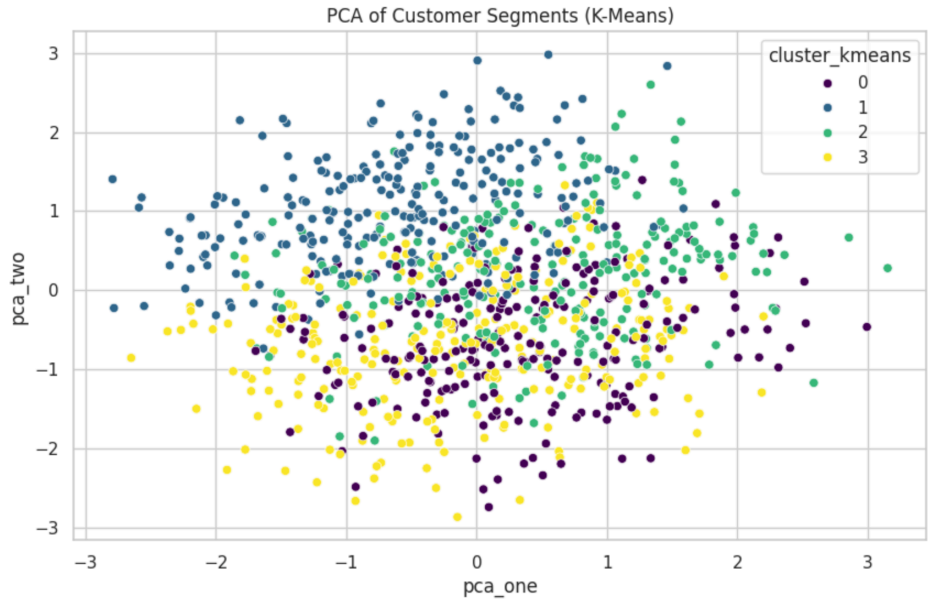
File Edit View Insert Runtime Tools Help All changes saved

Comment Share K

Google Account
KYLE MOLNAR
kmm704@scarletmail.rutgers.edu

+ Code + Text

[29]



[30]
```
# Analyze clusters
cluster_summary = df.groupby('cluster_kmeans').mean()
print(cluster_summary)

# Recommendations based on clusters
for cluster in cluster_summary.index:
    print(f"Cluster {cluster}:")
    print(f"  Average Age: {cluster_summary.loc[cluster, 'age']}")
    print(f"  Average Income: {cluster_summary.loc[cluster, 'income']}")
    print(f"  Average Spending Score: {cluster_summary.loc[cluster, 'spending_score']}")
    print(f"  Recommended Strategy: {'Provide discounts and offers' if cluster_summary.loc[cluster, 'spending_score'] < 50 else 'Offer premium products and services'}")
```

[30]

```
                          id        age    gender         income  \
cluster_kmeans
0               501.321739  30.669565  1.013043   98862.330435
1               518.334630  53.264591  1.116732   55591.821012
2               504.741935  56.741935  0.979839  121045.322581
3               478.520755  33.841509  0.935849   80966.513208

                spending_score  membership_years  purchase_frequency  \
cluster_kmeans
0                    23.839130          5.404348           28.386957
1                    43.319066          5.661479           25.762646
2                    54.358871          5.112903           25.455645
3                    77.690566          5.671698           26.916981

                preferred_category  last_purchase_amount  cluster_dbscan  \
cluster_kmeans
0                         2.130435            442.921174            -1.0
1                         2.038911            665.218405            -1.0
2                         2.028226            473.450685            -1.0
3                         2.090566            385.282642            -1.0

                cluster_agg   pca_one   pca_two
cluster_kmeans
0                  0.556522  0.348996 -0.627431
1                  1.762646 -0.613447  1.033744
2                  1.483871  0.483641  0.231860
3                  0.384906 -0.160590 -0.674960
Cluster 0:
  Average Age: 30.669565217391305
  Average Income: 98862.3304347826
  Average Spending Score: 23.839130434782607
  Recommended Strategy: Provide discounts and offers
Cluster 1:
  Average Age: 53.264591439688715
  Average Income: 55591.82101167315
  Average Spending Score: 43.31906614785992
  Recommended Strategy: Provide discounts and offers
Cluster 2:
  Average Age: 56.74193548387097
  Average Income: 121045.32258064517
  Average Spending Score: 54.358870967741936
  Recommended Strategy: Offer premium products and services
Cluster 3:
  Average Age: 33.841509433962266
  Average Income: 80966.51320754716
  Average Spending Score: 77.69056603773585
  Recommended Strategy: Offer premium products and services
```