Unsupervised Learning Project: Customer Segmentation

1. Introduction

This project focuses on customer segmentation using unsupervised learning techniques. The objective is to group customers into clusters based on their annual income and spending scores. We use two algorithms: KMeans and DBSCAN to analyze and compare the clustering results.

2. Dataset

We simulate a dataset that mirrors a common customer dataset used in segmentation tasks. It includes features such as 'Annual Income (k$)' and 'Spending Score (1-100)' for 200 customers.

3. Data Preprocessing

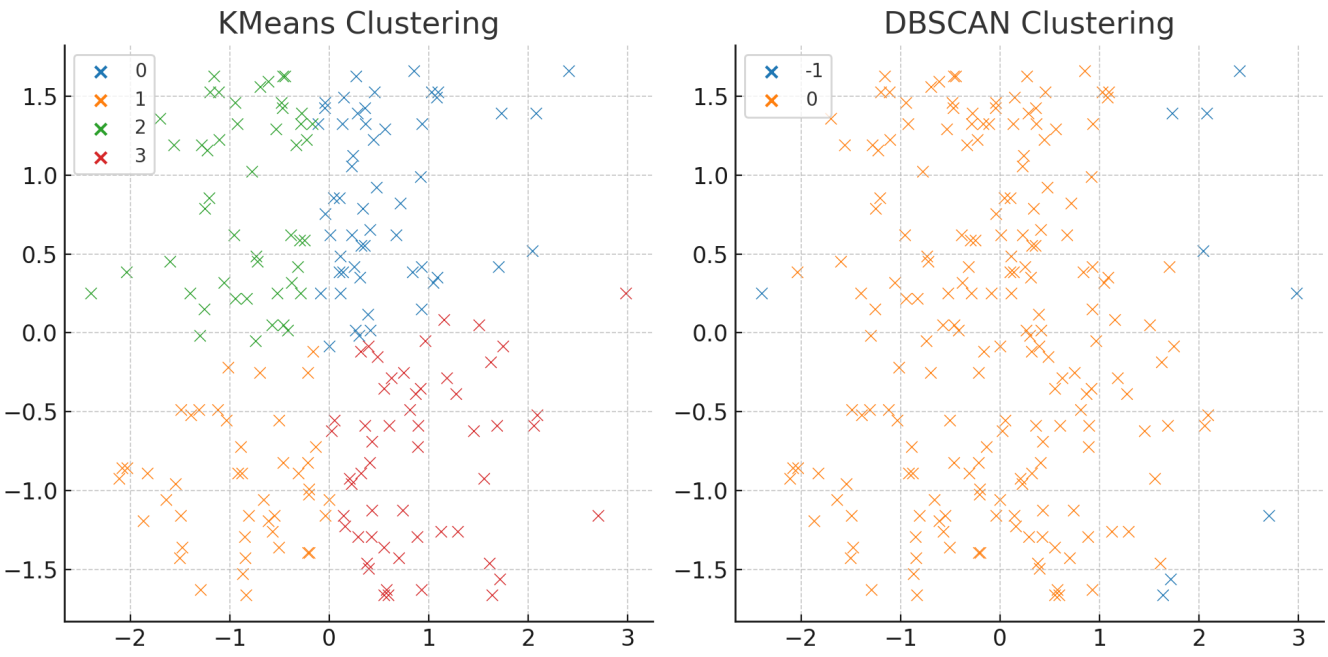
The selected features are standardized using StandardScaler to normalize the data for clustering.

4. Algorithms Used

- KMeans: A centroid-based clustering method that minimizes the within-cluster sum of squares.
- DBSCAN: A density-based clustering algorithm that finds core samples and expands clusters based on density.

5. Results

Unsupervised Learning Project: Customer Segmentation



KMeans formed 4 distinct clusters based on the income and spending data. DBSCAN identified dense areas as clusters and classified noise/outliers effectively. The two methods highlight different data patterns.

6. Discussion and Conclusion

KMeans works well when the number of clusters is known and data is evenly distributed. DBSCAN handles noise and irregular cluster shapes better. Future work could include using real datasets and evaluating more algorithms like Agglomerative Clustering or Gaussian Mixture Models.

7. Appendix - Source Code

```
{
"cells": [
{
"cell_type": "markdown",
"id": "b62200ad",
"metadata": {},
"source": [
"# Unsupervised Learning Project: Customer Segmentation"
]
},
{
"cell_type": "code",
"execution_count": null,
"id": "eb4bd65c",
"metadata": {},
"outputs": [],
"source": [
"import pandas as pd\n",
"import numpy as np\n",
```

Unsupervised Learning Project: Customer Segmentation

```
"from sklearn.preprocessing import StandardScaler\n",
"from sklearn.cluster import KMeans, DBSCAN\n",
"import matplotlib.pyplot as plt\n",
"import seaborn as sns"
]
},
{
"cell_type": "markdown",
"id": "c6f9afb8",
"metadata": {},
"source": [
"## Simulated Dataset Loading"
]
},
{
"cell_type": "code",
"execution_count": null,
"id": "413d1be4",
"metadata": {},
"outputs": [],
"source": [
"np.random.seed(42)\n",
"data = pd.DataFrame({\n",
"    \"Annual Income (k$)\": np.random.normal(60, 20, 200).clip(15, 150),\n",
"    \"Spending Score (1-100)\": np.random.randint(1, 101, 200)\n",
"})\n",
"data.head()"
]
},
{
"cell_type": "markdown",
"id": "e42850a3",
"metadata": {},
"source": [
"## Preprocessing"
]
},
{
"cell_type": "code",
"execution_count": null,
"id": "3d1dddc5",
"metadata": {},
"outputs": [],
"source": [
"scaler = StandardScaler()\n",
"scaled = scaler.fit_transform(data)"
]
},
{
"cell_type": "markdown",
"id": "02b257d2",
"metadata": {},
"source": [
"## KMeans Clustering"
]
},
{
"cell_type": "code",
"execution_count": null,
"id": "20e6f789",
"metadata": {},
"outputs": [],
```

Unsupervised Learning Project: Customer Segmentation


```
"source": [
"kmeans = KMeans(n_clusters=4, random_state=42)\n",
"kmeans_labels = kmeans.fit_predict(scaled)"
]
},
{
"cell_type": "markdown",
"id": "f866aca9",
"metadata": {},
"source": [
"## DBSCAN Clustering"
]
},
{
"cell_type": "code",
"execution_count": null,
"id": "ab1dfd35",
"metadata": {},
"outputs": [],
"source": [
"dbscan = DBSCAN(eps=0.5, min_samples=5)\n",
"dbscan_labels = dbscan.fit_predict(scaled)"
]
},
{
"cell_type": "markdown",
"id": "9909f4db",
"metadata": {},
"source": [
"## Visualization"
]
},
{
"cell_type": "code",
"execution_count": null,
"id": "90557058",
"metadata": {},
"outputs": [],
"source": [
"plt.figure(figsize=(10, 5))\n",
"plt.subplot(1, 2, 1)\n",
"sns.scatterplot(x=scaled[:, 0], y=scaled[:, 1], hue=kmeans_labels, palette=\"tab10\")\n",
"plt.title(\"KMeans Clustering\")\n",
"\n",
"plt.subplot(1, 2, 2)\n",
"sns.scatterplot(x=scaled[:, 0], y=scaled[:, 1], hue=dbscan_labels, palette=\"tab10\")\n",
"plt.title(\"DBSCAN Clustering\")\n",
"plt.tight_layout()\n",
"plt.show()"
]
}
],
"metadata": {},
"nbformat": 4,
"nbformat_minor": 5
}
```