MALL CUSTOMER SEGMENTATION

Problem statement

A mall owner wants to categorize customers based on their spending patterns, age, income, and behaviour. So, he asked a machine learning engineer to build a customer segmentation solution for the mall.

PREDICTION GOAL

Industry: Target Marketing

Step 1: domain selection - Machine learning

Step 2: learning type – Unsupervised Learning

Step 3: clustering

Tell basic info about the dataset

No of rows: 200

No of columns: 5

Important Features: **Annual Income (k$), Spending Score (1-100)**

All the research values of each algorithm

K means clustering

The Mall Customer dataset was analyzed using the K-Means clustering algorithm.
The optimal number of clusters was determined using the Elbow Method, where the minimum WCSS (inertia) was observed at **K = 5**.
A K-Means model was built with five clusters, and customers were successfully segmented into these groups.
The clustered data was saved as separate CSV files for further analysis.
Based on **income and spending behavior**, clusters **0, 1, and 2** were identified as the **target customer segments**.
Model performance was evaluated using **clustering evaluation metrics** and **visual inspection**, confirming that the model produces meaningful and well-separated clusters.

Agglomerative clustering

The dataset was also analyzed using **Agglomerative Hierarchical Clustering**.
A **dendrogram** was used to determine the optimal number of clusters. Since each data point starts as an individual cluster, the **Ward linkage method** was applied to measure inter-cluster distances.
From the dendrogram, the optimal number of clusters was identified as **5**, and the model was built accordingly.

The resulting clusters were similar to those obtained from **K-Means**, with clusters **0, 1, and 2** identified as the **target customer segments**.
Model performance was evaluated using the **Silhouette Score**, which showed results comparable to K-Means, indicating good clustering quality.

Affinity propagation clustering

The dataset was further analyzed using **Affinity Propagation clustering**.
Unlike K-Means or Hierarchical clustering, Affinity Propagation does not require specifying the number of clusters in advance; instead, it determines clusters based on **exemplars**, controlled by the **preference** parameter.
Preference values were derived from the **similarity matrix** computed using pairwise distances, and the **minimum, median, and maximum** values were tested.
By tuning the preference parameter, a configuration producing **5 clusters** was selected.
The model successfully identified five clusters, with performance evaluated using clustering metrics, achieving a score of **0.5**, which is comparable to the K-Means model and indicates good clustering quality.

AffinityPropagation				
	damping	cluster & no of clusters	silhouette_score metric	clustering remark
	0.5		0.432097161	Poor clustering
	0.6 to 0.9		0.442195717	Poor clustering
	damping=0.5,preference=-30.26622723099282, affinity='euclidean',random_state=0		0.555417624	GOOD clusters

BIRCH clustering

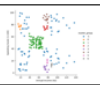
The dataset was also clustered using the **BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies)** algorithm.
Key hyperparameters such as the **branching factor** and **threshold** were tuned, with the number of clusters set to **5**.
The branching factor was fixed at **50**, and threshold values **0.5** and **0.4** were evaluated.
At **0.5**, one cluster was not well separated, while at **0.4**, all five clusters were clearly separated.
Visual inspection confirmed customer segmentation similar to other clustering algorithms.
The model showed good performance, with evaluation scores around **0.5 (nearly to 1)**, indicating good clustering.

DBSCAN clustering

The dataset was also analyzed using **DBSCAN (Density-Based Spatial Clustering of Applications with Noise)**.
The **epsilon (ε)** parameter was determined using the **k-distance graph**, with a stabilized value of **0.5**.
The model produced **three clusters**, of which two were well-separated and the remaining points

were considered **outliers/noise**.
Unlike previous algorithms, DBSCAN did not effectively identify all five customer segments, even after **parameter tuning and standardization**.
The **Davies-Bouldin score** was approximately **0.8**, indicating lower clustering quality.
Based on this, alternative clustering algorithms were considered for better segmentation.

DBSCAN

Tuning the 'eps' value		
epsilon	cluster	clustering remark
2		poor
5		not clustered well
6		not clustered well
10		not clustered well
16		poor

HDBSCAN clustering

The dataset was also analyzed using **HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise)**.
Unlike DBSCAN, HDBSCAN does not require an epsilon parameter. The **min_cluster_size** parameter was tuned from 5 to 10, with **8** selected as optimal. **min_samples** was set to None.
The model produced **5 clusters** along with several **outliers**, showing slightly better performance than DBSCAN.
Clustering quality was evaluated using a **hdbscan validity index**, yielding a score of **0.2**.
Based on this analysis, this algorithm performing moderately

HDBSCAN

min_cluster_size	min_samples	cluster	no of clusters	dbcv metric	clustering remark
5	None		0-5	0.186910859	poor
6	None		0-4	0.127005622	comparatively good
8	None		0-4	0.241205421	comparatively good
10	None		0-3	0.176324748	poor

Mean shift clustering

The dataset was analyzed using **Mean Shift clustering**, where the key parameter **bandwidth** was estimated using `estimate_bandwidth` from scikit-learn with a **quantile of 0.1**, resulting in a bandwidth of **0.64**.
The model was trained with **bin seeding enabled** for faster computation and produced **7 well-**

separated clusters.
Clustering analysis indicates that Mean Shift performs well on the customer dataset.
The final number of clusters can be adjusted based on **domain-specific requirements** for customer segmentation.

OPTICS clustering

The dataset was analyzed using **OPTICS (Ordering Points To Identify the Clustering Structure)**, which does not require specifying the number of clusters.
Key parameters **min_samples**, **xi**, and **min_cluster_size** were tuned manually: `min_samples=10`, `xi=0.01`, and `min_cluster_size=10`.
The model produced **6 clusters** along with many **outliers**, but the clustering was not well-separated.
Evaluation using the **Davies-Bouldin score** yielded **1.6**, indicating poor clustering performance for the Mall Customer dataset.

OPTICS

min_cluster_size	min_samples	xi	cluster & no of clusters	dbi metric	clustering remark
None	5	0.05	 0-10	2.077087884	Poor clustering
5	7	0.1	 0-8	3.655451887	Poor clustering
10	10	0.01	 0-5	1.678424119	Poor clustering
None	10	0.05	 0-5	2.022911907	Poor clustering

Spectral Clustering

The dataset was analyzed using **Spectral Clustering** with **n_clusters=5** (based on domain knowledge), `affinity='rbf'`, `gamma=1.0`, `n_neighbors=10`, `assign_labels='kmeans'`, and `random_state=0`.
The model produced **5 clusters**, but they were **not well-separated**, resulting in **moderate performance**.
Evaluation metrics showed a score of **0.4**, indicating that the clustering quality is average.
Final cluster selection can be adjusted according to **domain requirements**.

Final model for mall customer segmentation

Based on the analysis of the Mall Customer dataset, the following clustering algorithms performed well for customer segmentation:
K-Means, Agglomerative Hierarchical Clustering, Affinity Propagation, and BIRCH (Bridge) Clustering.
These algorithms produced well-separated clusters and identified target customer segments effectively.
Other algorithms like **DBSCAN, HDBSCAN, OPTICS, Mean Shift, and Spectral Clustering** showed moderate to poor performance or produced many outliers, making them less suitable for this dataset.

Code GitHub Repo: https://github.com/krthiksha/Machine-Learning-Culstering