DBSCAN

Density-Based Spatial Clustering of Applications with Noise

Chiara Peppicelli

Università di Firenze
Statistical Learning

September, 2024

Clustering

- **What**: Clustering is a method of organizing data into groups (clusters) where members of the same cluster share more similarities with each other than with members of different clusters.
- **Why**: It helps in identifying patterns, simplifying data, and is widely used in fields like customer segmentation, image recognition, and more.

Density-Based Clustering : DBSCAN

- **Basic Idea**: Clusters are dense regions in the data space, separated by regions of lower object density.
- **Method**: DBSCAN.
  - → Unlike methods like K-means , DBSCAN does not require specifying the number of clusters beforehand.
  - → Can find clusters of arbitrary shapes.

Density Definition

- Density is estimated for a particular point by counting how many points fall within a specified radius, known as Eps (ϵ), around that point.
- **Eps-Neighborhood**: For a point p, is the set of points located within a radius of ϵ from p.

$$N_\epsilon(p) : \{q | d(p, q) \leq \epsilon\}$$

Key Parameters of DBSCAN: Eps and MinPts.

- **Dense Region**: Eps-Neighborhood of a point that contains at least "*MinPts*" objects.

Definition (Key Parameters)

- **Eps**: The maximum distance between two points to be considered part of the same neighborhood.
- **MinPts**: The minimum number of points required to form a dense region.

Classification of Points

- **Core Points**: Points with at least *MinPts* neighbors (including itself) within a distance of *Eps*.
- **Border Points**: Points that are not core points but are within the Eps-Neighborhood of a core point.
- **Noise Points**: Points that are neither core points nor border points; these are considered outliers.

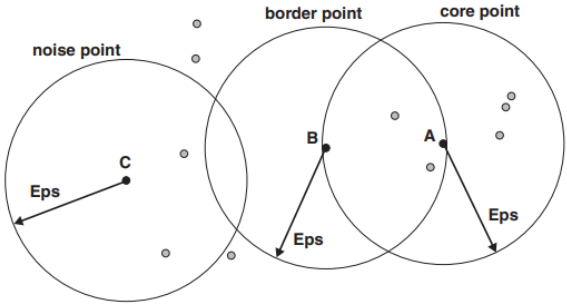


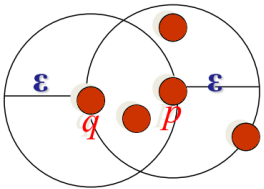
Figure. Core, border, and noise points.

Density-reachability

Definition (Directly Density-Reachable)

A point q is directly density-reachable from point p if p is a core object and q is in p's Eps-neighborhood.

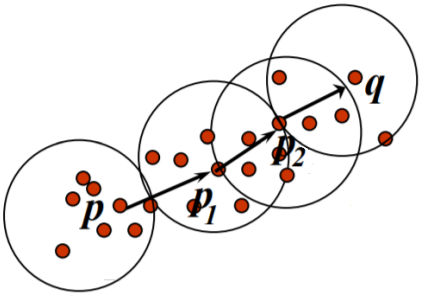
- Density-reachability can be asymmetric.
  - q is directly density-reachable from p.
  - p is not directly density-reachable from q.



Density-reachability

Definition (Density Reachable)

A point q is density-reachable from a point p if there is a chain of points $p_1, p_2, \ldots, p_n$ with $p_1 = p$ and $p_n = q$, such that p_{i+1} is directly density-reachable from p_i.



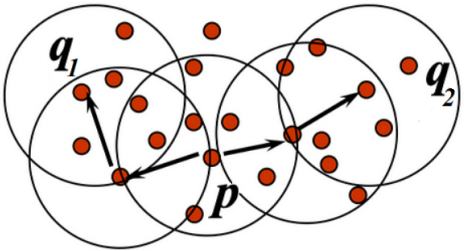
- Density-reachability extends direct density reachability.
- The relation is transitive but can be asymmetric.

Density-connectivity

Definition (Density-Connected)

A point q_1 is density-connected to a point q_2 with respect to *Eps* and *MinPts* if there exists a point p such that both q_1 and q_2 are density-reachable from p.

- Density-connectivity is symmetric.
- Can be defined for two border points.



DBSCAN Algorithm

Algorithm 1: DBSCAN Algorithm

Input: Data points X, radius ϵ, minimum points *MinPts*

Output: Clusters and noise points

1. Label all points as unvisited.

2. for *each point* $x_i \in X$ **do**

  if x_i *is unvisited* **then**

    Find all points within distance ϵ of x_i;

    if *number of neighbors* $\geq$ *MinPts* **then**

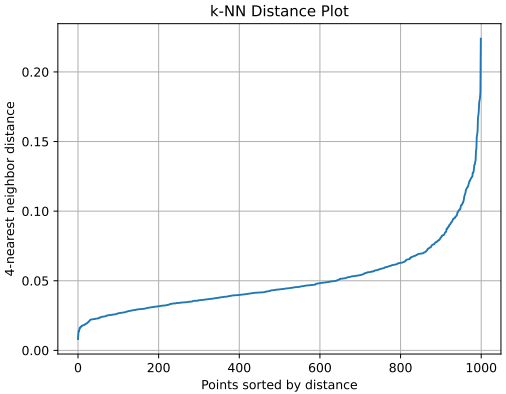
      Label x_i as a core point; Create a new cluster and include x_i; Expand the cluster with density-reachable points;

    else

      Label x_i as noise;

Parameters Estimation

- **MinPts**: The larger the data set, the larger the value of MinPts should be chosen. MinPts must be chosen at least 3.
- **Eps**: The value can be chosen by using a **k-distance graph**, plotting the distance to the k = MinPts nearest neighbor of each point in crescent order. Good values of Eps are where this plot shows a strong bend (elbow).



Strengths of DBSCAN

- **No need to predefine number of Clusters:** It does not require specifying the number of clusters beforehand.
- **Handles Noise:** Effectively identifies and manages noise and outliers in the dataset, marking them separately.
- **Identifies Arbitrarily Shaped Clusters:** It can find clusters of various shapes and sizes, unlike methods that assume spherical clusters.

The Two Moons Dataset

Characteristics :

- **Arbitrary Shapes**: The "Two Moons" dataset consists of two crescent-shaped groups.
- **Consistent Density Across Clusters**: Each moon has a relatively uniform density.
- **Separation**: The dataset has a distinct low-density area separating the two crescent groups.

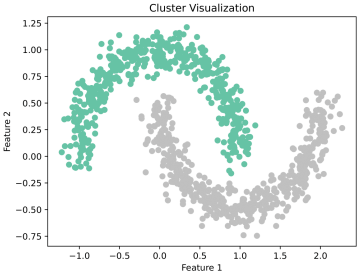


Figure: Data set "Two Moons"

The Two Moons Dataset

Comparison with K-Means

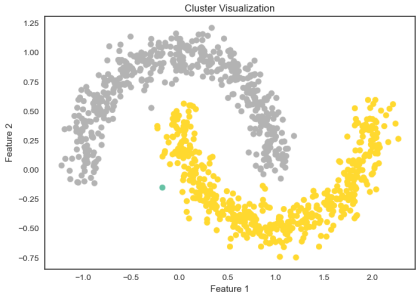


Figure: DBSCAN Clustering

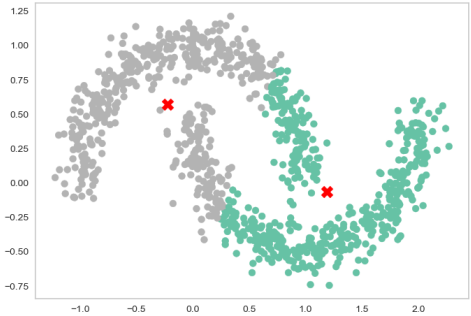


Figure: K-Means++ Clustering

The Two Moons Dataset

Comparison with other Clustering Methods

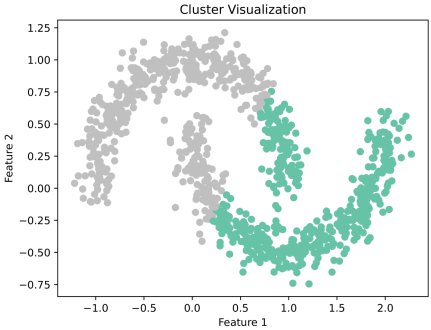


Figure: Spectral Clustering

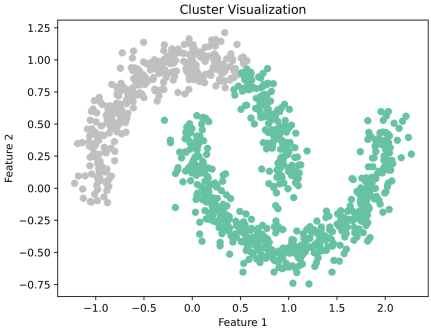


Figure: Hierarchical Clustering

Limitations of DBSCAN

- **Difficulty in Parameter Selection:** Determining the optimal values can be challenging and often requires domain knowledge or trial and error.
- **Challenges in High Dimensions:** In high-dimensional spaces, distance measures may lose their effectiveness, making it difficult for DBSCAN to accurately identify dense regions and clusters.

Data with Closely Positioned Clusters

Figure: Closely Positioned Clusters

- **Proximity Issues:** DBSCAN may struggle when clusters are very close to each other. It can merge nearby clusters if the distance between them is smaller than the specified ϵ parameter. This can lead to incorrect clustering results where distinct clusters are erroneously combined.

Data with Closely Positioned Clusters

Comparison with K-Means

Figure: DBSCAN Clustering

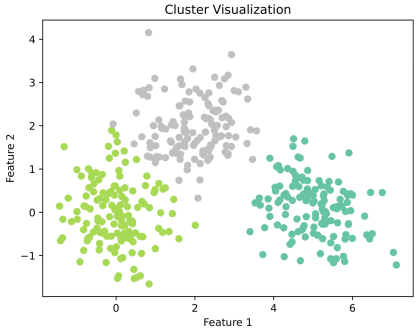


Figure: K-Means++ Clustering

Data with Closely Positioned Clusters

Comparison with other Clustering Methods

Figure: Spectral Clustering

Figure: Hierarchical Clustering

Varying Density Clusters

- **Problems with Varying Densities:** DBSCAN Struggles with datasets where clusters have significantly different densities, which may result in suboptimal clustering.

Figure: Clusters of Varying Densities

Varying Density Clusters

Tuning of the parameters to get better results

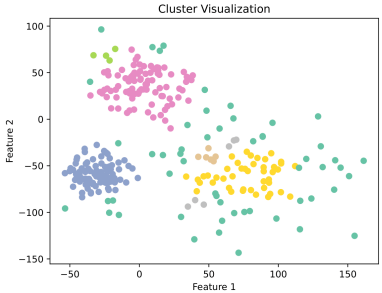


Figure: DBSCAN Clustering with bad parameters (eps=11, minpts=4)

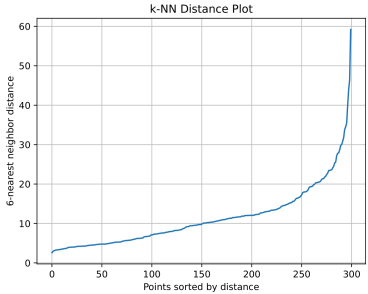


Figure: 6-Dist graph to choose a better eps

Varying Density Clusters

Comparison with K-Means

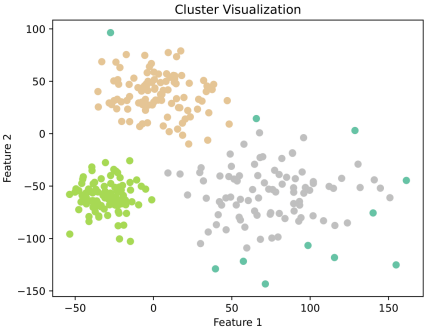


Figure: DBSCAN Clustering (eps=20, minpts=6)

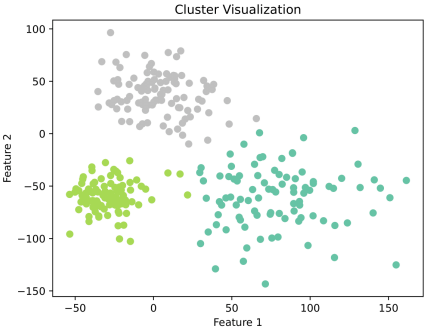


Figure: K-Means++ Clustering

Varying Density Clusters

Comparison with other Clustering Methods

Figure: Spectral Clustering

Figure: Hierarchical Clustering

Real Dataset

- **Dataset Overview**: The dataset "**Customer Personality Analysis**" includes 2240 datapoints (customers) and 29 features related to customer's demographics, purchase history, and responses to previous marketing campaigns.
- **Objective**: Segment customers based on their characteristics and behaviors using clustering algorithms.

Data Preprocessing

Key Features Creation

- I created a new column called "Age" by subtracting "Year_Birth" from 2024.
- I combined various product spending into a single feature "MntTotal".
- Aggregated online and store purchases into "Total_Online_Purchases" and "Total_Store_Purchases".
- Created "Num_Accepted" to track the number of accepted offers across all campaigns.
- Derived family size by combining marital status and the number of children ("Fam_size").
- Calculated the number of years each customer has been with the company ("Dt_Customer").

Data Preprocessing

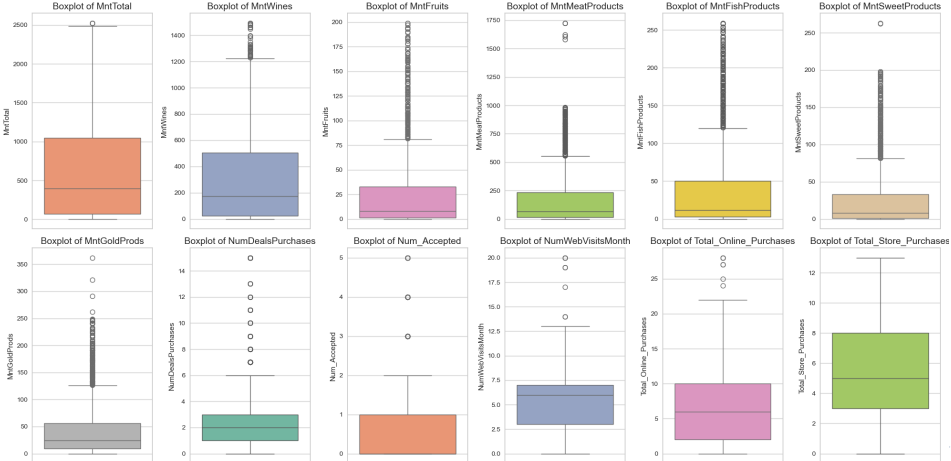
Simplifying the Data

- Converted "Education" values into numerical categories:
  0: Basic (Undergraduate) 1: Graduate 2: Postgraduate (Master, PhD)
- To simplify the dataset, irrelevant or redundant features were removed.

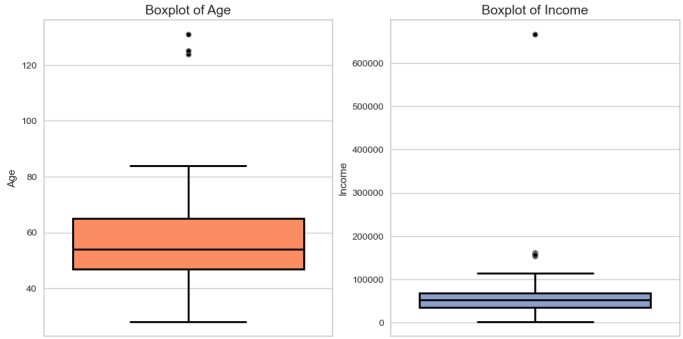
➔ This process resulted in a reduction of the dataset to **19 features**.

- Addressed missing values in the "Income" feature using mean imputation.

Data Visualization: Boxplots



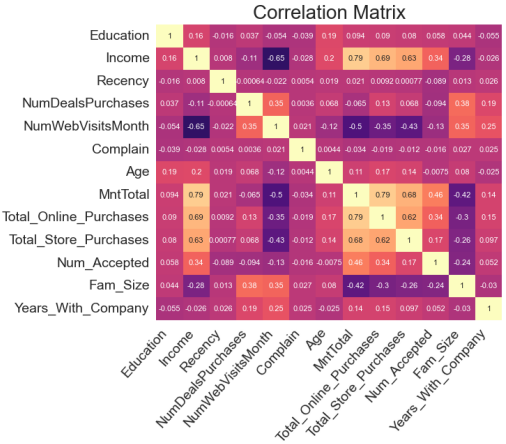
Data Visualization: Boxplots



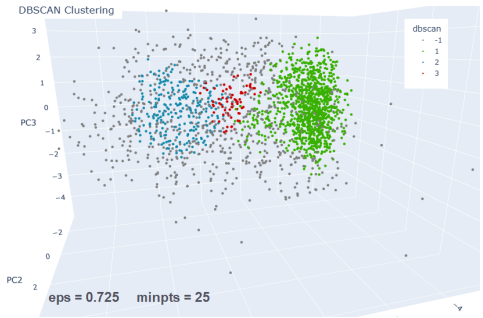
- In "Income" and "Age" there were suspicious values (e.g. an age of 120 years).
- Removed outliers in those features to enhance the accuracy of subsequent analyses.

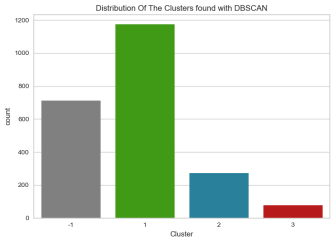
Data Visualization : Correlation Heatmap

- **Income**: Positive correlation with every purchase category but strong negative correlation with monthly website visits. Also negative correlation with family size.
- **Web Visits**: Negative correlation with spending across various product types, but positive with discounted purchases.



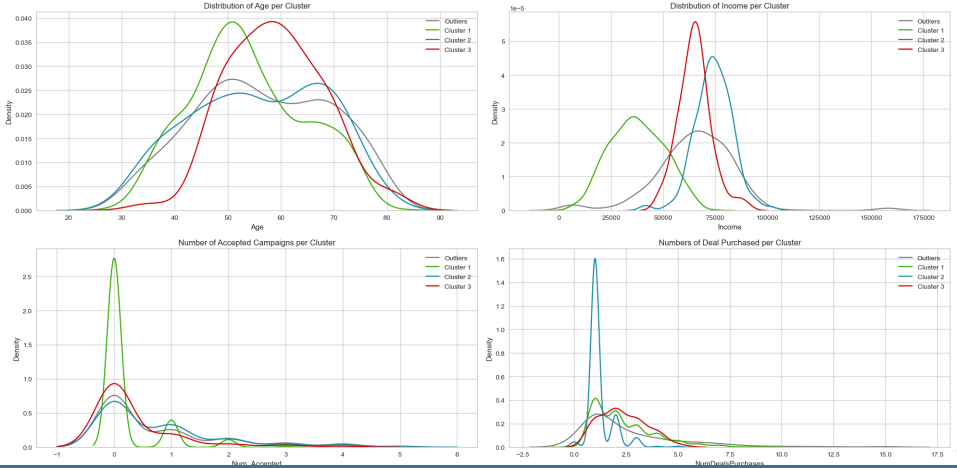
DBSCAN Clusters



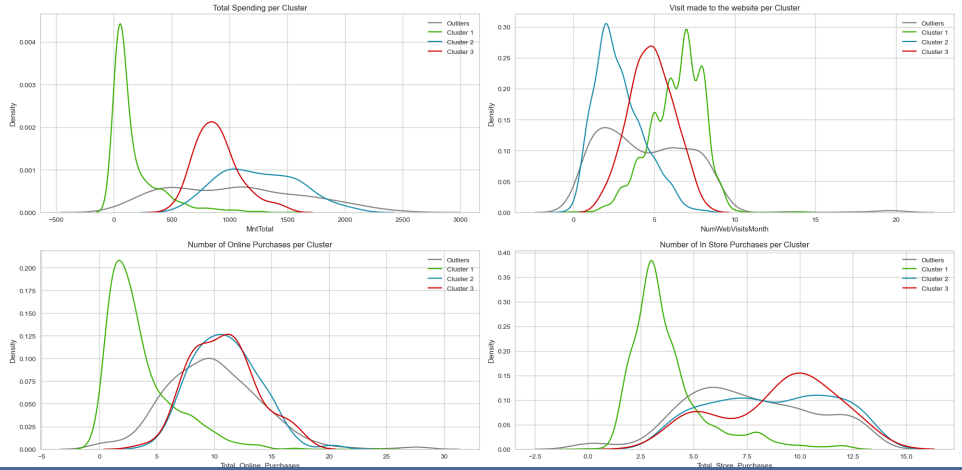


Noise	Cluster 1	Cluster 2	Cluster 3
713	1174	271	78

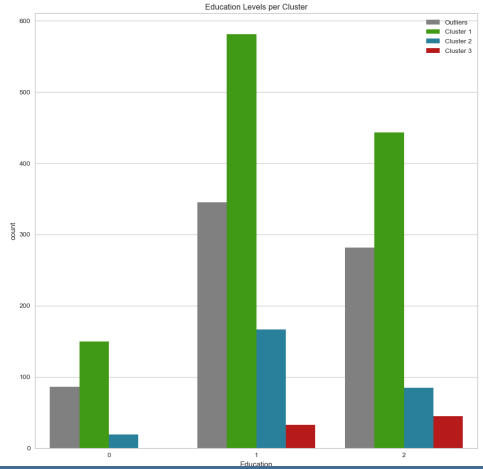
Analyzing Data Distributions: Density Plots

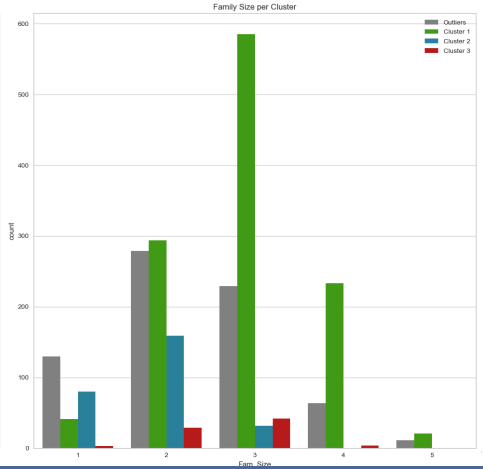


Analyzing Data Distributions : Density Plots



Analyzing Data Distributions : Bar Plots





Analyzing Data Distributions : Tables

- This table outlines **demographic** and **engagement metrics**.
- **Cluster 1**: The lowest income cluster, fewest online and store purchases but they are the ones that visits the website more and take advantages of deals.
- **Cluster 2**: Highest income but the lowest number of deals purchased.
- **Cluster 3**: Strong income, buy more deals than Cluster 2.

Clusters	Income	DealsPurchases	WebVisits	Online	Store
1	37649.35	2.34	6.39	3.58	3.80
2	74273.28	1.22	2.84	10.98	8.71
3	65230.21	2.35	4.83	10.82	8.97

Analyzing Data Distributions : Tables

- This table focuses on **spending behaviors** across different product categories.
- **Cluster 1**: Minimal ammount spent on all categories.
- **Cluster 2**: Highest total spending. They show a strong preference for wines and meat products.
- **Cluster 3**: Substantial spending, particularly on wines. Overall expenditure lower than Cluster 2.

Clusters	Total	Wines	Fruits	Meat	Fish	Sweet	Gold
1	169.74	97.56	5.25	33.80	7.65	5.34	20.13
2	1256.46	551.75	60.42	421.53	85.54	63.72	73.51
3	895.90	521.78	30.35	204.00	42.01	31.73	66.03

Clusters Profiling made with DBSCAN

- **Cluster 1**:
  - Profile : Lower-income, moderate-aged customers (average age 53.7 years).
  - Education: Slightly above average.
  - Spending: Low across all product categories.
  - Engagement: High web visits, rely more on store purchases.
  - Family Size: Larger family size (average of 2.91 members).
- **Marketing suggestions**: Offer promotions on essential products and budget-friendly loyalty programs to increase spending on basic items.

Clusters Profiling made with DBSCAN

- **Cluster 2**:
  - Profile: High-income, moderate-aged customers (average age 55.9 years), loyal big spenders.
  - Education: Average.
  - Spending: Highest spending on all product categories, but especially on wine and meat.
  - Engagement: Frequent online and store purchases, but low number of web visits and deals purchased.
  - Family Size: Smaller family size (average of 1.82 members).
- **Marketing suggestions**: Focus on premium services, luxury product promotions, and subscriptions for exclusive items to enhance the shopping experience for high-value customers.

Clusters Profiling made with DBSCAN

- **Cluster 3**:
  - Profile: Moderate to high-income, older customers (59.1 years).
  - Education: High.
  - Spending: High on wine, meat moderate on other categories.
  - Engagement: High online and store purchases, moderate number of web visits, relatively high purchasing of deals.
  - Family Size: Medium-sized families (average of 2.6 members).
- **Marketing suggestions**: Encourage loyalty through special offers and personalized incentives.

K-means++ Clusters

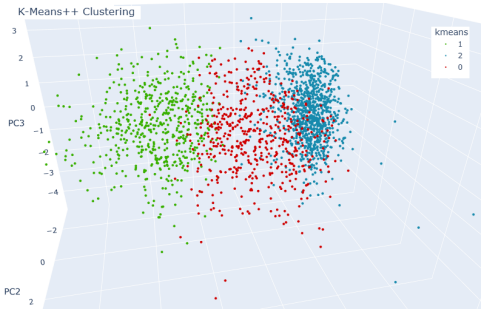


Figure: KMeans++ Clusters

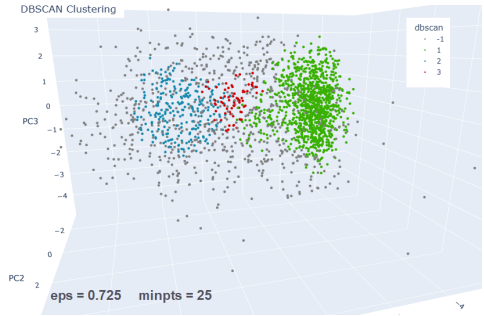


Figure: DBSCAN Clusters

Both algorithms identified key customer segments that follow similar patterns.

DBSCAN vs Kmeans++

- **KMeans**: Its clusters are distinct and well-defined, making it ideal for simple, interpretable segmentation. However, it assumes globular shapes, which may lead to less accurate segmentation with complex customer behaviors.
- **DBSCAN**: Excels at identifying customer groups with unusual behaviors through its noise cluster, uncovering diverse segments that KMeans may force into predefined clusters, potentially missing unique profiles and marketing opportunities.

Method	Silhouette Coefficient	Davies-Bouldin Index	Calinski-Harabasz Index
DBSCAN	0.3746	**0.8589**	1169.6773
KMeans++	**0.3805**	1.0314	**2111.9270**

Conclusion

- ✓ **Outlier Identification**: The algorithm effectively identified noise points, which can help in understanding market segments that are not well-represented or are outliers.
- ✓ **Natural Cluster Formation**: DBSCAN does not require a predefined number of clusters, allowing it to adapt to the data's inherent structure, allowing even flexible shapes.
- ✗ **Parameter sensitivity**: Small changes in parameter settings can lead to significant shifts in results, making it challenging to find the right balance.

References

Analytics India Magazine (2024).

A tutorial on various clustering evaluation metrics.
Accessed: 2024-09-07.

Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996).

A density-based algorithm for discovering clusters in large spatial databases with noise.
In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, page 226–231. AAAI Press.

Patel, A.

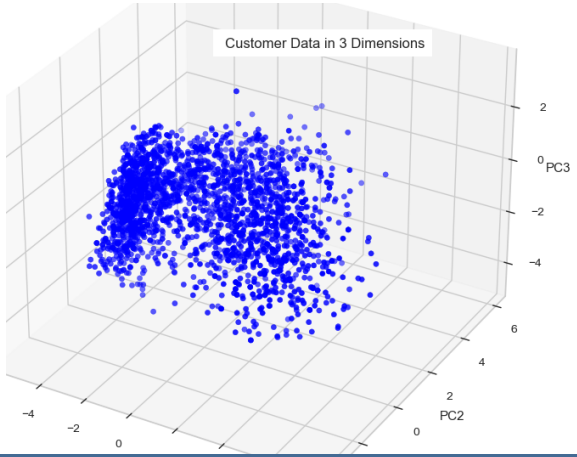
Kaggle.
https://www.kaggle.com/datasets/imakash3011/customer-personality-analysis.

DBSCAN Algorithm

Complexity

- **Time Complexity**: $O(n^2)$ (n number of data points) for naive approaches, but spatial indexing structures (such as k-d tree) can reduce it to $O(nlog(n))$ on average for low-dimensional data.
- **Memory Complexity**: $O(n)$, the algorithm needs to store the dataset and the labels for each point.

Principal Component Analysis (PCA)



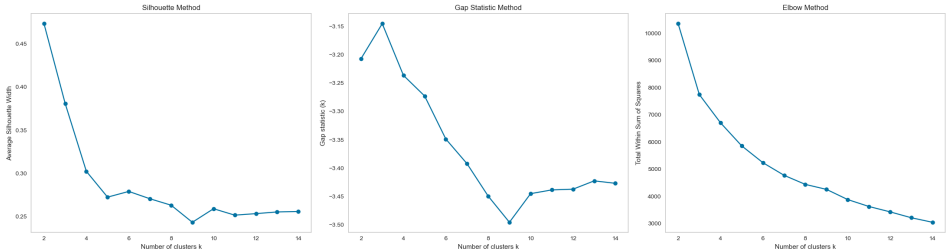
- Explained variance ratio for each principal component:
  PC1: 0.36
  PC2: 0.09
  PC3: 0.08

➛ **Total variance** explained by the first 3 PCs: **0.53**

K-means++

Parameters Estimation

- The Silhouette Method suggests k=2, but k=3 is also acceptable. The Gap Statistic peaks at k=3, while the Elbow Method shows that the elbow is reached further along, suggesting a bigger k (k=4 or k=5).
- Ultimately, choosing **k=3** seems like a reasonable compromise.



Clustering Evalutation Metrics: Silhouette Score

- The **Silhouette Score** measures how similar a data point is to its own cluster compared to other clusters. It is defined as:

$$S(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}$$

where:

  - $a(i)$ is the mean intra-cluster distance of point i,
  - $b(i)$ is the mean nearest-cluster distance of point i.

- It ranges from -1 to 1, where 1 indicates the sample is well-separated from other clusters, 0 means is close to the boundary between clusters, and -1 suggests is likely assigned to the wrong cluster.

Clustering Evalutation Metrics: Davies-Bouldin Index

- The **Davies-Bouldin Index** evaluates the average similarity ratio between each cluster and the one most similar to it. It is defined as:

$$DB = \frac{1}{n} \sum_{i=1}^{n} \max_{j \neq i} \left(\frac{s(i) + s(j)}{d(i,j)} \right)$$

  where:
  - $s(i)$ is the average distance between each point of cluster i and its centroid,
  - $d(i,j)$ is the distance between the centroids of clusters i and j.
- A lower Davies-Bouldin Index indicates better clustering performance.

Clustering Evalutation Metrics: Calinski-Harabasz Index

- The **Calinski-Harabasz Index**, also known as the Variance Ratio Criterion, is defined as:

$$CH = \frac{BCSS}{WCSS} \cdot \frac{n-k}{k-1}$$

where:

- BCSS (Between-Cluster Sum of Squares) $= \sum_{i=1}^{k} n_i \|c_i - c\|^2$
- WCSS (Within-Cluster Sum of Squares) $= \sum_{i=1}^{k} \sum_{x \in C_i} \|x - c_i\|^2$
- n is the number of samples,
- k is the number of clusters.

- Higher values of the Calinski-Harabasz Index indicate better clustering.