Machine Learning Project: Clustering on Buddymove Dataset

Ahmed Abdullah Shahid

Liverpool Hope University

COMM022 - Machine Learning Algorithms

24012028@hope.ac.uk

Table of Contents

Abstract

The BuddyMove dataset is a collection of user reviews that reflect people's preferences across different activities, such as sports, religious activities, nature outings, theater visits, shopping, and picnics. Think of it to understand how individuals engage in various leisure activities based on their ratings. This dataset is often used to explore machine learning and data analysis techniques, particularly for clustering and recommendation systems. It helps uncover patterns in user behavior, making it useful for businesses, travel agencies, or social platforms looking to personalize experiences.

The main goal of this analysis was to use two clustering techniques, K-Means and DBSCAN to group users based on their activity preferences. By identifying distinct user segments, this research aims to uncover patterns in behavior that can help create personal recommendations, targeted marketing strategies, and improved services. Understanding these user groups can provide valuable insights for businesses and platforms looking to tailor experiences to individual interests.

We will use this dataset to perform key data analysis tasks, such as summarizing the data, calculating statistical insights, and visualizing relationships between different variables. One important application is grouping users based on their activity preferences. By using clustering techniques like DBSCAN and K-Means, we can identify users with similar interests, making it easier to uncover patterns in their behavior and preferences.

Introduction

In today's data-driven world, understanding customer behaviour and preferences is crucial for businesses aiming to offer personalized and effective services. The BuddyMove dataset provides valuable insights into user activity preferences across different categories, including sports, religious activities, nature outings, theatre, shopping, and picnics. By analysing this dataset, we can uncover meaningful trends and patterns among different user groups, helping businesses tailor their services to better meet customer interests and needs. This report focuses on the application of two powerful clustering techniques, k-means and DBSCAN (Density-Based Spatial Clustering of Applications with Noise) to segment users based on their activity preferences. Clustering is a type of unsupervised machine learning that groups similar data points together, helping to identify natural clusters within a dataset without pre-labeled outcomes.

This analysis has two main objectives: first, to identify distinct groups of users who share similar activity preferences, and second, to compare the effectiveness of K-Means and DBSCAN in detecting these groups. K-Means is widely used because it is simple and effective at finding well-defined, circular clusters. However, it requires specifying the number of clusters in advance. On the other hand, DBSCAN is more flexible, as it can discover clusters of any shape and is better at handling outliers and noise, making it particularly useful for complex datasets.

Our goal is to provide actionable insights that help optimize resource allocation, enhance personalized services, and improve user engagement strategies. By understanding the unique interests and behaviors of different user groups, organizations can tailor their services to better meet customer needs. This personalized approach not only enhances user satisfaction but also fosters stronger loyalty and long-term engagement.

In the following sections of this report, we'll take a closer look at the step-by-step approach used in this analysis, including data preparation, applying K-Means and DBSCAN clustering, and evaluating the results. We'll also explore the key insights gained from these clusters and how they can be applied to enhance service delivery and improve targeted marketing strategies.

Methodology

Our first step will be data cleaning to ensure accuracy and consistency. Next, we'll explore the data to uncover patterns and relationships. Finally, we'll apply DBSCAN and K-Means clustering to group users based on their activity preferences.

1. Data Cleaning

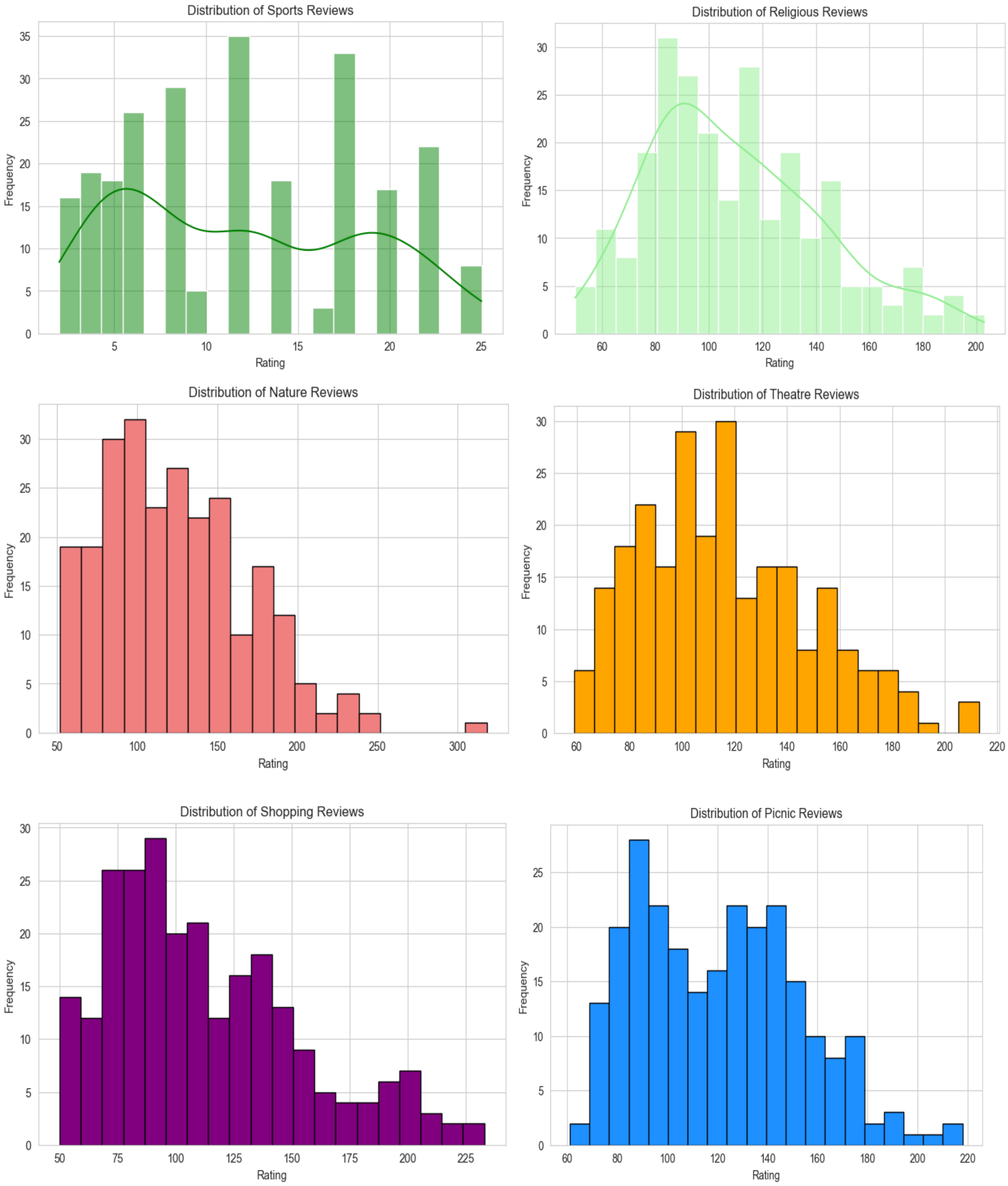
After running a few data cleaning functions, we can see that the data was already in good shape. These steps were mainly taken to ensure the data was fully prepared for analysis. The only adjustment made was removing the word "User" from each entry in the "User Id" column and converting the values from strings to integers. This small change helps streamline the data modeling process and ensures smoother analysis.

To remove the word user from the "User Id" column, I used str.replace('User ', ''), converting the values into integers for easier processing. Next, I explored the dataset's basic statistics using buddymove.describe(), which provided insights such as the number of rows, data types of each column, and memory usage. To check for missing values, I used buddymove.isnull().sum() and, if necessary, buddymove.dropna() and buddymove.drop_duplicates() to remove any incomplete and duplicate rows. However, after running these checks, I found that there were no missing values in the dataset, confirming that the data was already clean and ready for analysis.

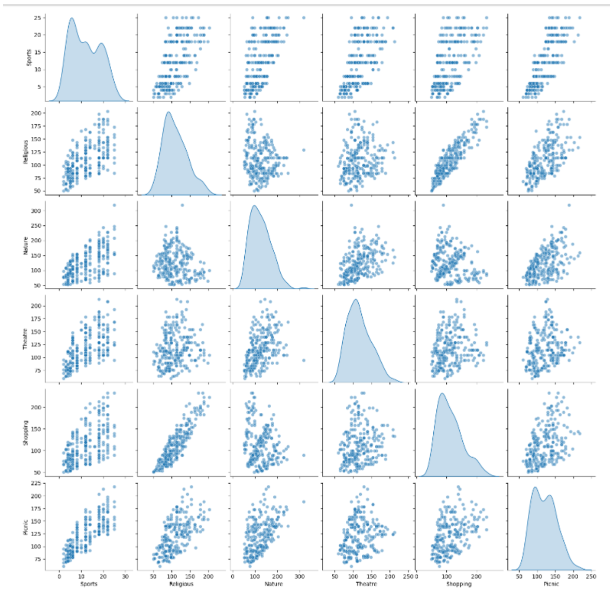
2. Data Exploration

In this section, I created six graphs, each comparing "User Id" with one of the other attributes individually. These visualizations represent user reviews across different categories, including sports, religion, nature, theater, shopping, and picnics. However, after analyzing the graphs in the Jupiter notebook file, it became clear that they don't reveal significant insights, as the "User Id" column itself doesn't provide meaningful relationships with these attributes.

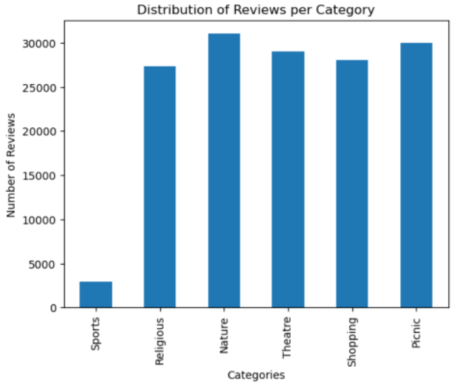
Next, I created histograms to visualize the distribution of reviews across different categories, including sports, religion, nature, theater, shopping, and picnics. These histograms illustrate how frequently users have reviewed each activity. From this data, we can uncover valuable insights. For example, we can identify which categories receive the most reviews, indicating popular interest among users. Similarly, we can see which categories have fewer reviews, highlighting fewer common preferences. This helps us understand overall engagement levels across different activities. The histograms below provide a clear visual representation of these trends.



I also created a Pair-plot, which displays scatter plots for all the attributes, allowing for a comprehensive visualization of their relationships.



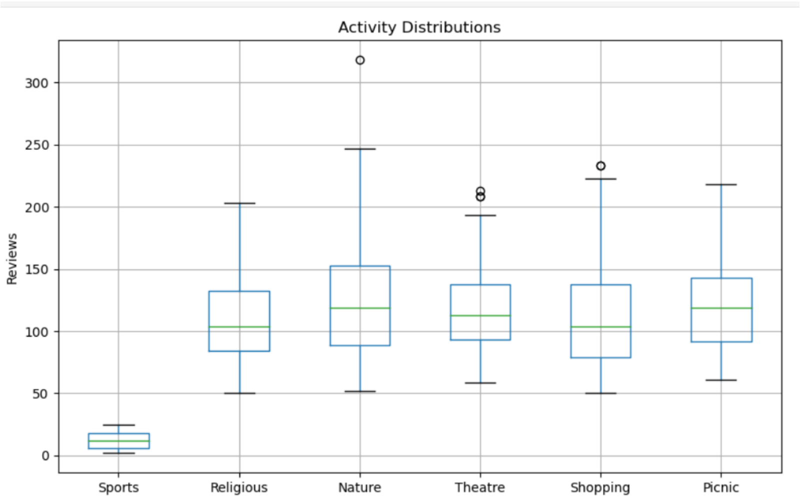
I also created a bar graph to visualize the distribution of reviews per category. This graph compares the total number of reviews across each activity category. For example, it clearly shows that, compared to the other categories, sports reviews are notably low.



The boxplot below provides a visual representation of the distribution of user reviews for each activity category. A few key observations can be made:

- The sports category may have a higher median review compared to religious activities.
- An outlier in the shopping category suggests that some users gave reviews that were much higher or lower than the rest.
- A broader IQR (Interquartile Range) for nature activities indicates greater diversity in user opinions.

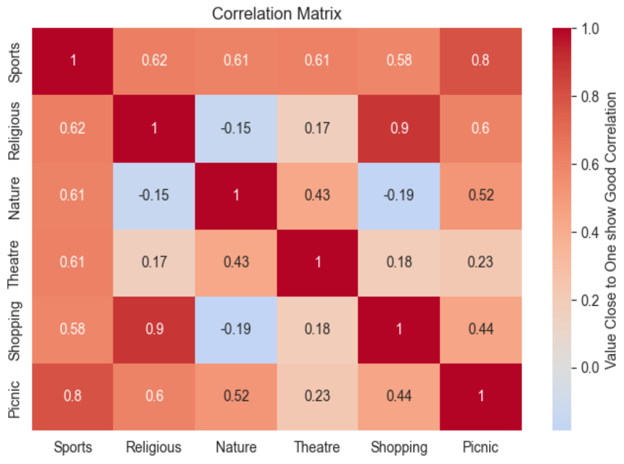
This boxplot helps us better understand the central tendency, distribution, and any outliers in the reviews across different activity categories in the BuddyMove dataset, providing useful insights into user behavior.



The heatmap allows you to visually assess the relationships between different activity categories. Values close to 1 indicate a strong positive correlation, meaning the two variables move together, while values closer to -1 suggest a weak or negative correlation, where an increase in one variable could lead to a decrease in the other. For instance, you might observe that:

- **"Religious and Shopping"** and **"Sports and Picnic"** show a strong positive correlation. This implies that if a user is highly engaged in one activity, they are likely to also be engaged in the other. For example, users who are active in religious activities tend to also have a strong interest in shopping, and those who enjoy sports activities may also enjoy participating in picnics.
- On the other hand, users who are often involved in nature-based activities may not show the same level of interest in shopping, as indicated by the weak or negative correlation between **"nature"** and **"shopping."** This suggests that people who enjoy spending time outdoors and engaging with nature might not prioritize shopping or might not be as enthusiastic about it compared to other activities.

These insights can be useful in understanding user behavior patterns and can help businesses tailor services or marketing strategies accordingly.



At the end of the data exploration process, I compiled a summary report highlighting key insights:

- Total Users: 249
- Total Reviews: 296,660
- Average Reviews per User: 595.7
- Most Popular Category: Nature
- Least Popular Category: Sports

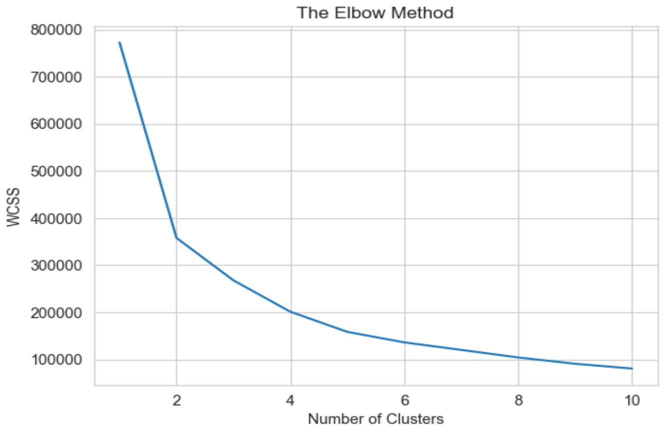
3. Data Modelling

Next, I performed data modeling using both k-means clustering and DBSCAN.

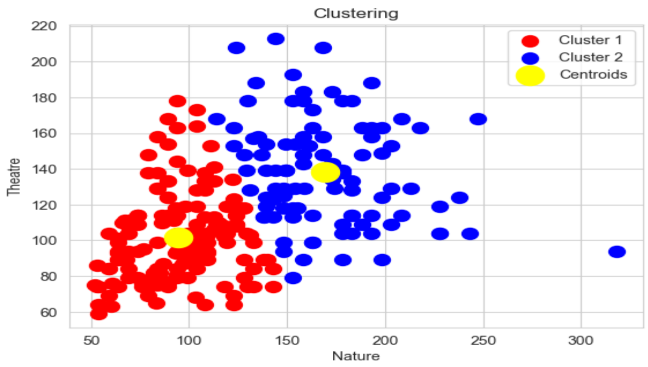
3.1 K-Means clustering

K-means clustering is a popular unsupervised machine learning technique used to partition a dataset into a predetermined number of clusters. The goal of K-means is to uncover hidden patterns or structures in the data by grouping similar data points together. In this analysis, I worked with six categories and created 15 unique combinations of them. For each combination, I applied K-means clustering. The first step in K-means clustering is determining the optimal value of *k*, which represents the number of clusters. To find this value, I used the elbow method for each combination, helping to identify the best *k* that minimizes the within cluster variance.

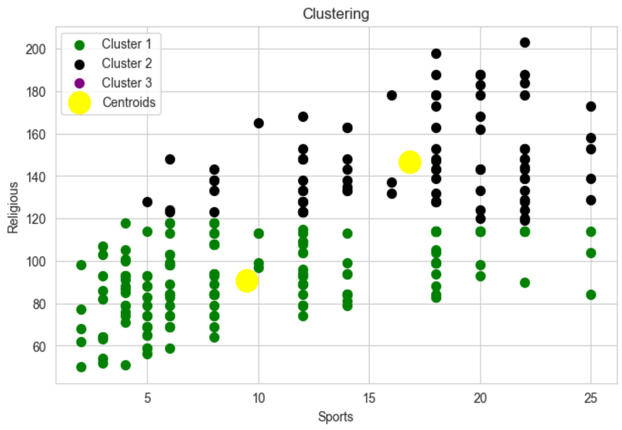
The elbow method is a practical approach for determining the optimal number of clusters (*k*) in K-means clustering. The idea behind it is to identify the point where adding more clusters no longer significantly improves the clustering performance. This point is referred to as the "elbow." In the graph below, which compares "Nature" and "Theatre," you can see how I applied the elbow method to find the best value for *k*. The sharp bend in the graph marks the elbow point, which in this case is when *k* equals 2. This indicates that 2 clusters would be the most appropriate for this combination of features.

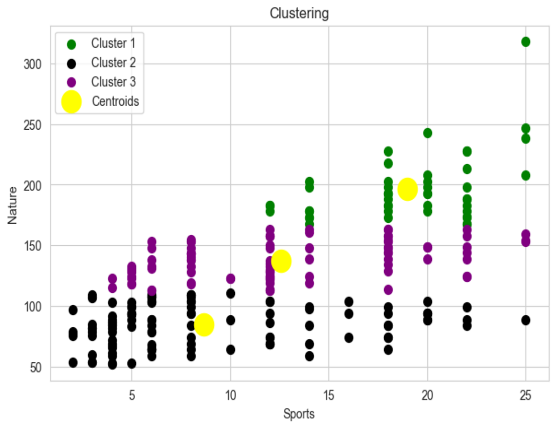


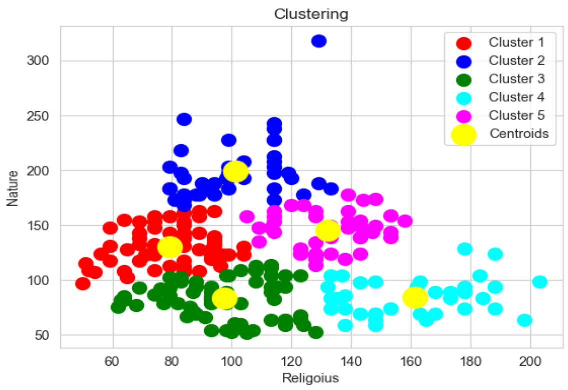
Then I have used k-mean clustering to make scatter plots for every combination. The scatter plot shows the results of clustering, with yellow points marking the centroids and red and blue points representing two distinct groups. The Silhouette Score of 0.4666 suggests that the clusters are fairly well-formed, though there is some overlap between them. Cluster 1 (red) consists of data points with lower values for "Nature" and "Theatre," while Cluster 2 (blue) contains higher values for these features.

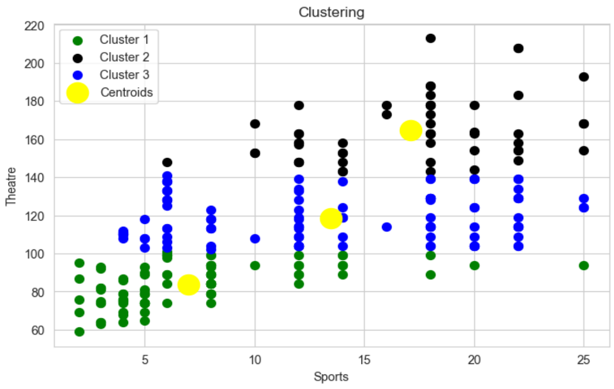

Clustering

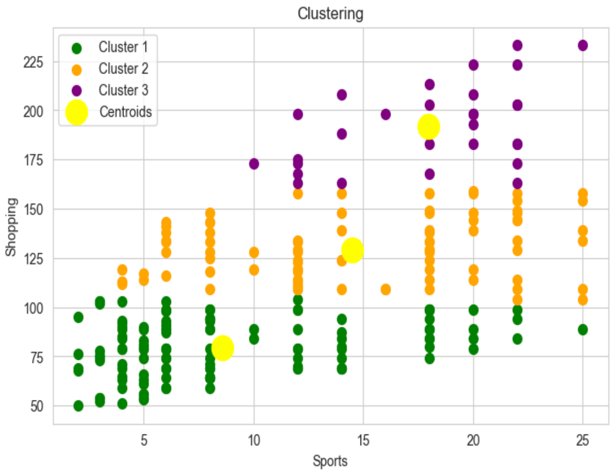
Here are the 14 additional category combinations, each analyzed using the same approach. As explained earlier, we can interpret the clusters in each plot similarly. To determine the optimal number of clusters, I used the Elbow Method and then evaluated the clustering accuracy using three different accuracy scores, Silhouette Score, Davies-Bouldin Index and Calinski-Harabasz Index. These measures helped assess how well the clusters were formed and separated, ensuring the reliability of the results.

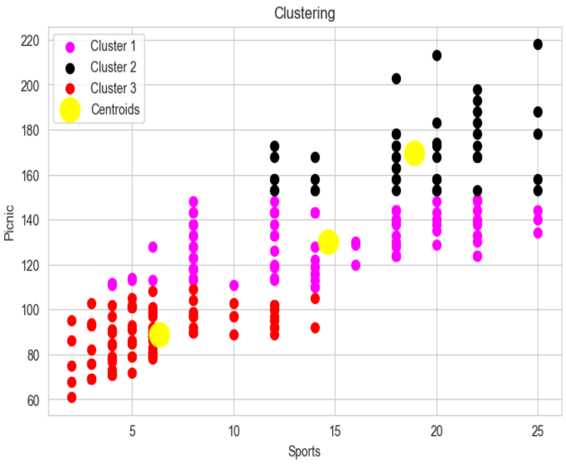

Clustering

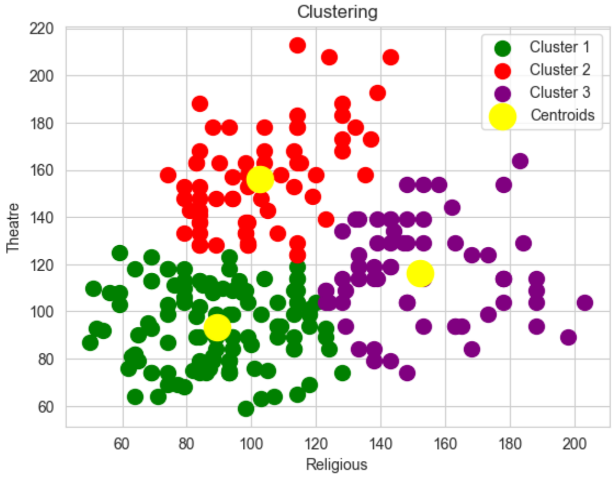

Clustering

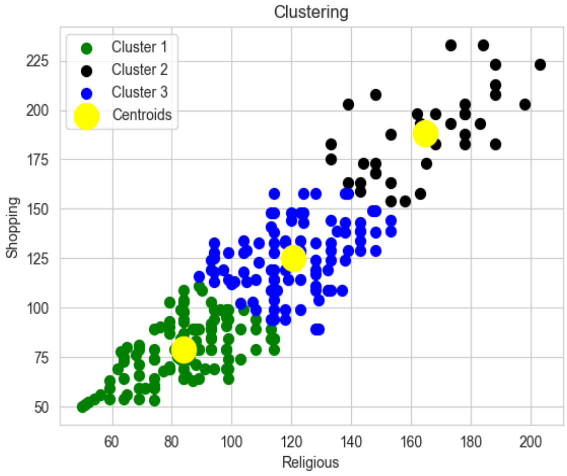

Clustering

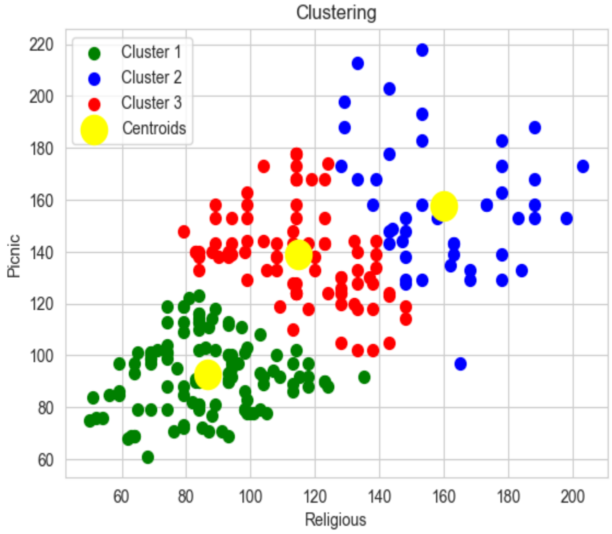

Clustering

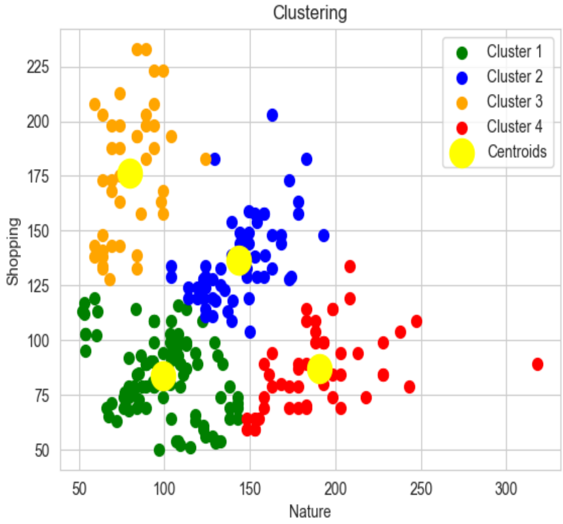

Clustering
Cluster 1
Cluster 2
Cluster 3
Centroids
Shopping
225
200
175
150
125
100
75
50
5
10
15
20
25
Sports

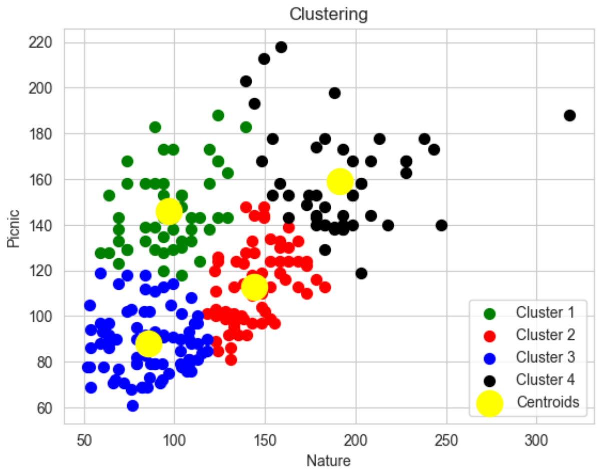

Clustering
Cluster 1
Cluster 2
Cluster 3
Centroids
Picnic
220
200
180
160
140
120
100
80
60
5
10
15
20
25
Sports

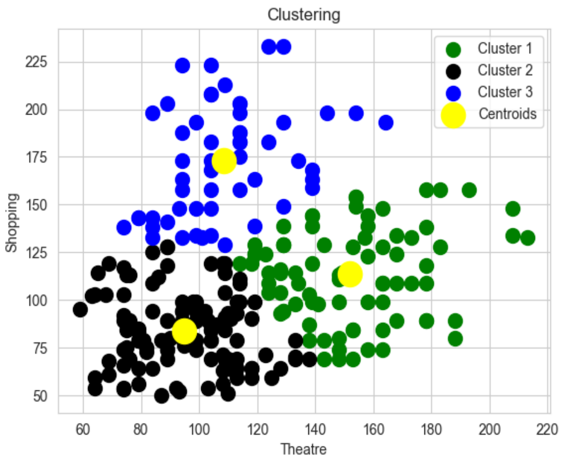

Clustering
Cluster 1
Cluster 2
Cluster 3
Centroids
Theatre
220
200
180
160
140
120
100
80
60
60
80
100
120
140
160
180
200
Religious

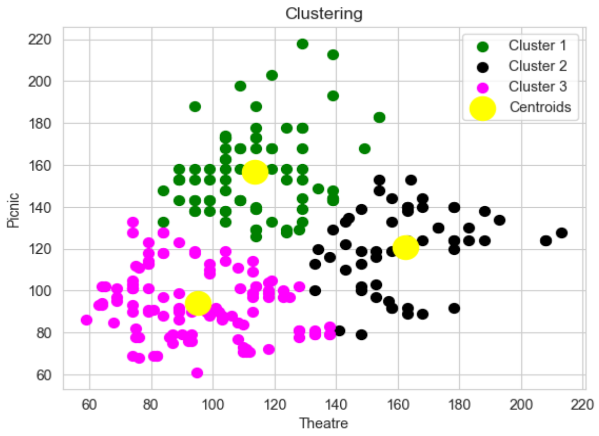

Clustering
Cluster 1
Cluster 2
Cluster 3
Centroids
Shopping
225
200
175
150
125
100
75
50
60
80
100
120
140
160
180
200
Religious

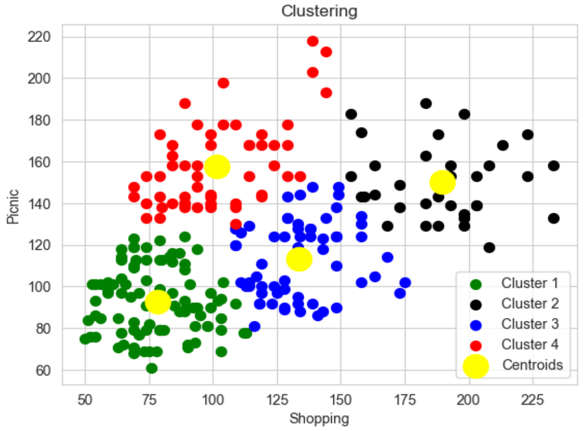

Clustering
Cluster 1
Cluster 2
Cluster 3
Centroids
Picnic
220
200
180
160
140
120
100
80
60
60
80
100
120
140
160
180
200
Religious


Clustering
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Centroids
Shopping
225
200
175
150
125
100
75
50
50
100
150
200
250
300
Nature

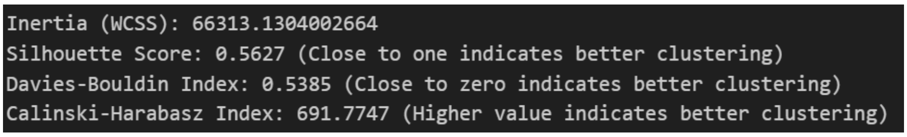






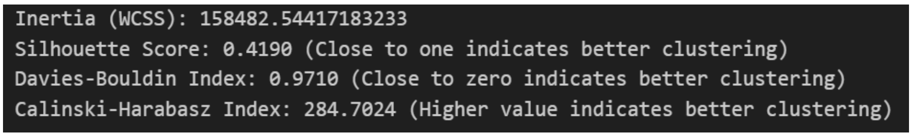


After running 15 clustering tests, the best result for distinguishing between shopping and sports was achieved with a Calinski-Harabasz Index of 691.77, a Silhouette Score of 0.5627, a Davies-Bouldin Index of 0.5385, and an Inertia value of 66,313.13. The lower Davies-Bouldin Index combined with higher Silhouette and Calinski-Harabasz scores indicates that the clusters were well-defined and distinct from each other.



```
Inertia (WCSS): 66313.1304002664
Silhouette Score: 0.5627 (Close to one indicates better clustering)
Davies-Bouldin Index: 0.5385 (Close to zero indicates better clustering)
Calinski-Harabasz Index: 691.7747 (Higher value indicates better clustering)
```

However, the lowest clustering performance was found between religion and picnic, with the Calinski-Harabasz Index equal to 284.70, the Davies-Bouldin Index equal to 0.9710, the Silhouette Score equal to 0.4190 and the Inertia equal to 158,482.54. The low Silhouette and Calinski-Harabasz scores along with the high Davies-Bouldin Index suggest significant overlap between clusters, meaning the separation was not as strong. This could indicate that these categories share similar patterns, making it harder for K-means to distinguish them effectively.



```
Inertia (WCSS): 158482.54417183233
Silhouette Score: 0.4190 (Close to one indicates better clustering)
Davies-Bouldin Index: 0.9710 (Close to zero indicates better clustering)
Calinski-Harabasz Index: 284.7024 (Higher value indicates better clustering)
```

These insights can be valuable for understanding patterns in the data, whether for customer segmentation, targeted marketing, or further analysis to refine decision-making.

3.2 DBSCAN

There are six categories in total, and from these, I created 15 different combinations to analyze using DBSCAN. DBSCAN stands for Density-Based Spatial Clustering of Applications with Noise is a popular clustering algorithm, particularly useful when clusters have varying densities or irregular shapes. Unlike K-means, DBSCAN does not require a predefined number of clusters and can also identify noise points (outliers) in the data. It works by grouping together points that are closely packed based on a density criterion, making it highly effective for datasets with complex structures.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) works by setting two key parameters:

- **minPts**: The minimum number of points required to form a dense region (also called a *core point*).
- **Epsilon (ε)**: The maximum distance between two points for them to be considered neighbors.

I have chosen an epsilon value of 0.5 and set the min_samples to 6 for DBSCAN, as these settings are yielding the best results in terms of accuracy. These values were selected based on the size and the variables of the dataset, ensuring they strike the right balance for defining clusters while minimizing noise.

- A **core point** is any point that has at least **minPts** other points (including itself) within its **ε** radius.
- These core points are typically found in the densest parts of the dataset, forming the foundation of clusters.

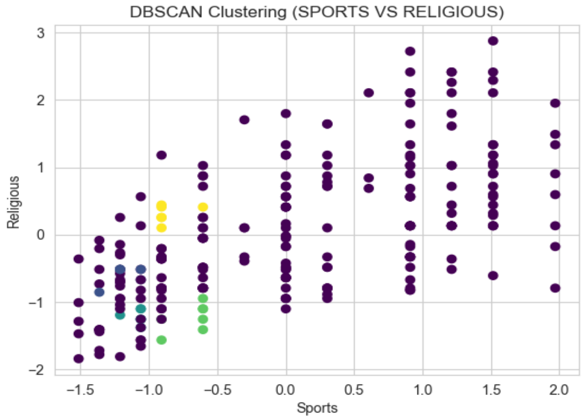
In simple terms, DBSCAN looks for clusters by identifying dense areas in the data, linking neighboring points, and distinguishing noise or outliers.

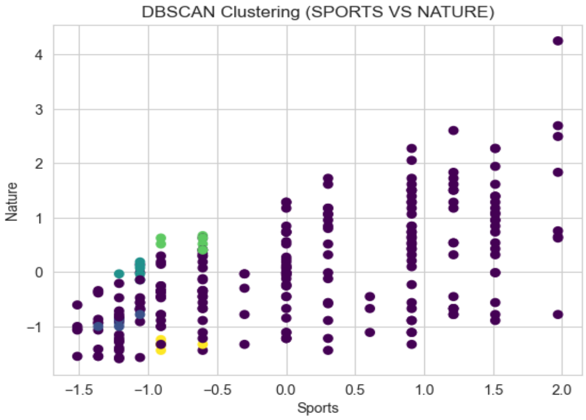
In DBSCAN clustering, **border points** are those that belong to a cluster but don’t have enough nearby neighbors to be considered core points. Instead, they are connected to a core point by falling within its ε-neighborhood. On the other hand, **noise points (outliers)** are neither core nor border points—they don’t meet the density requirement and aren’t close enough to any cluster, meaning they remain ungrouped. Essentially, core points form the dense center of a cluster, border points sit on the edges, and noise points are isolated from any cluster.

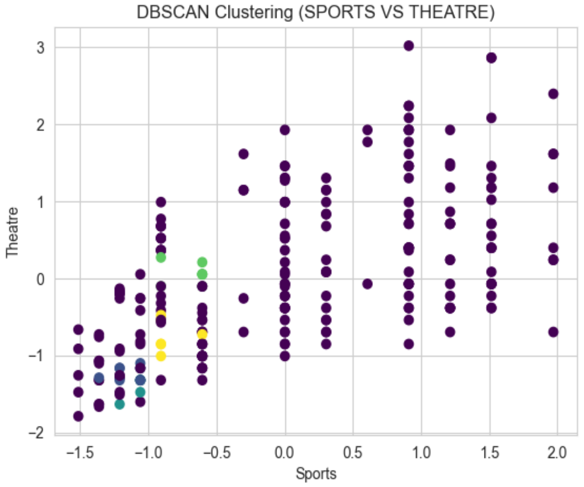
DBSCAN is robust to noise and capable of identifying clusters with irregular shapes and varying densities. However, it may struggle with datasets of varying densities or high-dimensional datasets due to the curse of dimensionality. Additionally, choosing appropriate values for epsilon and minPts can be challenging, and DBSCAN may not perform well if these parameters are not selected properly.

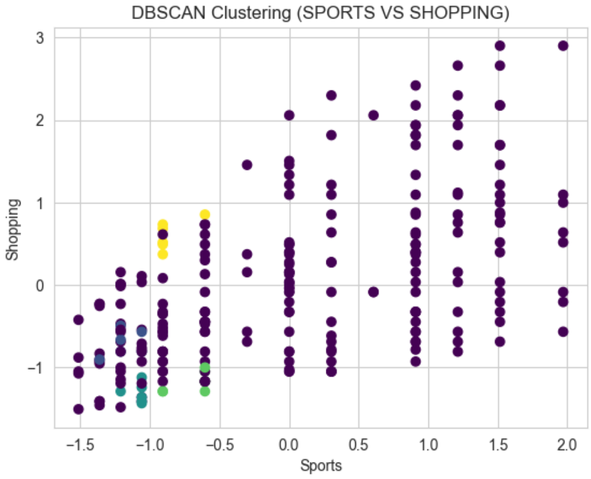
DBSCAN groups reviews that are closely packed together based on their density or similarity. Each cluster represents a set of reviews that share common themes or topics, such as sports or religious discussions, according to the chosen criteria. This way, similar reviews naturally form clusters, making it easier to identify patterns in the data.

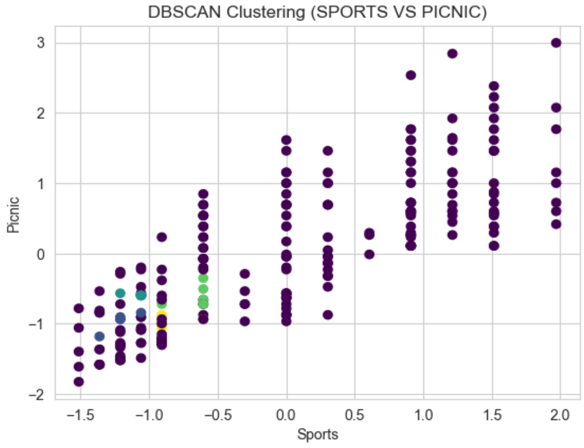
Below are the scatter plots for DBSCAN for all the combinations. We can read these graphs in the same way as for K-means clustering. Also, I have checked their accuracy score using Silhouette Score, Davies-Bouldin Index and Calinski-Harabasz Index.

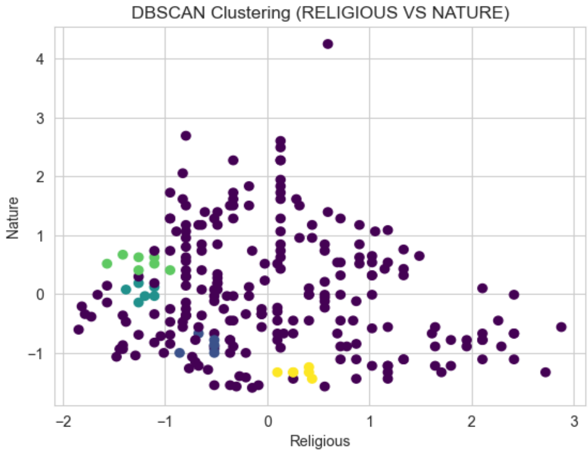

DBSCAN Clustering (SPORTS VS RELIGIOUS)
Religious
Sports
3
2
1
0
−1
−2
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0

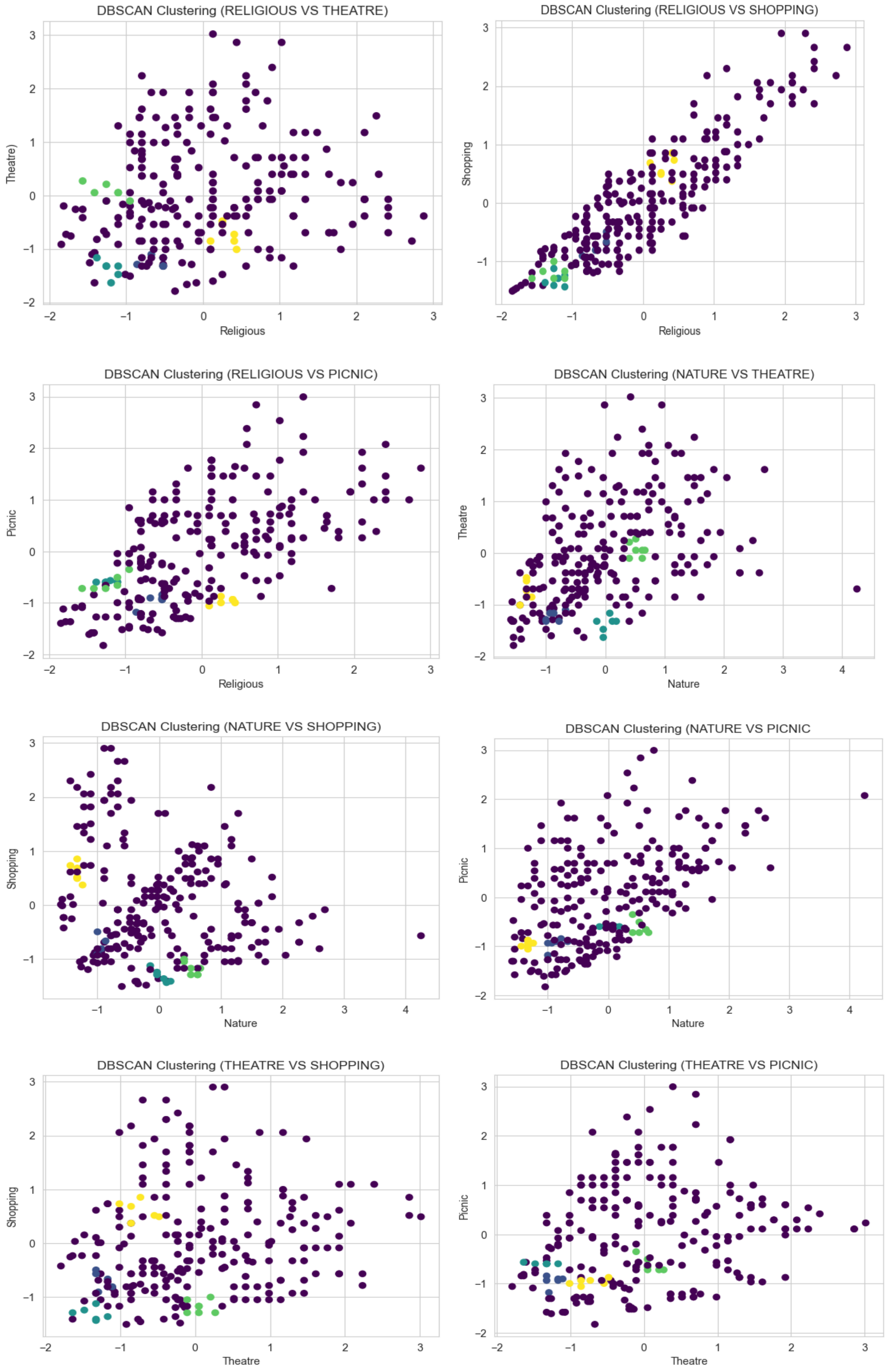

DBSCAN Clustering (SPORTS VS NATURE)
Nature
Sports
4
3
2
1
0
−1
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0


DBSCAN Clustering (SPORTS VS THEATRE)
Theatre
Sports
3
2
1
0
−1
−2
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0


DBSCAN Clustering (SPORTS VS SHOPPING)
Shopping
Sports
3
2
1
0
−1
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0


DBSCAN Clustering (SPORTS VS PICNIC)
Picnic
Sports
3
2
1
0
−1
−2
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0


DBSCAN Clustering (RELIGIOUS VS NATURE)
Nature
Religious
4
3
2
1
0
−1
−2
−1
0
1
2
3



DBSCAN Clustering (RELIGIOUS VS THEATRE)

DBSCAN Clustering (RELIGIOUS VS SHOPPING)

DBSCAN Clustering (RELIGIOUS VS PICNIC)

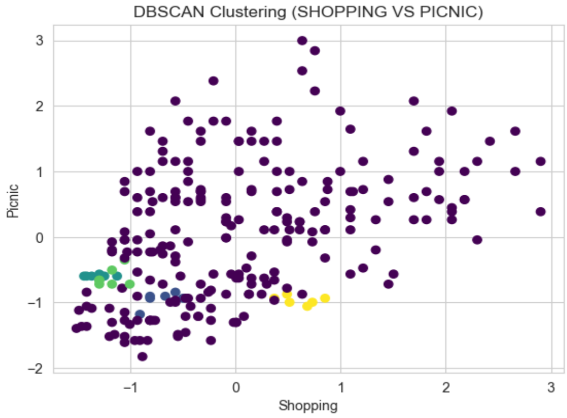
DBSCAN Clustering (NATURE VS THEATRE)

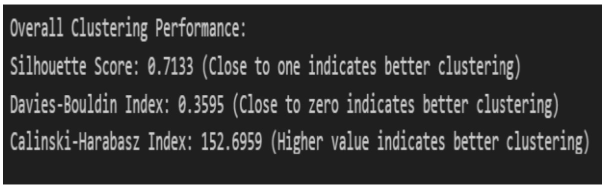
DBSCAN Clustering (NATURE VS SHOPPING)

DBSCAN Clustering (NATURE VS PICNIC

DBSCAN Clustering (THEATRE VS SHOPPING)

DBSCAN Clustering (THEATRE VS PICNIC)





```
Overall Clustering Performance:
Silhouette Score: 0.7133 (Close to one indicates better clustering)
Davies-Bouldin Index: 0.3595 (Close to zero indicates better clustering)
Calinski-Harabasz Index: 152.6959 (Higher value indicates better clustering)
```

In the data modelling process, we can see that K-means clustering is showing more accurate results than DBSCAN. The cluster boundaries are well-defined in k-means clustering while the boundaries are not properly defined in the DBSCAN part although its accuracy scores are better. So, we can say that K-means clustering is more suitable for 'Buddymove' dataset.

Critical Analysis

1. K-Mean Clustering

The K-Means clustering results, based on the Elbow Method, show moderate performance in capturing key themes like Nature, Theater, Sports, and others. While the clusters highlight meaningful patterns, metrics like the Silhouette Score, Davies-Bouldin Index, and Calinski-Harabasz Index suggest there's room for improvement in terms of cluster cohesion and separation.

1.1 Inertia (WCSS) Analysis

The wide range of inertia values (from 41,250 to 357,913) points to inconsistencies in how tightly the clusters are formed. Higher inertia values, such as 357,913.42 or 242,427.16, suggest that some clusters are more widespread and less compact, indicating significantly within-cluster variance. On the other hand, lower inertia values, like 41,250.02 and 66,313.13, reflect better cluster compactness, which is generally preferred in K-Means clustering. This variation suggests that while some clusters are well-defined, others may be too dispersed, possibly due to an unbalanced dataset.

1.2 Silhouette Score Analysis

The Silhouette Score, which ranges from 0.4052 to 0.5690, helps assess how well the clusters are defined. A score closer to 1 suggests better-defined clusters. Sports & Religion achieved the highest score of 0.5690, indicating a moderate balance of unity and separation between the clusters. In contrast, Nature & Religion scored the lowest at 0.4052, suggesting weak separation and overlap. Overall, most of the Silhouette Scores fall below 0.6, highlighting some overlap between clusters. This suggests that certain categories, like Nature, Religion, and Theater, may be harder to differentiate clearly using K-Means clustering.

1.3 Davies-Bouldin Index (DBI) Analysis

DBI values range from 0.5385 to 0.9710, with lower values indicating better clustering. Sports & Shopping achieved the best DBI of 0.5385, suggesting well-separated clusters. On the other hand, Religious & Picnic had the highest DBI of 0.9710, pointing to significant overlap and poor differentiation. The fact that many DBI values are above 0.7 suggests that several clusters lack clear separation. This could be due to sub-clusters within categories that K-Means struggles to distinguish, or similarities between categories, such as Religious & Picnic and Nature & Religious, making them harder to separate effectively.

1.4 Calinski-Harabasz Index (CHI) Analysis

CHI values range from 230.62 to 696.88, with higher values indicating better clustering. Sports & Picnic had the highest CHI at 696.88, showing well-separated clusters. In contrast, Religious & Theatre had the lowest CHI at 230.62, indicating poorly defined clusters. The variations in CHI suggest that some cluster pairs, like Sports & Picnic, are well-separated, while others, such as Religious & Theatre, have significant overlap.

Overall, the K-Means clustering results show a moderate level of effectiveness. While some clusters are well-defined, others, like Nature & Religious, Religious & Picnic, and Theatre & Shopping, are more difficult to distinguish. This suggests that K-Means might not be the best method for this dataset, as reflected by the high DBI values, low Silhouette Scores, and high inertia values, which point to considerable overlap between some clusters.

2. DBSCAN

K-Means and DBSCAN (Density-Based Spatial Clustering of Applications with Noise) operate quite differently. While K-Means focuses on partitioning data into clusters of similar sizes and shapes, DBSCAN is better at handling clusters of varying densities and irregular shapes. One of DBSCAN's key strengths is its ability to identify "noise" points that don't fit into any cluster, making it particularly useful for datasets with outliers or varying data distributions.

2.1 Silhouette Score (0.7133) Analysis

A Silhouette Score of 0.7133 shows that the clustering has greatly improved. Generally, a score closer to 1 means that the clusters are well-separated, and the points within each cluster are tightly grouped together. This higher score suggests that the clusters formed by DBSCAN are fairly distinct from one another, with strong cohesion within each cluster. While the score is quite good, it's not perfect, which means there could still be some slight overlap or room for further refinement in the clustering process.

2.2 Davies-Bouldin Index (0.3595) Analysis

A Davies-Bouldin Index of 0.3595 suggests a significant improvement in clustering quality. Since DBI values closer to zero indicate well-separated clusters, this lower score means the clusters are more distinct and separate from each other. The result of 0.3595 shows that DBSCAN has done a good job of creating unique, non-overlapping clusters with clear differentiation. While the score is promising and indicates strong performance, there might still be some minor overlap or shared characteristics between a few clusters.

2.3 Calinski-Harabasz Index (152.6959) Analysis

With a Calinski-Harabasz Index of 152.6959, the clustering shows stronger structure and definition. This higher CHI value suggests that the clusters are compact and well-separated, indicating a good level of clarity in the data division. When clusters are more densely packed with clear boundaries, as seen here, it means that DBSCAN has effectively grouped the data into meaningful segments. This aligns with the high Silhouette Score, reinforcing the idea that the clusters are distinct, cohesive, and well-formed.

The improvements in the clustering results are evident across all three metrics. With a Silhouette Score of 0.7133, the clusters are well-separated and tightly grouped, showing strong cohesion. The DBI score of 0.3595 further supports this by indicating that the clusters are clearly differentiated with minimal overlap. Additionally, the Calinski-Harabasz Index of 152.6959 highlights that the clusters are both compact and well-structured. Together, these results suggest that DBSCAN has performed well, creating distinct and well-separated clusters for this dataset.

Conclusion

In conclusion, K-Means provides clearer boundaries between clusters in our dataset, though both K-Means and DBSCAN have their strengths. While K-Means shows some variation in compactness, with some clusters being more spread out than others, it still does a good job of separating the clusters. Despite some overlapping in the Silhouette Score and Davies-Bouldin Index, the clusters remain distinguishable. On the other hand, DBSCAN doesn’t offer as clearly defined boundaries, but it excels at handling noise and identifying irregularly shaped clusters. The overlap in DBSCAN’s results, as shown by the Davies-Bouldin Index and Silhouette Score, suggests that while some clusters are well-separated and cohesive, others are not as distinct. In summary, DBSCAN offers more flexibility for managing different densities and noise, while K-Means excels in creating more clearly separated clusters.

Visualizations played a crucial role in helping us better understand user behavior and validate the unique characteristics of the clusters. The analysis revealed that both DBSCAN and K-Means were effective in grouping users based on their activity preferences. While DBSCAN uncovered more complex patterns and outliers, which can be valuable for offering personalized services, K-Means provided clear and easily interpretable clusters, ideal for broad marketing strategies. When combined, these insights can significantly boost customer satisfaction and engagement by enabling more targeted and customized service delivery. By leveraging these clustering methods, organizations can improve user satisfaction and loyalty, optimize resource allocation, and gain a deeper, more nuanced understanding of their audience.

References

1. Kaggle, (n.d.). *Buddymove Dataset*. Available at: https://www.kaggle.com/datasets/hemant2711/buddymove-holidayiq [Accessed 29 May 2024].
2. Kaggle, (n.d.). Analisa Clustering Kelompok 1. Available at: https://www.kaggle.com/code/muhammadrozy77/analisa-clustering-kelompok-1

3. Kaggle, (n.d.). Tugas Clustering Kelompok 1 AI. Available at: https://www.kaggle.com/code/rodhiyatunmardiah/tugas-clustering-kelompok-1-ai

4. Pandas Development Team. (2025). pandas.DataFrame.corr [online]. Available at: https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.corr.html

5. NumPy
   Oliphant, T.E., 2006. A guide to NumPy. Trelgol Publishing. Available at: https://numpy.org/doc/stable/

6. Pandas
   Pandas Development Team, 2025. pandas: powerful Python data analysis toolkit. Available at: https://pandas.pydata.org/pandas-docs/stable/

7. Scikit-learn (KMeans, silhouette_samples, silhouette_score, davies_bouldin_score, calinski_harabasz_score, DBSCAN)
   Pedregosa, F. et al., 2011. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, pp. 2825-2830. Available at: https://scikit-learn.org/stable/

8. Matplotlib
   Hunter, J.D., 2007. Matplotlib: A 2D Graphics Environment. Computing in Science & Engineering, 9(3), pp. 90-95. Available at: https://matplotlib.org/stable/

9. Seaborn
Waskom, M., 2021. Seaborn: statistical data visualization. Journal of Open Source Software, 6(60), p. 3021. Available at: https://seaborn.pydata.org/

10. Scikit-learn (StandardScaler)
Pedregosa, F. et al., 2011. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, pp. 2825-2830. Available at: https://scikit-learn.org/stable/