University
for the
Creative Arts

BERLIN SCHOOL OF
BUSINESS & INNOVATION

Essay / Assignment Title: User Behavior Analysis for Optimizing Engagement on Social Media Platforms

Programme title: Fundamentals of Data Analytics

Name: GUDIMETLLA BHARAT NAGENDRA REDDY (Q1076626)

Year: 2024-25

CONTENTS

Statement of compliance with academic ethics and the avoidance of plagiarism

I honestly declare that this dissertation is entirely my own work and none of its part has been copied from printed or electronic sources, translated from foreign sources and reproduced from essays of other researchers or students. Wherever I have been based on ideas or other people texts I clearly declare it through the good use of references following academic ethics.

(In the case that is proved that part of the essay does not constitute an original work, but a copy of an already published essay or from another source, the student will be expelled permanently from the postgraduate program).

Name and Surname (Capital letters):

GUDIMETLLA BHARAT NAGENDRA REDDY

Date: 2025/02/05

INTRODUCTION: Social Media: Breaking the Barriers of Mass Communication

In this 21st century Age where Digitization has pervaded almost every sphere of life, Social Media Platforms are rapidly becoming a crucial and indispensable in our day-to-day living. Social Media has become paramount for connecting individuals, facilitating communication, and enabling the exchange of information worldwide. Literally speaking, an explosion of information is being shared through online social networks. And that is what makes them such attractive sources of data, which can be used for social network research, and understanding user-behavior patterns to optimize engagement. (Abdesslem, Parris and Henderson, 2012). These platforms thrive on user-engagement and personalized content delivery, both of which are vital to adding, retaining users and improving overall user-satisfaction. Understanding user-behavior patterns is the key to achieving these objectives. By analyzing the activity data of users, Social Media platforms can uncover actionable insights that drive targeted recommendations, optimize features, and improve engagement strategies.

This task aims to analyze and categorize user- behavior patterns based on a dataset containing key metrics, including duration of daily usage, user age, geographic location, duration of daily usage, and total number of likes. The data set can be found on Kaggle Social Media Analysis. Through statistical analysis and machine learning techniques, the goal is to identify patterns, trends, and outliers that may influence user engagement. Additionally, this project seeks to define well-structured segments of users, thereby enabling more precise personalization of content and features.

Starting by cleaning and preparing the dataset to ensure quality, we can then fix any missing or incorrect data before we can explore it properly. When we apply statistical methods to the data, we generate statistical results, display how data values spread across a range, and discover abnormal data points to see how everything fits together. We then perform correlation analysis to reveal the link between main metrics, and understand which elements can affect user engagement. We organize users into specific groups using K-means and DBSCAN analysis to design focused engagement solutions that match their activities. We can also confirm that the models perform reliably as intended and adapt to increased data volumes for platform use. Social Media operators can use these findings to create better user experiences through data-based decisions about content delivery. Our method combines raw data with meaningful outcomes, to create a tailored and responsive Social Media setting.

CHAPTER ONE: Data Collection

Social Media Data Collection studies user behavior that detects patterns when people use these platforms.

Social Media networks track user interactions that take place on their service while they visit third-party sites and websites from advertisers. Social Media businesses need to collect precise interaction data so they can understand how users behave on their platforms. We need to track all interactions users have with platform features, while gathering complete data sets that include direct and indirect user-behavior patterns.

Platform operators study these measurements to discover user habits, which help improve their services and business plans. Such research focuses on tracking user behavior through site-visit frequency and duration, along with their activities on social media and their user engagement. This Study results from watching how people use social sites, thus enabling experts to develop more effective content delivery methods and interface upgrades (Benevenuto et al., 2009).

Specific Data Points to Collect

1. **Content Interaction Metrics**:
   - **Post Likes**: These indicate the popularity of content and the type of posts users prefer.
   - **Post Shares**: They reflect content that users find valuable enough to share withing their networks.
   - **Post Replies/Comments**: They highlight depth of engagement, showcasing topics users actively discuss or respond to.
   - **Native Video Views**: These measure consumption of video content, an increasingly dominant media type on social platforms.
2. **User Activity Metrics**
   - **Session Duration**: Tracks how long users remain active during a session.
   - **Login Frequency**: It provides insights into how often users access the platform.
   - **Time of Activity**: It Identifies peak activity hours and helps understand daily or weekly engagement patterns.

3. **Content Consumption Metrics**
   - **Click-Through Rates (CTR)**: Helps track how often users interact with links or advertisements.
   - **Saved Posts or Bookmarks**: This captures user interest in content for future reference.
   - **Browsing Patterns**: To map the sequence of interactions on the platform, such as transitions from viewing a video to liking a post.
4. **User Demographics and Preferences**:
   - **Age, Gender, and Location**: These parameters provide context to user behavior and helps in creating targeted content.
   - **Language Preferences**: This assists in delivering content in the user's preferred language.
   - **Interest Tags or Follows**: They indicate topics or accounts users are drawn to.
5. **Engagement with Recommendations**:
   - **Recommended Content Interaction**: Helps measure the effectiveness of personalized recommendations.
   - **Ad Engagement**: It tracks how users interact with sponsored content, providing insights into monetization strategies.
6. **Social Connections and Network Behavior**:
   - **Follower and Following Counts**: They together reflect a user’s network size and potential influence.
   - **Group Participation**: This indicates community engagement and related interests.
   - **Mentions and Tags**: Together they highlight social connections and collaborative activity.

Challenges in Data Collection

1. **Privacy Concerns and Regulations**:

- Researchers need to be aware of the Terms and Conditions when using social media data to avoid potential legal issues. Compliance with laws such as GDPR and CCPA requires platforms to ensure transparency in data collection, and obtain user consent. Privacy settings may also limit access to certain data. (Ahmed, Bath and Demartini, 2017).

2. **Data Accuracy and Completeness**:
    - Users often present curated or exaggerated versions of themselves, which leads to discrepancies between online behavior and real-world preferences. Additionally, missing, or incomplete data can skew analysis.
3. **Platform Limitations**:
    - Technical constraints, such as server downtimes or API rate limits, may impact the continuous and accurate collection of user activity data. Most platforms limit access to user data (Mayr and Weller, 2017).
4. **Bots and Fake Accounts**:
    - The presence of automated or fake accounts can distort engagement metrics, creating noise in the dataset.
5. **Contextual Ambiguity**:
    - Quantitative metrics like, likes and shares may not be able to fully capture the qualitative aspects of user intent or sentiment, adding a requirement of supplementary qualitative analysis.
6. **Behavioral Biases**:
    - Users tend to engage differently based on external factors, such as trending events or seasonal changes, complicating long-term behavioral predictions. In many cases there is bias in social media populations because it is hard to know exact user details. The different approaches used for the data collection induce biases because users choose what they wish to share including location tracking.

By addressing these challenges and collecting robust data points, social media platforms can build a wide array of understanding user behavior. This enables data-driven personalization, improved user satisfaction, and optimized delivery of content strategies.

CHAPTER TWO: Pre-processing Data Techniques

Data Pre-Processing helps us make important findings and decisions by improving the quality of user-data, before further analysis, on Social Media platforms.

Visualizations represent complex information as clear shapes and visuals, which help users make sense of it all. We can thus focus on transparent information delivery through data exploration, analysis, illustration, discovery, and communication. Line Charts and Bar Graphs show us clearly how user activity changes through time in our analysis.

Visualization of Data by means of Line Charts and Bar Graphs.

Line Charts help us observe how activities shift throughout time, when we need to track user numbers and behavior metrics. These visual representations reveal when user activity is highest and lowest, and detect brief increases or decreases, while in participation by users.

Bar Graphs, on the other hand, prove their value by identifying patterns among user groups by displaying data about their activities across different age segments and types of content.

Visual elements thus enhance data comprehension, by letting users quickly recognize meaningful trends based on what they see.

Outliers and missing values appear commonly in data collection processes, affecting both result reliability and accuracy of analysis. Dealing with broken or incomplete records of user activity serves as an essential part of pre-processing work. Data appears as missing when systems fail, or users stop engaging if privacy rules block access. We need to analyze how frequently data is missing if such gaps follow specific patterns in our data. When handling missing numerical data, we employ two kinds of imputation approaches: Basic methods that replace gaps with average statistics and more advanced techniques like K-nearest neighbors and Regression-based imputation methods (Kwak and Kim, 2017).

Categorical data gaps can be resolved by using the mode value or creating an additional unknown field. We need to treat outliers with special care whenever we find them in boxplots or scatterplots. Our evaluation handles data with actual behavior, much the same way as we handle mistakes by keeping or removing data points based on their characteristics. Researchers use trimming methods to handle outlier cases in their analysis.

Data processing with visualization tools and thorough clean-up procedures makes the dataset ready, for precise analysis that produces insights we can use effectively.

Results

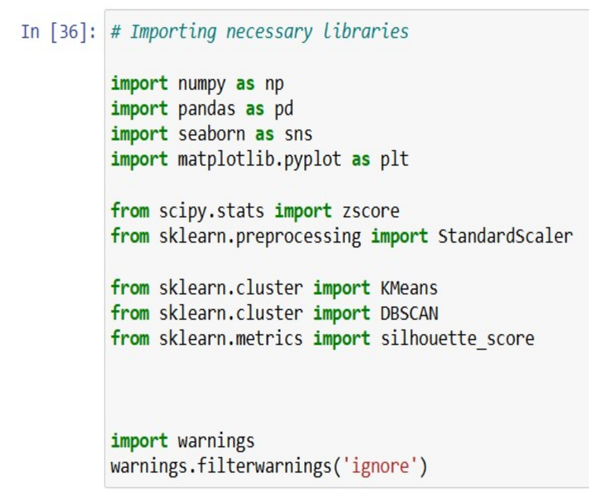
The dataset, with records of user activity data, reveals several important insights into user engagement patterns. The descriptive statistics show that the average Usage Duration is 4 hours per day, with values ranging from 1 hour to 13 hours. Age has a mean of 33.78 years, with users spanning from 18 to 60 years old. Total Likes per day has a mean of 5.32, with the minimum likes being 0 and the maximum being 28 likes. These figures provide a broad view of the typical user on the platform.

Visualizations indicate patterns in the data, like the fact that younger people use social media more than older ones. Young users spend more time on Social Media than old people based on the correlation of age and usage duration.

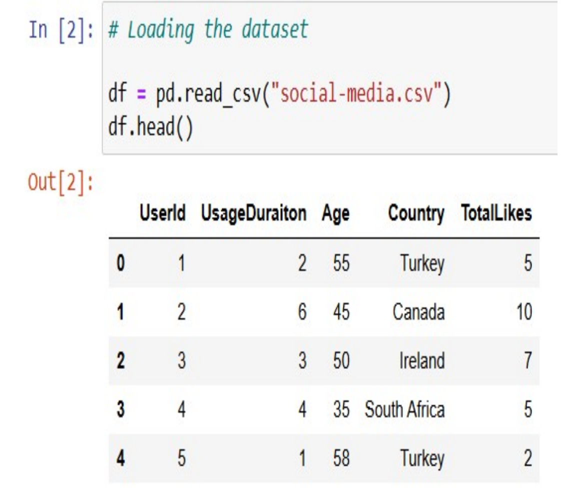
There are no missing values in this dataset. Upon conducting outlier detection using the Z-score method, we found a few outliers in UsageDuration (value 13) and TotalLikes (values 25, 25, and 28). These outliers suggest that certain users are more active than others, either in terms of usage or engagement with content.

The correlation analysis revealed a strong negative correlation between UsageDuration and Age (-0.64), indicating that younger users tend to spend more time on the platform. Additionally, there is a strong positive correlation between UsageDuration and TotalLikes (0.69), suggesting that users who engage more with the platform tend to like more posts.

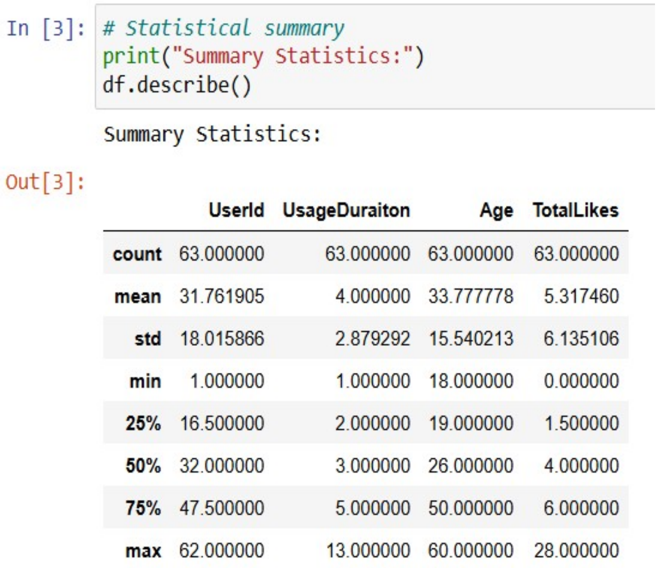
The clustering performance was evaluated using the Silhouette Score, where K-means achieved a score of 0.58, while DBSCAN scored 0.47. These values indicate that K-means performed slightly better in creating meaningful clusters based on user activity.



```
In [36]: # Importing necessary libraries

         import numpy as np
         import pandas as pd
         import seaborn as sns
         import matplotlib.pyplot as plt

         from scipy.stats import zscore
         from sklearn.preprocessing import StandardScaler

         from sklearn.cluster import KMeans
         from sklearn.cluster import DBSCAN
         from sklearn.metrics import silhouette_score



         import warnings
         warnings.filterwarnings('ignore')
```

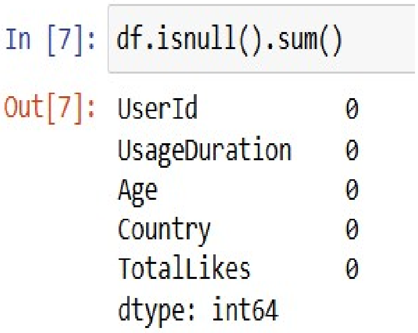


```
In [2]: # Loading the dataset

        df = pd.read_csv("social-media.csv")
        df.head()

Out[2]:
     UserId  UsageDuraiton  Age       Country  TotalLikes
  0       1              2   55        Turkey           5
  1       2              6   45        Canada          10
  2       3              3   50       Ireland           7
  3       4              4   35  South Africa           5
  4       5              1   58        Turkey           2
```

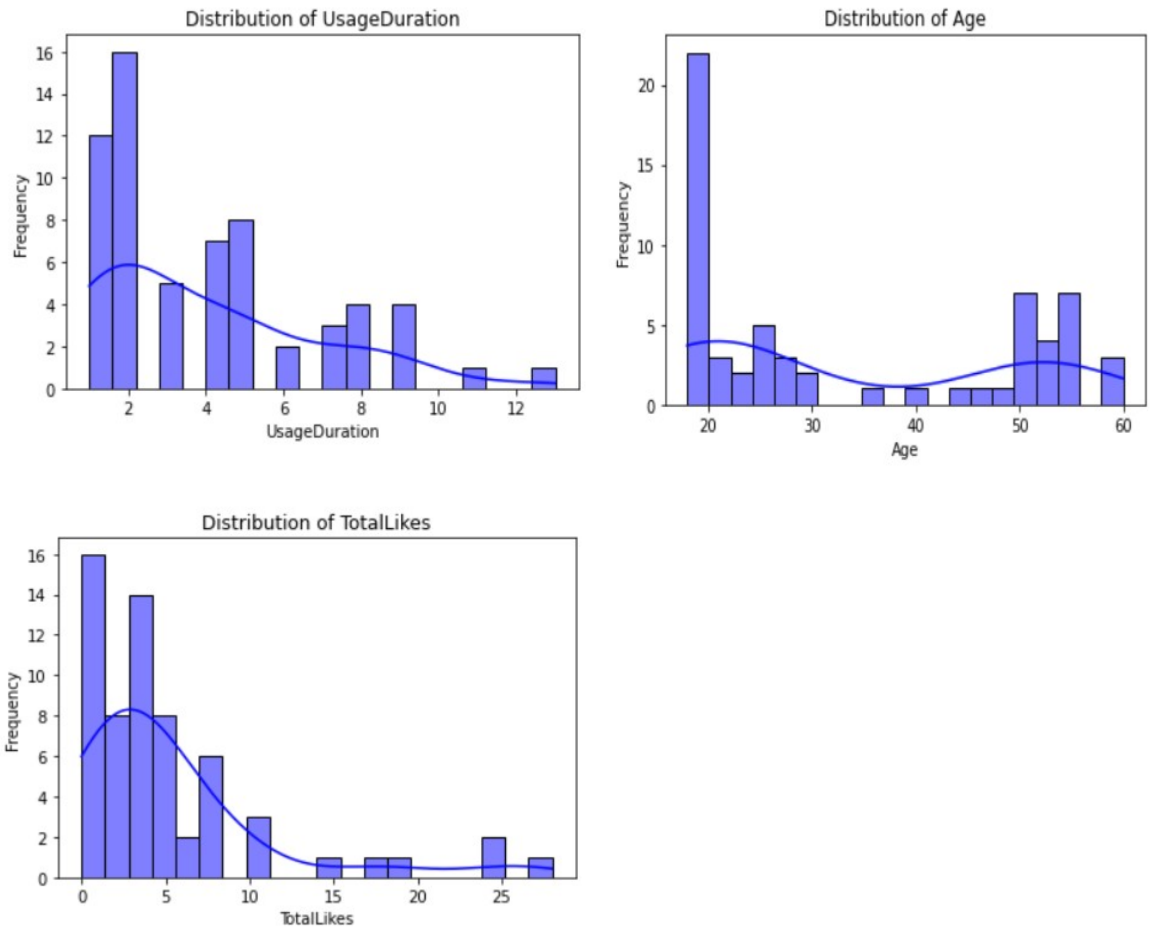


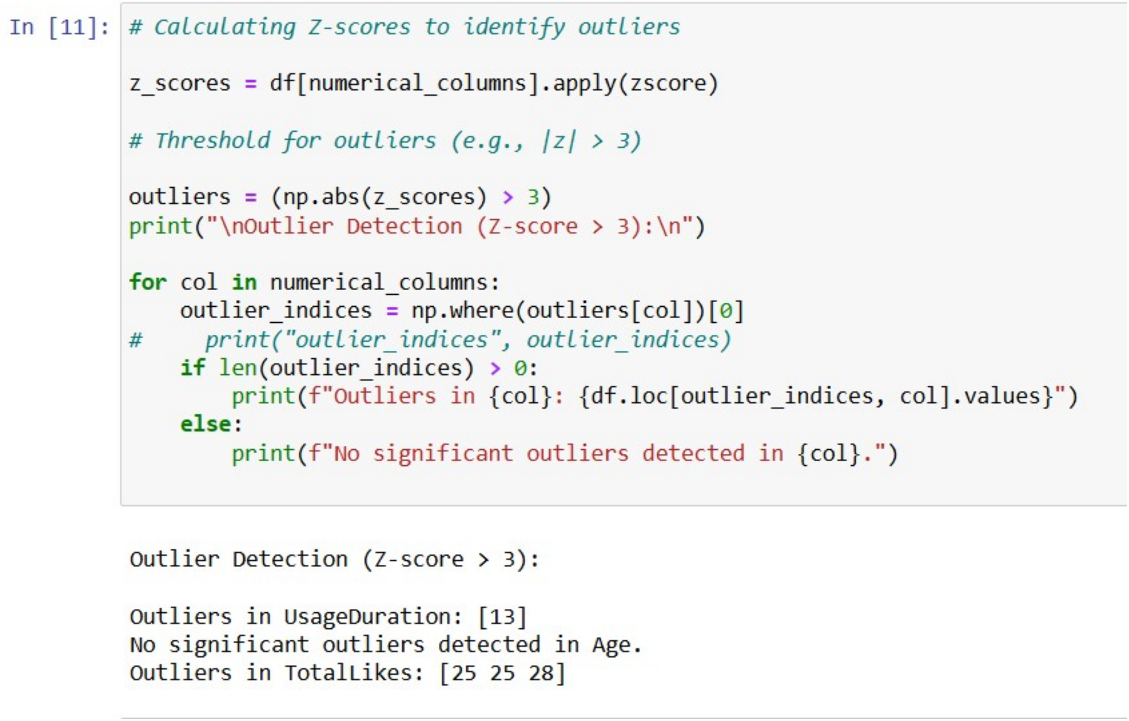
```
In [3]: # Statistical summary
        print("Summary Statistics:")
        df.describe()

        Summary Statistics:

Out[3]:
```

	UserId	UsageDuraiton	Age	TotalLikes
count	63.000000	63.000000	63.000000	63.000000
mean	31.761905	4.000000	33.777778	5.317460
std	18.015866	2.879292	15.540213	6.135106
min	1.000000	1.000000	18.000000	0.000000
25%	16.500000	2.000000	19.000000	1.500000
50%	32.000000	3.000000	26.000000	4.000000
75%	47.500000	5.000000	50.000000	6.000000
max	62.000000	13.000000	60.000000	28.000000



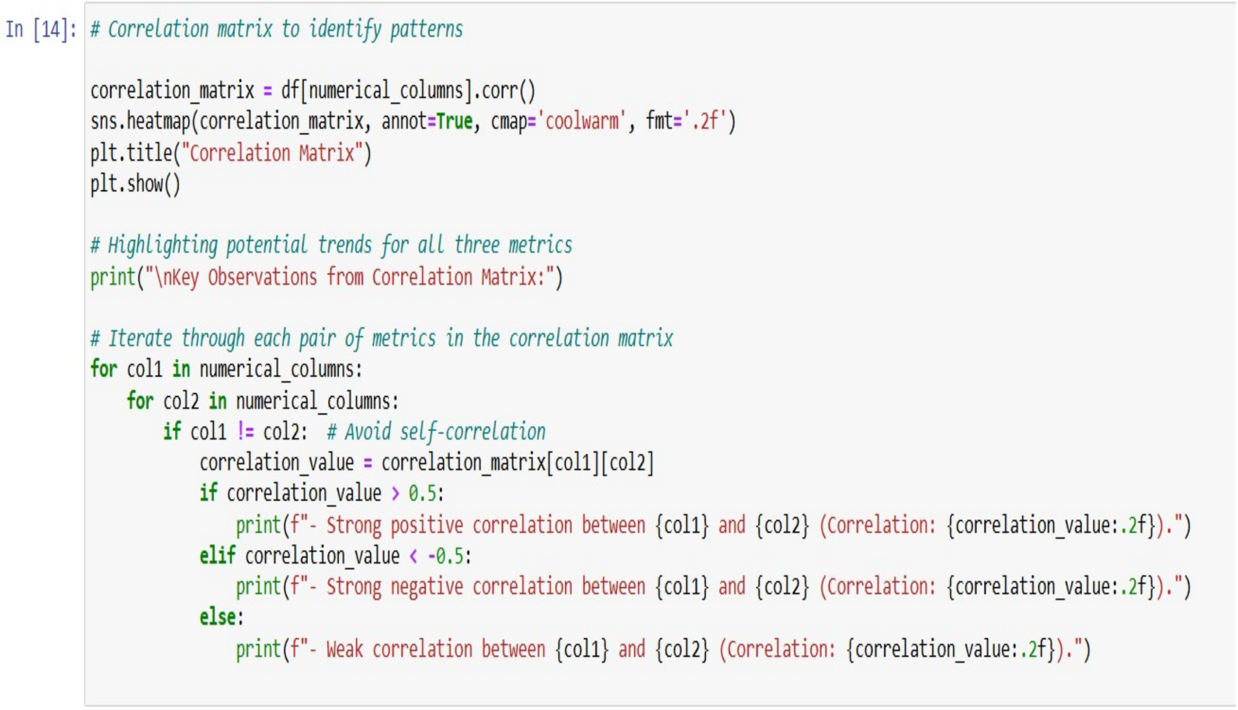
```
In [7]: df.isnull().sum()

Out[7]: UserId           0
        UsageDuration    0
        Age              0
        Country          0
        TotalLikes       0
        dtype: int64
```



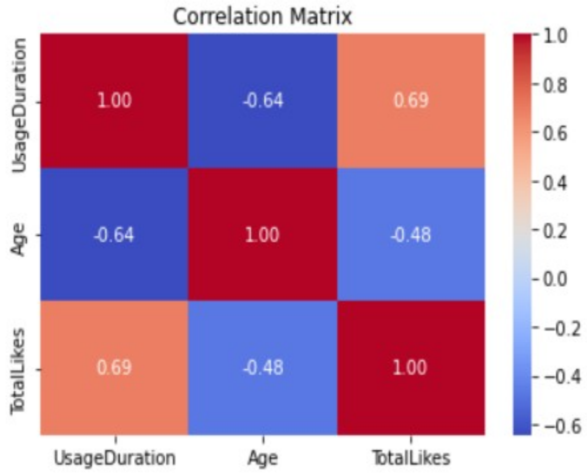


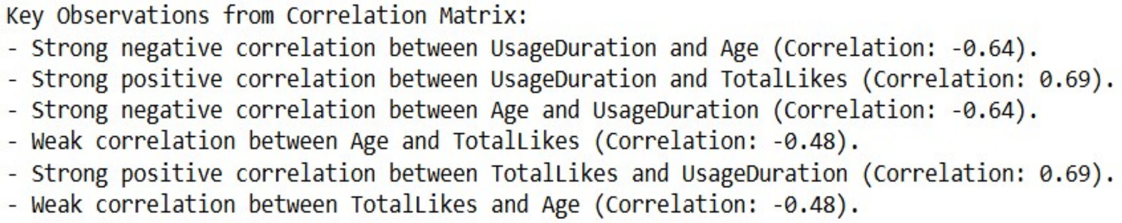
```
In [11]: # Calculating Z-scores to identify outliers

         z_scores = df[numerical_columns].apply(zscore)

         # Threshold for outliers (e.g., |z| > 3)

         outliers = (np.abs(z_scores) > 3)
         print("\nOutlier Detection (Z-score > 3):\n")

         for col in numerical_columns:
             outlier_indices = np.where(outliers[col])[0]
         #     print("outlier_indices", outlier_indices)
             if len(outlier_indices) > 0:
                 print(f"Outliers in {col}: {df.loc[outlier_indices, col].values}")
             else:
                 print(f"No significant outliers detected in {col}.")
```

```
Outlier Detection (Z-score > 3):

Outliers in UsageDuration: [13]
No significant outliers detected in Age.
Outliers in TotalLikes: [25 25 28]
```

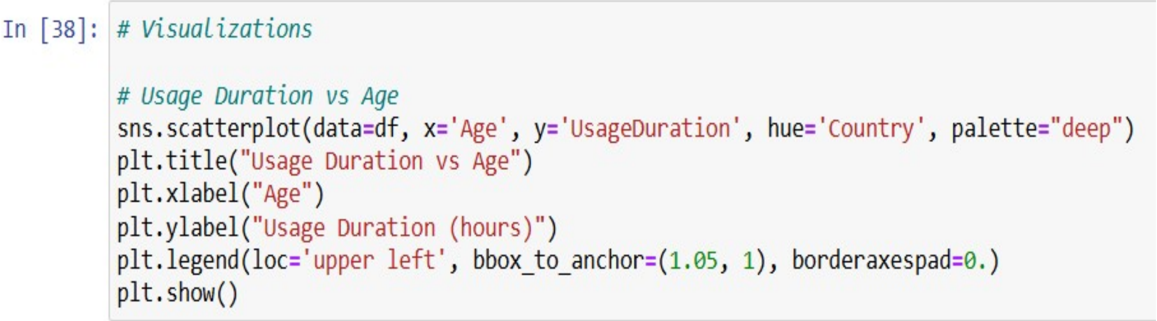


```
In [14]: # Correlation matrix to identify patterns

         correlation_matrix = df[numerical_columns].corr()
         sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt='.2f')
         plt.title("Correlation Matrix")
         plt.show()

         # Highlighting potential trends for all three metrics
         print("\nKey Observations from Correlation Matrix:")

         # Iterate through each pair of metrics in the correlation matrix
         for col1 in numerical_columns:
             for col2 in numerical_columns:
                 if col1 != col2:  # Avoid self-correlation
                     correlation_value = correlation_matrix[col1][col2]
                     if correlation_value > 0.5:
                         print(f"- Strong positive correlation between {col1} and {col2} (Correlation: {correlation_value:.2f}).")
                     elif correlation_value < -0.5:
                         print(f"- Strong negative correlation between {col1} and {col2} (Correlation: {correlation_value:.2f}).")
                     else:
                         print(f"- Weak correlation between {col1} and {col2} (Correlation: {correlation_value:.2f}).")
```

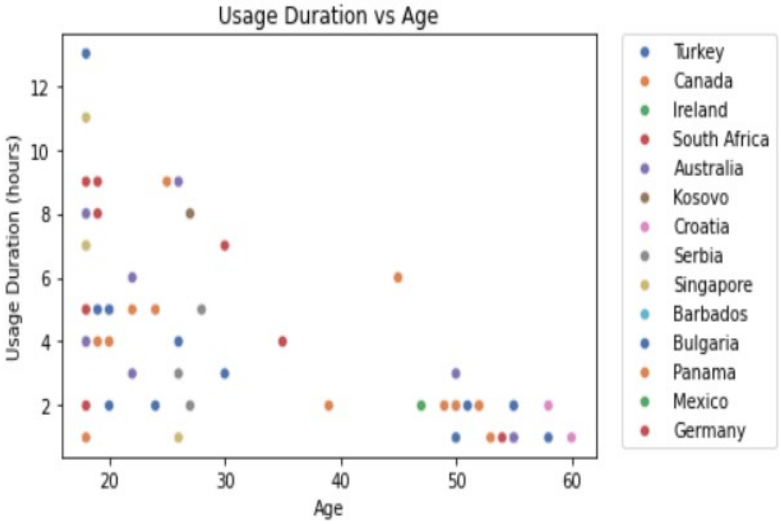




```
Key Observations from Correlation Matrix:
- Strong negative correlation between UsageDuration and Age (Correlation: -0.64).
- Strong positive correlation between UsageDuration and TotalLikes (Correlation: 0.69).
- Strong negative correlation between Age and UsageDuration (Correlation: -0.64).
- Weak correlation between Age and TotalLikes (Correlation: -0.48).
- Strong positive correlation between TotalLikes and UsageDuration (Correlation: 0.69).
- Weak correlation between TotalLikes and Age (Correlation: -0.48).
```



```
In [38]: # Visualizations

         # Usage Duration vs Age
         sns.scatterplot(data=df, x='Age', y='UsageDuration', hue='Country', palette="deep")
         plt.title("Usage Duration vs Age")
         plt.xlabel("Age")
         plt.ylabel("Usage Duration (hours)")
         plt.legend(loc='upper left', bbox_to_anchor=(1.05, 1), borderaxespad=0.)
         plt.show()
```



CHAPTER THREE: User Engagement Problems with Social Media Platforms

User Engagement refers to the quality of the user experience that highlights the positive aspects of interacting with Social Media on online platforms. In particular with a person's tendency to use the application for extended periods on a recurring basis. This is a critical concept in designing online applications for Desktop, Tablet, or Mobile Devices, driven by the realization that successful applications foster active engagement, rather than mere usage. Users dedicate their time, emotions and focus on technology, whose singular aim is to fulfill both practical and enjoyable entertainment needs. Evaluating user engagement through measurement is essential to gauge an application's prowess to captivate users. This factor helps conceive, guide, determine, and drive to execute its design and development. As a concept, User Engagement is multi-dimensional and inherently complex (Lalmas, O'Brien, H. and Yom-Tov, 2014). Identifying user engagement patterns is more often considered a clustering problem rather than a classification problem, due to the nature of its gathering of data and the objective of the analysis.

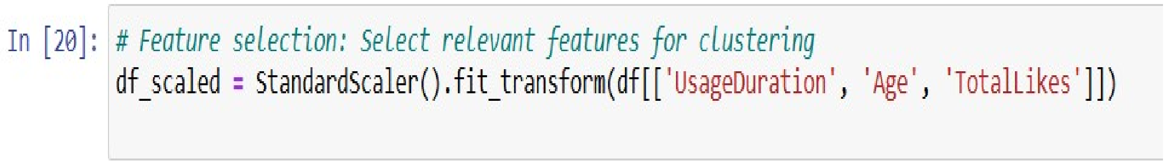
In clustering, the aim is to group users based on similarities in their behavior, without any predefined labels. Social media users' activity varies widely, and patterns emerge naturally from the data. These behaviors, such as time spent on the platform, frequency of engagement (likes, comments, shares), and content preferences, do not necessarily belong to predefined categories. Instead, users form diverse, overlapping groups that can be discerned through unsupervised learning techniques.

On the other hand, classification problems require predefined labels, and the objective is to assign users to these categories based on their features. Since user engagement patterns on Social Media are complex and dynamic, it is a complicated task to define them with fixed classes or labels. Users do not easily fit into a rigid classification system, which makes clustering a more appropriate approach for identifying user segments. Therefore, techniques like K-means and DBSCAN enable the discovery of natural groupings, allowing the platform to explore trends and behaviors without needing to impose artificial classifications.

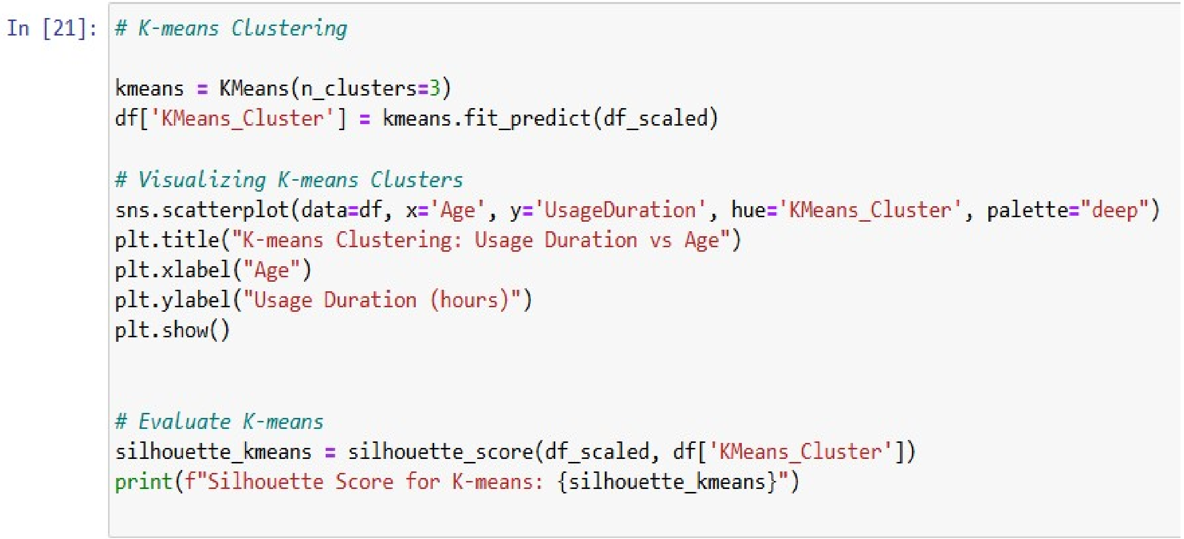
Well-organized user activity segments contribute significant advantages to Social Media platforms. As such, Platforms can then deliver better content by knowing their different user groups to create material that matches what they like and with which they wish to interact. Therefore, Marketers who target user segments deliver better ads that boost results and create content their users truly want to see. Platform operators improve user engagement when they study how each user segment interacts with platform elements, to develop personalized features that keep users actively engaged. Social platforms can then reorganize their resources by tracking what type of users interact how often with the platform. They can then provide direct support and processing power to these engaged and valuable groups (Benevenuto et al., 2009). By analyzing usage patterns of different user groups across various Social Media platforms, we can develop focused measures to encourage inactive users to stay active more often.

CHAPTER Four: Selection of Two Algorithms

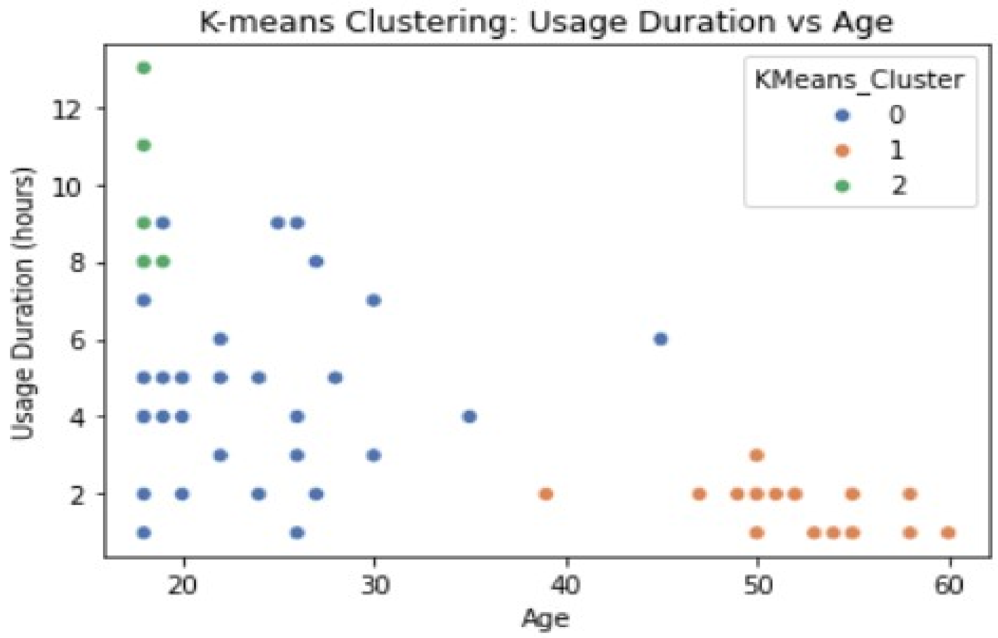
Our chosen Algorithms for clustering tasks are KMeans and DBSCAN. The KMeans algorithm stands tall as the preferred choice for data clustering applications. It splits input data into K separate clusters that someone selects before running (K). The algorithm starts with K random starting points, and keeps moving these centers to minimize how far data points are from their respective clusters. KMeans rapidly processes large datasets while efficiently handling Social Media user-data. Users must set the exact number of clusters at the start, but the method fails to handle irregular shapes and unevenly distributed data points (Ahmed, Seraj and Islam 2020). This algorithm needs careful initial centroid placement because poor starting points can lead to bad cluster results. DBSCAN tools cluster data points based on their points' density to form distinct groupings. Research shows that most tasks in clustering prefer the density-based method, because of its fast performance and easy implementation (Deng, 2020). Users who choose DBSCAN do not need to enter the number of clusters as input. DBSCAN detects clusters by identifying areas with high point density, where low-density spaces separate these dense regions. User-engagement analysis benefits from DBSCAN because it finds outliers, and spots rare user behaviors that don't match typical patterns. DBSCAN performs best when users select appropriate values for distance measurements and cluster formation requirements. The method faces two main challenges:
(a) finding good cluster divisions for different density levels in datasets and
(b) performing complex calculations as it processes. The method explained by Khan et al. (2014) experiences limitations in analyzing data where density differences are difficult to distinguish.



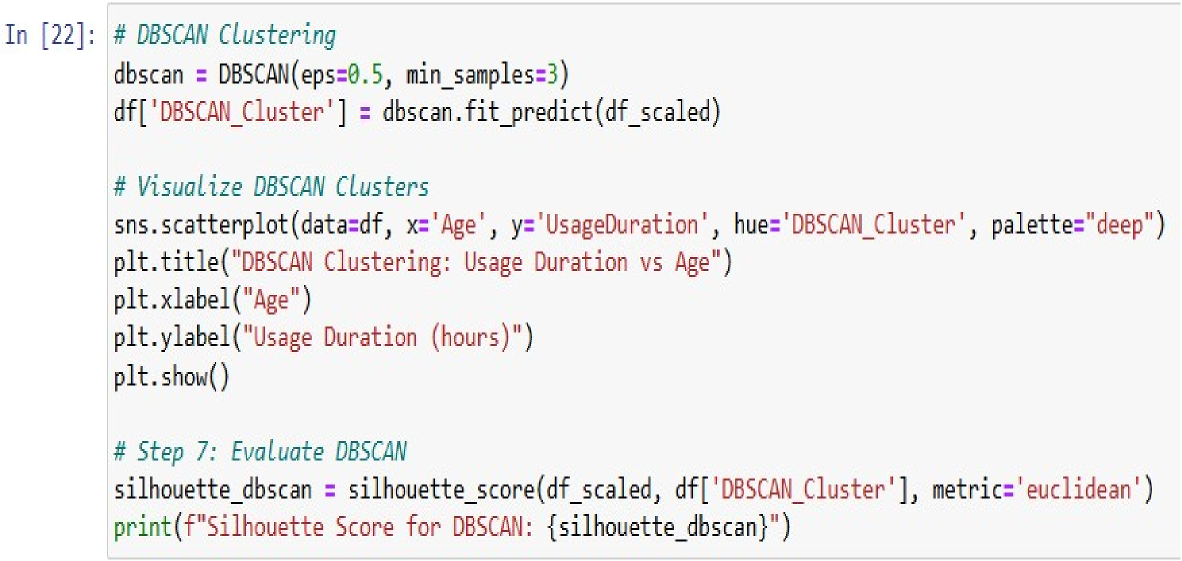
```
In [20]: # Feature selection: Select relevant features for clustering
         df_scaled = StandardScaler().fit_transform(df[['UsageDuration', 'Age', 'TotalLikes']])
```



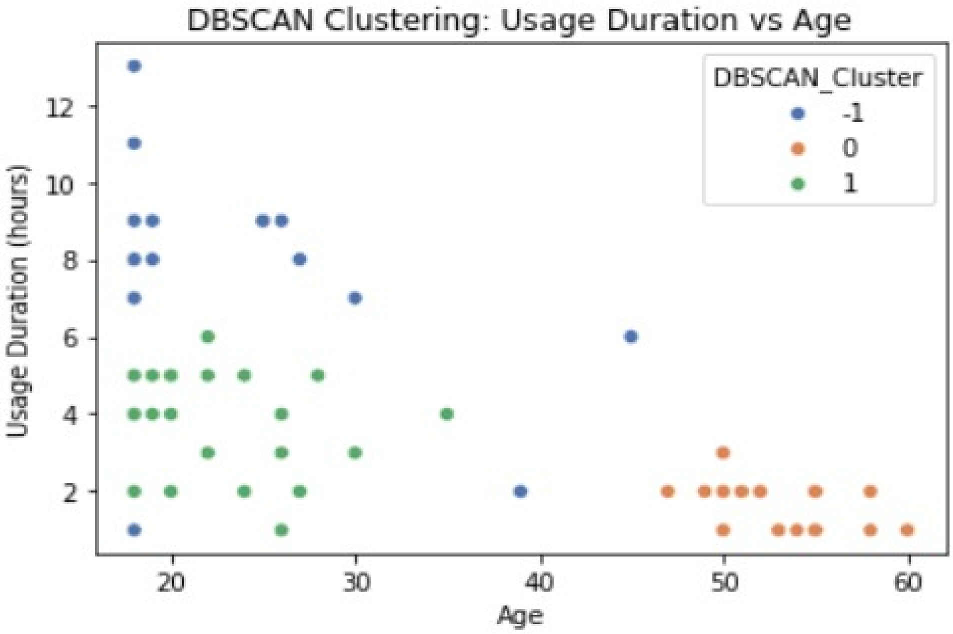
```
In [21]: # K-means Clustering

         kmeans = KMeans(n_clusters=3)
         df['KMeans_Cluster'] = kmeans.fit_predict(df_scaled)

         # Visualizing K-means Clusters
         sns.scatterplot(data=df, x='Age', y='UsageDuration', hue='KMeans_Cluster', palette="deep")
         plt.title("K-means Clustering: Usage Duration vs Age")
         plt.xlabel("Age")
         plt.ylabel("Usage Duration (hours)")
         plt.show()


         # Evaluate K-means
         silhouette_kmeans = silhouette_score(df_scaled, df['KMeans_Cluster'])
         print(f"Silhouette Score for K-means: {silhouette_kmeans}")
```

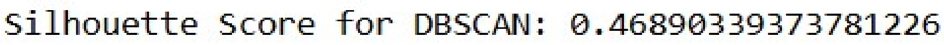


```
Silhouette Score for K-means: 0.5819035656047385
```



```
In [22]: # DBSCAN Clustering
         dbscan = DBSCAN(eps=0.5, min_samples=3)
         df['DBSCAN_Cluster'] = dbscan.fit_predict(df_scaled)

         # Visualize DBSCAN Clusters
         sns.scatterplot(data=df, x='Age', y='UsageDuration', hue='DBSCAN_Cluster', palette="deep")
         plt.title("DBSCAN Clustering: Usage Duration vs Age")
         plt.xlabel("Age")
         plt.ylabel("Usage Duration (hours)")
         plt.show()

         # Step 7: Evaluate DBSCAN
         silhouette_dbscan = silhouette_score(df_scaled, df['DBSCAN_Cluster'], metric='euclidean')
         print(f"Silhouette Score for DBSCAN: {silhouette_dbscan}")
```





```
Silhouette Score for DBSCAN: 0.46890339373781226
```

CHAPTER Five: Evaluating a specific Model

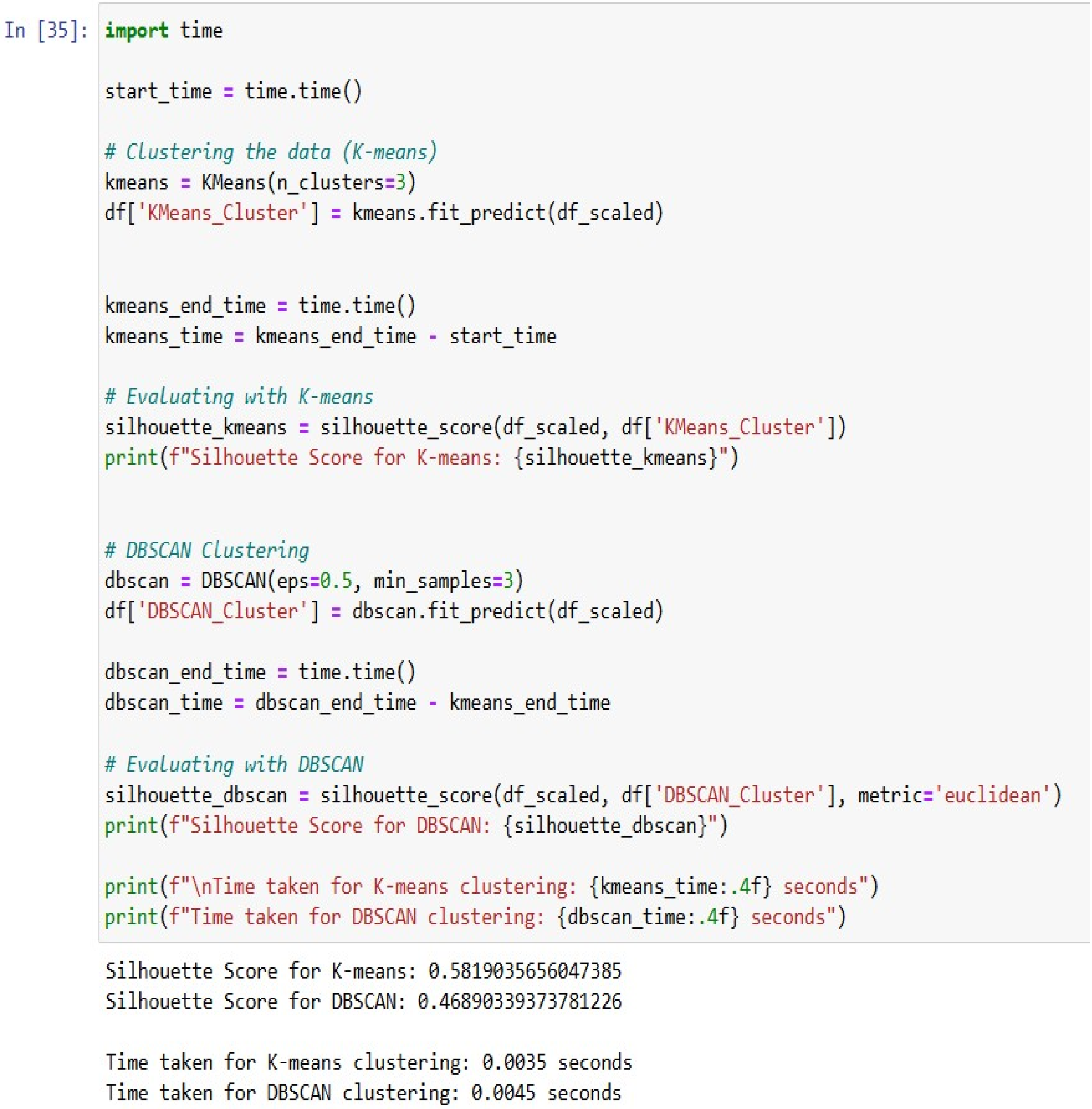
When evaluating clustering models such as K-means and DBSCAN, it is important to bear in mind that several factors such as accuracy, robustness, speed, interpretability, and scalability are essential. Clustering is unsupervised and the results are dependent on the nature of data, problem requirements, and the algorithm used (Kanavos, Karamitsos and Mohasseb, 2023).

The Silhouette Score provides a measure of cluster cohesion and separation. K-means achieved a higher Silhouette Score (0.58), indicating that its clusters are more compact and well-separated, when compared to DBSCAN (0.47). This suggests that K-means generated more meaningful clusters in this dataset.

However, DBSCAN's score is still quite reasonable, indicating that its density-based approach captured some distinct patterns, albeit with more noise or less well-defined clusters. On the other hand, K-means is sensitive to initial centroid placement and might struggle with non-spherical or varied-density clusters. While DBSCAN is more robust to noise and can handle irregular shapes, its performance is highly dependent on the choice of parameters (e.g., epsilon and min_samples).

The lower Silhouette score for DBSCAN reflects some challenges in handling the dataset's structure. Both algorithms performed relatively fast, with K-means taking 0.0035 seconds, and DBSCAN taking 0.0045 seconds. This indicates that for this dataset, both models are efficient and scalable, though K-means showed a slightly faster execution time.

K-means is highly interpretable, as it assigns each data point to one of k predefined clusters. The grouping in the results shows all points together for K-Means even if both algorithms had 3 clusters. DBSCAN's interpretability can be challenging, since it identifies noise and does not force all points into clusters. However, it can detect outliers, which K-means cannot do. K-means is generally more scalable to larger datasets due to its simplicity, while DBSCAN may struggle as dataset size increases due to its need to calculate pair-wise distances. Nevertheless, both algorithms performed efficiently on this smaller dataset.



In [35]:

```
import time

start_time = time.time()

# Clustering the data (K-means)
kmeans = KMeans(n_clusters=3)
df['KMeans_Cluster'] = kmeans.fit_predict(df_scaled)


kmeans_end_time = time.time()
kmeans_time = kmeans_end_time - start_time

# Evaluating with K-means
silhouette_kmeans = silhouette_score(df_scaled, df['KMeans_Cluster'])
print(f"Silhouette Score for K-means: {silhouette_kmeans}")


# DBSCAN Clustering
dbscan = DBSCAN(eps=0.5, min_samples=3)
df['DBSCAN_Cluster'] = dbscan.fit_predict(df_scaled)

dbscan_end_time = time.time()
dbscan_time = dbscan_end_time - kmeans_end_time

# Evaluating with DBSCAN
silhouette_dbscan = silhouette_score(df_scaled, df['DBSCAN_Cluster'], metric='euclidean')
print(f"Silhouette Score for DBSCAN: {silhouette_dbscan}")

print(f"\nTime taken for K-means clustering: {kmeans_time:.4f} seconds")
print(f"Time taken for DBSCAN clustering: {dbscan_time:.4f} seconds")
```

```
Silhouette Score for K-means: 0.5819035656047385
Silhouette Score for DBSCAN: 0.46890339373781226

Time taken for K-means clustering: 0.0035 seconds
Time taken for DBSCAN clustering: 0.0045 seconds
```

CONCLUDING REMARKS

This Assignment has endeavoured to explore clustering techniques to analyze and categorize User-engagement patterns on Social Media platforms.

1. By leveraging K-means and DBSCAN, one was able to identify distinct user-activity segments, resulting in valuable insights into user behavior.

2. The higher Silhouette Score for K-means (0.58) indicated that it produced more cohesive and well-separated clusters, making it an effective tool for segmenting users based on engagement.

3. DBSCAN, while also useful for identifying dense regions and outliers, showed a lower score (0.47), reflecting some challenges in clustering the data.

4. Both algorithms demonstrated impressive speed, with minimal execution times of 0.0035 and 0.0045 seconds, making them efficient for larger datasets.

5. Ultimately, the insights gained through clustering can assist in personalizing content, optimizing marketing efforts, and improving user experience on the platform.

6. Future work could involve fine-tuning parameters and testing additional clustering algorithms for better segmentation accuracy.

BIBLIOGRAPHY

Abdesslem, F.B., Parris, I. and Henderson, T., 2012. Reliable online social network data collection. *Computational social networks: Mining and visualization*, pp.183-210.

Ahmed, M., Seraj, R. and Islam, S.M.S., 2020. The k-means algorithm: A comprehensive survey and performance evaluation. *Electronics*, *9*(8), p.1295.

Ahmed, W., Bath, P.A. and Demartini, G., 2017. Using Twitter as a data source: An overview of ethical, legal, and methodological challenges. *The ethics of online research*, *2*, pp.79-107.

Benevenuto, F., Rodrigues, T., Cha, M. and Almeida, V., 2009, November. Characterizing user behavior in online social networks. In *Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement* (pp. 49-62).

Deng, D., 2020, September. DBSCAN clustering algorithm based on density. In *2020 7th international forum on electrical engineering and automation (IFEEA)* (pp. 949-953). IEEE.

Kanavos, A., Karamitsos, I. and Mohasseb, A., 2023. Exploring clustering techniques for analyzing user engagement patterns in Twitter data. *Computers*, *12*, p.124.

Khan, K., Rehman, S.U., Aziz, K., Fong, S. and Sarasvady, S., 2014, February. DBSCAN: Past, present and future. In *The fifth international conference on the applications of digital information and web technologies (ICADIWT 2014)* (pp. 232-238). IEEE.

Khan, M. and Khan, S.S., 2011. Data and information visualization methods, and interactive mechanisms: A survey. *International Journal of Computer Applications*, *34*(1), pp.1-14.

Kwak, S.K. and Kim, J.H., 2017. Statistical data preparation: management of missing values and outliers. *Korean journal of anesthesiology*, *70*(4), pp.407-411.

Lalmas, M., O'Brien, H. and Yom-Tov, E., 2014. *Measuring user engagement*.

Mayr, P. and Weller, K., 2017. Think before you collect: Setting up a data collection approach for social media studies. *The SAGE handbook of social media research methods*.