Assessment Report

on

“Internet Usage Clustering

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY DEGREE

SESSION 2024-25

in

CSE (AI)

By

Name : Raamanjal Singh Gangwar

Roll Number : 202401100300187

Section: C

Under the supervision of

“MAYANK LAKHOTIA”

KIET Group of Institutions, Ghaziabad

April, 2025

Introduction

In the modern digital era, internet usage has become an integral part of everyday life. People use the internet for various purposes including education, entertainment, communication, and work. However, users differ widely in terms of how much time they spend online, the types of websites they visit, and how frequently they access the internet. Analyzing and understanding these usage patterns can provide valuable insights for improving user experience, optimizing network resources, and designing targeted services.

Manual analysis of user behavior on a large scale is inefficient and error-prone. This is where **machine learning**, particularly **unsupervised learning**, plays a key role. By leveraging clustering algorithms like **K-Means**, we can group users based on their usage patterns without needing any labelled data. These clusters can represent distinct user profiles such as casual users, moderate users, and heavy users, each with their own characteristics.

In this project, we analyze a dataset containing user-level internet usage data and perform clustering based on:

- **Daily usage time (in hours)**
- **Number of different site categories visited**
- **Number of browsing sessions per day**

Objectives

- To analyze internet usage data of users using data science techniques.
- To apply **feature scaling** for accurate clustering.
- To use the **Elbow Method** to determine the optimal number of user clusters.
- To apply **K-Means clustering** to group users with similar online behavior.

Methodology

The methodology followed in this project consists of several well-defined steps, designed to extract meaningful clusters from internet usage data. Here's a breakdown of each step:

1. Data Loading and Exploration

- The dataset (internet_usage.csv) containing user information was loaded using the pandas library.
- The features include:
    - daily_usage_hours: Number of hours spent online per day
    - site_categories_visited: Count of different categories of websites accessed
    - sessions_per_day: Number of browsing sessions per day

2. Data Preprocessing

- Since the features are on different scales, **standardization** was applied using Standard Scaler from scikit-learn to ensure that each feature contributes equally to the clustering process.
- Standardization transforms the data so that it has a mean of 0 and a standard deviation of 1.

3. Determining Optimal Number of Clusters

- To decide how many clusters (user groups) should be formed, the **Elbow Method** was used.
- K-Means was run for values of k from 1 to 10, and the **inertia** (sum of squared distances to the nearest cluster center) was plotted.
- The "elbow point" in the plot (where inertia starts to decrease more slowly) was chosen as the optimal number of clusters.

4. Clustering with K-Means

- The KMeans algorithm from scikit-learn was applied with the optimal number of clusters.
- Each user was assigned a cluster label based on their internet usage pattern.

5. Visualization

- A **pair plot** using seaborn was created to visualize the distribution of users across clusters based on the three features.
- Cluster centers and behavior were analyzed by calculating the **mean values** of each feature per cluster.

Code

```
# Importing essential libraries
import pandas as pd  # For handling tabular data
import matplotlib.pyplot as plt  # For data visualization
import seaborn as sns  # For statistical graphics
from sklearn.cluster import KMeans  # Clustering algorithm
from sklearn.preprocessing import StandardScaler  # Normalizes features
```

```
# Loading the dataset
df = pd.read_csv('/content/internet_usage.csv')

# Checking data info and summary
print("Data Info:")
print(df.info())
print("\nSummary Stats:")
print(df.describe())
```

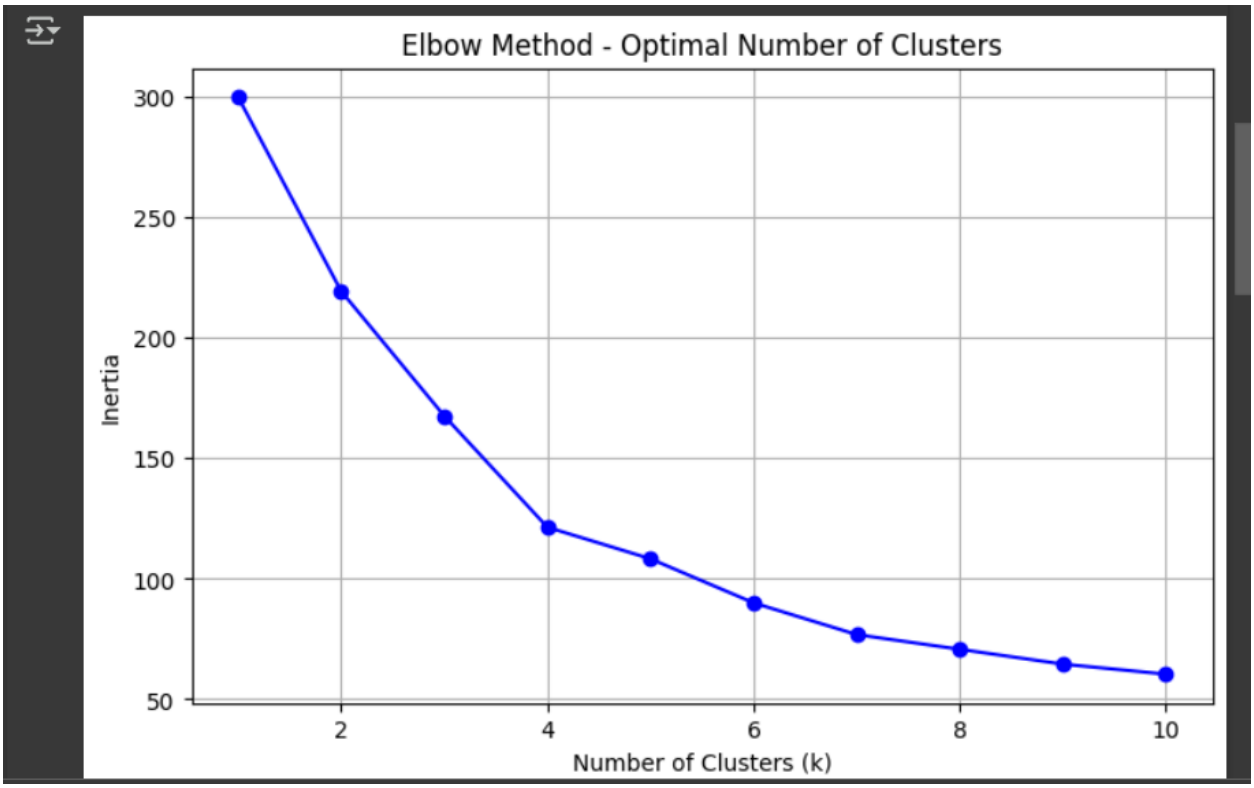
```
# Feature Scaling (Normalization)
# KMeans performs better when features are scaled (0 mean, unit variance)
scaler = StandardScaler()
scaled_data = scaler.fit_transform(df)
# Using Elbow Method to choose optimal number of clusters (k)
inertia = []

for k in range(1, 11):
    km = KMeans(n_clusters=k, random_state=42)
    km.fit(scaled_data)
    inertia.append(km.inertia_)

# Plotting the elbow curve
plt.figure(figsize=(8, 5))
plt.plot(range(1, 11), inertia, marker='o', linestyle='-', color='blue')
plt.title("Elbow Method - Optimal Number of Clusters")
plt.xlabel("Number of Clusters (k)")
plt.ylabel("Inertia")
plt.grid(True)
plt.show()
```

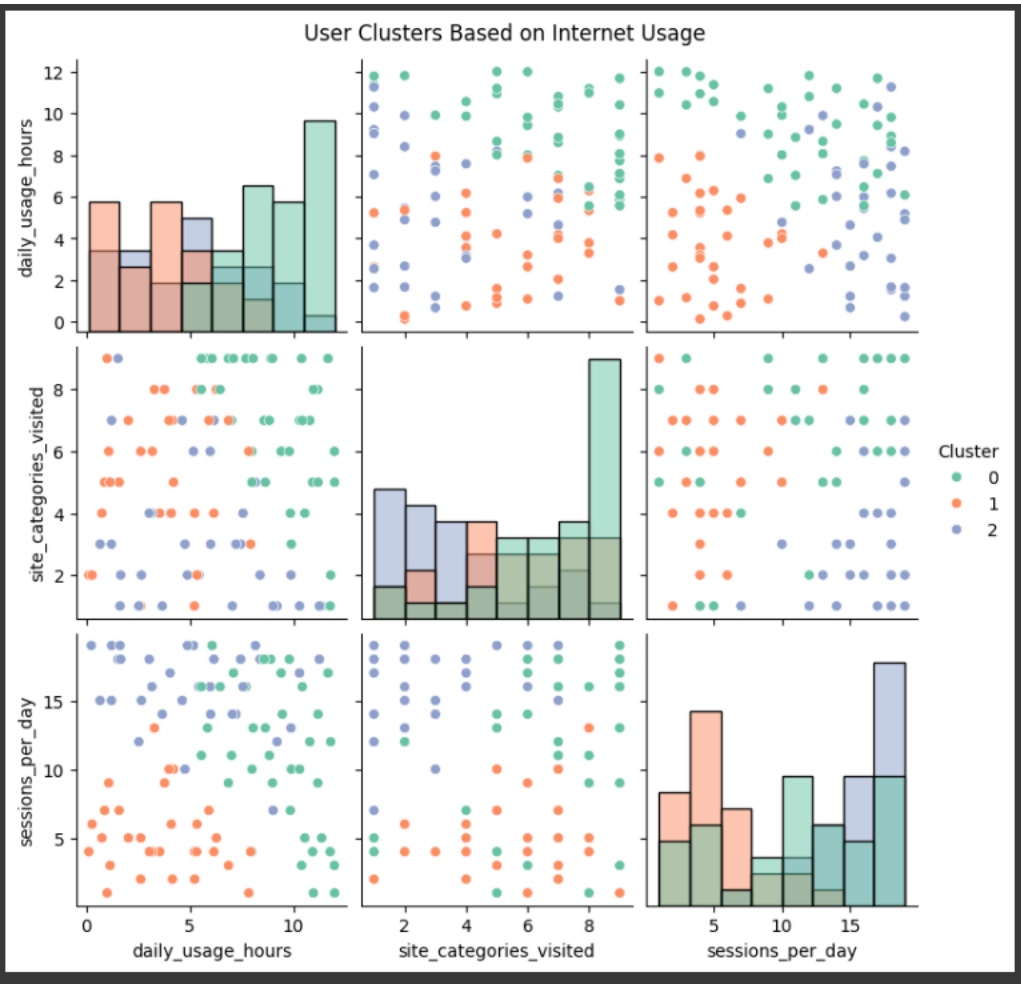
```
# Fit KMeans with optimal clusters
kmeans = KMeans(n_clusters=3, random_state=42)
df['Cluster'] = kmeans.fit_predict(scaled_data)

# Converting cluster to int
df['Cluster'] = df['Cluster'].astype(int)

# Visualizing Clusters with pairplot
sns.pairplot(df, hue='Cluster', palette='Set2', diag_kind='hist')
plt.suptitle("User Clusters Based on Internet Usage", y=1.02)
plt.show()
```

```
# Understanding characteristics of each cluster
print("\nCluster-wise Averages:")
print(df.groupby('Cluster').mean())
```

Output

```
Data Info:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 100 entries, 0 to 99
Data columns (total 3 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   daily_usage_hours       100 non-null    float64
 1   site_categories_visited  100 non-null    int64
 2   sessions_per_day        100 non-null    int64
dtypes: float64(1), int64(2)
memory usage: 2.5 KB
None

Summary Stats:
       daily_usage_hours  site_categories_visited  sessions_per_day
count         100.000000               100.000000        100.000000
mean            6.298375                 5.100000         10.870000
std             3.448911                 2.653376          5.799086
min             0.143016                 1.000000          1.000000
25%             3.494349                 3.000000          5.000000
50%             6.169502                 5.000000         11.500000
75%             9.069780                 7.000000         16.000000
max            11.988594                 9.000000         19.000000
```






```
Cluster-wise Averages:
         daily_usage_hours  site_categories_visited  sessions_per_day
Cluster
0                 9.307566                 6.631579         11.131579
1                 3.666101                 5.166667          5.200000
2                 5.192716                 3.218750         15.875000
```

Reference

Dataset: Internet_Usage.csv

Libraries used:

- Pandas
- Matplotlib
- Seaborn
- Scikit-learn