Dr. Vishwanath Karad MIT WORLD PEACE UNIVERSITY | PUNE

TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

Third Year B. Tech (EL & CE)

Semester: VI **Subject:** Data Science for Engineering

Name: Shreerang Mhatre **Class: TY**

Roll No: 52 **Batch: A2**

Experiment No: 07

Name of the Experiment: Clustering using Python

Performed on: 25/04/2024

Submitted on: 25/04/2024

Problem Statement:

Aim:

Write a python program to perform Clustering: We have the data for workout as below.

Date	Distance_km	Duration_min	Delta_last_workout	Day_category
10/17/17	4.3	21.58	1	0
11/04/17	1.9	9.25	18	1
11/18/17	1.9	9.0	14	1
11/23/17	1.9	8.93	5	0
11/28/17	2.3	11.94	5	0
11/29/17	2.8	14.05	1	0

To keep track of your performance you need to identify similar workout sessions. Clustering can help you group the data into distinct groups, guaranteeing that the data points in each group are similar to each other. Perform following steps:

i. Load the Data

ii. Data Exploratory Analysis: Pair Plot and Distance versus workout duration, distance versus duration with the number of days, correlation (Scatter plot) to get idea about correlation between different features.

iii. Select K-means clustering for model and get the clusters.

iv. Evaluate the performance of the model.

localhost:8888/notebooks/DSE%20Practicals/EXP%20-%207/Shreerang%20Mhatre_52_A2%20exp%20-%207.ipynb

jupyter Shreerang Mhatre_52_A2 exp - 7 Last Checkpoint: Last Tuesday at 10:18 AM (autosaved)

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

In [1]:
```
!pip install numpy --user
!pip install pandas --user
!pip install matplotlib --user
!pip install seaborn --user
!pip install scikit-learn --user
```

```
Requirement already satisfied: numpy in c:\users\shreerang\anaconda3\lib\site-packages (1.24.4)
Requirement already satisfied: pandas in c:\users\shreerang\anaconda3\lib\site-packages (1.4.4)
Requirement already satisfied: pytz>=2020.1 in c:\users\shreerang\anaconda3\lib\site-packages (from pandas) (2022.1)
Requirement already satisfied: numpy>=1.18.5 in c:\users\shreerang\anaconda3\lib\site-packages (from pandas) (1.24.4)
Requirement already satisfied: python-dateutil>=2.8.1 in c:\users\shreerang\anaconda3\lib\site-packages (from pandas) (2.8.2)
Requirement already satisfied: six>=1.5 in c:\users\shreerang\anaconda3\lib\site-packages (from python-dateutil>=2.8.1->pandas)
(1.16.0)
Requirement already satisfied: matplotlib in c:\users\shreerang\anaconda3\lib\site-packages (3.5.2)
Requirement already satisfied: packaging>=20.0 in c:\users\shreerang\anaconda3\lib\site-packages (from matplotlib) (21.3)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\shreerang\anaconda3\lib\site-packages (from matplotlib) (4.25.0)
Requirement already satisfied: numpy>=1.17 in c:\users\shreerang\anaconda3\lib\site-packages (from matplotlib) (1.24.4)
Requirement already satisfied: cycler>=0.10 in c:\users\shreerang\anaconda3\lib\site-packages (from matplotlib) (0.11.0)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\shreerang\anaconda3\lib\site-packages (from matplotlib) (2.8.2)
Requirement already satisfied: pyparsing>=2.2.1 in c:\users\shreerang\anaconda3\lib\site-packages (from matplotlib) (3.0.9)
Requirement already satisfied: pillow>=6.2.0 in c:\users\shreerang\anaconda3\lib\site-packages (from matplotlib) (9.2.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\shreerang\anaconda3\lib\site-packages (from matplotlib) (1.4.2)
Requirement already satisfied: six>=1.5 in c:\users\shreerang\anaconda3\lib\site-packages (from python-dateutil>=2.7->matplotli
b) (1.16.0)
Requirement already satisfied: seaborn in c:\users\shreerang\anaconda3\lib\site-packages (0.11.2)
Requirement already satisfied: numpy>=1.15 in c:\users\shreerang\anaconda3\lib\site-packages (from seaborn) (1.24.4)
Requirement already satisfied: scipy>=1.0 in c:\users\shreerang\anaconda3\lib\site-packages (from seaborn) (1.9.1)
Requirement already satisfied: pandas>=0.23 in c:\users\shreerang\anaconda3\lib\site-packages (from seaborn) (1.4.4)
Requirement already satisfied: matplotlib>=2.2 in c:\users\shreerang\anaconda3\lib\site-packages (from seaborn) (3.5.2)
Requirement already satisfied: pillow>=6.2.0 in c:\users\shreerang\anaconda3\lib\site-packages (from matplotlib>=2.2->seaborn)
(9.2.0)
Requirement already satisfied: cycler>=0.10 in c:\users\shreerang\anaconda3\lib\site-packages (from matplotlib>=2.2->seaborn)
(0.11.0)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\shreerang\anaconda3\lib\site-packages (from matplotlib>=2.2->se
```

Type here to search 37°C ENG 05:38 PM 25-04-2024

localhost:8888/notebooks/DSE%20Practicals/EXP%20-%207/Shreerang%20Mhatre_52_A2%20exp%20-%207.ipynb

jupyter Shreerang Mhatre_52_A2 exp - 7 Last Checkpoint: Last Tuesday at 10:18 AM (autosaved)

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

In [2]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn
from sklearn.cluster import KMeans
```

In [3]:
```
df = pd.read_csv('data_workout.csv')
```

In [4]:
```
df.shape
```

Out[4]:
```
(6, 5)
```

In [5]:
```
df.head(6)
```

Out[5]:

	date	distance_km	duration_min	delta_last_workout	day_category
0	17-10-2017	4.3	21.58	1	0
1	04-11-2017	1.9	9.25	18	1
2	18-11-2017	1.9	9.00	14	1
3	23-11-2017	1.9	8.93	5	0
4	28-11-2017	2.3	11.94	5	0
5	29-11-2017	2.8	14.05	1	0

In [6]:
```
df.dtypes
```

Out[6]:
```
date                  object
distance_km          float64
duration_min         float64
delta_last_workout     int64
day_category           int64
dtype: object
```

Type here to search 37°C ENG 05:38 PM 25-04-2024

localhost:8888/notebooks/DSE%20Practicals/EXP%20-%207/Shreerang%20Mhatre_52_A2%20exp%20-%207.ipynb

jupyter Shreerang Mhatre_52_A2 exp - 7 Last Checkpoint: Last Tuesday at 10:18 AM (autosaved)

File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel)

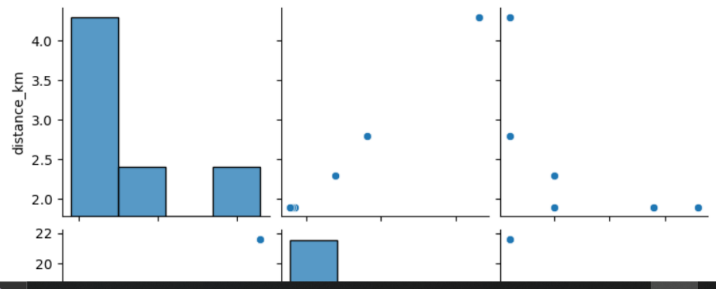
In [7]:
```
# Remove unnecessary attributes
df1=df[['distance_km', 'duration_min', 'delta_last_workout']]
df1
```

Out[7]:

	distance_km	duration_min	delta_last_workout
0	4.3	21.58	1
1	1.9	9.25	18
2	1.9	9.00	14
3	1.9	8.93	5
4	2.3	11.94	5
5	2.8	14.05	1

In [8]:
```
seaborn.pairplot(df1)
```

Out[8]:
```
<seaborn.axisgrid.PairGrid at 0x23c0acdee20>
```

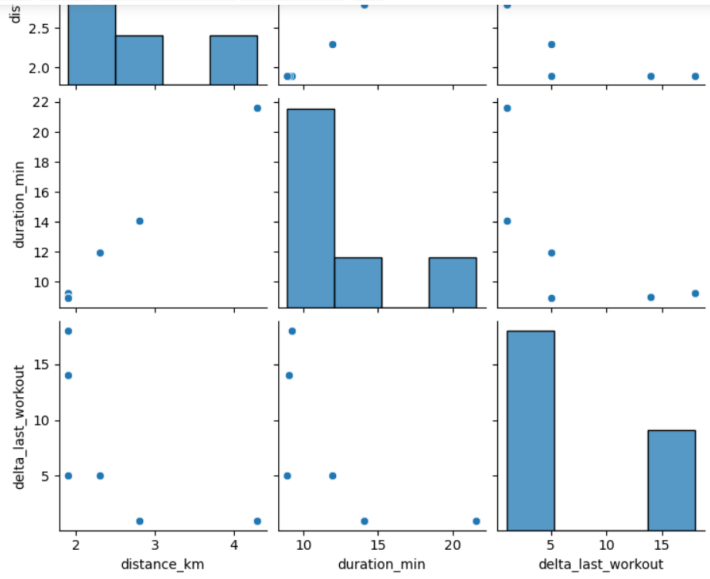


localhost:8888/notebooks/DSE%20Practicals/EXP%20-%207/Shreerang%20Mhatre_52_A2%20exp%20-%207.ipynb

jupyter Shreerang Mhatre_52_A2 exp - 7 Last Checkpoint: Last Tuesday at 10:18 AM (autosaved)

File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel)



localhost:8888/notebooks/DSE%20Practicals/EXP%20-%207/Shreerang%20Mhatre_52_A2%20exp%20-%207.ipynb

jupyter Shreerang Mhatre_52_A2 exp - 7 Last Checkpoint: Last Tuesday at 10:18 AM (autosaved)

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

In [9]:
```
#Set number of clusters at initialisation time
Kmeans = KMeans(n_clusters=2)
```

In [10]:
```
#Run the clustering algorithm
Kmeans.fit(df1)
```

```
C:\Users\SHREERANG\anaconda3\lib\site-packages\sklearn\cluster\_kmeans.py:1446: UserWarning: KMeans is known to have a memory l
eak on Windows with MKL, when there are less chunks than available threads. You can avoid it by setting the environment variabl
e OMP_NUM_THREADS=1.
  warnings.warn(
```

Out[10]:
```
KMeans(n_clusters=2)
```

In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

In [11]:
```
#Generate cluster predictions and store in y_hat
df2=Kmeans.predict(df1)
df2
```

Out[11]:
```
array([0, 1, 1, 1, 1, 0])
```

In [12]:
```
centroids = Kmeans.cluster_centers_
print(centroids)
```

```
[[ 3.55  17.815  1.   ]
 [ 2.     9.78  10.5  ]]
```

In [13]:
```
labels = Kmeans.labels_
print(labels)
```

```
[0 1 1 1 1 0]
```

In [15]:
```
#To define number of clusters

Sum_of_squared_distances = []
```

Type here to search 37°C ENG 05:38 PM 25-04-2024

localhost:8888/notebooks/DSE%20Practicals/EXP%20-%207/Shreerang%20Mhatre_52_A2%20exp%20-%207.ipynb

jupyter Shreerang Mhatre_52_A2 exp - 7 Last Checkpoint: Last Tuesday at 10:18 AM (autosaved)

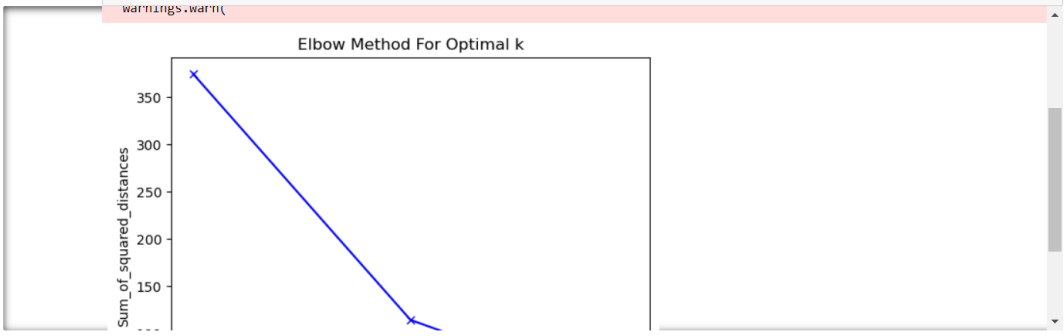
File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

In [15]:
```
#To define number of clusters

Sum_of_squared_distances = []
K = range(1,4)
for k in K:
    km = KMeans(n_clusters=k)
    km = km.fit(df1)
    Sum_of_squared_distances.append(km.inertia_)

plt.plot(K, Sum_of_squared_distances, 'bx-')
plt.xlabel('k')
plt.ylabel('Sum_of_squared_distances')
plt.title('Elbow Method For Optimal k')
plt.show()
```

```
  warnings.warn(
```



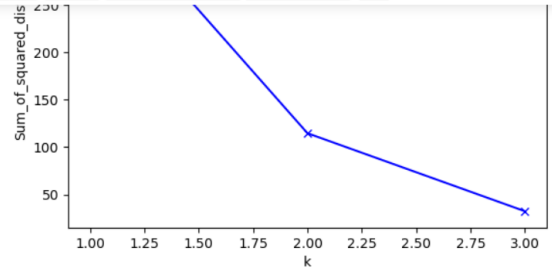
In [16]:
```
# Performance Alayisi
```

Type here to search 37°C ENG 05:38 PM 25-04-2024

localhost:8888/notebooks/DSE%20Practicals/EXP%20-%207/Shreerang%20Mhatre_52_A2%20exp%20-%207.ipynb

jupyter Shreerang Mhatre_52_A2 exp - 7 Last Checkpoint: Last Tuesday at 10:18 AM (autosaved)

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)



In [16]:
```
# Performance Alayisi
from sklearn.metrics import accuracy_score
dfo1 = df[['day_category']]

score = accuracy_score(dfo1, df2)
print('Accuracy:{0:f}'.format(score))
```

```
Accuracy:0.666667
```

In [ ]:

Type here to search 37°C ENG 05:38 PM 25-04-2024

MIT-WPU
MIT WORLD PEACE UNIVERSITY | PUNE

Exp 7 - DSE

Name - Shreerang Mhatre
Rollno - 52

* Post Lab Questions

Q1) What are some real-world applications of clustering algorithms, & how do they benefit from clustering?

→ ① Customer segmentation in Marketing
- clustering algorithms like K-means clustering can segment customers based on their purchasing behavior, demographics, or preferences. By identifying anomalies, cybersecurity systems can detect potential security breaches, fraud attempts, or malicious activities etc.

② Image segmentation in Computer vision.
- K-Means clustering or Mean shift clustering can segment images into distinct regions based on pixel similarity or color intensity. Image segmentation is crucial for object recognition, image analysis, and medical imaging applications, enhancing the accuracy of image processing.

Q2) Define clustering & k-means clustering.

→ ① clustering -

clustering is a machine learning technique used to group similar data points together based on certain characteristics or features. The goal of clustering is to create clusters or groups where data points within each cluster are more similar to each other than to data points in other clusters.

② k-Means Clustering -

k-means is an iterative algorithm that partitions a dataset into k clusters, where k is a pre defined number chosen by the user. The algorithm works by assigning each data point to the nearest cluster centroid, based on a distance metric such as Euclidean distance.