**Experiment Number : 5 - Applying similarity measures on the numeric datasets**

**Batch: FDS-2 Roll Number: 16010422234 Name: Chandana Ramesh Galgali**

**Aim of the Experiment:** Applying similarity measures on the numeric datasets and textual datasets

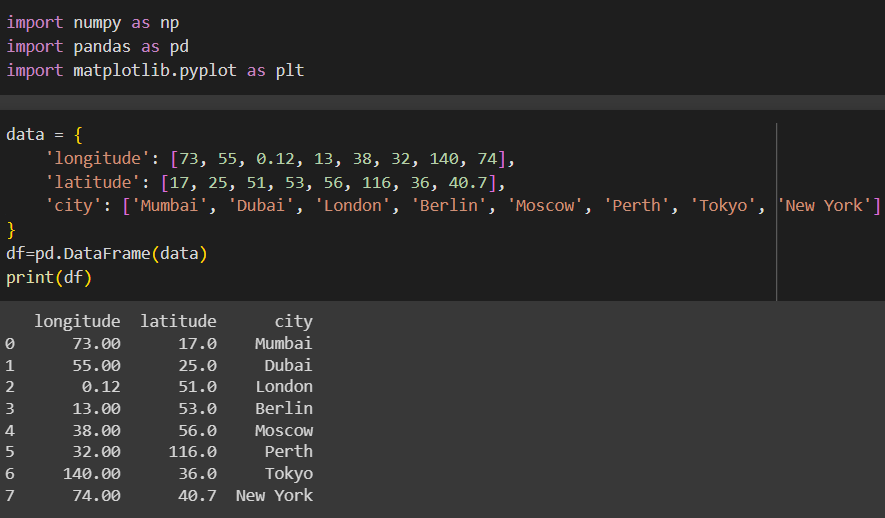
**Program/ Steps:**

Identify the suitable attributes to apply the numeric similarity measures and write python code to calculate

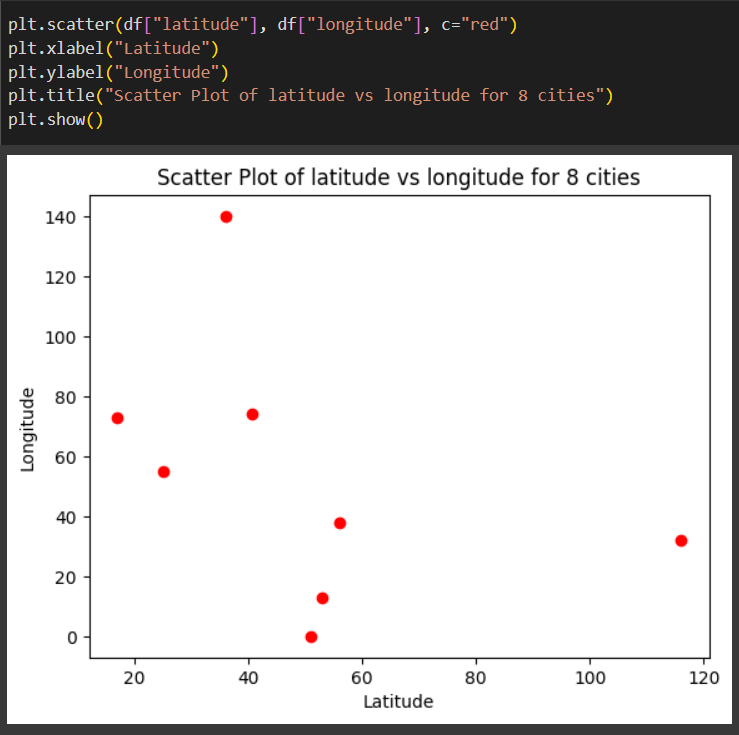
Euclidean, Manhattan similarity measures on it.

**Code with Output/Result:**

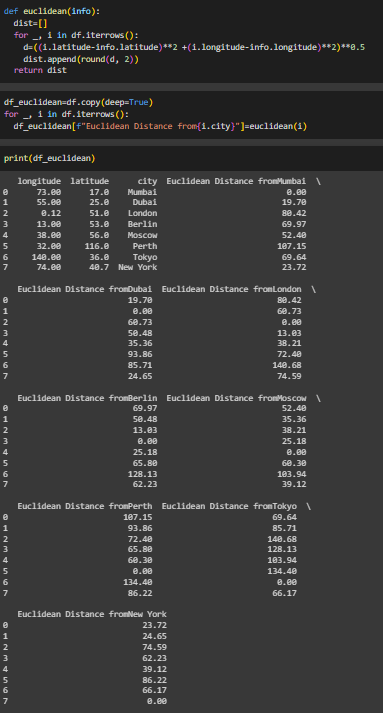
**1. Importing Libraries and creating dataset:**

****

**2. Scatter Plot:**

****

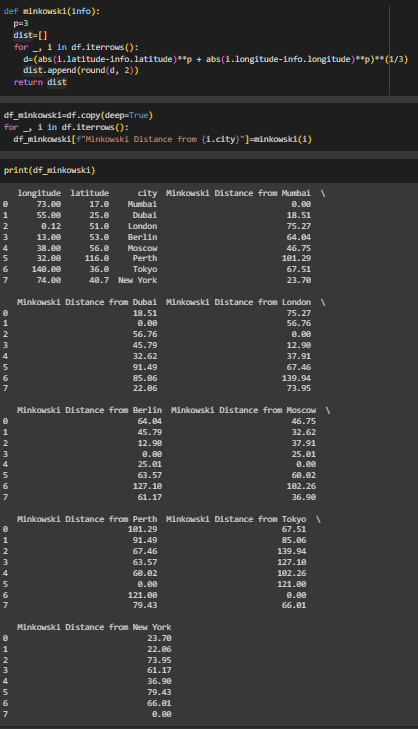
**3. Euclidean Distance:**

****

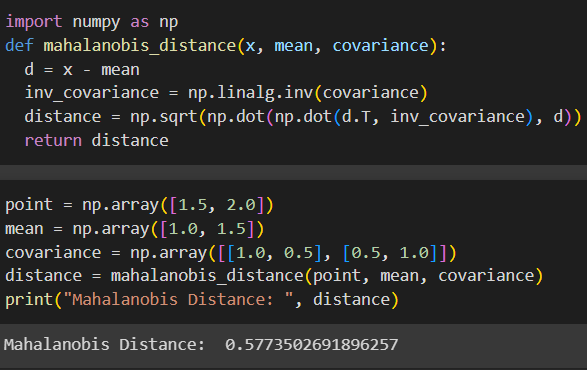
**4. Manhattan Distance:**

****

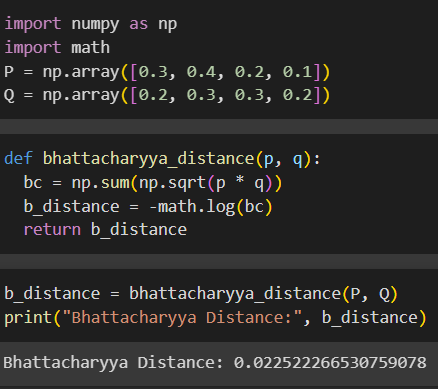
**5. Minkowski Distance:**

****

**6. Mahalanobis Distance:**

****

**6. Bhattacharyya Distance:**

****

**import numpy as np**

**import pandas as pd**

**import matplotlib.pyplot as plt**

**data = pd.read\_csv(r'C:\Users\daxay\Downloads\Flight\_delay.csv')**

**data\_array = data.to\_numpy()**

**dataframe = {**

**'Flight Number': data\_array[:, 7],**

**'Arrival Delay': data\_array[:, 12],**

**'Departure Delay': data\_array[:, 13]**

**}**

**df = pd.DataFrame(dataframe)**

**print("Dataframe:\n", df)**

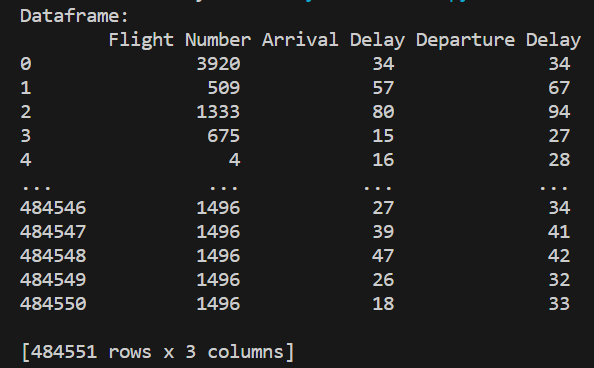
**plt.scatter(df["Arrival Delay"], df["Departure Delay"], c="red")**

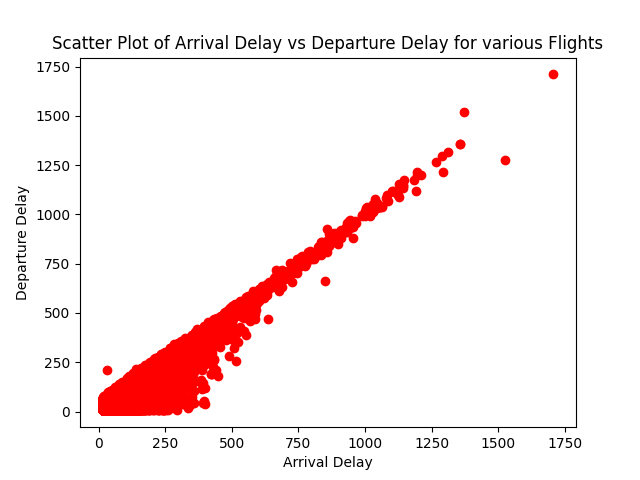
**plt.xlabel("Arrival Delay")**

**plt.ylabel("Departure Delay")**

**plt.title("Scatter Plot of Arrival Delay vs Departure Delay for various Flights")**

**plt.show()**

****

****

**import numpy as np**

**import pandas as pd**

**data = pd.read\_csv(r'C:\Users\daxay\Downloads\Flight\_delay.csv')**

**data\_array = data.to\_numpy()**

**dataframe = {**

**'Flight Number': data\_array[:4, 7],**

**'Arrival Delay': data\_array[:4, 12],**

**'Departure Delay': data\_array[:4, 13],**

**}**

**df = pd.DataFrame(dataframe)**

**print("Dataframe:\n", df)**

**def euclidean(info, df):**

**dist = []**

**for \_, i in df.iterrows():**

**d = ((i['Arrival Delay'] - info['Arrival Delay'])\*\*2 + (i['Departure Delay'] - info['Departure Delay'])\*\*2)\*\*0.5**

**dist.append(round(d, 2))**

**return dist**

**df\_euclidean = df.copy(deep=True)**

**for \_, i in df.iterrows():**

**df\_euclidean[f"Euclidean Distance from {i['Flight Number']}"] = euclidean(i, df)**

**print(df\_euclidean)**

**def manhattan(info, df):**

**dist = []**

**for \_, i in df.iterrows():**

**d = (abs(i['Arrival Delay'] - info['Arrival Delay']) + abs(i['Departure Delay'] - info['Departure Delay']))**

**dist.append(round(d, 2))**

**return dist**

**df\_manhattan = df.copy(deep=True)**

**for \_, i in df.iterrows():**

**df\_manhattan[f"Manhattan Distance from {i['Flight Number']}"] = manhattan(i, df)**

**print(df\_manhattan)**

**def minkowski(info, df):**

**p = 3**

**dist = []**

**for \_, i in df.iterrows():**

**d = (abs(i['Arrival Delay'] - info['Arrival Delay'])\*\*p + abs(i['Departure Delay'] - info['Departure Delay'])\*\*p)\*\*(1/3)**

**dist.append(round(d, 2))**

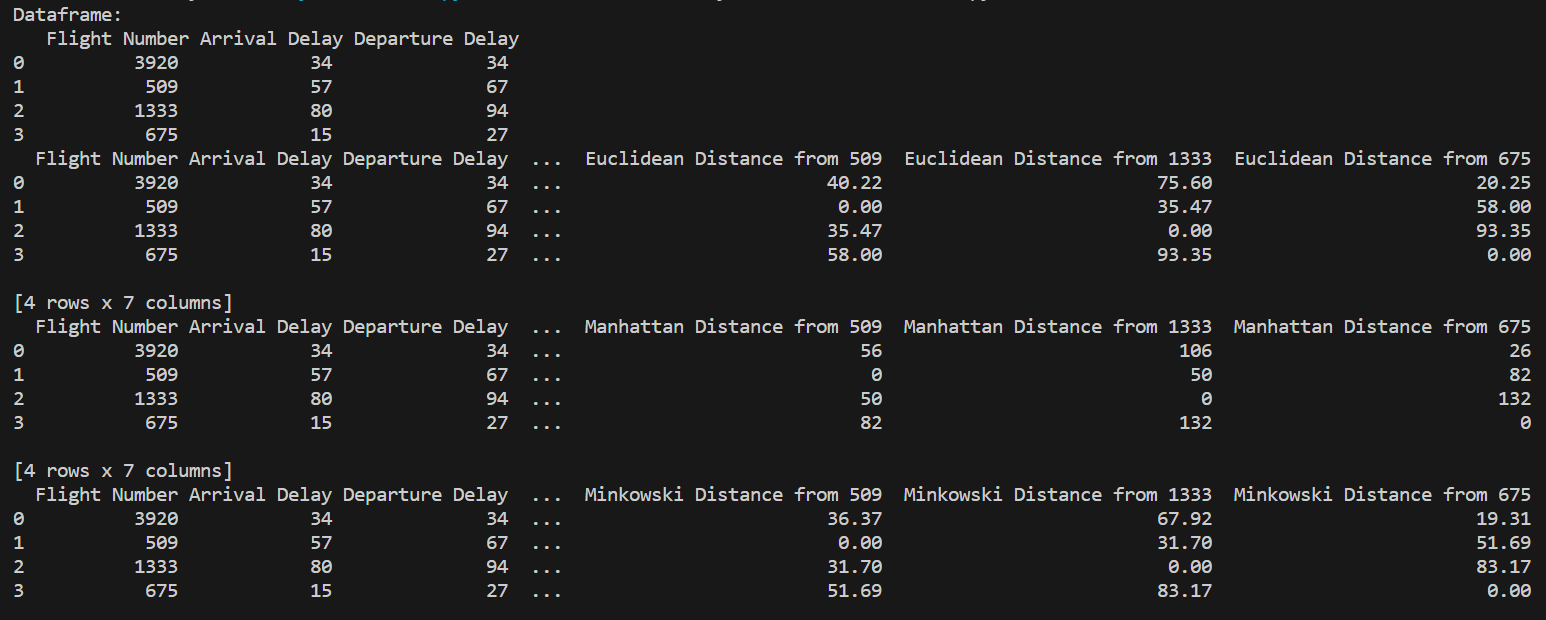
**return dist**

**df\_minkowski = df.copy(deep=True)**

**for \_, i in df.iterrows():**

**df\_minkowski[f"Minkowski Distance from {i['Flight Number']}"] = minkowski(i, df)**

**print(df\_minkowski)**

****

**Post Lab Question-Answers:**

**1. What is distance in Data Science and what is its importance?**

**Ans:** In the context of data science, distance refers to a measure of dissimilarity or similarity between two data points. It quantifies the separation or similarity between observations in a dataset. Distance metrics are used in various data science tasks, such as clustering, classification, and anomaly detection.

The importance of distance in data science lies in its ability to provide a quantitative measure of how different or similar data points are. It allows us to compare and analyze patterns, relationships, and structures within the data. By calculating distances, we can identify similar data points that belong to the same group or cluster, or detect outliers that are significantly different from the rest of the data.

Distance metrics, such as Euclidean distance, Manhattan distance, or cosine similarity, enable us to perform calculations and make informed decisions based on the proximity or dissimilarity of data points. They are fundamental tools in data science algorithms and techniques, helping us uncover insights, make predictions, and solve real-world problems.

**2. What are the different applications of Numeric similarity measure?**

**Ans:** Numeric similarity measures, also known as distance metrics, have various applications in data science. Some of the key applications include:

1. Clustering: Numeric similarity measures are used to group similar data points together in clustering algorithms. By calculating the distances between data points, clustering algorithms can identify natural groupings or clusters within a dataset.

2. Classification: Similarity measures are used in classification algorithms to determine the similarity between a new data point and existing labeled data points. This similarity is then used to assign a class label to the new data point.

3. Anomaly detection: Numeric similarity measures can help identify anomalies or outliers in a dataset. Data points that are significantly different from the majority of the data can be detected by calculating their distances from the rest of the data points.

4. Recommender systems: Similarity measures are used in recommender systems to find items or products that are similar to a user's preferences. By calculating the similarity between user profiles or item features, recommender systems can provide personalized recommendations.

5. Dimensionality reduction: Similarity measures are used in dimensionality reduction techniques, such as t-SNE or PCA, to preserve the similarity relationships between data points in lower-dimensional representations.

6. Time series analysis: Similarity measures are used to compare and analyze time series data. By calculating the similarity between different time series, patterns, trends, or anomalies can be identified.

These are just a few examples of the applications of numeric similarity measures in data science. The versatility of these measures allows them to be applied in various domains and problem-solving scenarios.

**3. Why use Mahalanobis distance if Euclidean distances are available? Give suitable examples with justification.**

**Ans:** The Mahalanobis distance is used when there are correlations or dependencies between variables in the dataset, which cannot be captured by the Euclidean distance. Unlike the Euclidean distance, the Mahalanobis distance takes into account the covariance structure of the data, making it a more appropriate choice in certain scenarios. Here are a few examples where the Mahalanobis distance is preferred over the Euclidean distance:

1. Outlier detection: In outlier detection, the Mahalanobis distance is useful when the variables in the dataset are correlated. By considering the covariance structure, the Mahalanobis distance can identify outliers that deviate from the expected patterns, even if they are not far away in terms of Euclidean distance.

2. Multivariate analysis: When analyzing multivariate data, the Mahalanobis distance is used to measure the similarity or dissimilarity between observations. It accounts for the correlations between variables, allowing for a more accurate assessment of the distance between data points.

3. Anomaly detection in high-dimensional data: In high-dimensional datasets, the Euclidean distance becomes less reliable due to the curse of dimensionality. The Mahalanobis distance, on the other hand, can handle high-dimensional data by considering the covariance structure, making it more suitable for anomaly detection in such cases.

4. Classification with imbalanced data: In classification tasks with imbalanced data, where the number of samples in different classes is significantly different, the Mahalanobis distance can help address the issue. By considering the covariance structure, the Mahalanobis distance can give more weight to the minority class, leading to better classification performance.

In summary, the Mahalanobis distance is preferred over the Euclidean distance when there are correlations or dependencies between variables in the dataset. It provides a more accurate measure of distance by considering the covariance structure, making it suitable for outlier detection, multivariate analysis, anomaly detection in high-dimensional data, and classification with imbalanced data.

**Outcomes: Comprehend descriptive and proximity measures of data**

**Conclusion (based on the Results and outcomes achieved):**

**References:**

Books/ Journals/ Websites

1. Han, Kamber, "Data Mining Concepts and Techniques", Morgan Kaufmann 3nd Edition