# Task 5: Dimensionality Reduction Analysis

## 1. Introduction

This document outlines the approach taken to analyze pairwise similarity among data objects after applying dimensionality reduction techniques using Principal Component Analysis (PCA). The task involves repeating the pairwise similarity analysis from Task 4, but with a reduced-dimensional dataset, and providing insights on the effectiveness of dimensionality reduction.

## 2. Dimensionality Reduction

### 2.1 Dataset:

The dataset used for this analysis is the 'Air Quality UCI' dataset, which includes various air quality metrics.

### 2.2 Dimensionality Reduction Technique:

Principal Component Analysis (PCA) was applied to reduce the number of features in the dataset. PCA is a technique that transforms the data into a set of orthogonal components, capturing the most variance with fewer dimensions.

### 2.3 PCA Implementation:

Number of Components: 5  
Preprocessing Steps:  
- Replaced missing values (-200) with NaN.  
- Filled missing values in numeric columns with the median of the respective columns.  
- Standardized numeric features.

## Python Code:

import pandas as pd  
import numpy as np  
from sklearn.preprocessing import StandardScaler  
from sklearn.metrics.pairwise import cosine\_similarity  
from sklearn.decomposition import PCA  
  
# Load and preprocess the dataset  
df = pd.read\_excel('AirQualityUCI.xlsx')  
  
# Replace -200 with NaN  
df.replace(-200, np.nan, inplace=True)  
  
# Separate numeric columns  
numeric\_df = df.select\_dtypes(include=[np.number])  
  
# Compute medians for numeric columns and fill missing values  
df[numeric\_df.columns] = numeric\_df.fillna(numeric\_df.median())  
  
# Reset index and sample 20 data objects  
df.reset\_index(drop=True, inplace=True)  
sampled\_df = df.sample(n=20, random\_state=1)  
  
# Extract numeric features for similarity calculation  
features = sampled\_df.select\_dtypes(include=[np.number])  
  
# Standardize features  
scaler = StandardScaler()  
scaled\_features = scaler.fit\_transform(features)  
  
# Compute Cosine Similarity with original features  
cosine\_sim\_original = cosine\_similarity(scaled\_features)  
  
def get\_max\_similarity\_pair(similarity\_matrix):  
 np.fill\_diagonal(similarity\_matrix, 0) # Set diagonal to 0 to avoid self-similarity  
 max\_similarity\_idx = np.unravel\_index(np.argmax(similarity\_matrix, axis=None), similarity\_matrix.shape)  
 max\_similarity\_score = similarity\_matrix[max\_similarity\_idx]  
 return max\_similarity\_idx, max\_similarity\_score  
  
def print\_pair\_info(df, idx, measure\_name, score):  
 # Extract numeric columns for displaying pairs  
 numeric\_columns = df.select\_dtypes(include=[np.number]).columns  
 pair\_1 = df.iloc[idx[0]][numeric\_columns]  
 pair\_2 = df.iloc[idx[1]][numeric\_columns]  
   
 print(f"\n{measure\_name} Similarity:")  
 print(f"Pair with maximum similarity (Index {idx[0]} and {idx[1]}):")  
 print(f"Pair 1:\n{pair\_1}\n")  
 print(f"Pair 2:\n{pair\_2}\n")  
 print(f"Similarity Score: {score:.4f}")  
  
 # Check if they are really similar  
 if score > 0.8: # You can adjust this threshold based on your context  
 print("The pairs are really similar.")  
 else:  
 print("The pairs are not very similar.")  
  
# Get the pair with maximum similarity and the similarity score in the original dataset  
original\_max\_idx, original\_max\_score = get\_max\_similarity\_pair(cosine\_sim\_original)  
print\_pair\_info(sampled\_df, original\_max\_idx, 'Cosine Similarity (Original)', original\_max\_score)  
  
# Apply PCA for dimensionality reduction  
pca = PCA(n\_components=5) # Reduce to 5 dimensions  
pca\_features = pca.fit\_transform(scaled\_features)  
  
# Compute Cosine Similarity with PCA-reduced features  
cosine\_sim\_pca = cosine\_similarity(pca\_features)  
  
# Get the pair with maximum similarity and the similarity score after PCA  
pca\_max\_idx, pca\_max\_score = get\_max\_similarity\_pair(cosine\_sim\_pca)  
print\_pair\_info(sampled\_df, pca\_max\_idx, 'Cosine Similarity (PCA)', pca\_max\_score)

## 3. Results

### 3.1 Cosine Similarity (Original Dataset):

Pair with maximum similarity (Index 0 and 2):  
Pair 1:  
CO(GT) 1.1  
PT08.S1(CO) 1047.333333  
NMHC(GT) 74.0  
C6H6(GT) 4.932008  
PT08.S2(NMHC) 760.0  
NOx(GT) 64.0  
PT08.S3(NOx) 1032.0  
NO2(GT) 74.0  
PT08.S4(NO2) 1378.666667  
PT08.S5(O3) 1003.0  
T 11.466667  
RH 61.433333  
AH 0.830289  
  
Pair 2:  
CO(GT) 1.8  
PT08.S1(CO) 1129.5  
NMHC(GT) 56.0  
C6H6(GT) 5.191654  
PT08.S2(NMHC) 773.0  
NOx(GT) 70.0  
PT08.S3(NOx) 1130.25  
NO2(GT) 82.0  
PT08.S4(NO2) 1451.75  
PT08.S5(O3) 1050.5  
T 12.1  
RH 61.100001  
AH 0.860316  
  
Similarity Score: 0.9681  
Insight: The pairs are highly similar, showing strong consistency in the high-dimensional feature space.

### 3.2 Cosine Similarity (PCA-Reduced Dataset):

Pair with maximum similarity (Index 10 and 16):  
Pair 1:  
CO(GT) 1.8  
PT08.S1(CO) 1125.0  
NMHC(GT) 150.0  
C6H6(GT) 12.170581  
PT08.S2(NMHC) 1055.5  
NOx(GT) 179.8  
PT08.S3(NOx) 645.0  
NO2(GT) 109.0  
PT08.S4(NO2) 1715.5  
PT08.S5(O3) 929.25  
T 33.825001  
RH 29.225  
AH 1.514256  
  
Pair 2:  
CO(GT) 2.1  
PT08.S1(CO) 1180.75  
NMHC(GT) 150.0  
C6H6(GT) 12.816446  
PT08.S2(NMHC) 1077.5  
NOx(GT) 159.0  
PT08.S3(NOx) 706.0  
NO2(GT) 122.0  
PT08.S4(NO2) 1869.75  
PT08.S5(O3) 1139.75  
T 35.4  
RH 27.825  
AH 1.573265  
  
Similarity Score: 0.9854

**Answer :** The pairs really are indeed very similar. By examining the individual attributes, it is evident that all the points are relatively close to each other.

Insight: The similarity score is slightly higher after PCA, indicating that PCA has preserved or enhanced the similarity between the pairs. The pairs are still highly similar, suggesting that PCA effectively captures the essential relationships within the data.

## 4. Conclusion

The analysis demonstrates that PCA effectively reduces dimensionality while retaining the critical relationships within the dataset. Both the original and PCA-reduced datasets show high similarity scores between the most similar pairs. The slight increase in similarity score post-PCA suggests that dimensionality reduction can enhance data representation by removing noise and focusing on the principal components that capture the most variance.