**Linear Algebra Assignment 4**

**Sriram Radhakrishna PES1UG20CS435 Section : ‘H’**

**Principal Component Analysis applied on MNIST Dataset**

**Python code (executed on a kaggle notebook, space separated based on cell content) :**

# initial library import

import numpy as np

import pandas as pd

import seaborn as sns

# data import

data = pd.read\_csv('../input/mnist-data/train.csv')

data.head()

# dropping unnecessary labels

label = data['label'] # save label data for later use

data.drop('label', axis = 1, inplace = True)

data.head()

# scaling data to have a mean of 0 and standard deviation of 1

from sklearn.preprocessing import StandardScaler

data\_standardized = StandardScaler().fit\_transform(data)

data\_standardized

# covariance matrix to determine dimensional relationships

covMatrix = np.matmul(data\_standardized.T ,data\_standardized)

covMatrix

# eigenvalue & eigenvector calculation to determine principal components

from scipy.linalg import eigh

values, vector = eigh(covMatrix,eigvals=(782,783))

vector = vector.T

values

# projecting vector on standardized data

projectedData = np.matmul(vector, data\_standardized.T)

projectedData

# preparing stacked data for visualization

reducedData = np.vstack((projectedData, label)).T

reducedData = pd.DataFrame(reducedData, columns = ['pca\_1', 'pca\_2', 'label'])

# data visualization

sns.FacetGrid(reducedData, hue = 'label', size = 8).map(sns.scatterplot, 'pca\_1', 'pca\_2').add\_legend()

# visualization of what the dataset actually represents

import matplotlib.pyplot as plt

index = 1234 # random index chosen for representation purposes

fig\_data = np.array(data.iloc[index]).reshape(28,28)

plt.imshow(fig\_data, interpolation = None, cmap = 'gray')

plt.show()

print('Digit represented : ', label[index])

**Output screenshots :**

















