| In [1]:  | Principal Component Analysis (PCA)  # Importing Libraries  |
|--|--|
|  | <pre>import pandas as pd import numpy as np import matplotlib.pyplot as plt from sklearn.preprocessing import StandardScaler</pre>   |
|  | <pre>from scipy.linalg import eigh import seaborn as sns import plotly.express as px import warnings warnings.filterwarnings('ignore')</pre>   |
| In [2]:  | <pre># Loading MNIST Dataset  mnist = pd.read_csv('train.csv') mnist.head()</pre>  |
| Out[2]:  | label         pixel0         pixel1         pixel2         pixel3         pixel6         pixel6         pixel74         pixel775         pixel775         pixel776         pixel778         pixel779         pixel780         pixel781         pixel782         pixel783           0         1         0   |
|  | 2       1       0  |
| In [3]:  | "Store the label in seperate variable  |
|  | <pre>lab = mnist['label']  # drop label and store date in data data = mnist.drop('label',axis =1)</pre>  |
| In [4]:  | <pre>#Check the shape of data print(lab.shape) print(data.shape)  (42000,)</pre>   |
| In [5]:  | (42000, 784)   |
|  | <pre>grid_data = data.iloc[idx].values.reshape(28,28) plt.imshow(grid_data, interpolation = "none", cmap = "gray") plt.show()</pre>  |
|  | 0 -<br>5 -   |
|  | 10 -   |
|  |  |
|  |  |
| In [6]:  |  |
| In [7]:  | 9 # Standardizing the data   |
| To [0].  | <pre>standardized_data = StandardScaler().fit_transform(data) print(standardized_data.shape)  (42000, 784)</pre>   |
| In [8]:  | <pre># Calculating the Covariance data = standardized_data covar_matrix = np.matmul(data.T,data) print("The shape of Covarince matrix = ",covar_matrix.shape)</pre>  |
| In [9]:  | <pre>values , vectors = eigh(covar_matrix , eigvals=(782,783)) print("shape", vectors.shape)</pre>   |
|  | <pre>vectors = vectors.T print("updated shape", vectors.shape)  shape (784, 2) updated shape (2, 784)</pre>  |
| In [10]:   | <pre># projecting original data to plane new_coordinates = np.matmul(vectors, data.T) print("resultant new datapoints shape", vectors.shape ,"X", data.T.shape)  resultant new datapoints shape (2, 784) X (784, 42000)</pre>  |
| In [11]:   | # Appending label to 2d projected data new_coordinates = np.vstack((new_coordinates,lab)).T  |
| In [12]:   | <pre>dataframe = pd.DataFrame(data=new_coordinates, columns=("1st Principal" , "2nd Principal" , "labels")) print(dataframe.head())  1st Principal 2nd Principal labels</pre>  |
|  | 1st Principal 2nd Principal labels 0   |
| In [13]:   | <pre># Plotting 2d data using seaborn sns.FacetGrid(dataframe , hue ="labels", size = 6).map(plt.scatter, "1st Principal" , "2nd Principal").add_legend() plt.show()</pre>   |
|  | 30 -   |
|  | Iabels 0.0 1.0 2.0   |
|  | 9 3.0<br>4.0<br>5.0<br>6.0<br>7.0  |
|  | 0-   |
|  | -10 - 10 10 20 1st Principal   |
| In [14]:   | fig = px.scatter(dataframe, x="1st Principal", y="2nd Principal", color="labels") fig.show()   |
|  | labels 9   |
|  | 25<br>20   |
|  | Jed Principal 15  10  4  |
|  |  |
|  |  |
|  | -15 -10 -5 0 5 10 15 20 25 30  1st Principal   |
| In [15]:   | PCA using Scikit-Learn  # Initializing the PCA   |
| In [16]:   | <pre>from sklearn import decomposition pca = decomposition.PCA()  # Configuring the parameters</pre>   |
|  | <pre># Number of components = 2 pca.n_components = 2 pca_data = pca.fit_transform(data)</pre>  |
|  | <pre>print("shape of pca_reduced =", pca_data.shape)</pre>   |
| In [17]:   | <pre>print("shape of pca_reduced =", pca_data.shape) shape of pca_reduced = (42000, 2)  # Appending label to 2d projected data pca_data = np.vstack((pca_data.T,lab)).T</pre>  |
| In [18]:   | shape of pca_reduced = (42000, 2)  # Appending label to 2d projected data  |
|  | <pre>shape of pca_reduced = (42000, 2)  # Appending label to 2d projected data pca_data = np.vstack((pca_data.T,lab)).T  # Creating new dataframe for plotting labelled points</pre>   |
| In [18]:   | <pre>shape of pca_reduced = (42000, 2)  # Appending label to 2d projected data pca_data = np.vstack((pca_data.T,lab)).T  # Creating new dataframe for plotting labelled points pca_dataframe = pd.DataFrame(data=pca_data, columns=("1st Principal" , "2nd Principal" , "labels"))  # Plotting 2d data using seaborn sns.FacetGrid(pca_dataframe , hue ="labels", size = 6).map(plt.scatter, "1st Principal" , "2nd Principal").add_legend()</pre>   |
| In [18]:   | shape of pca_reduced = (42000, 2)  # Appending label to 2d projected data pca_data = np.vstack((pca_data.T,lab)).T  # Creating new dataframe for plotting labelled points pca_dataframe = pd.DataFrame(data=pca_data, columns=("ist Principal", "2nd Principal", "labels"))  # Plotting 2d data using seaborn sns.FacetGrid(pca_dataframe, hue = "labels", size = 6).map(plt.scatter, "ist Principal", "2nd Principal").add_legend() plt.show()  |
| In [18]:   | shape of pca_reduced = (42880, 2)  # Appending label to 2d projected data pca_data = np.vstack((pca_data.T,lab)).T  # Creating new dataframe for plotting labelled points pca_dataframe = pd.bataFrame(data=pca_data, columns=("ist Principal" , "2nd Principal" , "labels"))  # Plotting 2d data using seaborn sns.FacetGrid(pca_dataframe , hue ="labels", size = 6).map(plt.scatter, "ist Principal" , "2nd Principal").add_legend() plt.show()    Document   Do |
| In [18]:   | shape of pca_reduced = (42800, 2)  # Appending label to 2d projected data pca_data = np.vstack((pca_data.T,lab)).T  # Creating new dataframe for plotting labelled points pca_dataframe = pd DataFrame(data=pca_data, columns=("ist Principal", "labels"))  # Plotting 2d data using seaborn sns.FacetOrid(pca_dataframe , hue ="labels", size = 6).map(plt.scatter, "ist Principal", "2nd Principal").add_legend() plt.show()    Data   |
| In [18]:   | shape of pca_reduced = (42800, 2)  # Appending label to 2d projected data pca_data = np.vstack((pca_data.1,lab)).1  # Creation now data frame for plotting labelled points pca_dataframe = pd.Outsframe(datapca_data, columns=("ist Principal", "2nd Principal", "labels"))  # Flotting 2d data using scapers ans.Facetrid(pca_dataframe, hue = "labels", size = 6).map(plt.scatter, "ist Principal", "2nd Principal").add_legend()  plt.snow()  # Libets  |
| In [18]:  In [19]:   | shape of pea_reduced = (42000, 2)  # Appending label to 2d projected data pea_data = np.vstack((pea_data.T,lab)).T  # Creating new dataframe for plotting labelied points pea_dataframe = pd Gentaframe(data=pea_data, columns=("ist Principal" , "2nd Principal" , "labels"))  # Planting 2d data using seaborn sea_faceteriat(pea_dataframe , hue ="labels", size = 6).map(plt.scatter, "ist Principal" , "2nd Principal").add_tegend() plt.shau()  # Principal  |
| In [18]:  In [19]:   | shape of per reduced = (47800, 7)  * Appending label to 20 monoperate data post_data = np.vstack((per_estat., Jab)).T  * Creating now dataframe for obtaing labelled points post_datarrame = per.bettsFrame(data-per_estat., Columns("sst Principal", "2nd Principal", "laseis"))  * Floting 2d data using season sin: Recetric(per_estatrame = nue = labels", size = 8).map(pit.scatter, "ist Principal", "2nd Principal").add_legend()  # Floting 2d data using season sin: Recetric(per_estatrame = nue = labels", size = 8).map(pit.scatter, "ist Principal", "2nd Principal").add_legend()  ### Principal   P |
| In [18]:  In [19]:   | Shape of pos_reduced = (4000, 2)  # Appearance Abbasic to deprotected data pos_cata = ns_vertack([pos_cata.r.lab)).T  ## Committing wise Abbasic (notes that it appeal matrix and post post post post post post post post  |
| In [18]:  In [19]:   | Shape of pos_reduced = (4000, 2)  # Appearance Abbasic to deprotected data pos_cata = ns_vertack([pos_cata.r.lab)).T  ## Committing wise Abbasic (notes that it appeal matrix and post post post post post post post post  |
| In [18]:  In [19]:   | Change of polynomials (1020, 2)  # Appending shade to 20 projected and polynomials (1020, 2015)   10 polynomials (1020, 2015)  |
| In [18]:  In [19]:   | Stopp of parameters dead 1.00 purposes of dead parameters produced = (LTDM, 7)  # Creating can distance for classing labelled pourse produced and parameters produced and para |
| In [18]:  In [19]:   | where of your returned ~ (NOW, 2)  A Apparature place of the 2 projected does  A Commission of the Com |
| In [19]:  In [20]:   | thape or pan_returns = (42000, 2)  # Cypen_pan_returns = (42000, 2)  # Cypen_pan_returns = (10.78 to pan_returns do not performed to pan_returns = (10.78 to pan_returns = (10 |
| In [19]:  In [20]:   | expected to a return of control of comparison of the process of th |
| In [19]:  In [20]:   | these of pautocean - readon, 2)  # Administration for participation of the participation of t |
| In [19]:  In [20]:   | Processing and the Processing Annual Processing  |
| In [19]:  In [20]:  In [21]:   | Secretary Secretary and a state of the control of t |
| In [19]:  In [20]:  In [21]:   | Applied to the control of the contro |
| In [19]:  In [20]:  In [21]:  In [24]:  Out [24]:                                    | ### Committed in 1928, 19   Committed in 1928, 19   Committed in 1928, 19   Committed in 1928, 1 |
| In [19]: In [20]: In [21]: In [24]:  | ### Committed in 1928, 19   Committed in 1928, 19   Committed in 1928, 19   Committed in 1928, 1 |
| In [18]:  In [19]:  In [20]:  In [21]:  In [24]:  In [25]:                           | The content of the co |
| In [18]:  In [29]:  In [21]:  In [22]:   | And the control and the contro |
| In [18]:  In [29]:  In [21]:  In [22]:   | However the content of common and content of |
| In [18]:  In [29]:  In [20]:  In [21]:  In [23]:                                     | And the content of th |
| In [18]:  In [19]:  In [20]:  In [21]:  In [24]:  In [25]:                           | The Control of State of Control of State of Stat |
| In [18]:  In [19]:  In [20]:  In [21]:  In [24]:  In [25]:                           | The Control of the Co |
| In [18]:  In [19]:  In [20]:  In [21]:  In [24]:  In [25]:                           | The second secon |
| In [18]:  In [19]:  In [20]:  In [21]:  In [24]:  In [25]:  In [26]:                 | Part      |
| In [18]:  In [19]:  In [20]:  In [21]:  In [22]:  In [23]:  In [26]:  In [26]:       | Company   Comp   |
| In [29]:  In [20]:  In [21]:  In [24]:  In [25]:  In [26]:                           | Secretary and an extension of the control of the co |
| In [28]:  In [20]:  In [22]:  In [24]:  In [25]:  In [26]:  In [27]:                 | AND THE PROPERTY OF THE PROPER |
| In [20]:  In [21]:  In [22]:  In [23]:  In [26]:  In [26]:  In []:  In []:           | AND THE PROPERTY OF THE PROPER |
| In [20]:  In [21]:  In [22]:  In [23]:  In [24]:  In [25]:  In [26]:  In []:  In []: | AND THE PROPERTY OF THE PROPER |
| In [20]:  In [21]:  In [22]:  In [23]:  In [24]:  In [25]:  In [26]:  In []:  In []: | AND THE PROPERTY OF THE PROPER |
| In [20]:  In [21]:  In [22]:  In [23]:  In [24]:  In [25]:  In [26]:  In []:  In []: | AND THE PROPERTY OF THE PROPER |