PCA Lab Exercise

Lab Exercise 1: Understanding PCA with a Simple Dataset

Objective: To understand the basic principles of PCA by reducing the dimensionality of a simple dataset.

Instructions:

1. **Load the Dataset:**
    - Use the Iris dataset (available in sklearn or seaborn) which contains 4 features and 3 classes.
    - Perform a quick exploratory data analysis (EDA) to visualize the data and identify potential patterns.
2. **Standardization:**
    - Standardize the features so that they have a mean of 0 and a standard deviation of 1.
3. **Compute the Covariance Matrix:**
    - Calculate the covariance matrix of the standardized data.
4. **Eigenvalues and Eigenvectors:**
    - Calculate the eigenvalues and eigenvectors of the covariance matrix.
5. **PCA Transformation:**
    - Sort the eigenvectors by the magnitude of their corresponding eigenvalues.
    - Project the data onto the first two principal components.
6. **Visualization:**
    - Create a scatter plot of the data in the new 2D space defined by the first two principal components. Use different colors for each class in the Iris dataset.

Lab Exercise 2: PCA for Image Compression

Objective: Use PCA to compress and then reconstruct an image, demonstrating the power of dimensionality reduction in data compression.

Instructions:

1. **Load an Image:**
    - Load a grayscale image (e.g., a 256x256 image of a face or any simple object).
2. **Reshape the Image:**
    - Treat the image as a matrix and flatten it to a 2D matrix where each row is a pixel and each column is a feature (intensity values of pixels).
3. **Apply PCA:**
    - Perform PCA on the image data, reducing the number of principal components used for reconstruction.
4. **Reconstruction:**
    - Reconstruct the image using a different number of principal components (e.g., 5, 20, 50, 100).
5. **Visualize Results:**
    - Display the original image and the reconstructed images at various levels of dimensionality reduction.

Lab Exercise 3: PCA for Feature Reduction in a Classification Task

Objective: Apply PCA as a preprocessing step to reduce the feature space for a classification problem and compare its effect on model performance.

Instructions:

1. **Load a Dataset:**
    - Use a dataset with many features, such as the Wine dataset from sklearn, which has 13 features.
2. **Split the Data:**
    - Split the data into training and testing sets (e.g., 70% training, 30% testing).
3. **Baseline Model (No PCA):**
    - Train a classification model (e.g., Logistic Regression or SVM) on the raw dataset without applying PCA. Evaluate its performance using accuracy, precision, and recall.
4. **Apply PCA:**
    - Apply PCA to the training data, retaining different numbers of components (e.g., 2, 5, 10).
    - Project the test data onto the same principal components.
5. **Train and Evaluate:**
    - Train the same classification model on the reduced dataset.
    - Compare the performance of the models trained with different numbers of principal components.
6. **Visualization:**
    - Plot a graph showing how accuracy changes as the number of components increases.

Questions:

- How does the number of components affect model performance?
- Is there a point where adding more components doesn't improve accuracy?