Experiment No. 5

1. **Aim:**

   To understand and implement the Principal Component Analysis (PCA) algorithm on a real-world dataset to reduce dimensionality and analyze the results.

2. **Objectives:**

   1. To grasp the mathematical foundation of PCA, including eigenvectors and eigenvalues.
   2. To apply PCA for dimensionality reduction on a high-dimensional dataset.
   3. To visualize the data in reduced dimensions and interpret the significance of principal components.
   4. To evaluate the variance retained after applying PCA.

3. **Course Outcomes:**

   By the end of this experiment, students should be able to:

   1. Explain the PCA algorithm and its importance in reducing dimensionality.
   2. Implement PCA on a real-world dataset.
   3. Interpret the results of PCA, including the explained variance by principal components.
   4. Use PCA to enhance the performance of machine learning models by reducing noise and redundancy in data.

4. **Hardware / Software Required:**

   1. **Hardware:** A computer with at least 8 GB of RAM for handling large datasets.
   2. **Software:** Python (with libraries like NumPy, pandas, matplotlib, scikit-learn), R

5. **Theory:**

   Principal Component Analysis (PCA) is a dimensionality reduction technique used to reduce the number of variables in a dataset while retaining the maximum possible variance. PCA identifies the directions (principal components) in which the data varies the most and projects the data onto these directions.

   Mathematical Concepts:

   - **Covariance Matrix:** Measures the variance and the relationship between different variables in the data.
   - **Eigenvectors and Eigenvalues:** Eigenvectors of the covariance matrix point to the directions of maximum variance (principal components), and the corresponding eigenvalues represent the magnitude of variance in these directions.

   Steps in PCA:

   1. **Standardization:** Standardize data to have a mean of 0 and a variance of 1.

2. **Covariance Matrix Computation:** Compute the covariance matrix to understand how the variables of the dataset relate to each other.
3. **Eigen Decomposition:** Compute the eigenvectors and eigenvalues of the covariance matrix.
4. **Feature Vector Formation:** Select the top k eigenvectors corresponding to the largest eigenvalues to form a feature vector.
5. **Projection of Data:** Transform the original dataset using the feature vector to obtain the principal components.

6. **Algorithm / Design / Procedure / Flowchart / Analysis:**

Algorithm:

1. **Standardize the Dataset:**

$$Z = \frac{X - \mu}{\sigma}$$

Z is the standardized data, X is the original data, μ is the mean, and σ is the standard deviation.

2. **Compute the Covariance Matrix:**

$$\Sigma = \frac{1}{n-1} Z^T Z$$

Σ is the covariance matrix.

3. **Calculate Eigenvectors and Eigenvalues:** Solve the equation:

$$\Sigma v = \lambda v$$

λ are the eigenvalues and v are the eigenvectors.

4. **Sort and Select Principal Components :** Choose the top k eigenvectors corresponding to the largest eigenvalues.

5. **Transform the Data:** Project the original data onto the new k-dimensional space:

$$Y = Z \cdot V_k$$

Where V_k is the matrix with the selected eigenvectors.

7. **Procedure:**

1. Load the dataset and standardize it.
2. Calculate the covariance matrix.
3. Perform eigen decomposition on the covariance matrix.
4. Select the top k principal components.
5. Project the data onto these components.
6. Visualize the transformed data in the reduced dimensions.

Consider a dataset with two features. After applying PCA, you might find that 95% of the variance is explained by the first principal component. The original two-dimensional data can be effectively reduced to one dimension, simplifying further analysis.

8. **Results/Output Analysis:**

Present the principal components obtained, the explained variance, and the resulting reduced-dimensional data. Analyze how much of the original variance is retained and how PCA has simplified the dataset.

9. **Conclusions:**

Summarize the effectiveness of PCA in reducing dimensionality while preserving important information. Discuss how PCA can improve the efficiency of machine learning algorithms by removing noise and redundant features.

10. **Viva Questions:**
    1. What is the purpose of PCA?
    2. How do eigenvectors and eigenvalues relate to PCA?
    3. Why is it important to standardize data before applying PCA?
    4. Can PCA be used for non-linear data?

11. **References:**

1. Jolliffe, I. T. (2002). Principal component analysis and factor analysis. *Principal component analysis*, 150-166.
2. Alpaydin, E. (2020). *Introduction to machine learning*. MIT press.
3. Shalev-Shwartz, S., & Ben-David, S. (2014). *Understanding machine learning: From theory to algorithms*. Cambridge university press.