**Principal Component Analysis (PCA) in Machine Learning**

Having too many features in data can cause problems like overfitting (good on training data but poor on new data), slower computation, and lower accuracy. This is called the [curse of dimensionality](https://www.geeksforgeeks.org/videos/curse-of-dimensionality-in-machine-learning/), where more features exponentially increase the data needed for reliable results.

Principal Component Analysis (PCA) is a **dimensionality reduction** technique used in machine learning and statistics. It transforms a dataset with many correlated features into a smaller set of uncorrelated features, called **principal components**, while retaining as much variance as possible.

**Why Use PCA?**

1. **Curse of Dimensionality** – When dealing with high-dimensional data, many algorithms suffer from inefficiency and overfitting.
2. **Feature Reduction** – PCA helps in reducing the number of features while preserving important information.
3. **Noise Reduction** – By removing less significant components, PCA reduces the effect of noise in the dataset.
4. **Data Visualization** – It enables visual representation of high-dimensional data in 2D or 3D.

**How PCA Works?**

PCA follows these steps:

1. **Standardize the Data**
   * Since PCA is affected by scale, we normalize the dataset so that all features have zero mean and unit variance.
2. **Compute the Covariance Matrix**
   * The covariance matrix captures the relationships between different features.
3. **Compute the Eigenvalues and Eigenvectors**
   * Eigenvectors determine the direction of the new feature space (principal components).
   * Eigenvalues represent the amount of variance carried by each principal component.
4. **Select the Top k Principal Components**
   * Choose the number of components (k) that explain most of the variance.
5. **Transform the Data**
   * The original data is projected onto the new feature space defined by the selected principal components.

**Mathematical Representation**

Let x be a dataset with n samples and p features.

1. **Compute Mean and Standardize the Data:**

X′=X−μ

where μ is the mean of each feature.

1. **Compute Covariance Matrix C:**



1. **Compute Eigenvalues λ and Eigenvectors V:**



The eigenvectors define the new axes, and eigenvalues represent the variance captured.

1. **Select k Principal Components**
   * Sort eigenvalues in descending order and select the top k.
2. **Transform the Data:**



where Vk contains the top k eigenvectors.

**Choosing the Right Number of Components**

* The explained variance ratio tells how much variance each principal component retains.
* A common method is to select the number of components where the cumulative explained variance is around **95%**.
* **Scree Plot**: A plot of eigenvalues helps in deciding the number of principal components.

**Advantages of PCA**

* Reduces computational cost.
* Removes multicollinearity among features.
* Enhances visualization in lower dimensions.

**Limitations of PCA**

* May lead to loss of interpretability.
* Assumes that principal components are linear combinations of original features.
* PCA is sensitive to outliers.

**Applications of PCA**

* Image Compression
* Face Recognition
* Feature Extraction in Machine Learning
* Noise Filtering

**Conclusion**

PCA is a powerful tool for dimensionality reduction that balances variance retention and computational efficiency. However, it should be applied carefully, especially when interpretability is essential.