PCA

- As the number of features or dimensions in a dataset increases, the amount of data required to obtain a statistically significant result increases exponentially. This can lead to issues such as overfitting, increased computation time, and reduced accuracy of machine learning models this is known as the curse of dimensionality problems that arise while working with high-dimensional data.
- **Principal Component Analysis (PCA)** is an **unsupervised technique** used in machine learning to **reduce the dimensionality** of a dataset.
- Some **machine learning** algorithms can be sensitive to the number of dimensions, requiring more data to achieve the same level of accuracy as lower-dimensional data.

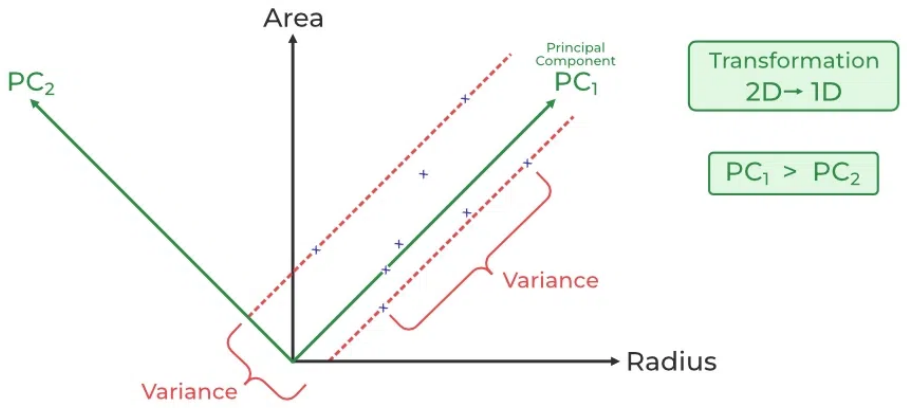
What is Principal Component Analysis (PCA)?

Principal Component Analysis (PCA) technique was introduced by the mathematician **Karl Pearson** in 1901**.** It works on the condition that while the data in a higher dimensional space is mapped to data in a lower dimension space, the variance of the data in the lower dimensional space should be maximum.

- **Principal Component Analysis (PCA)** is a statistical procedure that uses an orthogonal transformation that converts a set of correlated variables to a set of uncorrelated variables.PCA is the most widely used tool in exploratory data analysis and in machine learning for predictive models. Moreover,
- Principal Component Analysis (PCA) is an **unsupervised learning** algorithm technique used to examine the interrelations among a set of variables. It is also known as a general factor analysis where regression determines a line of best fit.
- The main goal of Principal Component Analysis (PCA) is to reduce the dimensionality of a dataset while preserving the most important patterns or

relationships between the variables without any prior knowledge of the target variables.

- Principal Component Analysis (PCA) is used to reduce the dimensionality of a data set by finding a new set of variables, smaller than the original set of variables, retaining most of the sample's information, and useful for the **regression and classification** of data.



1. Principal Component Analysis (PCA) is a technique for dimensionality reduction that identifies a set of orthogonal axes, called principal components, that capture the maximum variance in the data. The principal components are linear combinations of the original variables in the dataset and are ordered in decreasing order of importance. The total variance captured by all the principal components is equal to the total variance in the original dataset.
2. The first principal component captures the most variation in the data, but the second principal component captures the maximum **variance** that is **orthogonal** to the first principal component, and so on.
3. Principal Component Analysis can be used for a variety of purposes, including data visualization, feature selection, and data compression. In data visualization, PCA can be used to plot high-dimensional data in two or three dimensions, making it easier to interpret. In feature selection, PCA can be used to identify the most important variables in a dataset. In data compression, PCA can be used to reduce the size of a dataset without losing important information.

4. In Principal Component Analysis, it is assumed that the information is carried in the variance of the features, that is, the higher the variation in a feature, the more information that features carries.

Step-By-Step Explanation of PCA (Principal Component Analysis)

Step 1: Standardization

First, we need to **standardize** our dataset to ensure that each variable has a mean of 0 and a standard deviation of 1.

Step 2: Covariance Matrix Computation

Covariance measures the strength of joint variability between two or more variables, indicating how much they change in relation to each other.

The value of covariance can be positive, negative, or zeros.

- Positive: As the x1 increases x2 also increases.
- Negative: As the x1 increases x2 also decreases.
- Zeros: No direct relation

Step 3: Compute Eigenvalues and Eigenvectors of Covariance Matrix to Identify Principal Components

---

What is PCA used for?

On its own, PCA is used across a variety of use cases:

1. **Visualize multidimensional data**. Data visualizations are a great tool for communicating multidimensional data as 2- or 3-dimensional plots.
2. **Compress information**. Principal Component Analysis is used to compress information to store and transmit data more efficiently. For example, it can be used to compress images without losing too much quality, or in signal processing. The technique has successfully been applied across a wide range of compression problems in pattern recognition (specifically face recognition), image recognition, and more.
3. **Simplify complex business decisions**. PCA has been employed to simplify traditionally complex business decisions. For example, traders use over 300

financial instruments to manage portfolios. The algorithm has proven successful in the risk management of interest rate derivative portfolios, lowering the number of financial instruments from more than 300 to just 3-4 principal components.

4. **Clarify convoluted scientific processes**. The algorithm has been applied extensively in the understanding of convoluted and multidirectional factors, which increase the probability of neural ensembles to trigger action potentials.

When PCA is used as part of preprocessing, the algorithm is applied to:

1. **Reduce the number of dimensions** in the training dataset.
2. **De-noise** the data. Because PCA is computed by finding the components which explain the greatest amount of variance, it captures the signal in the data and omits the noise.

Advantages of PCA:

1. **Easy to compute**. PCA is based on linear algebra, which is computationally easy to solve by computers.
2. **Speeds up other machine learning algorithms**. Machine learning algorithms converge faster when trained on principal components instead of the original dataset.
3. **Counteracts the issues of high-dimensional data**. High-dimensional data causes regression-based algorithms to overfit easily. By using PCA beforehand to lower the dimensions of the training dataset, we prevent the predictive algorithms from overfitting.
4. **Noise Reduction**: Principal Component Analysis can be used to reduce the noise in data. By removing the principal components with low variance, which are assumed to represent noise, Principal Component Analysis can improve the signal-to-noise ratio and make it easier to identify the underlying structure in the data.
5. **Data Compression**: Principal Component Analysis can be used for data compression. By representing the data using a smaller number of principal components, which capture most of the variation in the data, PCA can reduce the storage requirements and speed up processing.

Disadvantages of PCA:

1. **Low interpretability of principal components**. Principal components are linear combinations of the features from the original data, but they are not as easy to interpret. For example, it is difficult to tell which are the most important features in the dataset after computing principal components.
2. **The trade-off between information loss and dimensionality reduction**. Although dimensionality reduction is useful, it comes at a cost. Information loss is a necessary part of PCA. Balancing the trade-off between dimensionality reduction and information loss is unfortunately a necessary compromise that we have to make when using PCA.
3. **Non-linear Relationships**: Principal Component Analysis assumes that the relationships between variables are linear. However, if there are non-linear relationships between variables, Principal Component Analysis may not work well.
4. **Computational Complexity**: Computing Principal Component Analysis can be computationally expensive for large datasets. This is especially true if the number of variables in the dataset is large.

The assumptions and limitations of PCA

1. **PCA assumes a correlation between features**. If the features (or dimensions or columns, in tabular data) are not correlated, PCA will be unable to determine principal components.
2. **PCA is sensitive to the scale of the features**. Imagine we have two features - one takes values between 0 and 1000, while the other takes values between 0 and 1. PCA will be extremely biased towards the first feature being the first principle component, regardless of the *actual* maximum variance within the data. This is why it's so important to standardize the values first.
3. **PCA is not robust against outliers**. Similar to the point above, the algorithm will be biased in datasets with strong outliers. This is why it is recommended to remove outliers before performing PCA.
4. **PCA assumes a linear relationship between features**. The algorithm is not well suited to capturing non-linear relationships. That's why it's advised to turn non-linear features or relationships between features into linear, using the standard methods such as log transforms.

5. **Technical implementations often assume no missing values**. When computing PCA using statistical software tools, they often assume that the feature set has no missing values (no empty rows). Be sure to remove those rows and/or columns with missing values, or impute missing values with a close approximation