# Pca   (Principal Component Analysis)

dimensionality reduction and machine learning method used to simplify a large data set into a smaller set while still maintaining significant patterns and

1. **Standardize the Data**: PCA begins by standardizing the data, which means scaling it so that each feature has a mean of 0 and a standard deviation of 1. This step is crucial when the features have different units or scales.
2. **Compute the Covariance Matrix**: The covariance matrix captures how the features in the dataset vary with each other. It's a square matrix giving the covariance between each pair of features.
3. **Calculate the Eigenvalues and Eigenvectors**: The eigenvalues and eigenvectors of the covariance matrix are computed. The eigenvectors determine the directions of the new feature space, and the eigenvalues represent the magnitude of the variance in these directions.
4. **Sort Eigenvectors by Eigenvalues**: Eigenvectors are sorted by their corresponding eigenvalues in descending order. This step helps in identifying which components (eigenvectors) explain the most variance in the data.
5. **Select Principal Components**: The top k eigenvectors (where k is the number of dimensions you want to keep) are selected. These eigenvectors form a new feature space.
6. **Transform the Data**: The original data is projected onto the new feature space defined by the selected eigenvectors. This results in a reduced-dimensionality representation of the data.

# data warehouse vs database

smaller number of more complex queries over multiple large data stores.

- Databases are structured as efficiently as possible, with no duplicate information in multiple tables. Data warehouse information is typically denormalized, prioritizing read operations ahead of write operations
- **Databases** are optimized for handling real-time transactional data, supporting daily operations with high efficiency for read/write tasks.
- **data Warehouses** are optimized for analyzing large volumes of historical data, supporting complex queries, reporting, and data analysis for business intelligence purposes.

# How to show missing values ?

## technic

1. MCAR - Missing completely at random

This happens if all the variables and observations have the same probability of being missing. Imagine providing a child with Lego of different colors to build a house. Each Lego represents a piece of information, like shape and color. The child might lose some Legos during the game. These lost legos represent missing information, just like when they can't remember the shape or the color of the Lego they had. That information was lost randomly, but they do not change the information the child has on the other Legos.

2. MAR - Missing at random

   For MAR, the probability of the value being missing is related to the value of the variable or other variables in the dataset. This means that not all the observations and variables have the same chance of being missing. An example of MAR is a survey in the Data community where data scientists who do not frequently upgrade their skills are more likely not to be aware of new state-of-the-art algorithms or technologies, hence skipping certain questions. The missing data, in this case, is related to how frequently the data scientist upskills.

3. MNAR- Missing not at random

MNAR is considered to be the most difficult scenario among the three types of missing data. It is applied when neither MAR nor MCAR apply. In this situation, the probability of being missing is completely different for different values of the same variable, and these reasons can be unknown to us. An example of MNAR is a survey about married couples. Couples with a bad relationship might not want to answer certain questions as they might feel embarrassed to do so.

# Function

Isnull()

Notnull()

Info()

Isna()

# Normalization and standardization

Standardization centers data around a mean of zero and a standard deviation of one.

normalization scales data to a set range, often [0, 1], by using the minimum and maximum values, out lyres is abscale