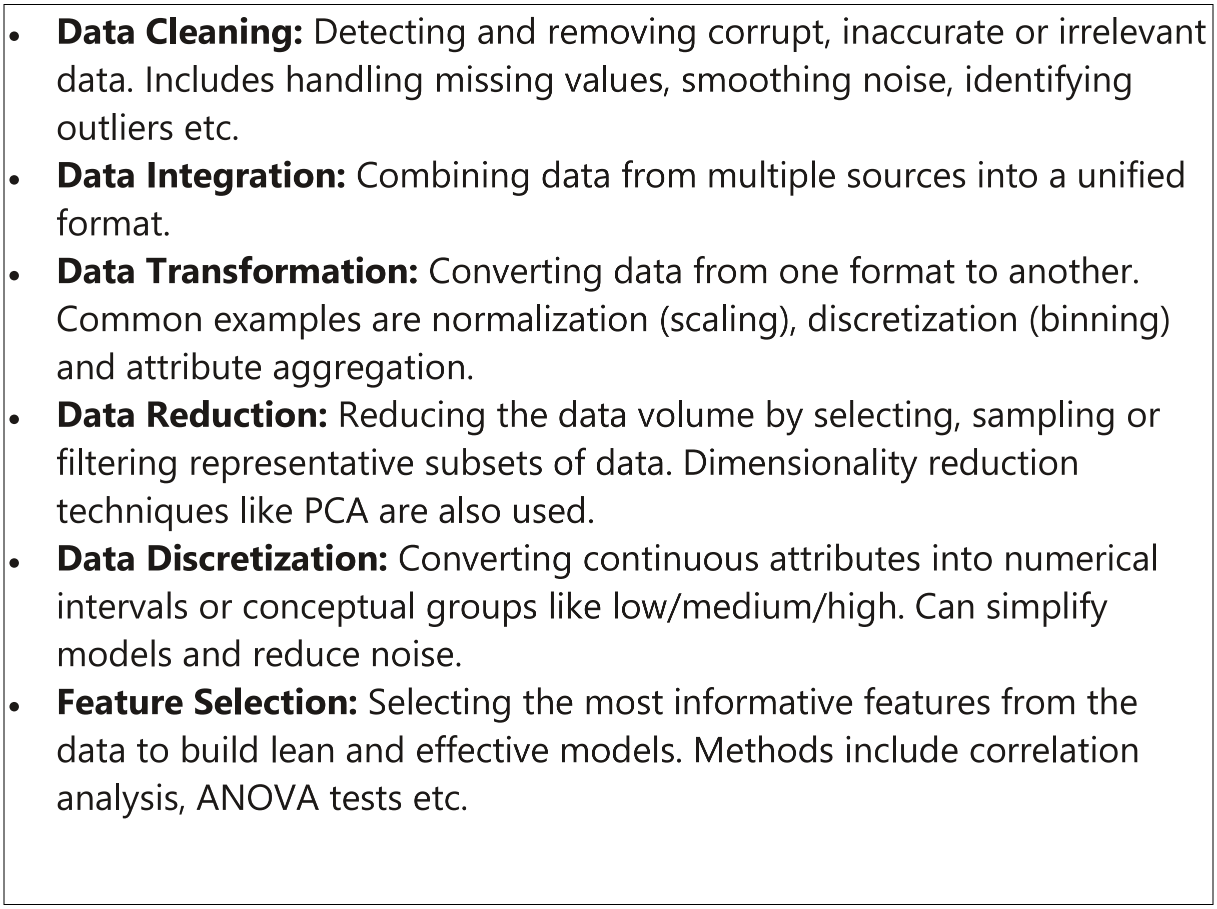
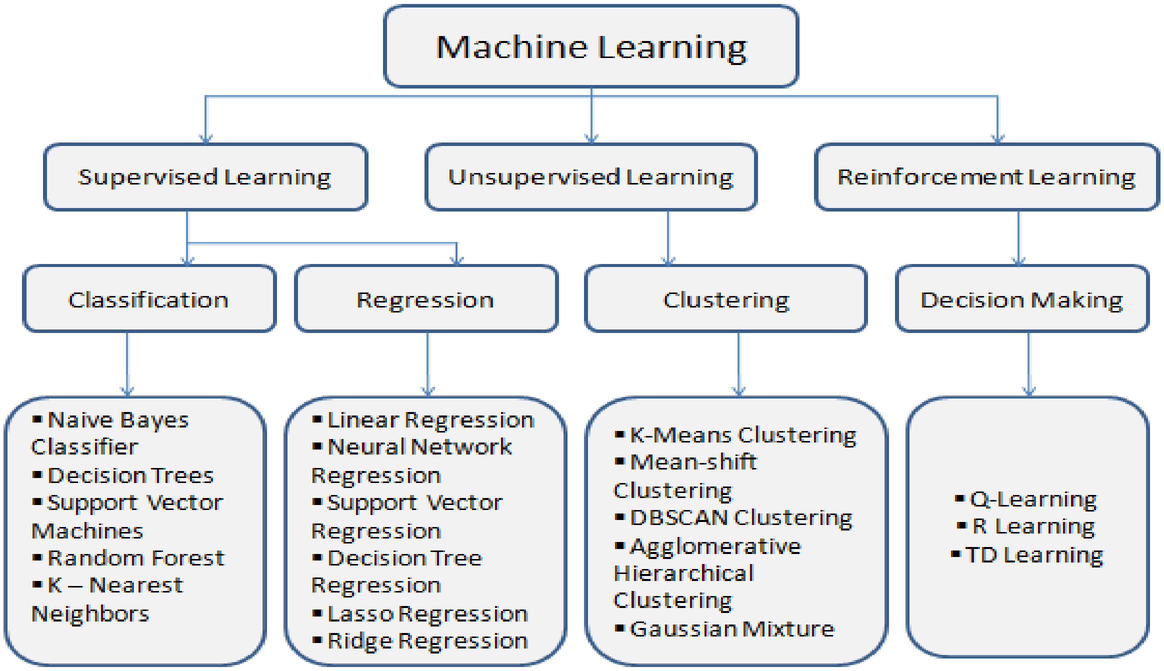
ML

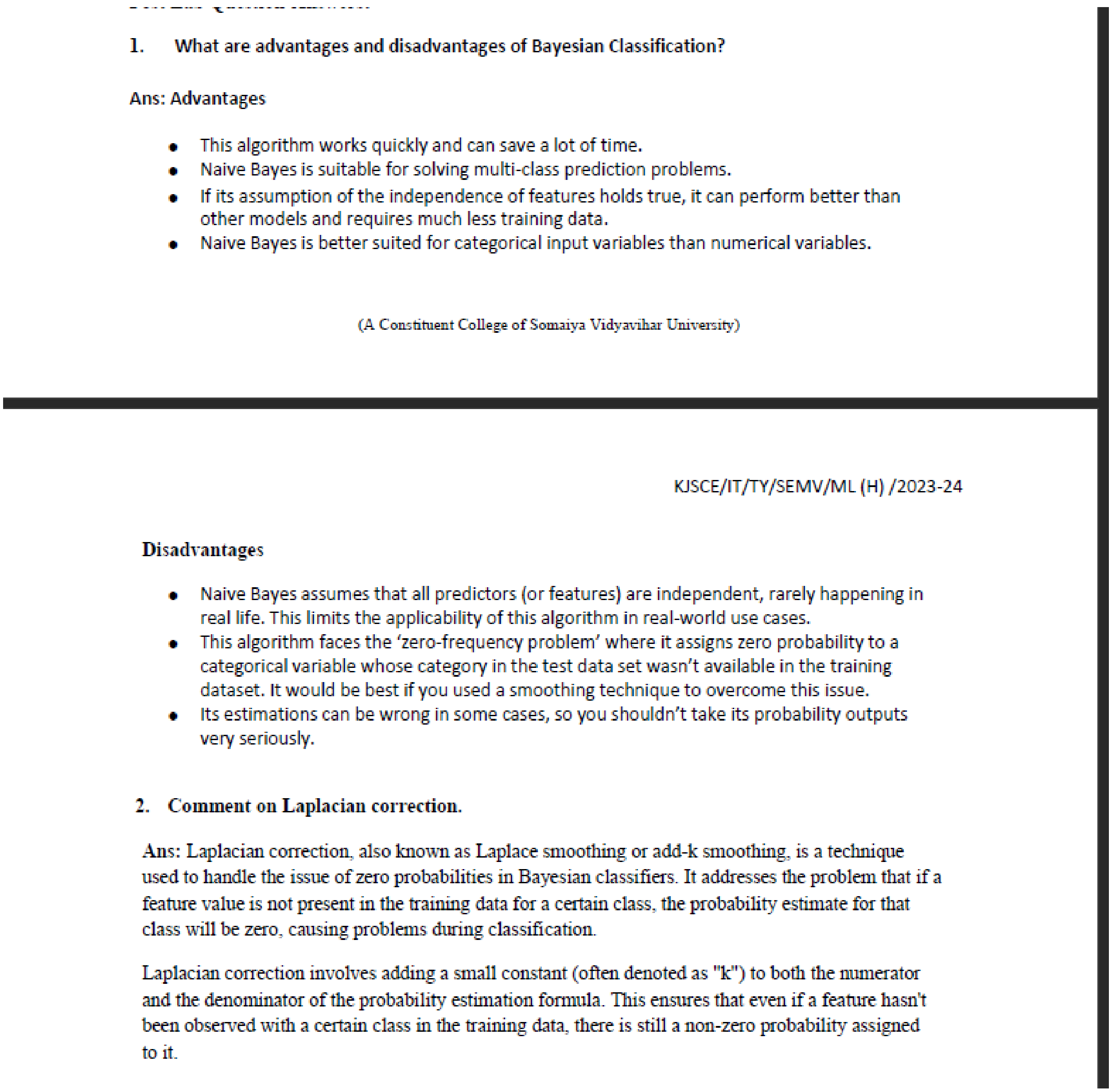
Data pre-processing techniques:



- **Data Cleaning:** Detecting and removing corrupt, inaccurate or irrelevant data. Includes handling missing values, smoothing noise, identifying outliers etc.
- **Data Integration:** Combining data from multiple sources into a unified format.
- **Data Transformation:** Converting data from one format to another. Common examples are normalization (scaling), discretization (binning) and attribute aggregation.
- **Data Reduction:** Reducing the data volume by selecting, sampling or filtering representative subsets of data. Dimensionality reduction techniques like PCA are also used.
- **Data Discretization:** Converting continuous attributes into numerical intervals or conceptual groups like low/medium/high. Can simplify models and reduce noise.
- **Feature Selection:** Selecting the most informative features from the data to build lean and effective models. Methods include correlation analysis, ANOVA tests etc.

In Bayesian statistics, a maximum a posteriori probability (MAP) estimate is an estimate of an unknown quantity, that equals the mode of the posterior distribution. The MAP can be used to obtain a point estimate of an unobserved quantity on the basis of empirical data.


Machine Learning
Supervised Learning
Unsupervised Learning
Reinforcement Learning
Classification
Regression
Clustering
Decision Making
▪ Naive Bayes Classifier
▪ Decision Trees
▪ Support Vector Machines
▪ Random Forest
▪ K – Nearest Neighbors
▪ Linear Regression
▪ Neural Network Regression
▪ Support Vector Regression
▪ Decision Tree Regression
▪ Lasso Regression
▪ Ridge Regression
▪ K-Means Clustering
▪ Mean-shift Clustering
▪ DBSCAN Clustering
▪ Agglomerative Hierarchical Clustering
▪ Gaussian Mixture
▪ Q-Learning
▪ R Learning
▪ TD Learning



1. **What are advantages and disadvantages of Bayesian Classification?**

Ans: Advantages

- This algorithm works quickly and can save a lot of time.
- Naive Bayes is suitable for solving multi-class prediction problems.
- If its assumption of the independence of features holds true, it can perform better than other models and requires much less training data.
- Naive Bayes is better suited for categorical input variables than numerical variables.

(A Constituent College of Somaiya Vidyavihar University)

KJSCE/IT/TY/SEMV/ML (H) /2023-24

Disadvantages

- Naive Bayes assumes that all predictors (or features) are independent, rarely happening in real life. This limits the applicability of this algorithm in real-world use cases.
- This algorithm faces the 'zero-frequency problem' where it assigns zero probability to a categorical variable whose category in the test data set wasn't available in the training dataset. It would be best if you used a smoothing technique to overcome this issue.
- Its estimations can be wrong in some cases, so you shouldn't take its probability outputs very seriously.

2. **Comment on Laplacian correction.**

Ans: Laplacian correction, also known as Laplace smoothing or add-k smoothing, is a technique used to handle the issue of zero probabilities in Bayesian classifiers. It addresses the problem that if a feature value is not present in the training data for a certain class, the probability estimate for that class will be zero, causing problems during classification.

Laplacian correction involves adding a small constant (often denoted as "k") to both the numerator and the denominator of the probability estimation formula. This ensures that even if a feature hasn't been observed with a certain class in the training data, there is still a non-zero probability assigned to it.

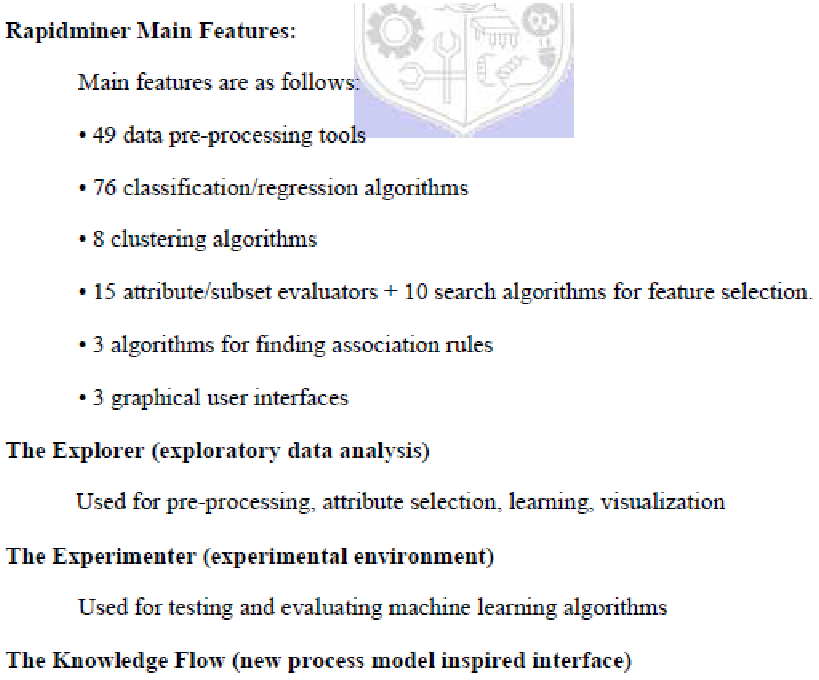
Decision trees:

- Decision trees model data as a tree structure, where each node represents a feature test on an attribute, each branch represents an outcome of the test, and leaves represent the final classification.
- To build a tree, recursive partitioning is used to split the training data into subsets based on the attribute values.

KNN-Algorithm:

It is a type of instance-based learning, where the algorithm makes predictions based on the majority class or average value of the k-nearest data points in the feature space.

Lazy Learner firstly stores the training dataset and wait until it receives the test dataset.



Rapidminer Main Features:

Main features are as follows:

- 49 data pre-processing tools
- 76 classification/regression algorithms
- 8 clustering algorithms
- 15 attribute/subset evaluators + 10 search algorithms for feature selection.
- 3 algorithms for finding association rules
- 3 graphical user interfaces

The Explorer (exploratory data analysis)

Used for pre-processing, attribute selection, learning, visualization

The Experimenter (experimental environment)

Used for testing and evaluating machine learning algorithms

The Knowledge Flow (new process model inspired interface)

Target variable -- The “target variable” is the variable whose values are to be modeled and predicted by other variables.

K-means Clustering Algorithm:

K-Means clustering is a popular unsupervised machine learning algorithm used for partitioning a dataset into distinct, non-overlapping subgroups or clusters. The primary goal of K-Means is to group similar data points together and discover underlying patterns in the data.

K-value :(techniques)

Elbow Method:

The Elbow Method involves running K-Means clustering on the dataset for a range of K values and then plotting the sum of squared distances from each point to its assigned center (inertia or distortion) against the number of clusters.

The "elbow" in the plot represents a point where adding more clusters does not significantly reduce the sum of squared distances. The K value corresponding to this elbow is often considered as a reasonable choice.

Silhouette Score:

The Silhouette Score measures how well-separated the clusters are. It assigns a score to each data point based on the distance between the point and the nearest neighboring cluster, compared to the distance between the point and its own cluster.

Support Vector Machine:

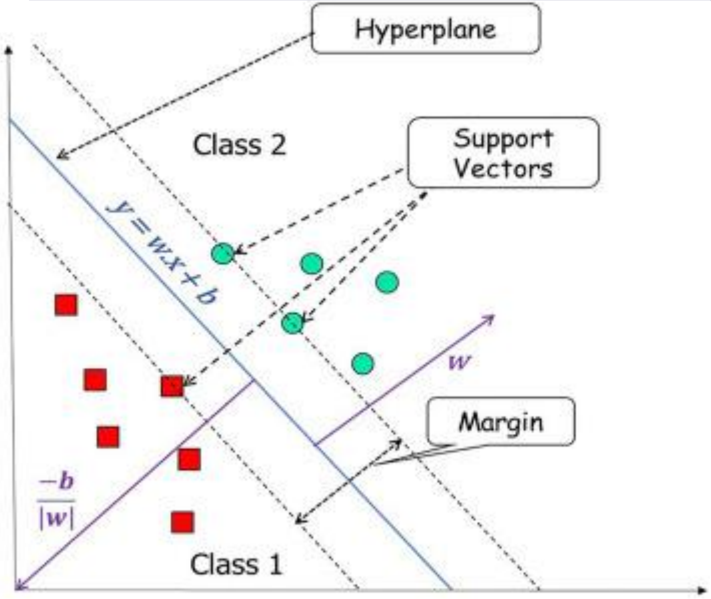
SVM is particularly effective in high-dimensional spaces and is widely used in various applications, including image classification, text classification, and bioinformatics. The primary objective of SVM is to find a hyperplane that best separates data points of different classes.

Margin:

- The margin is the distance between the hyperplane and the nearest data point from either class.

Support Vectors:

- Support vectors are the data points that are closest to the hyperplane and influence the position and orientation of the hyperplane.



Kernel:

Kernel Selection: Choose a kernel function to transform the data into a higher-dimensional space. SVM uses the kernel trick to find non-linear decision boundaries. Commonly used kernels are:
Linear Kernel: For linearly separable data (the default kernel when no other kernel is specified).
Polynomial Kernel: Suitable for data with polynomial patterns.
Radial Basis Function (RBF) Kernel (Gaussian Kernel): Suitable for non-linear data and is the most popular kernel choice.
Sigmoid Kernel: Useful for neural network-inspired architectures.
Parameter Selection: SVM has two crucial parameters to be set:
C (Cost parameter): Controls the trade-off between maximizing the margin and minimizing the classification error on the training data. A smaller C allows for a wider margin but may result in misclassifying some training examples, while a larger C aims to classify all examples correctly, potentially leading to a narrower margin.

Overfitting:

Overfitting is a common problem in machine learning that occurs when a model learns the training data too well, to the point that it begins to capture noise or random fluctuations in the data rather than the underlying patterns. This leads to a model that performs very well on the training data but fails to generalize effectively to unseen or new data. Overfitting is often considered the opposite of underfitting, where the model is too simple to capture the underlying patterns in the data.

What are advantages and disadvantages of SVM?

Ans. Sensitivity to Scaling: SVMs are sensitive to the scaling of input features. It's essential to

preprocess data properly to ensure that all features are on similar scales. Computational Intensity:

Training an SVM can be computationally intensive, especially for large datasets. The complexity of the training process is O(n^2) or O(n^3), depending on the kernel used, where 'n' is the number of data points. Difficulty in Tuning: Choosing the right kernel and setting appropriate hyperparameters can be challenging, and the performance of SVMs is highly dependent on these choices. Binary Classification: SVMs are inherently binary classifiers. Extending them to multi-class classification requires using techniques like one-vs-all or one-vs-one, which can lead to increased complexity.

ID3(Decision trees Algorithm)

- ID3 can overfit the training data (to avoid overfitting, smaller decision trees should be preferred over larger ones).

- This algorithm usually produces small trees, but it does not always produce the smallest possible tree.

- ID3 is harder to use on continuous data

-

Decision Trees:

- Decision Trees are prone to overfitting, especially when they are deep and complex.
- A deep Decision Tree can memorize the training data, capturing noise in the data instead of the underlying patterns.
- Overfitting occurs when the tree is too specific to the training data and does not generalize well to unseen data.

ID3:

- ID3 is a specific algorithm used to create Decision Trees.
- It constructs trees using a top- down, greedy approach.
- ID3 tends to overfit more because it creates a tree by selecting the attribute that provides the best information gain at each step.
- This means it can create branches for noise in the data, especially if some attributes have a large number of possible values (high cardinality).
- ID3 does not have mechanisms like pruning, which are

Pruning in the context of Decision Trees refers to the process of cutting off some branches (subtrees) from the tree, i.e., removing parts of the tree that do not provide significant predictive power. Pruning is important in Decision Tree algorithms for several reasons:

Regression:

Machine Learning Regression is a technique for investigating the relationship between independent variables or features and a dependent variable or outcome. It's used as a method for predictive modelling in machine learning, in which an algorithm is used to predict continuous outcomes.

Aspect	Linear Regression	Nonlinear Regression
Mathematical	Y = β0 + β1X + ε	Y = f(β0, β1, β2, ..., βn, X) + ε
Linearity Assumption	Assumes a linear relationship between the predictor(s) and the response variable.	Does not assume a linear relationship; the relationship can be any functional form.
Parameter Estimation	Typically involves estimating coefficients (β0, β1, etc.) using methods like Ordinary Least Squares (OLS).	Involves estimating the parameters (β0, β1, β2, etc.) of a chosen nonlinear function, which may require more complex optimization techniques.
Model Complexity	Generally simpler models with fewer parameters.	Can involve complex and highly flexible functional forms, allowing for a better fit to data but potentially leading to overfitting.
Model Interpretability	Coefficients (β0, β1, etc.) are interpretable as the change in the response variable associated with a unit change in the predictor(s).	Parameter interpretation can be challenging, as the relationship is nonlinear and may not have simple interpretations.
Residual Analysis	Linear regression often assumes homoscedasticity (constant variance of errors) and independence of residuals.	Residuals may not exhibit homoscedasticity or independence, and model assumptions may need to be checked carefully.
Use Cases	Well-suited for modeling linear relationships, such as simple trends, correlations, and linear patterns.	Appropriate for modeling more complex, nonlinear relationships, such as exponential growth, sigmoidal curves, or polynomial functions.
Assumption Violations	Sensitivity to violations of linearity and other assumptions, which can lead to biased estimates and unreliable predictions.	More robust to violations of linearity, but the choice of the nonlinear function should be guided by domain knowledge.
Model Selection	Typically involves selecting the predictor(s) and checking for multicollinearity.	Involves selecting both the functional form and the predictor(s), which can be more challenging.
Examples	Simple linear regression, multiple linear regression.	Logistic regression, exponential growth models, polynomial regression.

Bias-Variance Trade-off

$$E(MSE) = noise^2 + bias^2 + variance$$

Unavoidable error

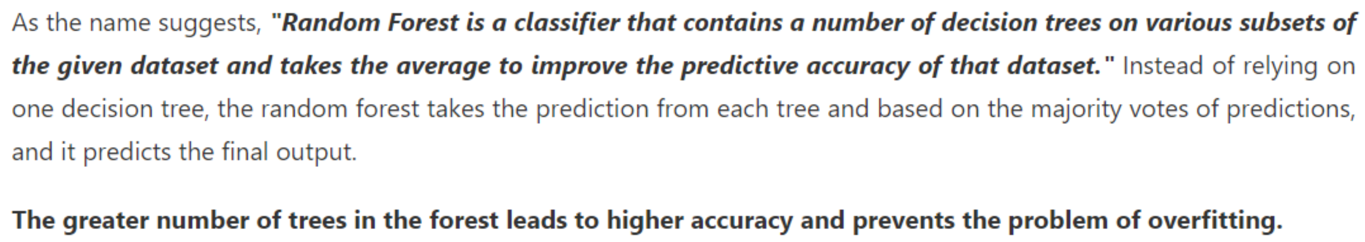
Error due to incorrect assumptions

Error due to variance of training samples

Issues (2): Evaluating Classification Methods

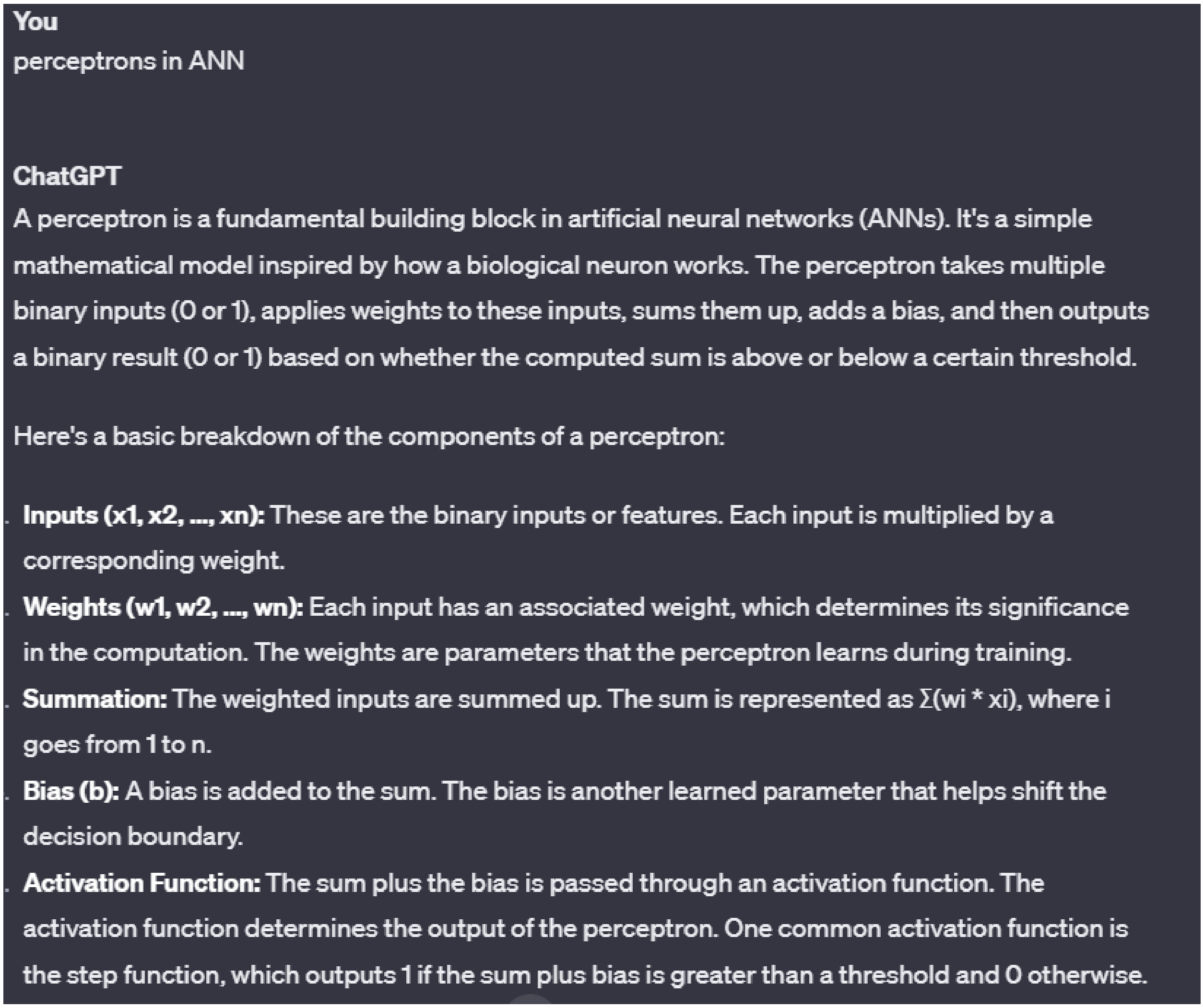
- Predictive accuracy
- Speed and scalability
  - time to construct the model
  - time to use the model
- Robustness
  - handling noise and missing values
- Scalability
  - efficiency in disk-resident databases
- Interpretability:
  - understanding and insight provded by the model
- Goodness of rules
  - decision tree size
  - compactness of classification rules

Random Forest Classifier:



As the name suggests, ***"Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset."*** Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.



You

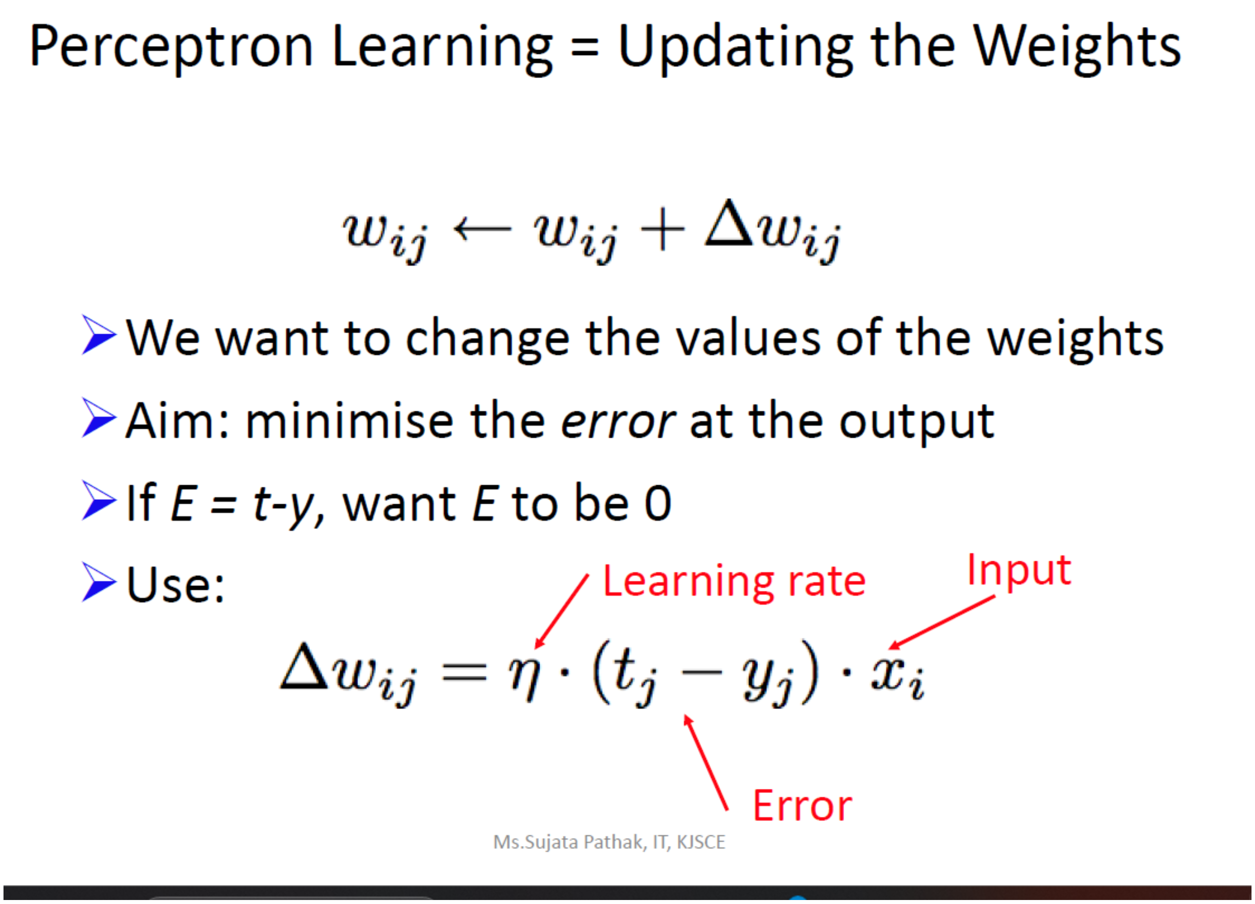
perceptrons in ANN

ChatGPT

A perceptron is a fundamental building block in artificial neural networks (ANNs). It's a simple mathematical model inspired by how a biological neuron works. The perceptron takes multiple binary inputs (0 or 1), applies weights to these inputs, sums them up, adds a bias, and then outputs a binary result (0 or 1) based on whether the computed sum is above or below a certain threshold.

Here's a basic breakdown of the components of a perceptron:

- **Inputs (x1, x2, ..., xn):** These are the binary inputs or features. Each input is multiplied by a corresponding weight.
- **Weights (w1, w2, ..., wn):** Each input has an associated weight, which determines its significance in the computation. The weights are parameters that the perceptron learns during training.
- **Summation:** The weighted inputs are summed up. The sum is represented as Σ(wi * xi), where i goes from 1 to n.
- **Bias (b):** A bias is added to the sum. The bias is another learned parameter that helps shift the decision boundary.
- **Activation Function:** The sum plus the bias is passed through an activation function. The activation function determines the output of the perceptron. One common activation function is the step function, which outputs 1 if the sum plus bias is greater than a threshold and 0 otherwise.



Perceptron Learning = Updating the Weights

$$w_{ij} \leftarrow w_{ij} + \Delta w_{ij}$$

- We want to change the values of the weights
- Aim: minimise the *error* at the output
- If $E = t-y$, want E to be 0
- Use:

Learning rate Input

$$\Delta w_{ij} = \eta \cdot (t_j - y_j) \cdot x_i$$

Error

2.3 Linear Separability

- Outputs are:

$$y_j = \mathrm{sign}\left(\sum_{i=1}^{n} w_{ij} x_i\right)$$

$$\Rightarrow \mathbf{w} \cdot \mathbf{x} > 0$$

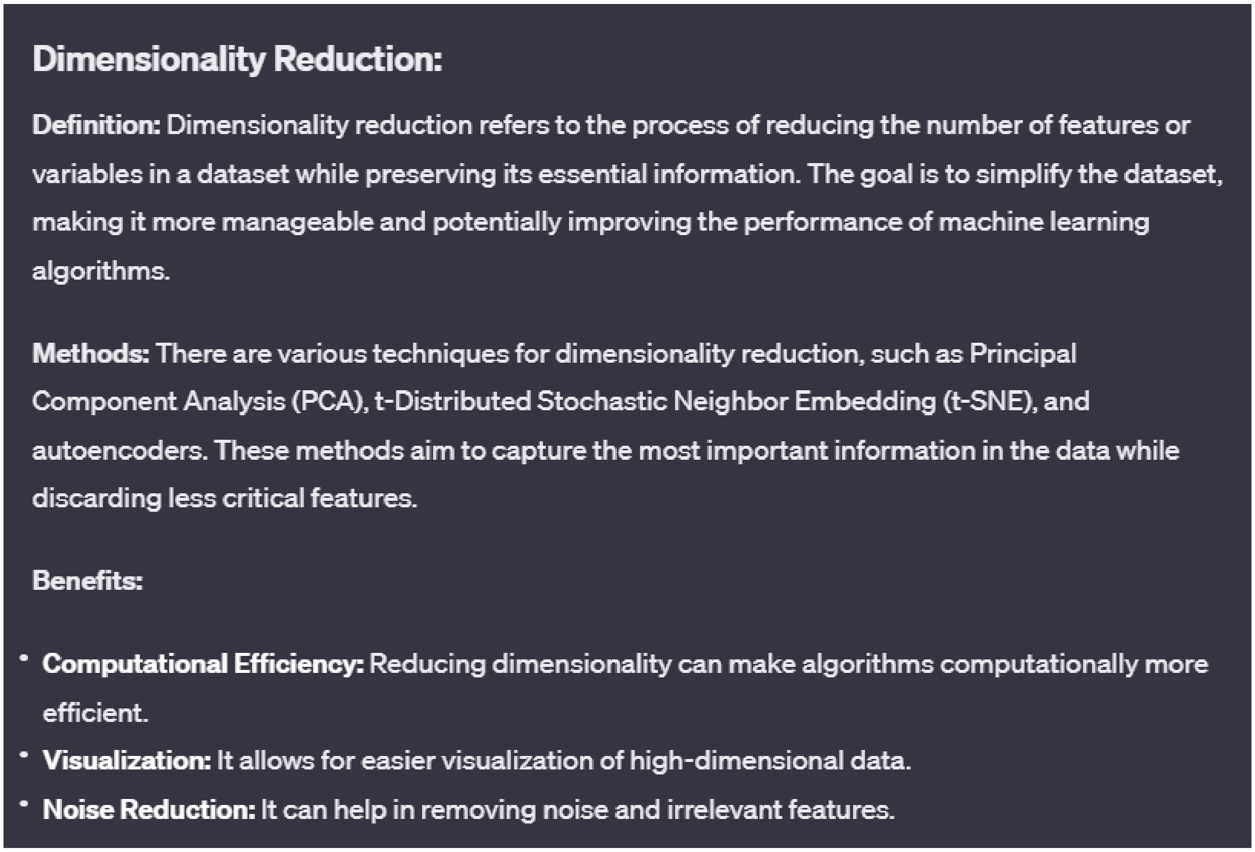
where

$$\mathbf{w} \cdot \mathbf{x} = \|\mathbf{w}\| \times \|\mathbf{x}\| \cos \alpha$$

and α is the angle between vectors **x** and **w**.

Hebb’s Rule

- Hebb’s rule says that-
  - “Changes in the strength of synaptic connections are proportional to the correlation in the firing of the two connecting neurons.
  - So if two neurons consistently fire simultaneously, then any connection between them will change in strength, becoming stronger.
  - However, if the two neurons never fire simultaneously, the connection between them will die away.



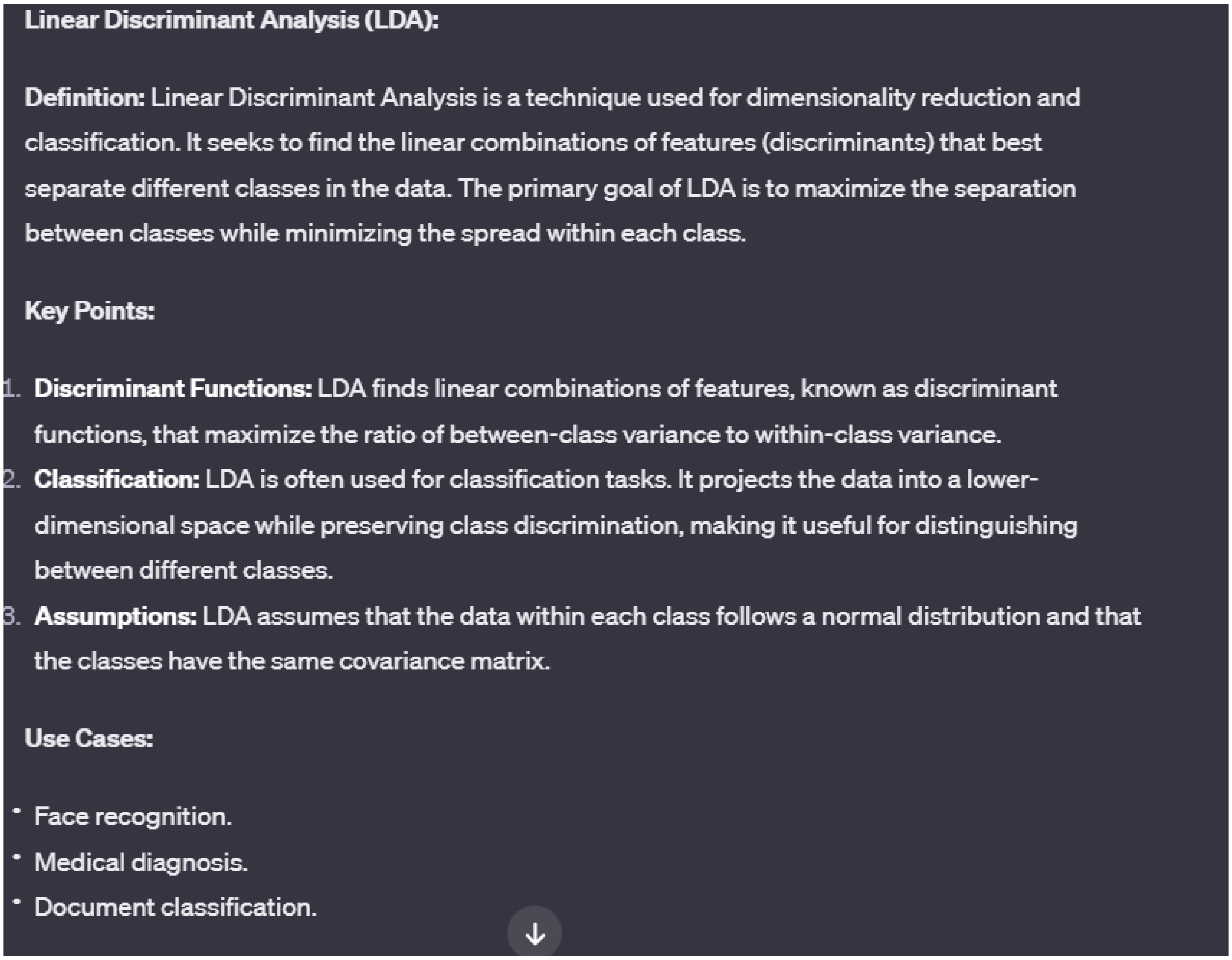
Dimensionality Reduction:

Definition: Dimensionality reduction refers to the process of reducing the number of features or variables in a dataset while preserving its essential information. The goal is to simplify the dataset, making it more manageable and potentially improving the performance of machine learning algorithms.

Methods: There are various techniques for dimensionality reduction, such as Principal Component Analysis (PCA), t-Distributed Stochastic Neighbor Embedding (t-SNE), and autoencoders. These methods aim to capture the most important information in the data while discarding less critical features.

Benefits:

- **Computational Efficiency:** Reducing dimensionality can make algorithms computationally more efficient.
- **Visualization:** It allows for easier visualization of high-dimensional data.
- **Noise Reduction:** It can help in removing noise and irrelevant features.



Linear Discriminant Analysis (LDA):

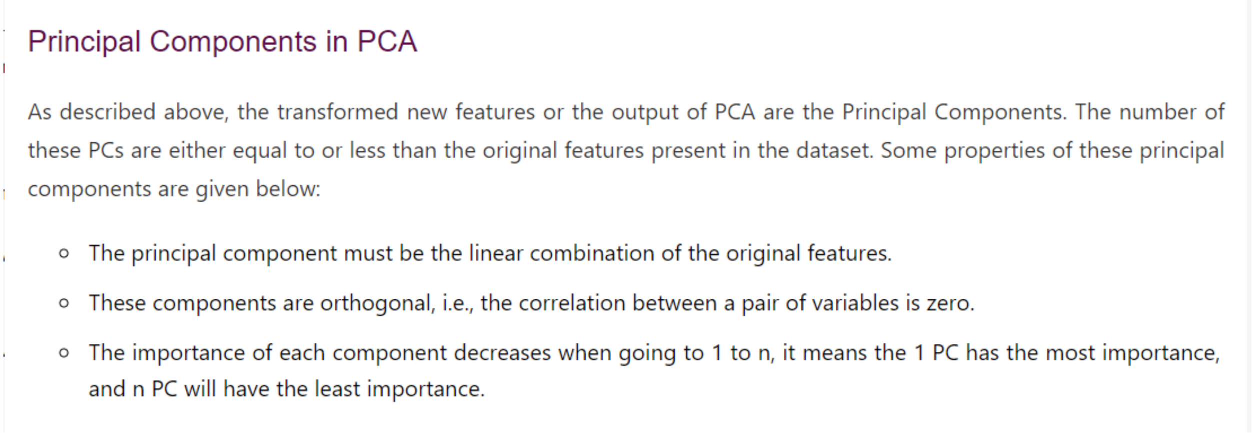
Definition: Linear Discriminant Analysis is a technique used for dimensionality reduction and classification. It seeks to find the linear combinations of features (discriminants) that best separate different classes in the data. The primary goal of LDA is to maximize the separation between classes while minimizing the spread within each class.

Key Points:

1. **Discriminant Functions:** LDA finds linear combinations of features, known as discriminant functions, that maximize the ratio of between-class variance to within-class variance.
2. **Classification:** LDA is often used for classification tasks. It projects the data into a lower-dimensional space while preserving class discrimination, making it useful for distinguishing between different classes.
3. **Assumptions:** LDA assumes that the data within each class follows a normal distribution and that the classes have the same covariance matrix.

Use Cases:

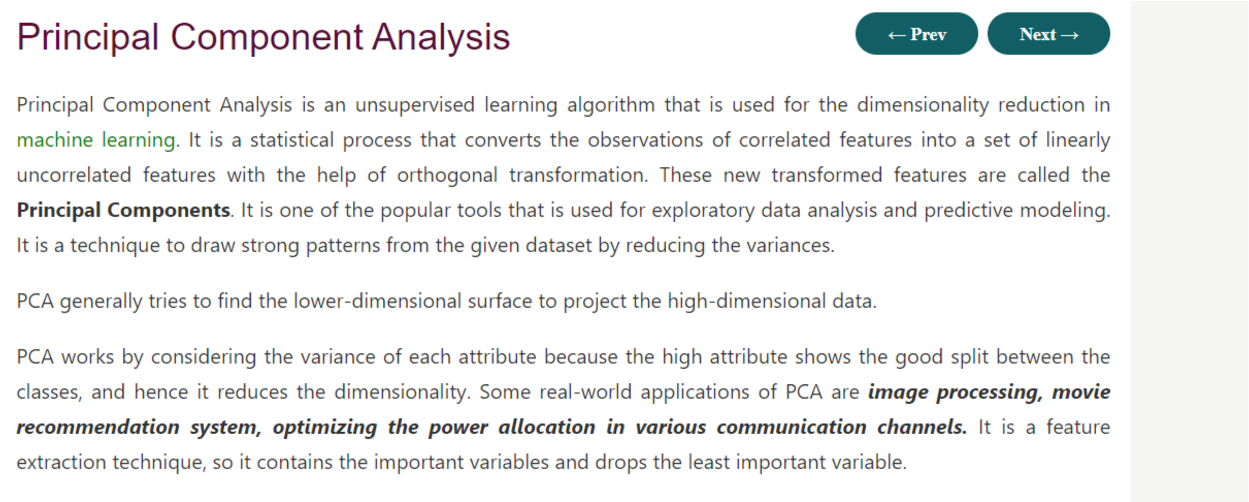
- Face recognition.
- Medical diagnosis.
- Document classification.



Principal Components in PCA

As described above, the transformed new features or the output of PCA are the Principal Components. The number of these PCs are either equal to or less than the original features present in the dataset. Some properties of these principal components are given below:

- The principal component must be the linear combination of the original features.
- These components are orthogonal, i.e., the correlation between a pair of variables is zero.
- The importance of each component decreases when going to 1 to n, it means the 1 PC has the most importance, and n PC will have the least importance.

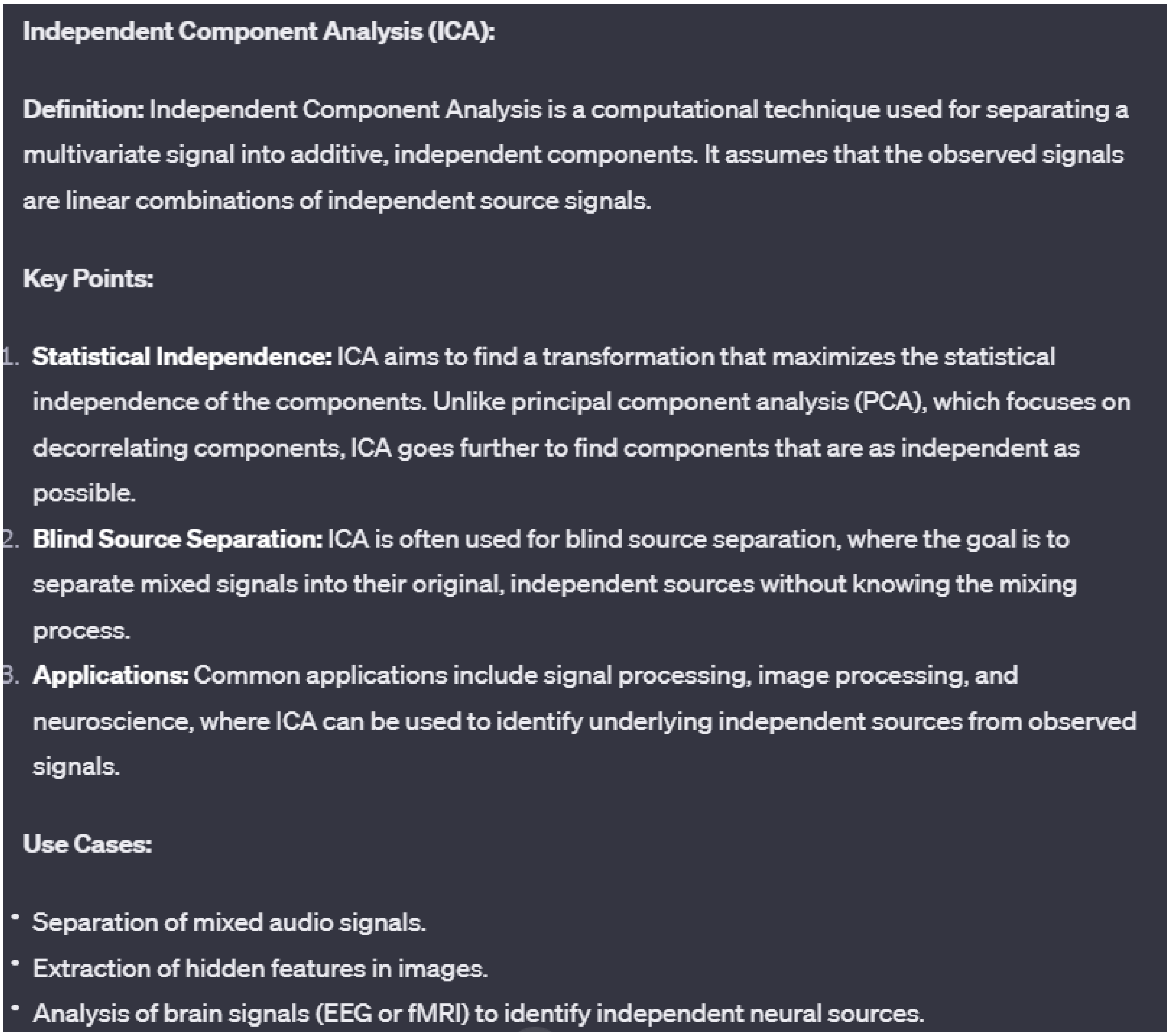


Principal Component Analysis

Principal Component Analysis is an unsupervised learning algorithm that is used for the dimensionality reduction in machine learning. It is a statistical process that converts the observations of correlated features into a set of linearly uncorrelated features with the help of orthogonal transformation. These new transformed features are called the **Principal Components**. It is one of the popular tools that is used for exploratory data analysis and predictive modeling. It is a technique to draw strong patterns from the given dataset by reducing the variances.

PCA generally tries to find the lower-dimensional surface to project the high-dimensional data.

PCA works by considering the variance of each attribute because the high attribute shows the good split between the classes, and hence it reduces the dimensionality. Some real-world applications of PCA are ***image processing, movie recommendation system, optimizing the power allocation in various communication channels.*** It is a feature extraction technique, so it contains the important variables and drops the least important variable.



Independent Component Analysis (ICA):

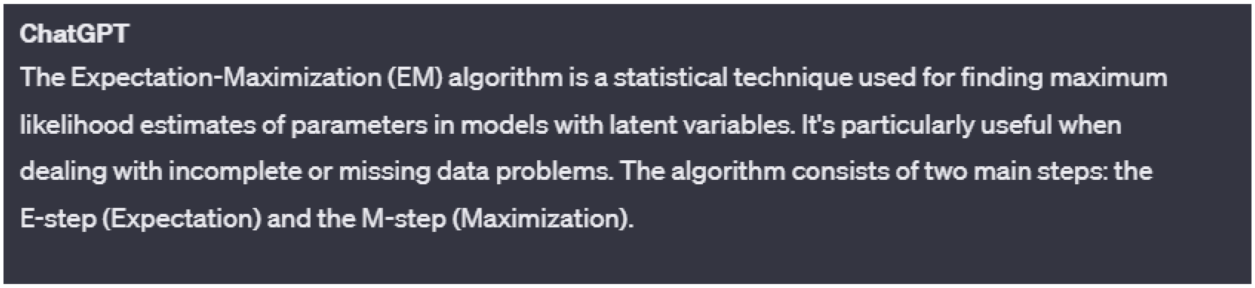
Definition: Independent Component Analysis is a computational technique used for separating a multivariate signal into additive, independent components. It assumes that the observed signals are linear combinations of independent source signals.

Key Points:

1. **Statistical Independence:** ICA aims to find a transformation that maximizes the statistical independence of the components. Unlike principal component analysis (PCA), which focuses on decorrelating components, ICA goes further to find components that are as independent as possible.
2. **Blind Source Separation:** ICA is often used for blind source separation, where the goal is to separate mixed signals into their original, independent sources without knowing the mixing process.
3. **Applications:** Common applications include signal processing, image processing, and neuroscience, where ICA can be used to identify underlying independent sources from observed signals.

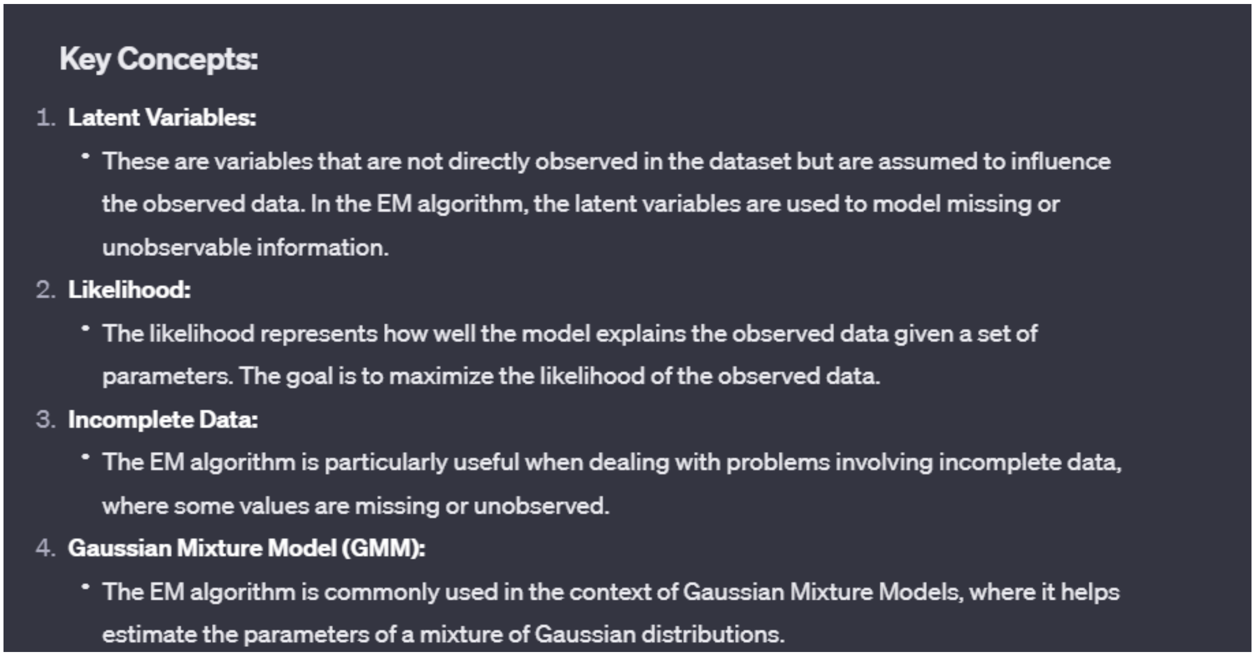
Use Cases:

- Separation of mixed audio signals.
- Extraction of hidden features in images.
- Analysis of brain signals (EEG or fMRI) to identify independent neural sources.



ChatGPT

The Expectation-Maximization (EM) algorithm is a statistical technique used for finding maximum likelihood estimates of parameters in models with latent variables. It's particularly useful when dealing with incomplete or missing data problems. The algorithm consists of two main steps: the E-step (Expectation) and the M-step (Maximization).



Key Concepts:

1. **Latent Variables:**
   - These are variables that are not directly observed in the dataset but are assumed to influence the observed data. In the EM algorithm, the latent variables are used to model missing or unobservable information.
2. **Likelihood:**
   - The likelihood represents how well the model explains the observed data given a set of parameters. The goal is to maximize the likelihood of the observed data.
3. **Incomplete Data:**
   - The EM algorithm is particularly useful when dealing with problems involving incomplete data, where some values are missing or unobserved.
4. **Gaussian Mixture Model (GMM):**
   - The EM algorithm is commonly used in the context of Gaussian Mixture Models, where it helps estimate the parameters of a mixture of Gaussian distributions.

Nearest Neighbor methods are a family of algorithms used for classification and regression tasks. These algorithms are based on the idea that instances of the same class or with similar target values tend to be close to each other in the feature space. The two main types of nearest neighbor methods are k-Nearest Neighbors (k-NN) for classification and regression problems. Here's an overview of each: