

INTRODUCTION TO MACHINE LEARNING (CSL2010)

COURSE PROJECT

Group Members:


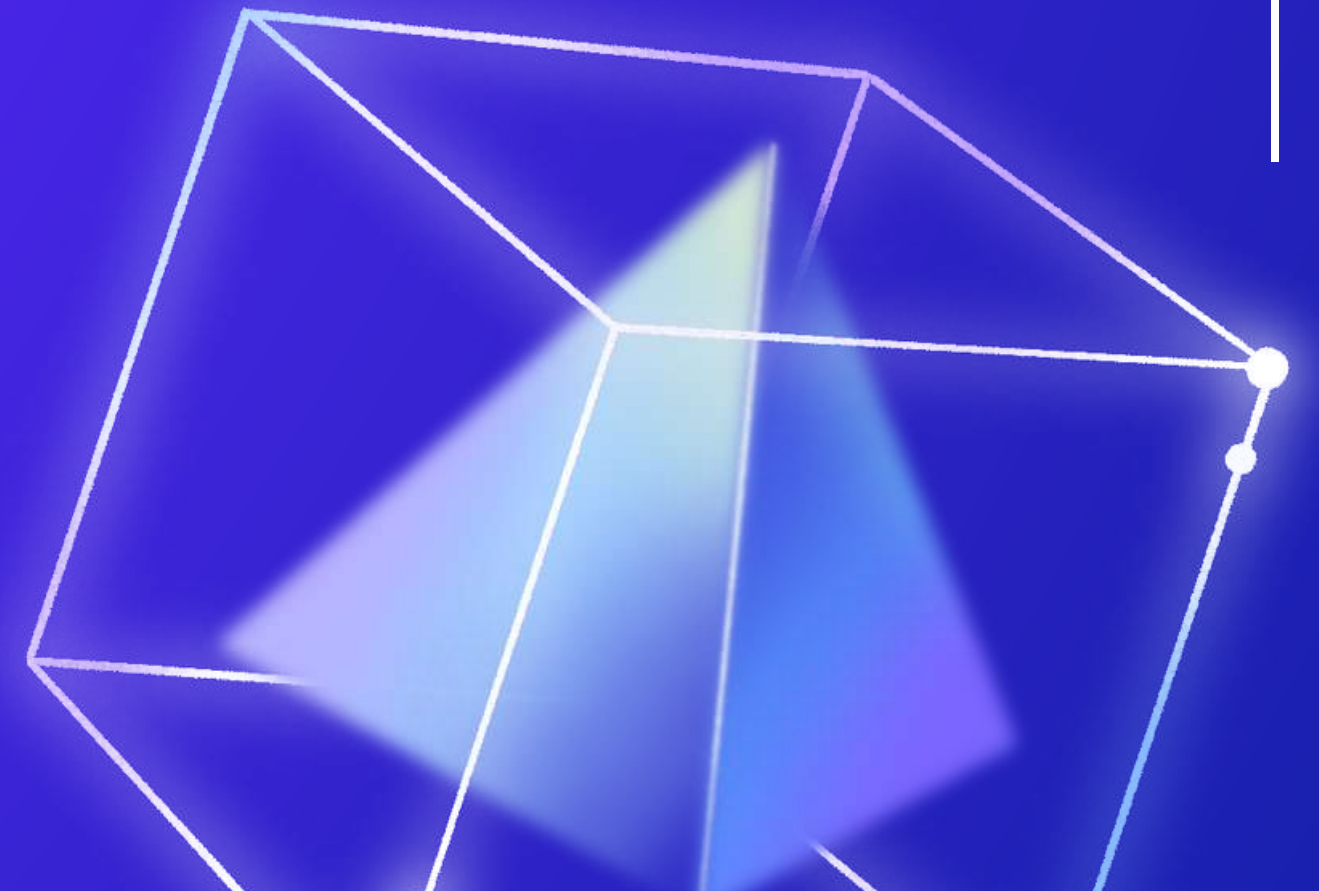
- Ishaan Pandey (B22CI017)
 - G Dileep (B22CH011)
 - Siddhartha Singh (B22BB041)
 - Yash Kumar (B22MT047)
 - Deekshant Singh Rajawat (B22CI010)
- 





TABLE OF CONTENTS

• Introduction	01
◦ Background	02
◦ Objective	03
• Theory	04
• Methodology	05
• Model Explanation	06
◦ Hyper Parameter Tuning	07
◦ Accuracy Scores	
• Conclusion	



INTRODUCTION

Background of Alzheimer's disease

Alzheimer's disease is responsible for gradual decline in cognitive function, memory loss, and changes in behaviour. People suffering from Alzheimer's have profound impacts on the lives and their families.. Early diagnosis is crucial for effective intervention.

Objectives of our project

This project aims to explore the application of machine learning, specifically analysis of handwriting features, for the detection of Alzheimer's disease. We seek to create a tool that can aid in the early diagnosis of Alzheimer's, allowing for timely intervention and personalized care.



MACHINE LEARNING

Automation

In the realm of Alzheimer's disease detection, automation plays a crucial role in efficiently collecting handwriting samples for analysis. Automated tools can streamline the process of gathering a diverse range of handwriting samples, ensuring a comprehensive dataset for training machine learning algorithms.

Algorithm

The core of this project lies in the development of robust machine learning algorithms capable of extracting meaningful features from handwriting samples. Researchers are focused on designing algorithms that can discern subtle patterns and anomalies in the way individuals write, with a specific emphasis on characteristics associated with early-stage Alzheimer's disease.



BENIFIT OF ML IN ALZIEMER

01

Machine learning algorithms excel at identifying subtle patterns and anomalies in handwriting that may indicate the early stages of Alzheimer's disease.

02

By minimizing subjective interpretation and human biases, ML algorithms contribute to reliable and standardized diagnostic assessments.

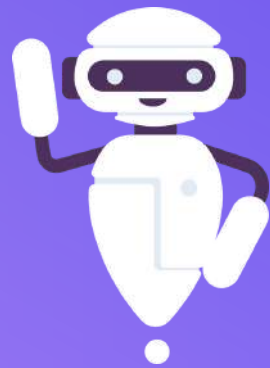
03

The precision in analysis allows healthcare professionals to tailor interventions based on specific cognitive challenges and needs, optimizing patient care.

04

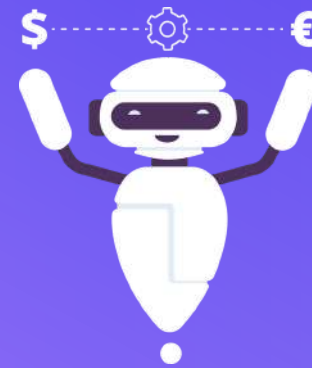
ML algorithms can process large datasets rapidly, providing a cost-effective solution for widespread screening efforts. This efficiency is crucial for reaching a broader population and detecting Alzheimer's cases in a timely manner.

PROJECT OBJECTIVES



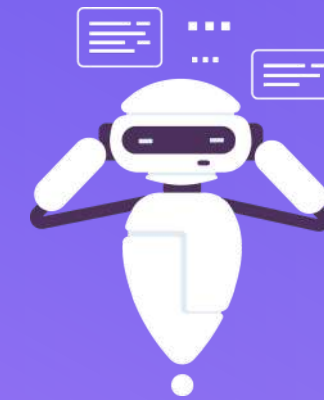
OBJECTIVE 01

Develop a machine learning model for early detection of Alzheimer's, focusing on analyzing relevant data (e.g., handwriting features) to enable timely intervention.



OBJECTIVE 02

Utilize machine learning algorithms to assess and predict Alzheimer's risk based on diverse datasets, incorporating factors like genetic markers and lifestyle data.



OBJECTIVE 03

Explore machine learning applications for personalizing treatment plans, analyzing patient-specific data to tailor interventions and optimize the effectiveness of care strategies.

Methodology

We started off by importing the necessary libraries, modules, and our training and testing dataset. After importing, we scaled the datasets using the `StandardScaler()` function. Then we performed exploratory data analysis to find the distribution of target classes in both of our datasets.

Then we did data analysis, in which we did feature extraction using Principal Component Analysis (PCA). As PCA assumes that data is captured in the variance of features, we captured 80% variance and reduced the number of features from more than 400 to just 48. Then we plotted the cumulative and individual explained variance graphs for our 48 principal components.

We used these 5 different classifiers for this classification task: Support Vector Machine (SVM) using RBF kernel, Random Forest classifier, Decision tree classifier, K-Nearest Neighbours (KNN), Xtreme Gradient Boost (XGboost).

Methodology (continued)

We found out the performance of different classifiers using the following metrics: accuracy score, recall, precision, f1-score. To improve the performance of our models, we used GridSearchCV for hyper-parameter tuning.

For SVM we varied the values of 'C' and 'Gamma'.

For Decision Tree classifier, we varied the values of 'criterion', 'max_depth', 'min_samples_split', 'min_samples_leaf'.

For Random Forest classifier, we varied the values of 'n_estimators' and 'max_depth'.

For KNN classifier, we varied the values of 'n_neighbours', 'weights', 'metric'.

For XGboost classifier, we varied the values of 'n_estimators' and 'max_depth'.

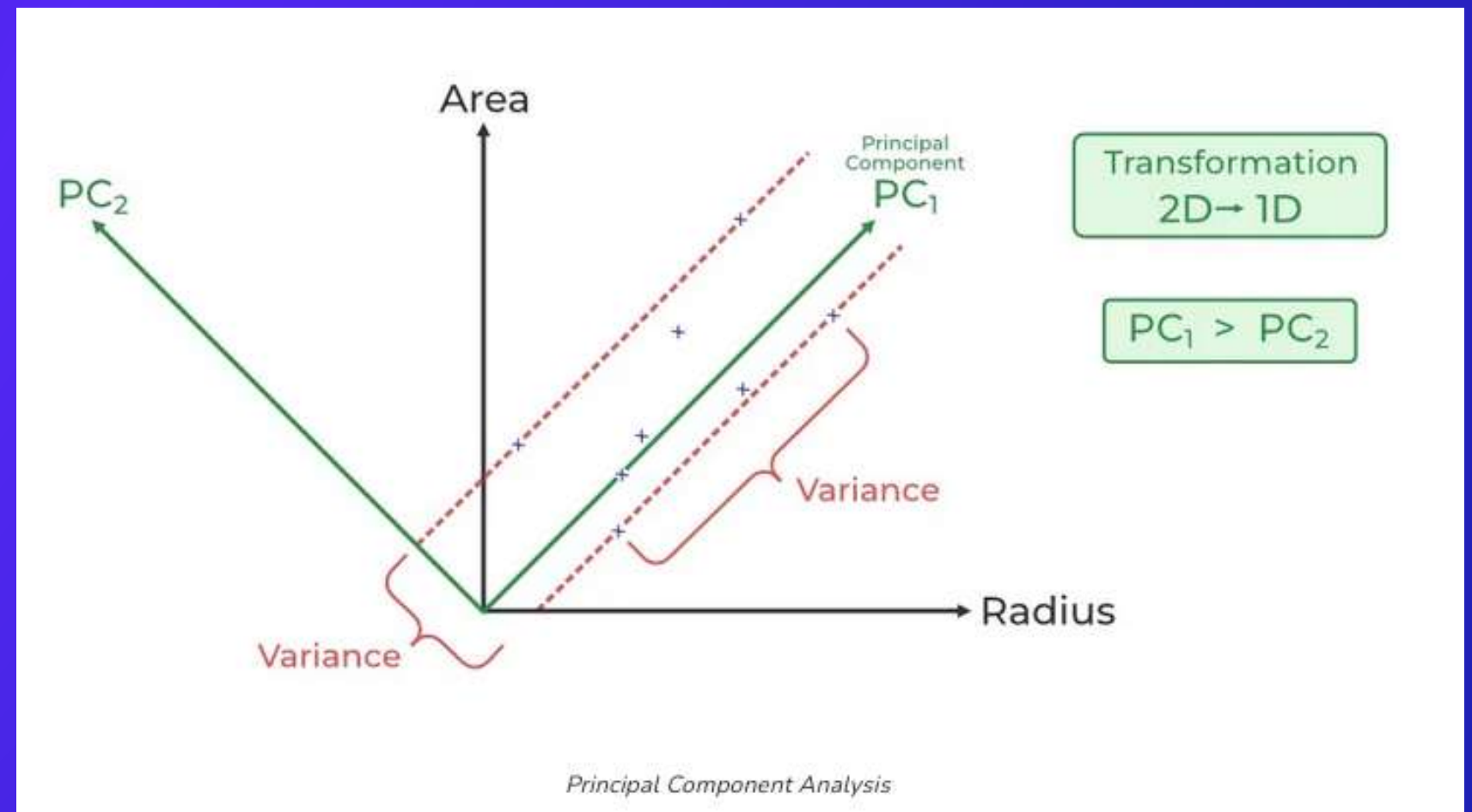
After evaluating the fine-tuned models, we found out that the best model for this classification task is

-

Principal Component Analysis

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. To solve this issue, we use a popular dimensionality reduction technique known as Principal Component Analysis (PCA)

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.



Advantages of Principal Component Analysis

- Dimensionality Reduction: Principal Component Analysis is a popular technique used for reducing number of dimensions, which is the process of reducing the number of variables in a dataset. By reducing the number of variables, PCA simplifies data analysis, improves performance, and makes it easier to visualize data.
- Feature Selection: Principal Component Analysis can be used for feature selection, which is the process of selecting the most important variables in a dataset. This is useful in machine learning, where the number of variables can be very large, and it is difficult to identify the most important variables.
- Data Visualization: Principal Component Analysis can be used for data visualization. By reducing the number of variables, PCA can plot high-dimensional data in two or three dimensions, making it easier to interpret.

MODELS USED

(From next slide)

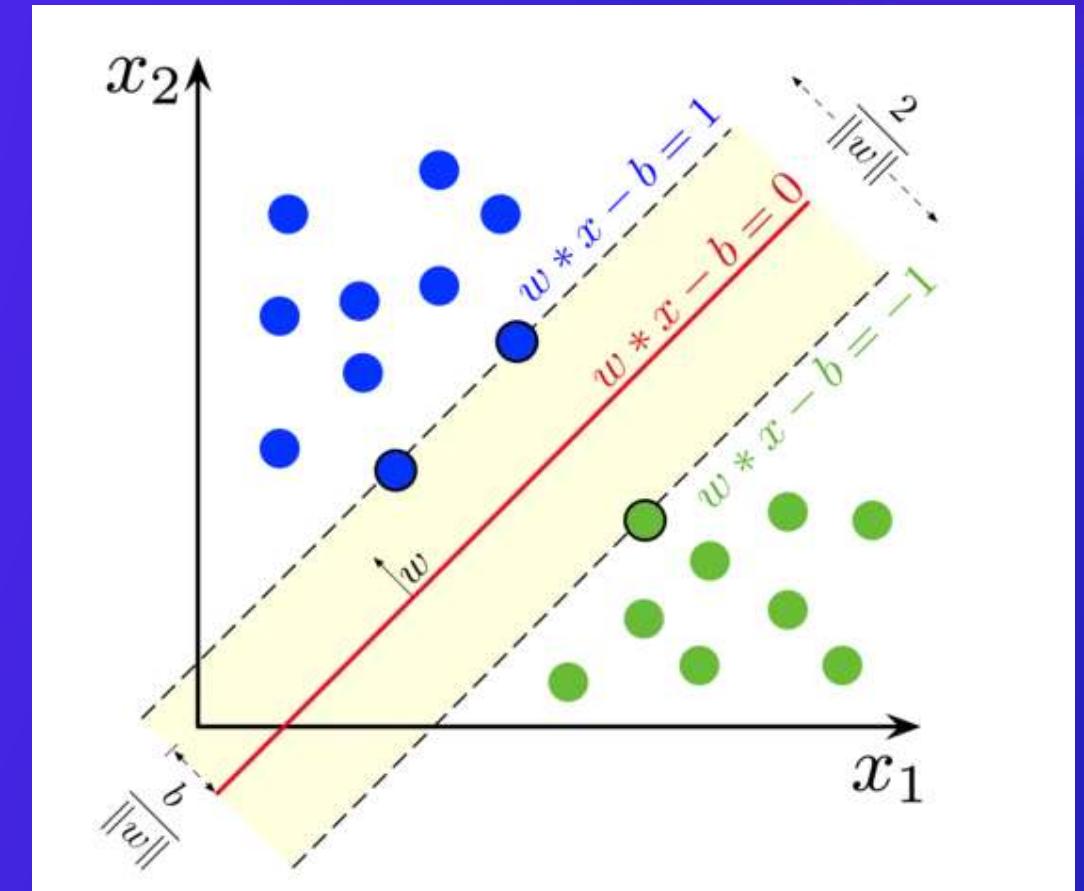
SUPPORT VECTOR MACHINE

The Support Vector Machine(SVM) is a supervised learning algorithm initially proposed by Vladimir Vapnik in 1992. It is one of the widely used algorithms for classification tasks although it can handle regression tasks as well.

The Objective of a Support Vector Machine is to find the hyperplane that has the maximum distance between the data points coming from each of the classes.

The basic steps to create a Support Vector Machine can be divided into 3 major tasks:-

- select two hyperplanes which separates the data with no points between them
- maximize their margin
- Find the average line which is midway from both the support vectors created in step 1. This line is called decision boundary(red line in figure)



SUPPORT VECTOR MACHINE

What happens when we are required to apply this to a data where we cannot apply a linearly separable decision boundary?

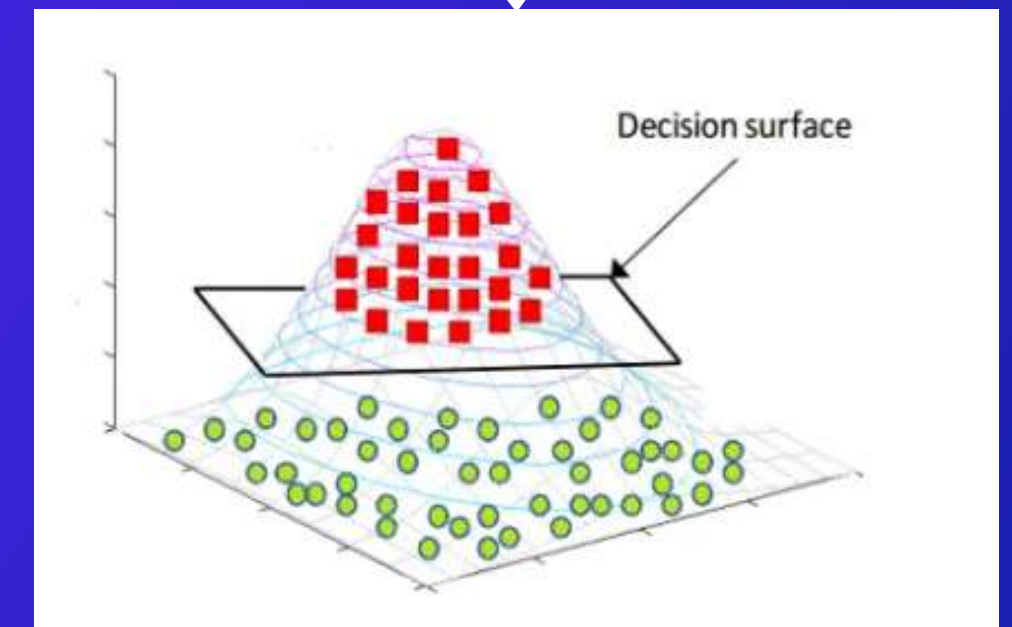
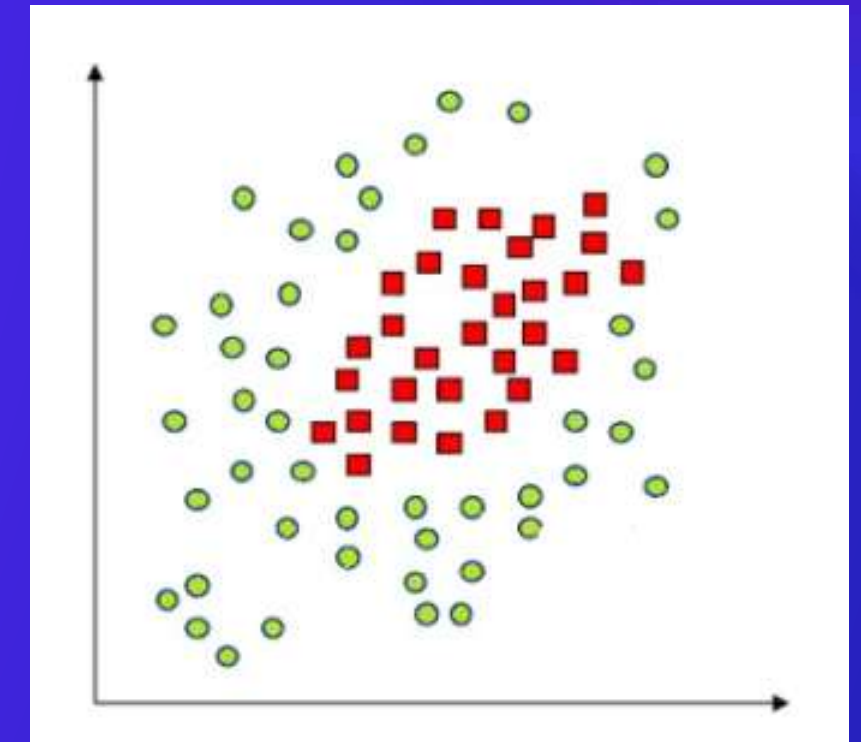
We map all the data points to a higher dimension (in this case, 3 dimension), find the boundary, and make the classification. We accomplish this by using "Kernel SVM"

In order to overcome this, we use a mathematical tool which is called the Kernel trick, which allows us to operate in the original feature space without spending much of our computational resources trying to calculate tedious data in the higher dimension.

Some of the famous kernels include:-

- Linear kernel
- Polynomial kernel
- Radial basis function (RBF)/Gaussian kernel

We have used RBF kernel in our project since it can deal with complex data pretty well compared to other kernels like linear, polynomial etc.

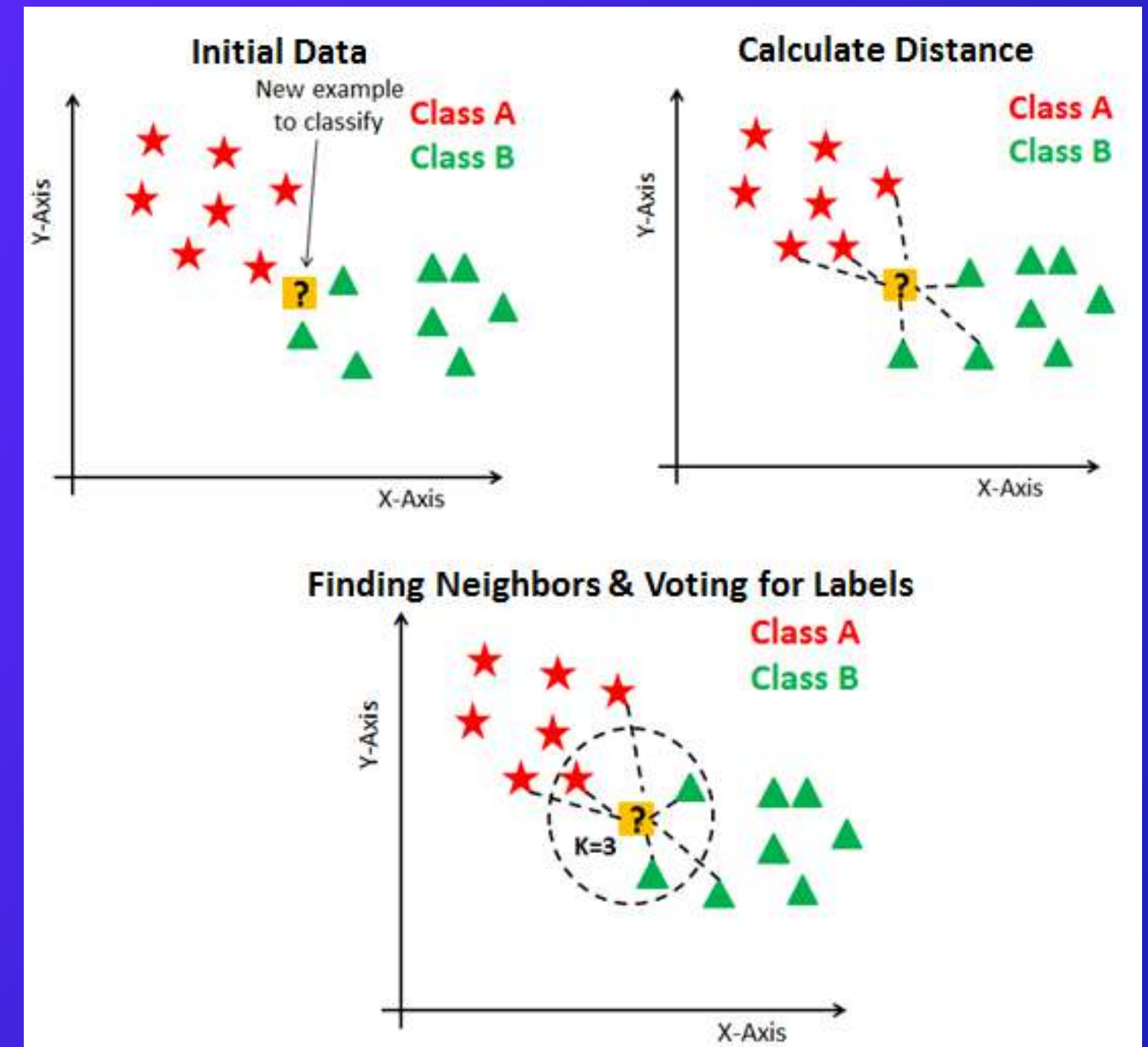


K-NEAREST NEIGHBOURS

K-Nearest Neighbors (KNN) is a simple and widely used classification and regression algorithm in machine learning.

It is a type of distance-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. KNN is a non-parametric and lazy learning algorithm, i.e it doesn't make assumptions about the underlying data distribution, and it defers the learning process until predictions need to be made.

However, it can be computationally expensive, especially with large datasets, since it requires calculating distances between the new data point and all training examples.



Brief overview of how KNN works

1. Training Phase: Where we try to store all the training examples in memory.

2. Prediction Phase:

- Given a new input data point, calculate its distance to all the training examples. The distance metric used (commonly Euclidean distance) depends on the nature of the data.
- Identify the k-nearest neighbors to the input data point based on the calculated distances.

3. Classification (for KNN classification):

- For classification problems, assign the most frequent class label among the k-nearest neighbors to the input data point.

4. Regression (for KNN regression):

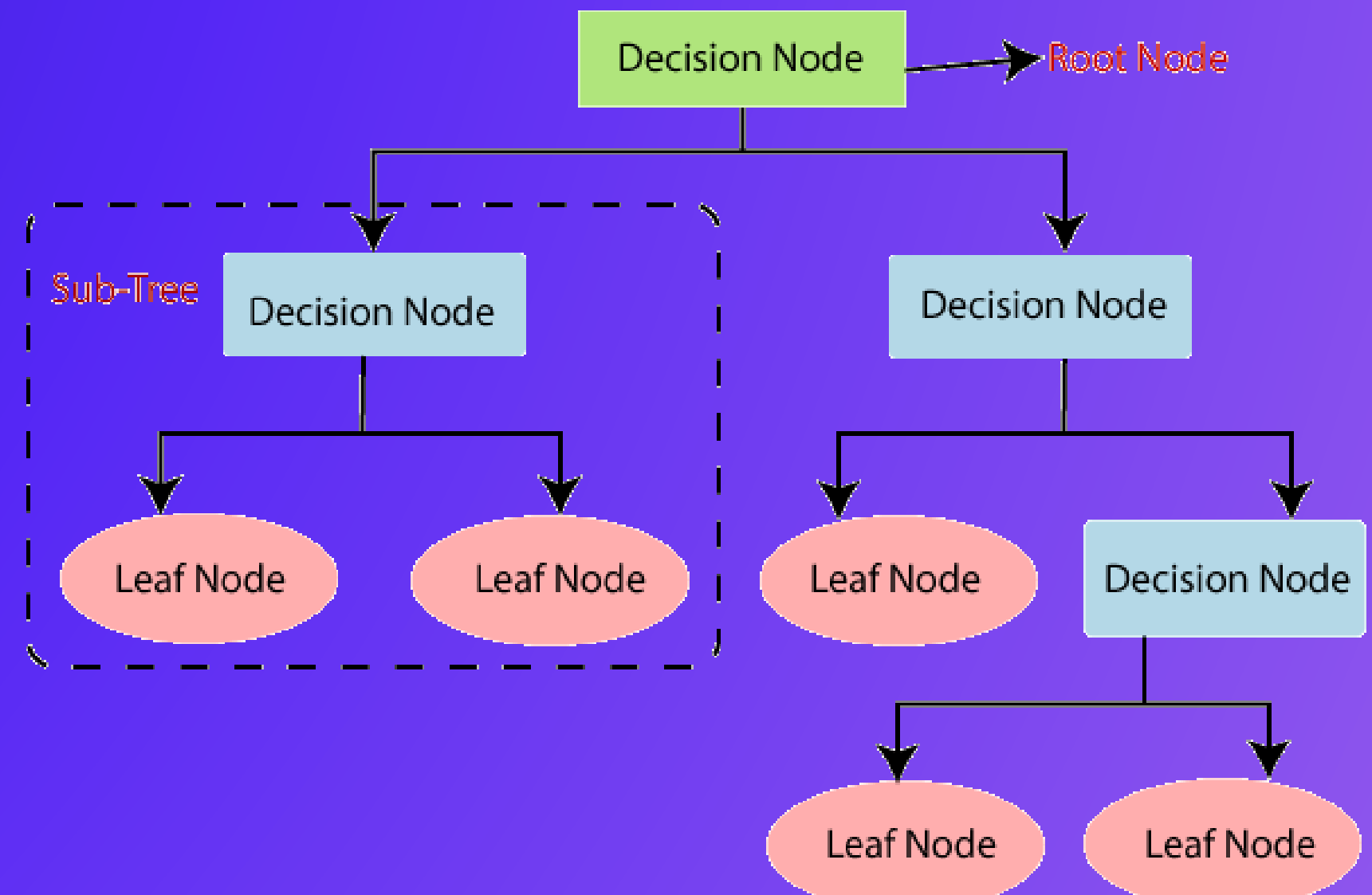
- For regression problems, calculate the average of the target values of the k-nearest neighbors and assign this average as the predicted value for the input data point.

- Choice of K:

- The choice of the parameter k (number of neighbors) is crucial and can significantly impact the performance of the algorithm. A small k value makes the algorithm sensitive to noise, while a large k value may smooth out patterns

Decision Trees

Decision trees are a popular machine learning algorithm that can be used for both regression and classification tasks. The decisions or the test are performed on the basis of features of the given dataset. We have majorly four types of Decision trees – ID3, CART (Classification and Regression Trees), Chi-Square and Reduction in Variance.



Reason to use decision Trees are

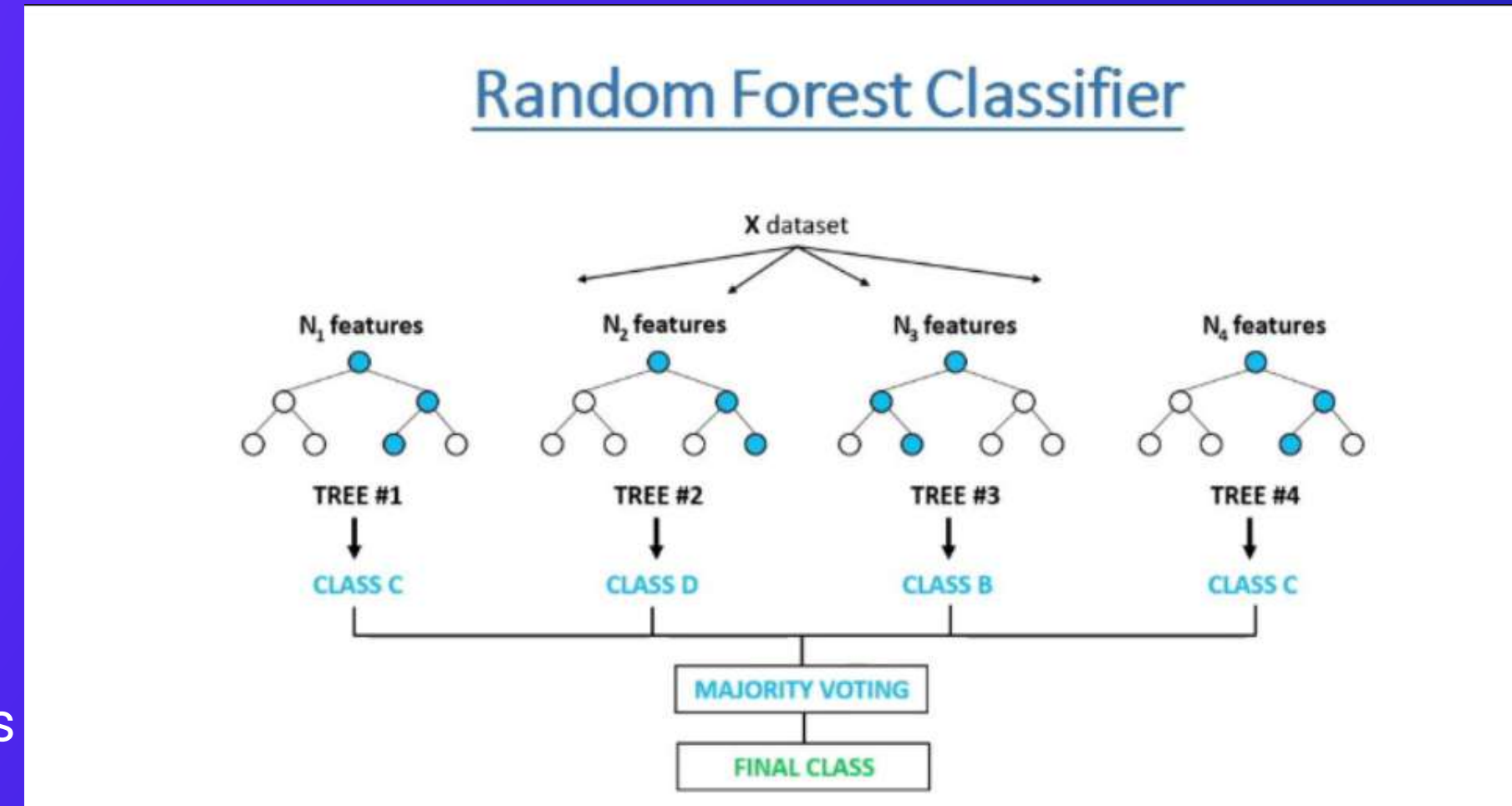
- Decision Trees usually mimic human thinking ability while making a decision, so it is easy to understand.
- The logic behind the decision tree can be easily understood because it shows a tree-like structure.

And also random Forest is an Example of Decision trees and the main Aim of random forest algorithm is to solve overfitting issue

RandomForest

Random forest machine learning algorithm is a part of an symbol learning category which is used for both classification and regression. The algorithm is a collection of decision trees which are formed during training and each tree is trained on a random subset of the data which is inturn trained on random subset of features during the prediction phase. The results from each tree are combined through voting process when the prediction are averaged.

The ensemble nature of Random Forest enhances its robustness and generalization capabilities, making it less prone to overfitting compared to individual decision trees. This algorithm is known for its versatility, scalability, and effectiveness in handling high-dimensional data, making it a popular choice in various machine learning applications.

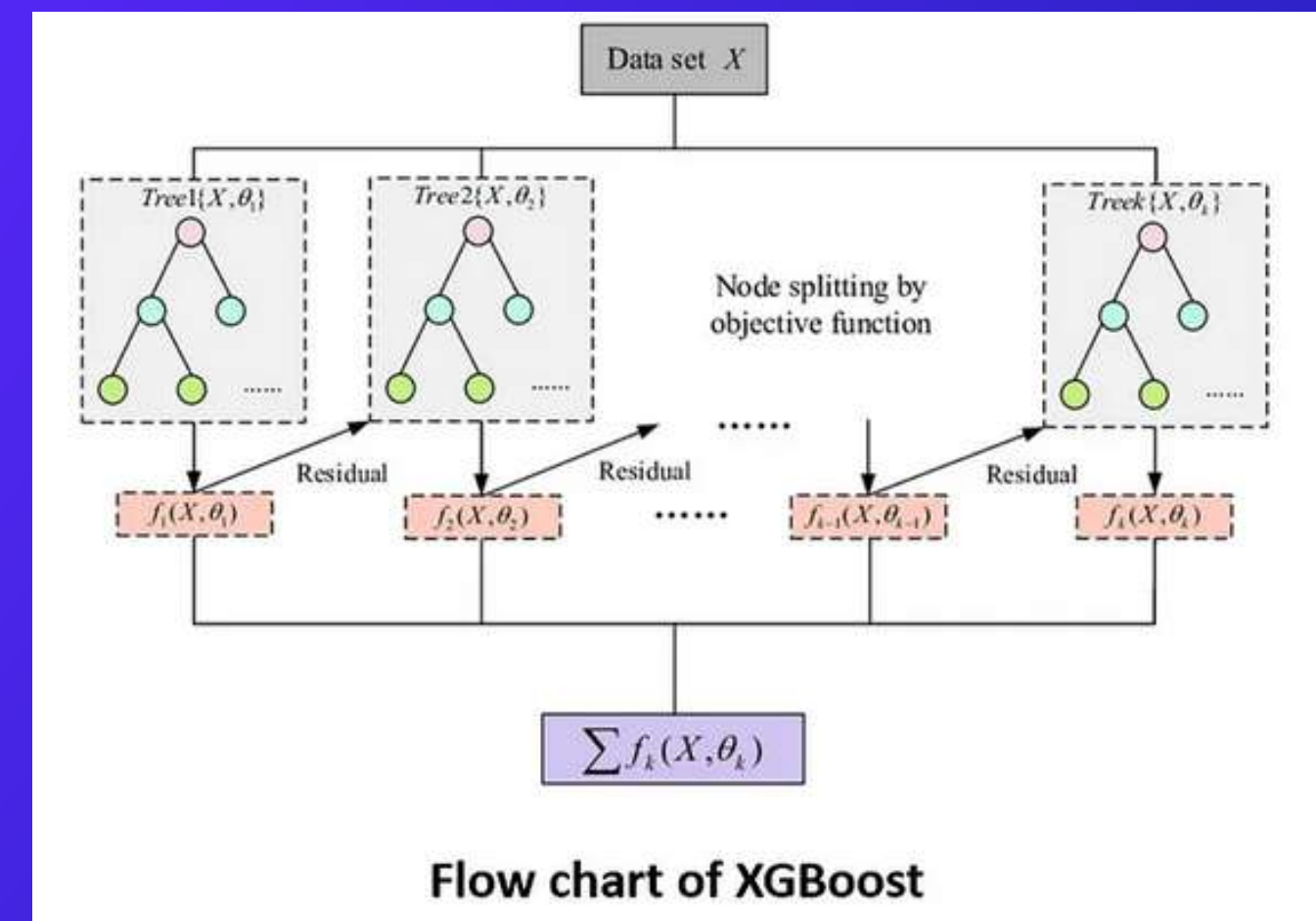


XG Boost

XGBoost (eXtreme Gradient Boosting) is a machine learning algorithm known for its exceptional performance in classification and regression tasks.

It operates on an ensemble of decision trees, building them sequentially and correcting errors from previous iterations. XGBoost minimizes a cost function by employing gradient boosting, where each subsequent tree focuses on minimizing the gradient of the loss function.

It introduces regularization terms to prevent overfitting and handles missing data effectively. XGBoost's unique features include parallel computation, pruning, and advanced optimization techniques, making it scalable and efficient. Its success is attributed to its ability to combine the strengths of multiple weak learners, resulting in a robust, accurate, and interpretable model suitable for various machine learning applications.



FINAL RESULTS OF ALL 5 CLASSIFIERS

	Classifier	Accuracy
(01)	SVM	76%
(02)	Random Forest	66%
(03)	Decision Trees	53%
(04)	XGboost	71%
(05)	KNN	55%

REFERENCES

- Documentations:-
 - Scikit-learn: <https://scikit-learn.org/0.21/documentation.html>
 - Matplotlib: <https://matplotlib.org/stable/index.html>
- YouTube videos:
- GeeksForGeeks articles: <https://www.geeksforgeeks.org/>
- Medium articles: <https://medium.com/>
- Week of ML Kaggle Competition (hosted by RAID, IIT-Jodhpur): <https://www.kaggle.com/competitions/week-of-ml-kaggle-competition>
- Stackoverflow: <https://stackoverflow.com/>

THANK YOU!

Hope you liked our
presentation!

