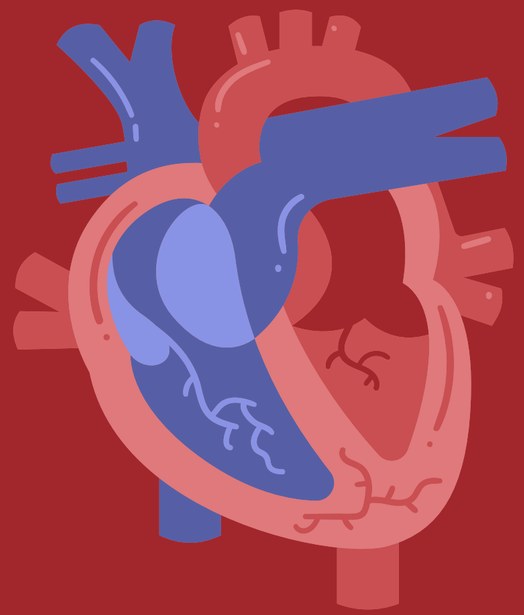
HEART DISEASE PREDICTION USING ML TECHNIQUES



Machine learning methods
for predicting heart disease

HEART DISEASE PREDICTION: A MACHINE LEARNING APPROACH

- **Global Health Challenge:** Heart disease remains a leading cause of death worldwide, with millions of lives affected annually.

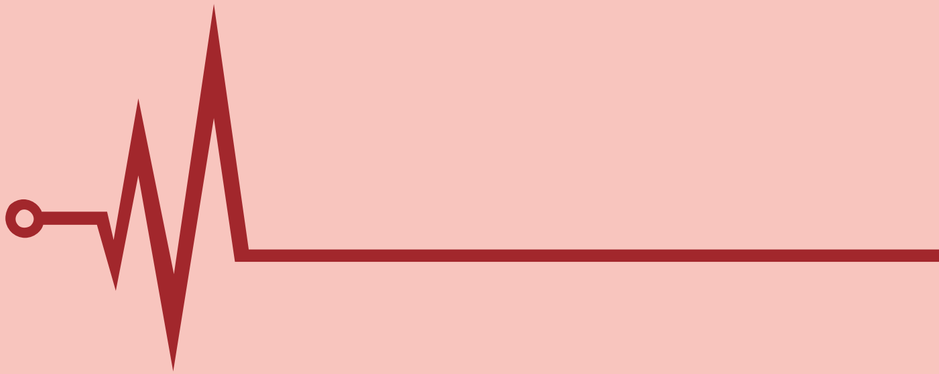
- **Importance of Early Detection:** Early prediction and diagnosis of heart disease can drastically improve treatment outcomes and reduce mortality rates.

- **Objective of the Research:** This study aims to explore the most effective machine learning (ML) models for heart disease prediction by synthesizing findings from two studies.

- **Focus on Top ML Models:**
  We examine four widely used and high-performing ML models:
  - KNN (K Nearest Neighbour)
  - Support Vector Machines (SVM)
  - Random Forest (RF)
  - Artificial Neural Networks (ANN)

PROBLEM STATEMENT:

Develop a machine learning model to predict the likelihood of heart disease in patients based on clinical features such as age, gender, chest pain type, cholesterol levels, and other relevant health indicators. The goal is to classify patients into two categories: those with heart disease and those without, enabling early diagnosis and intervention.



OUR DATASET

We have used a **Kaggle Heart Disease Dataset**, a well-curated dataset with over 1000 entries and 14 features.

S. No	Observation	Description	Values
1.	Age	Age in Years	Continuous
2.	Sex	Sex of Subject	Male/Female
3.	CP	Chest Pain	Four Types
4.	Trestbps	Resting Blood Pressure	Continuous
5.	Chol	Serum Cholesterol	Continuous
6.	FBS	Fasting Blood Sugar	< ,or > 120 mg/dl
7.	Restecg	Resting Electrocardiograph	Five Values
8.	Thalach	Maximum Heart Rate Achieved	Continuous
9.	Exang	Exercise Induced Angina	Yes/No
10.	Oldpeak	ST Depression when Workout compared to the Amount of Rest Taken	Continuous
11.	Slope	Slope of Peak Exercise ST segment	up/ Flat /Down
12.	Ca	Gives the number of Major Vessels Coloured by Fluoroscopy	0-3
13.	Thal	Defect Type	Reversible/Fixed/Normal
14.	Num(Disorder)	Heart Disease	Not Present /Present in the Four Major types.

OUR DATASET

These are the first 15 rows taken from our dataset:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
2	52	1	0	125	212	0	1	168	0	1	2	2	3	0
3	53	1	0	140	203	1	0	155	1	3.1	0	0	3	0
4	70	1	0	145	174	0	1	125	1	2.6	0	0	3	0
5	61	1	0	148	203	0	1	161	0	0	2	1	3	0
6	62	0	0	138	294	1	1	106	0	1.9	1	3	2	0
7	58	0	0	100	248	0	0	122	0	1	1	0	2	1
8	58	1	0	114	318	0	2	140	0	4.4	0	3	1	0
9	55	1	0	160	289	0	0	145	1	0.8	1	1	3	0
10	46	1	0	120	249	0	0	144	0	0.8	2	0	3	0
11	54	1	0	122	286	0	0	116	1	3.2	1	2	2	0
12	71	0	0	112	149	0	1	125	0	1.6	1	0	2	1
13	43	0	0	132	341	1	0	136	1	3	1	0	3	0
14	34	0	1	118	210	0	1	192	0	0.7	2	0	2	1
15	51	1	0	140	298	0	1	122	1	4.2	1	3	3	0

DATA PRE-PROCESSING

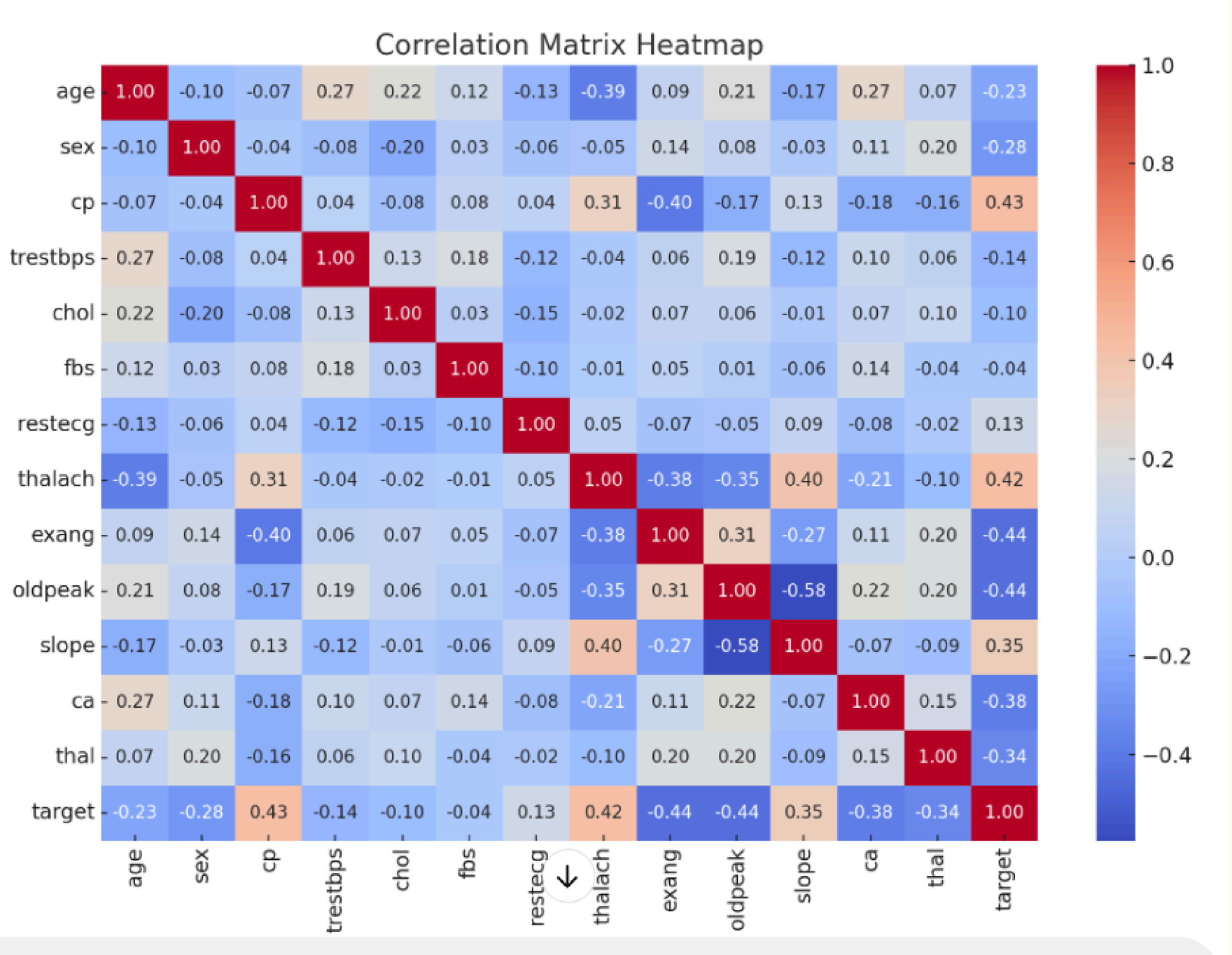
Key preprocessing steps include:

- **Outlier Removal:**
  - Eliminating extreme values to enhance the quality and reliability of the data.
  - Ensures more accurate model training by reducing the impact of anomalous points.
- **Handling Missing Values:**
  - **.dropna**() method is used for minimal missing data.
  - To fill the missing values Mean/Median is used for numerical data and Mode is used for categorical data.
- **Handling Duplicated Values:**
  - Duplicated values are identified using **.duplicated**() method and **.drop_duplicated**() method is used to remove them.

DATA PRE-PROCESSING

Key preprocessing steps include:

- **Feature selection:**
  - Use Correlation Matrix to identify the relationship between features and target.
  - Example: Attributes like chest pain (Cp), thalach showed strong correlations with the target.
- **Data Splitting:**
  - Dividing the data into training and testing sets.
  - Common splits: 80-20
  - Ensures fair evaluation and prevents over-fitting.


Correlation Matrix Heatmap
age 1.00 -0.10 -0.07 0.27 0.22 0.12 -0.13 -0.39 0.09 0.21 -0.17 0.27 0.07 -0.23
sex -0.10 1.00 -0.04 -0.08 -0.20 0.03 -0.06 -0.05 0.14 0.08 -0.03 0.11 0.20 -0.28
cp -0.07 -0.04 1.00 0.04 -0.08 0.08 0.04 0.31 -0.40 -0.17 0.13 -0.18 -0.16 0.43
trestbps 0.27 -0.08 0.04 1.00 0.13 0.18 -0.12 -0.04 0.06 0.19 -0.12 0.10 0.06 -0.14
chol 0.22 -0.20 -0.08 0.13 1.00 0.03 -0.15 -0.02 0.07 0.06 -0.01 0.07 0.10 -0.10
fbs 0.12 0.03 0.08 0.18 0.03 1.00 -0.10 -0.01 0.05 0.01 -0.06 0.14 -0.04 -0.04
restecg -0.13 -0.06 0.04 -0.12 -0.15 -0.10 1.00 0.05 -0.07 -0.05 0.09 -0.08 -0.02 0.13
thalach -0.39 -0.05 0.31 -0.04 -0.02 -0.01 0.05 1.00 -0.38 -0.35 0.40 -0.21 -0.10 0.42
exang 0.09 0.14 -0.40 0.06 0.07 0.05 -0.07 -0.38 1.00 0.31 -0.27 0.11 0.20 -0.44
oldpeak 0.21 0.08 -0.17 0.19 0.06 0.01 -0.05 -0.35 0.31 1.00 -0.58 0.22 0.20 -0.44
slope -0.17 -0.03 0.13 -0.12 -0.01 -0.06 0.09 0.40 -0.27 -0.58 1.00 -0.07 -0.09 0.35
ca 0.27 0.11 -0.18 0.10 0.07 0.14 -0.08 -0.21 0.11 0.22 -0.07 1.00 0.15 -0.38
thal 0.07 0.20 -0.16 0.06 0.10 -0.04 -0.02 -0.10 0.20 0.20 -0.09 0.15 1.00 -0.34
target -0.23 -0.28 0.43 -0.14 -0.10 -0.04 0.13 0.42 -0.44 -0.44 0.35 -0.38 -0.34 1.00
age sex cp trestbps chol fbs restecg thalach exang oldpeak slope ca thal target
1.0 0.8 0.6 0.4 0.2 0.0 −0.2 −0.4

K – NEAREST NEIGHBOUR (KNN)

1. KNN tries to find similarities between predictors and values that are within the dataset.

2. KNN uses a non-parametric method as there is not a particular finding of parameters to a particular functional form.

3. It does not make any type of assumptions about the features and output of the dataset.

HOW KNN WORKS ?

- Initially, we select a value for K in our KNN algorithm.
- Find the euclidean distance of k neighbours.
- Check all the neighbours to the new point we have given and see which is nearest to our point.
- We see to which class there is the highest number obtained. The max number is chosen and we assign our new point to that class.

K – NEAREST NEIGHBOUR (KNN)

Example :-

Age (x_1)	Cholesterol (x_2)	RestECG (x_3)	Target (y)
45	200	0	0
50	240	1	1
60	180	1	0
55	220	0	1
65	260	1	1

We want to classify a **new patient** with the following features:

$$\text{New Point } P = (x_1 = 58, x_2 = 210, x_3 = 0).$$

Solution:

Euclidean Distance formula:

$$d(P,Q) = \sqrt{(x_1^{(P)} - x_1^{(Q)})^2 + (x_2^{(P)} - x_2^{(Q)})^2 + (x_3^{(P)} - x_3^{(Q)})^2}$$

d(P, Q1) = 16.40
d(P, Q2) = 31.05
d(P, Q3) = 30.08
d(P, Q4) = 10.44
d(P, Q5) = 50.50

K – NEAREST NEIGHBOUR (KNN)

Rank the points by their distance to P:

Point	Distance (d)	Target (y)
Q_4	10.44	1
Q_1	16.40	0
Q_3	30.08	0
Q_2	31.05	1
Q_5	50.50	1

Choose $k = 3$. The nearest neighbors are:

$$\{Q_4(y = 1), Q_1(y = 0), Q_3(y = 0)\}$$

The targets of the neighbors are: $\{1, 0, 0\}$.

The majority vote is $y = 0$ (No Heart Disease).

Final predicted output = No heart disease (y=0)

ADVANTAGES

- We can implement the algorithm with ease.
- Very effective against noisy data by averaging k-nearest neighbours.
- Better suited for non-linear dataset.
- Doesn’t require a separate training phase.

DISADVANTAGES

- Computationally expensive, especially for large datasets.
- KNN may struggle with high-dimensional data.
- Slow, especially for large datasets as it includes distance calculation.
- Very sensitive to the presence of irrelevant parameters.

SUPPORT VECTOR MACHINE (SVM)

A supervised machine learning algorithm that classifies data by finding an optimal line or hyperplane that maximizes the distance between each class in an N-dimensional space.

HOW SVM WORKS ?

Goal: Given a set of labeled training data, SVM aims to find a hyperplane (decision boundary) that best separates the data into different classes.

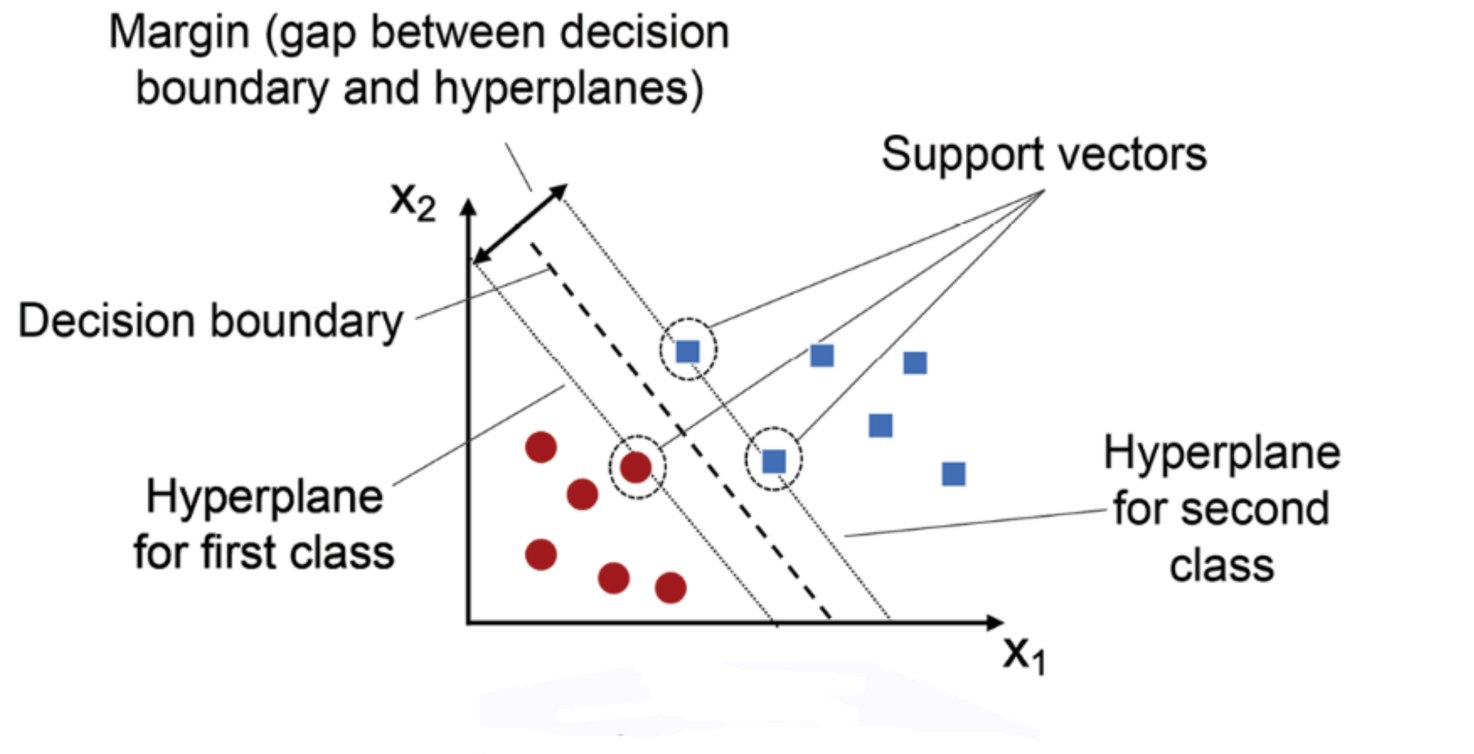
Input: A set of feature vectors (data points) with corresponding labels.

Output: A decision boundary (hyperplane) and the classification of new data points.

SUPPORT VECTOR MACHINE (SVM)

OBJECTIVE OF SVM:

To find an optimal hyperplane that has a maximum margin of separation between two nearest data points of both the classes.



SUPPORT VECTOR MACHINE (SVM)

- **Visualization of Data**
  - If the data is in 2D, we can plot it as points on a graph.
  - In binary classification, we have two classes of points, and SVM will try to find a hyperplane (a straight line in 2D) that separates them.

- **Finding the Hyperplane**
  - Hyperplane Equation: In general, the equation of a hyperplane in n-dimensional space is

$$\mathbf{w}^T \mathbf{x} + b = 1$$

where:

w is the weight vector (perpendicular to the hyperplane),

x is the input feature vector,

b is the bias term (which shifts the hyperplane).

SUPPORT VECTOR MACHINE (SVM)

Maximize the Margin

- The margin is the distance between the hyperplane and the nearest data points from either class.
- Mathematically, the margin is maximized by minimizing the norm of the weight vector ||w|| while satisfying the following constraints:

$$y_i(w^T \cdot x_i + b) \geq 1 \quad \text{for all data points} (x_i, y_i)$$

where y_i is the class label (+1 or -1), and x_i is the feature vector of the i^{th} data point.

Optimization Problem

- The SVM optimization problem is to minimize the following objective function, which balances the margin and misclassification errors:

$$\text{minimize } \frac{1}{2}\|w\|^2 \quad \text{subject to } y_i(w^T x_i + b) \geq 1 \; for \; i = 1, 2, 3, \cdots, m$$

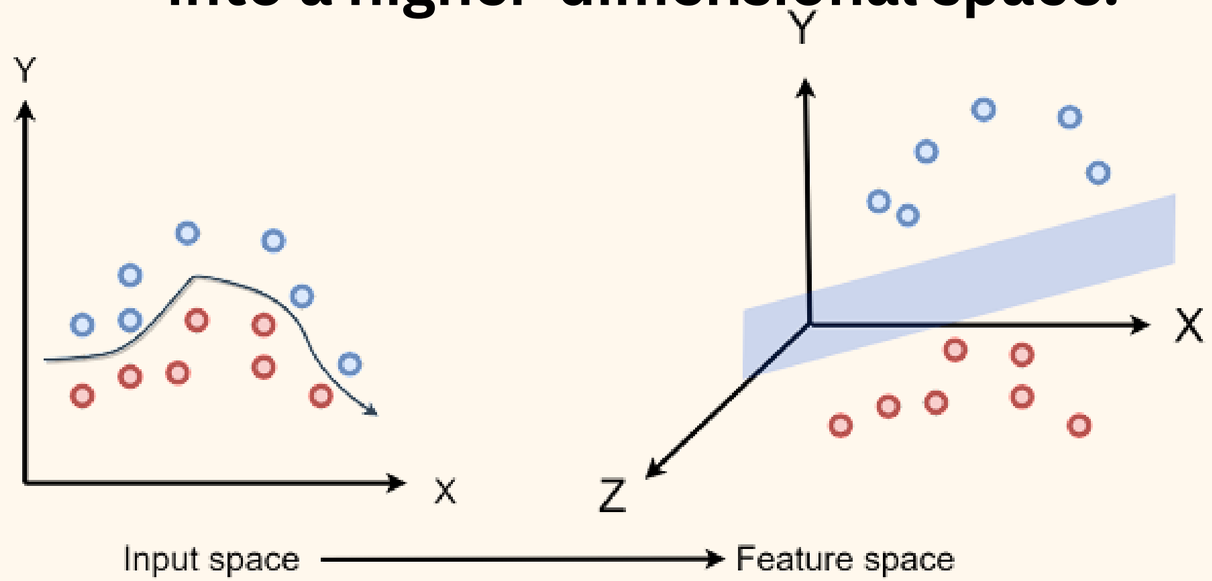
SUPPORT VECTOR MACHINE (SVM)

The new objective becomes:

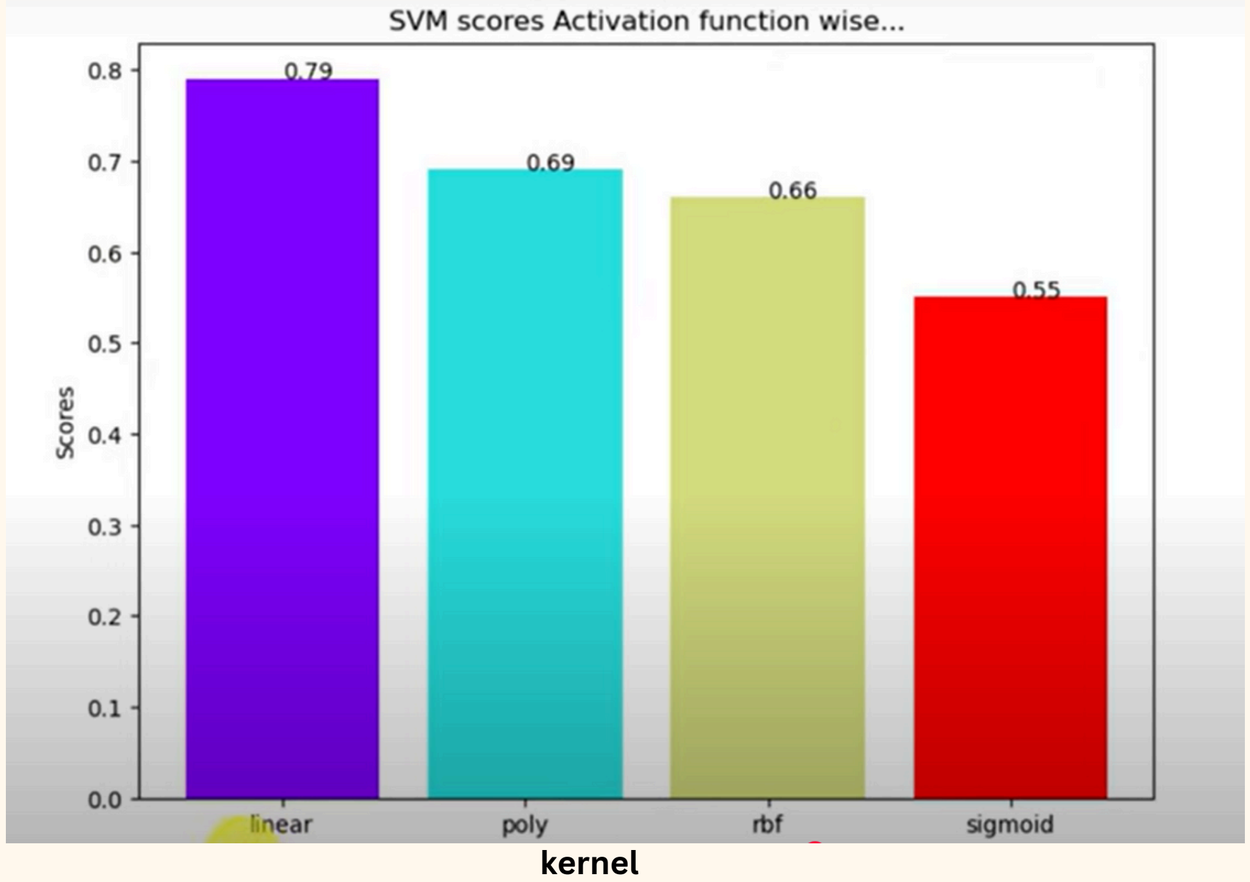
$$\text{Minimize} \quad \frac{1}{2}||w||^2 + C\sum_{i=1}^{n}\xi_i$$

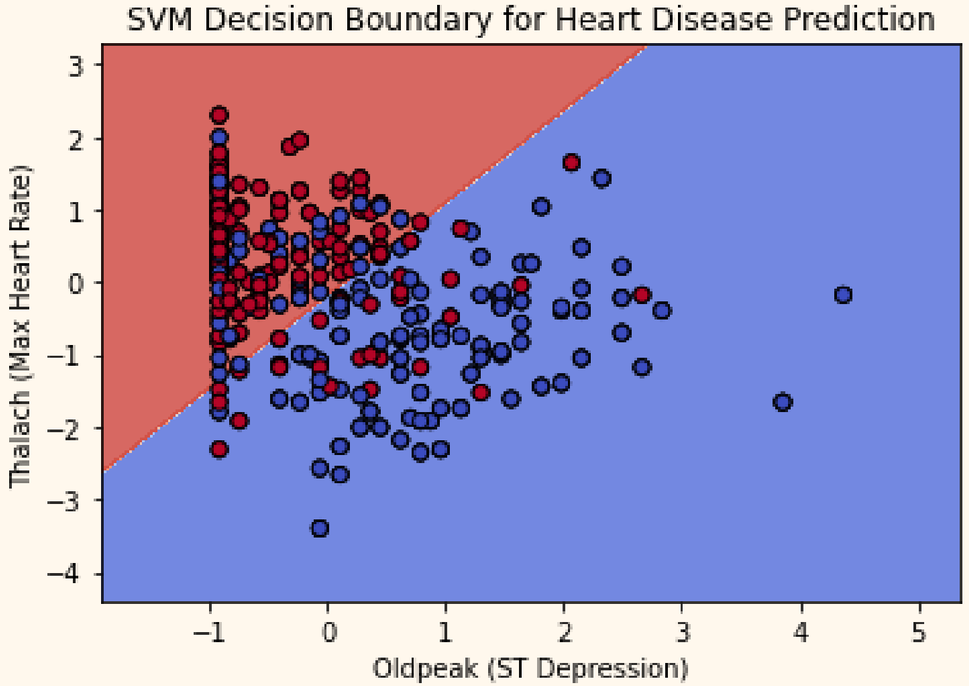
SVM KERNEL

In SVM, a kernel is a mathematical function that transforms the input data into a higher-dimensional space.



SUPPORT VECTOR MACHINE (SVM)





ADVANTAGES

- Effective in High-Dimensional Spaces
- Use kernel functions to handle non-linear relationships.
- Robust to Overfitting in high-dimensional spaces.
- Clear Margin of Separation between the classes, which helps in making accurate predictions

DISADVANTAGES

- Computational Complexity especially for large datasets.
- Not ideal for noisy data as SVMs are sensitive to outliers.
- Hard to interpret.
- Less Scalable as when working with a large dataset training time can increase exponentially with number of samples.

RANDOM FOREST (RF)

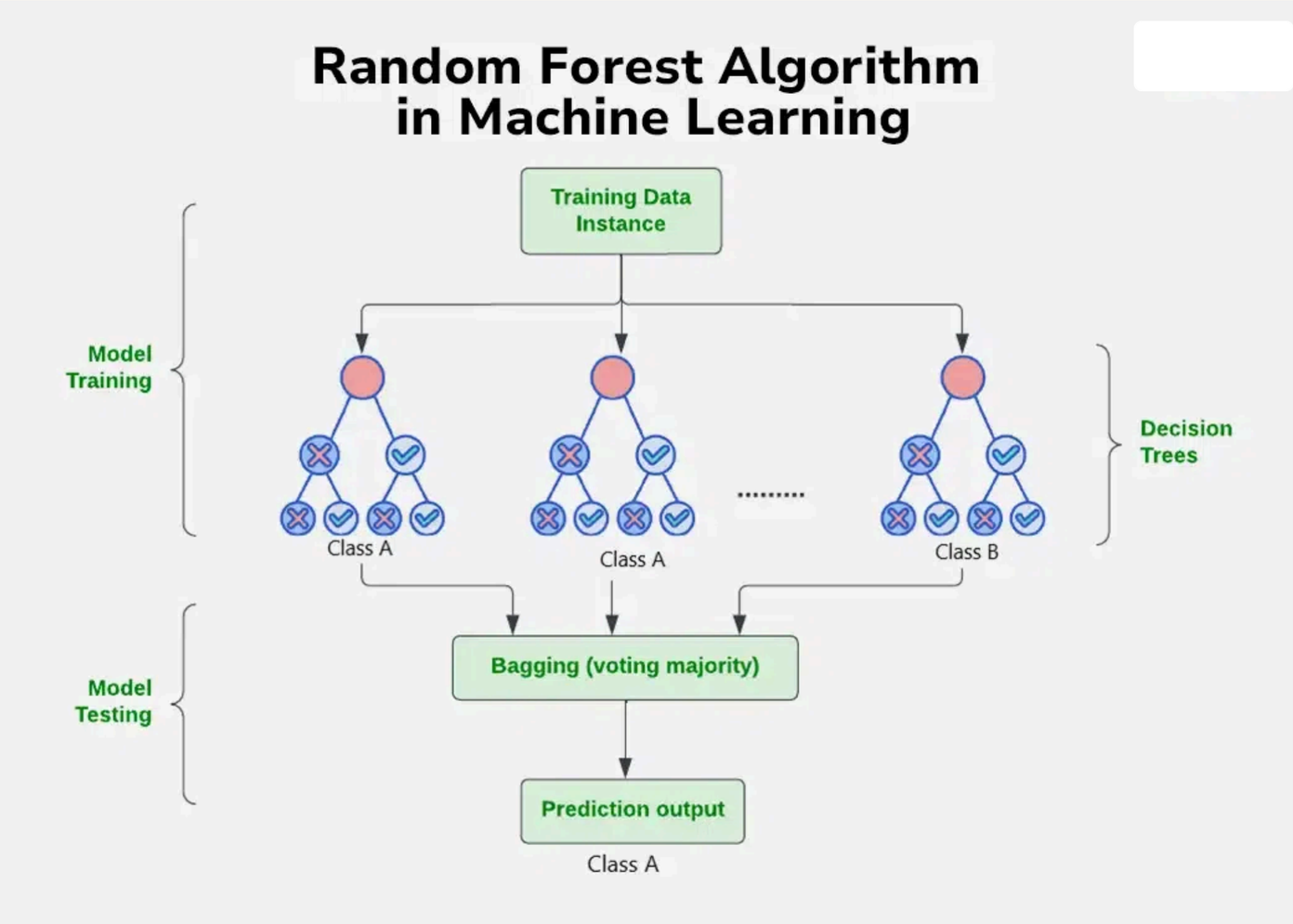
-A popular machine learning technique that combines multiple decision trees to improve the accuracy and robustness of predictions.

-It is a type of ensemble learning method, where the model uses several individual models (in this case, decision trees) to make a final prediction.

HOW RF WORKS ?

Goal: Improve accuracy and reduce overfitting compared to a single decision tree, by building a "forest" of decision trees, each trained on different subsets of data and features.
Input: Training data with nnn samples and mmm features, plus corresponding labels for supervised tasks.
Output: Predicted class (classification) or value (regression) based on the ensemble's decisions.

RANDOM FOREST (RF)



RANDOM FOREST (RF)

NOW HOW IT WORKS:

Bootstrap Aggregation (Bagging):

- Random Forest creates multiple decision trees by sampling the training data with replacement (bootstrapping).
- Each tree is trained on a different random subset of the data.

Random Feature Selection:

- For each split in a tree, the algorithm considers a random subset of features rather than all features. This ensures that the trees are diverse and reduces correlation among them.

RANDOM FOREST (RF)

Tree Growth:

- Each decision tree is grown to its full depth (or another stopping criterion like maximum depth or minimum samples per leaf).

Prediction:

- For classification: Each tree makes a prediction, and the majority vote determines the final prediction.

Ensemble Decision:

- The forest aggregates the results from all trees to produce a more accurate and robust output.

RANDOM FOREST (RF)

WE HAVE CHOSEN A SAMPLE DATA SET:

Row	Age	Sex	CP	Trestbps	Chol	FBS	Restecg	Thalach	Exang	Oldpeak	Slope	CA	Thal	Target
1	52	1	0	125	212	0	1	168	0	1	2	2	3	0
2	53	1	0	140	203	1	0	155	1	3.1	0	0	3	0
3	58	0	0	100	248	0	0	122	0	1	1	0	2	1
4	71	0	0	112	149	0	1	125	0	1.6	1	0	2	1
5	34	0	1	118	210	0	1	192	0	0.7	2	0	2	1

FROM THAT WE HAVE CREATED BOOTSTRAP SAMPLES:

Age	Sex	CP	Trestbps	Chol	FBS	Restecg	Thalach	Exang	Oldpeak	Slope	CA	Thal	Target
52	1	0	125	212	0	1	168	0	1	2	2	3	0
58	0	0	100	248	0	0	122	0	1	1	0	2	1
71	0	0	112	149	0	1	125	0	1.6	1	0	2	1
71	0	0	112	149	0	1	125	0	1.6	1	0	2	1

Age	Sex	CP	Trestbps	Chol	FBS	Restecg	Thalach	Exang	Oldpeak	Slope	CA	Thal	Target
34	0	1	118	210	0	1	192	0	0.7	2	0	2	1
52	1	0	125	212	0	1	168	0	1	2	2	3	0
34	0	1	118	210	0	1	192	0	0.7	2	0	2	1
53	1	0	140	203	1	0	155	1	3.1	0	0	3	0

Age	Sex	CP	Trestbps	Chol	FBS	Restecg	Thalach	Exang	Oldpeak	Slope	CA	Thal	Target
58	0	0	100	248	0	0	122	0	1	1	0	2	1
71	0	0	112	149	0	1	125	0	1.6	1	0	2	1
58	0	0	100	248	0	0	122	0	1	1	0	2	1
52	1	0	125	212	0	1	168	0	1	2	2	3	0

Age	Sex	CP	Trestbps	Chol	FBS	Restecg	Thalach	Exang	Oldpeak	Slope	CA	Thal	Target
53	1	0	140	203	1	0	155	1	3.1	0	0	3	0
34	0	1	118	210	0	1	192	0	0.7	2	0	2	1
71	0	0	112	149	0	1	125	0	1.6	1	0	2	1
34	0	1	118	210	0	1	192	0	0.7	2	0	2	1

RANDOM FOREST (RF)

BUILD DECISION TREES FOR EACH TABLE:

WE GOT THIS AS OUR RESULT

IF TRESTBPS≤118.5TRESTBPS <= 118.5TRESTBPS≤118.5 → TARGET = 1
IF TRESTBPS>118.5TRESTBPS > 118.5TRESTBPS>118.5 → TARGET = 0

SO WHEN WE WILL TAKE A NEW DATA SET IT WILL PASS THROUGH ALL THE TREES AND AGGREGATE THESE PREDICTIONS USING MAJORITY VOTING.

ADVANTAGES

- High Accuracy for both classification and regression tasks because it averages multiple decision trees, reducing variance and overfitting.
- Robustness to Overfitting by combining several decision trees and using techniques like bagging and random feature selection.
- Handles Large Datasets Well with high dimensionality, providing good performance even with a large number of features.
- Feature Importance as it can provide insights into which features are most important for making predictions, helping with feature selection.

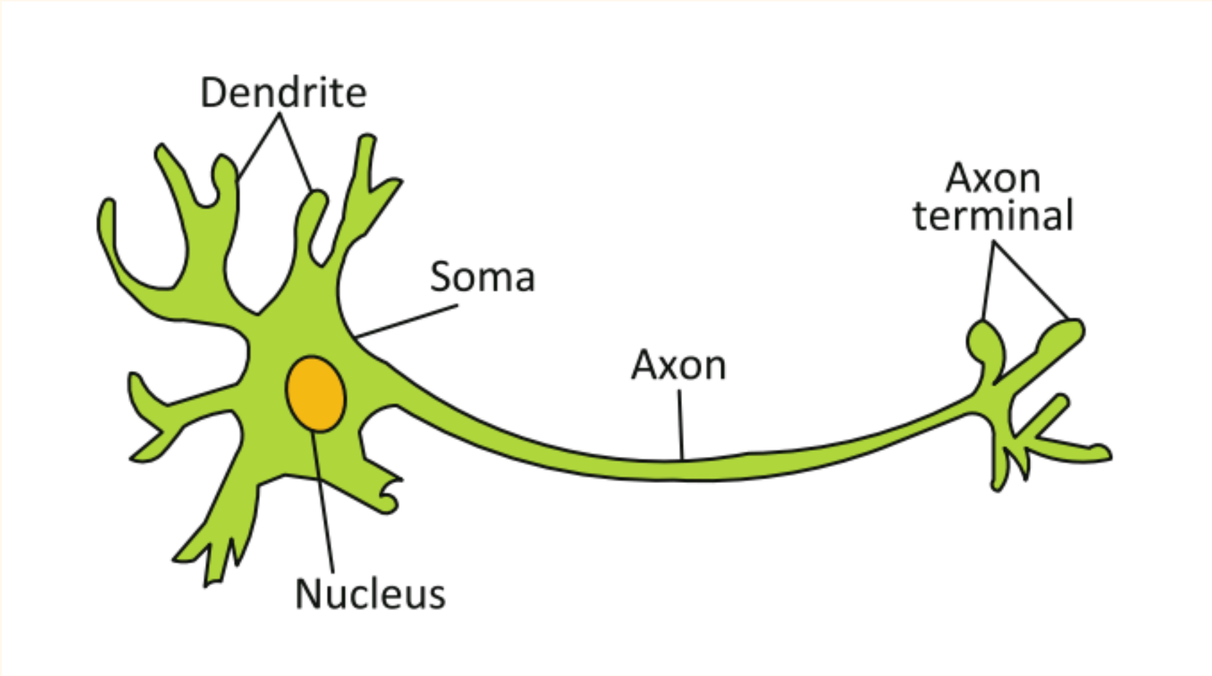
DISADVANTAGES

- Computationally Expensive especially when working with large datasets or complex models with many trees.
- Memory Usage which may be a concern when working with large datasets or deploying the model in resource-constrained environments.
- Slower Predictions when there are a large number of trees, as predictions require each tree to be evaluated.
- Poor Performance on Sparse Data (i.e., have many missing or zero values), especially if the feature space is high-dimensional.

Artificial Neural Network(ANN)

What is a Neuron?

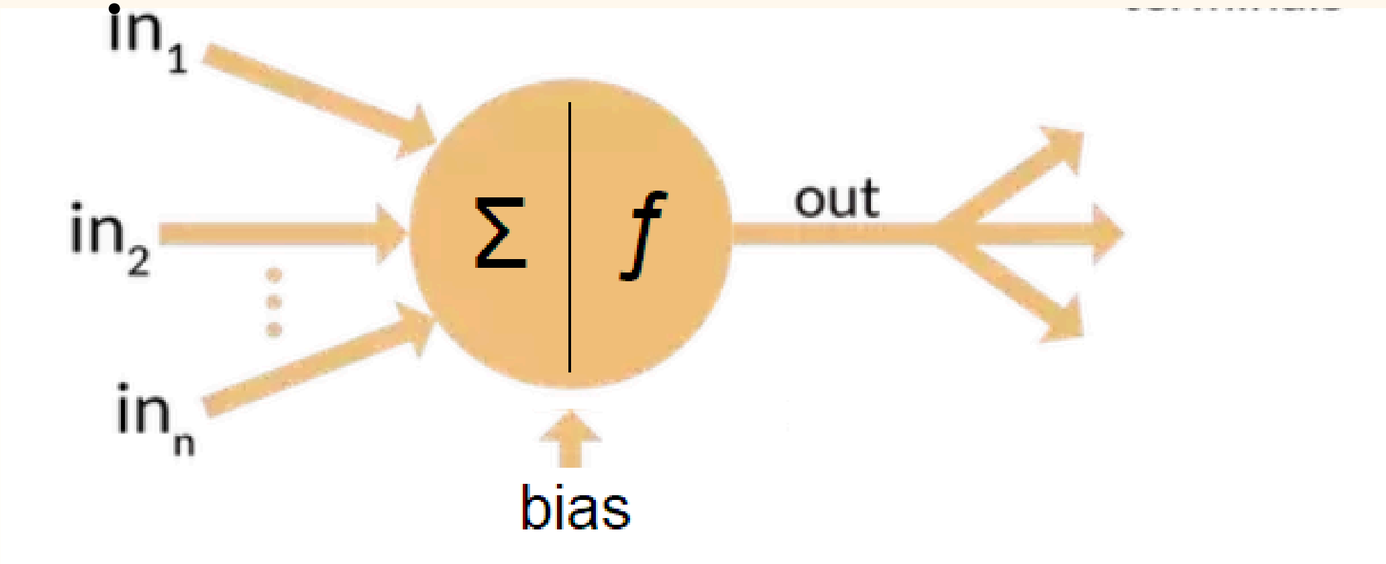
- Basic working unit of human brain.
- These are specialized cells responsible for sending and receiving signals from the brain.



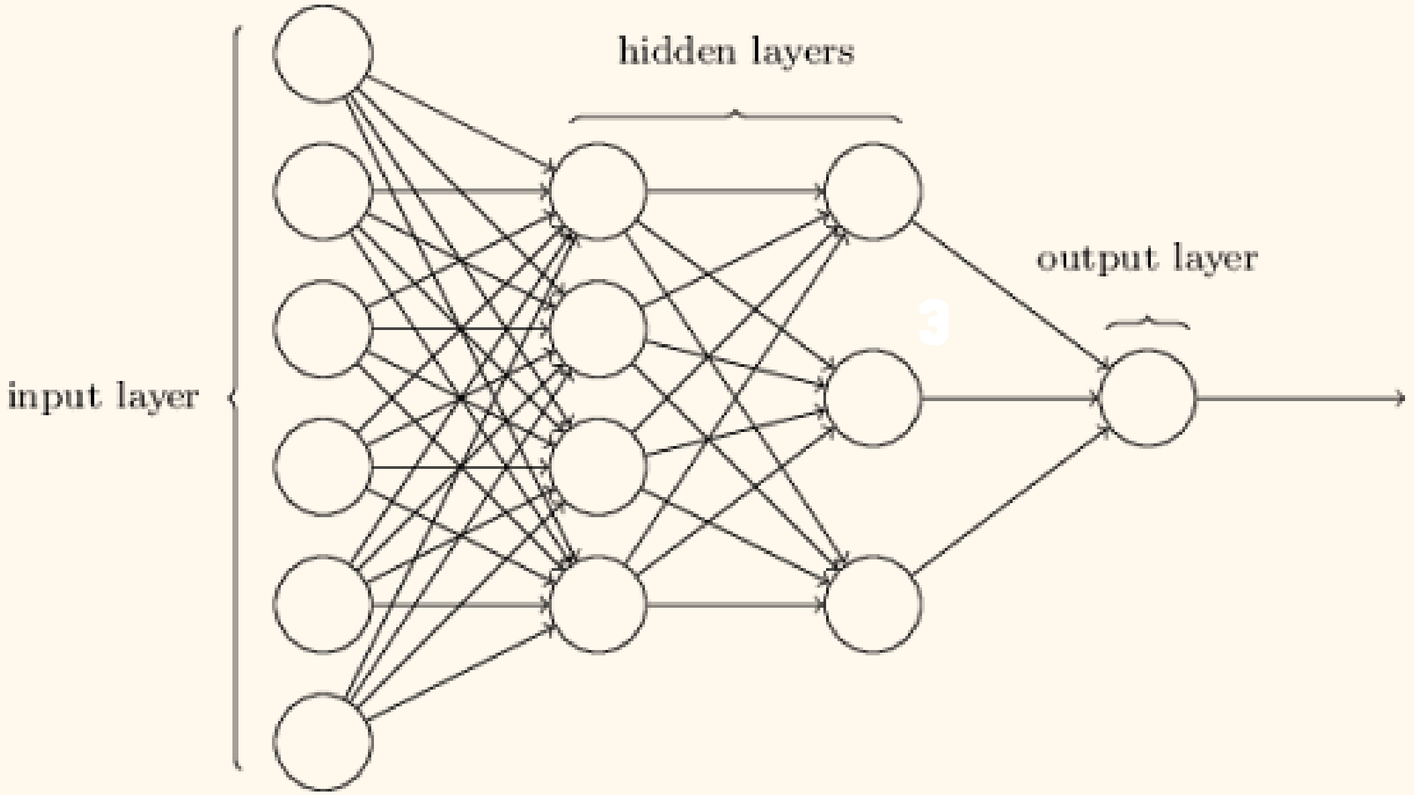
Artificial Neural Network(ANN)

Artificial Neuron:

An artificial neuron is a mathematical function based on the model of biological neurons.



KEY COMPONENTS OF ANN



A Multi-Layer Neuron

KEY COMPONENTS OF ANN

INPUT LAYER:
- Receives the input data (features) and passes it to the next layer.
- Each neuron in this layer represents one feature of the input.

HIDDEN LAYERS:
- Perform computations and extract patterns from the input data.
- Each neuron in these layers applies a weight, a bias, and an activation function to transform the data.
- The number of hidden layers and neurons depends on the complexity of the task.

OUTPUT LAYER:
- Produces the final prediction or result.
- The number of neurons in this layer corresponds to the number of output classes or values.

WEIGHTS AND BIASES:
- Weights determine the importance of inputs.
- Bias shifts the output to improve model performance.

ACTIVATION FUNCTION:
- Activation functions are an integral building block of neural networks that enable them to learn complex patterns in data. They transform the input signal of a node in a neural network into an output signal that is then passed on to the next layer.

HOW DOES ANN WORK?

1-Data Input:

- Each input is passed to the input layer of the neural network. Each node (or neuron) in this layer represents one feature of the input data.

2-Weights and Biases Initialization:

- Each connection between neurons has a weight associated with it, and each neuron has a bias.
- These weights and biases are initialized randomly at the start of training.

3-Forward Propagation (**Passing Data through Layers**)

- The data is passed through the network layer by layer, from the input layer to the output layer.
- The data passed to each neuron is multiplied by the weight of the connection, and the bias is added. This is done for all neurons in each layer.
- The result of this summation is then passed through an activation function. The activation function determines the output of the neuron.

HOW DOES ANN WORK?

4-Activation Function

- The activation function applies a mathematical operation to the weighted sum of inputs to each neuron.
- It introduces non-linearity into the network, enabling it to learn complex patterns.

5-Output

- After the data has passed through all the hidden layers, it reaches the output layer.
- The output layer provides the final result or prediction of the network.

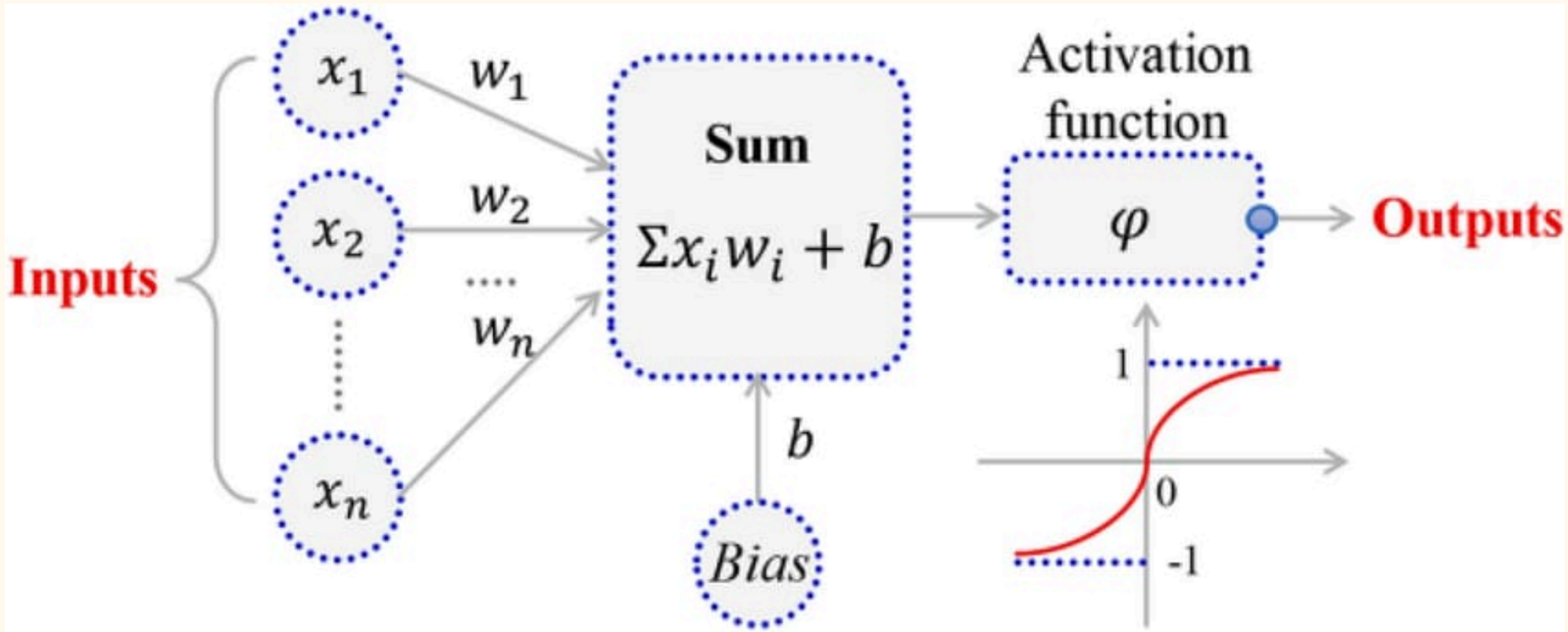
6-Loss Calculation

- The model's prediction is compared with the actual target (true value) using a loss function. The loss function calculates the error (or difference) between the predicted output and the actual target value.

HOW DOES ANN WORK?

7-Backpropagation (**Learning from Error**)

- Backpropagation is the process used to update the weights and biases in the network in response to the error.
- The goal is to minimize the error (loss).



ADVANTAGES

- Learning Complex Patterns: ANNs can model complex, non-linear relationships.
- Flexibility: Suitable for various tasks like image recognition, speech recognition, and time-series forecasting.
- Scalability: Artificial neural networks can be used for large datasets and high-dimensional problems.

DISADVANTAGES

- Computationally Expensive: Requires significant computational resources.
- Data Hungry: Needs a large amount of labeled data to train effectively.
- Hard to Interpret: Often considered a "black-box" model, making it difficult to explain decisions or understand the inner workings.

ENSEMBLE LEARNING

- Ensemble learning is a supervised learning technique used in machine learning to improve overall performance by combining the predictions from multiple models.

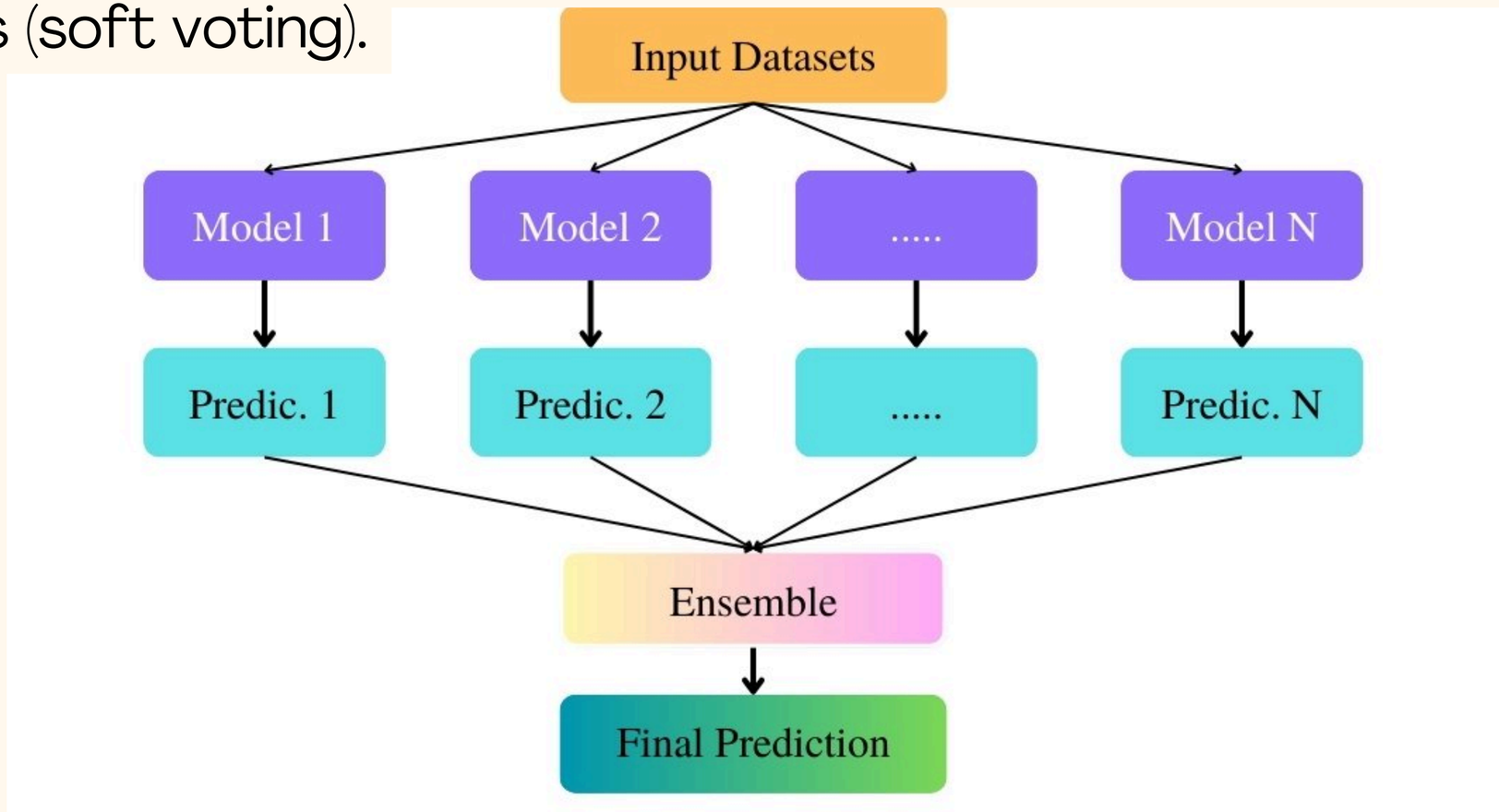
- The idea behind ensemble learning is that by combining multiple models, each with its strengths and weaknesses, the ensemble can achieve better results than any single model alone.

- Here we have combined KNN, SVM, Random Forest and ANN using voting(averaging) method of ensemble learning.

ENSEMBLE LEARNING

Voting (Averaging):

The voting technique in ensemble learning combines predictions from multiple models to make a final decision, improving accuracy and robustness by using either majority class votes (hard voting) or averaged probabilities (soft voting).



CONCLUSION

Machine learning models such as KNN, SVM, Random Forest, and ANN have demonstrated significant potential in accurately predicting heart disease using clinical features. Combining these models through ensemble techniques like soft voting further enhances predictive accuracy by leveraging the strengths of individual models. These advancements provide valuable tools for early and accurate heart disease detection, enabling timely interventions and improving patient outcomes while reducing mortality rates.