PREDICTIVE ANALYTICS USING PYTHON

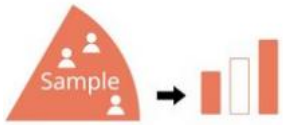
PRESENTED BY – REMYA R S

Analyzing Data for a Business

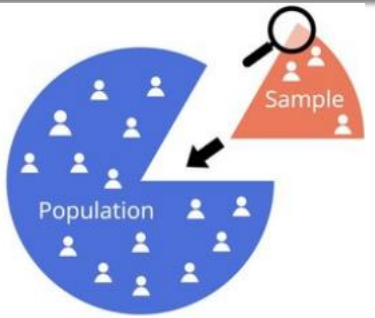
Understanding Problem Statement

Data Collection

Data Cleaning



Descriptive Analysis



Inferential Analysis

Data Visualization

Predictive Analysis

Introduction to Predictive Analytics

Predictive Analytics

- Predictive analysis is a branch of data analytics that uses historical data, statistical algorithms, and machine learning techniques
- Its used to identify the likelihood of future outcomes based on historical data.
- The goal is to go beyond knowing what has happened to providing a best assessment of what will happen in the future.

Key Concepts in Predictive Analysis

Key Concepts

Historical Data: Past data that serves as the basis for analysis.

Statistical Algorithms: Mathematical formulas used to analyze data trends.

Machine Learning: A subset of artificial intelligence that involves training algorithms to learn from and make predictions based on data.

Steps in Predictive Analysis

Steps

Data Collection: Gathering historical data relevant to the problem.

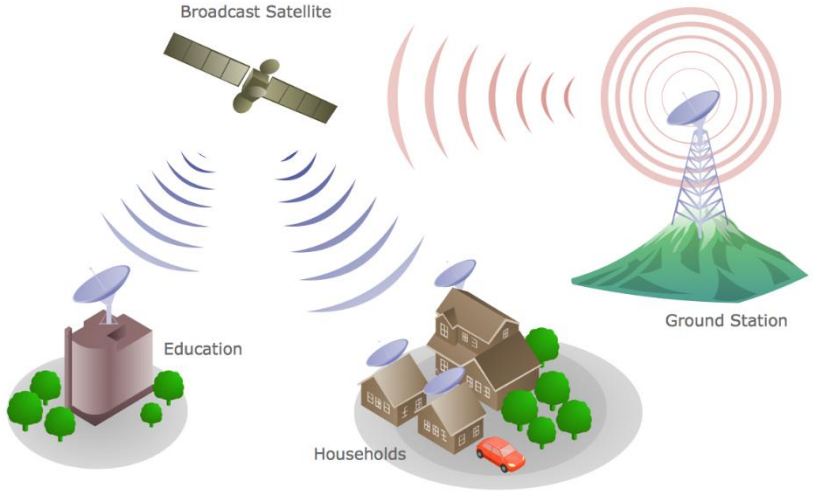
Data Cleaning: Removing noise and correcting inconsistencies in the data.

Data Analysis: Identifying patterns and relationships in the data using statistical techniques.

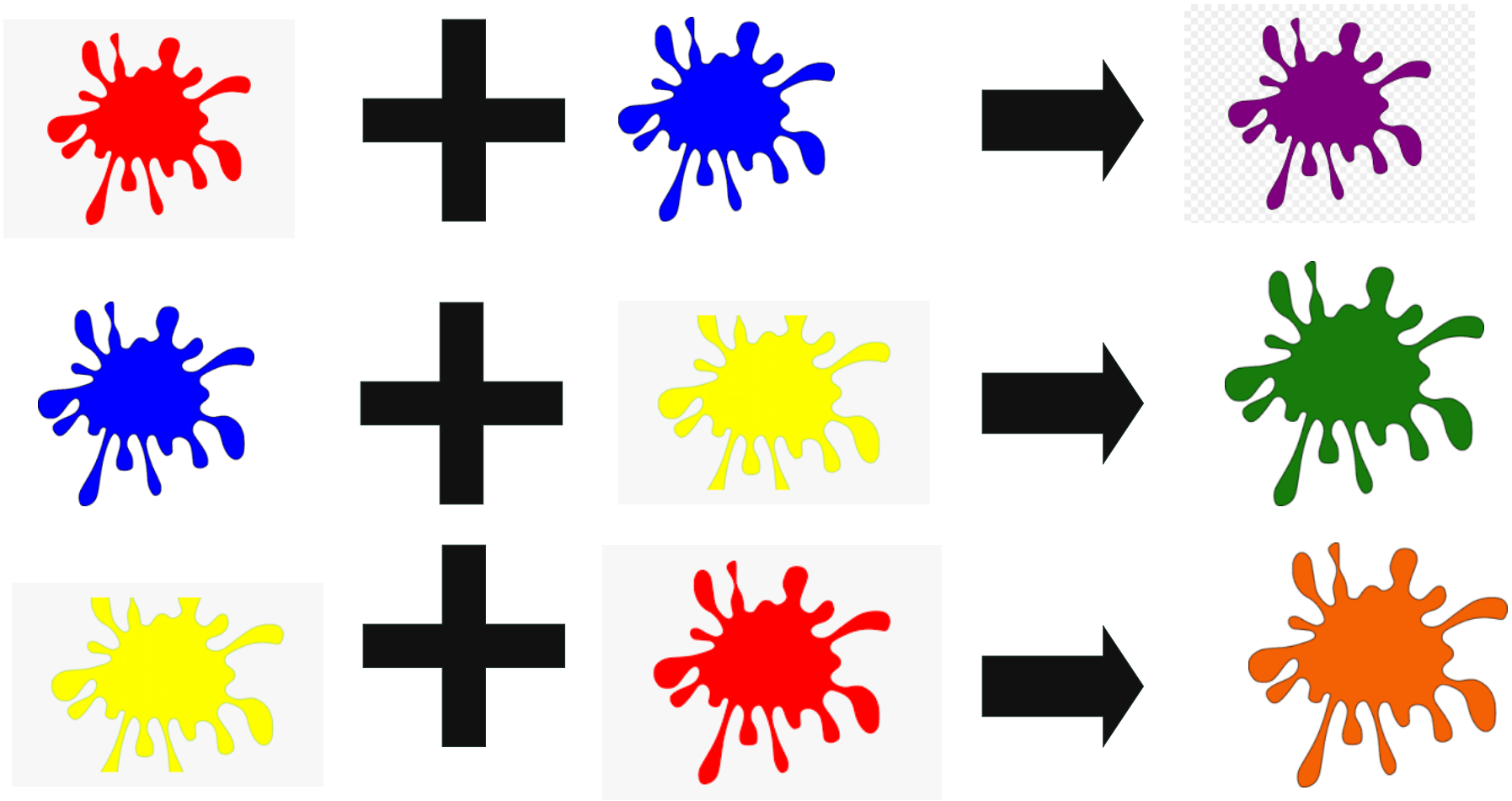
Model Building: Developing a predictive model using machine learning algorithms.

Model Validation: Testing the model on a separate dataset to ensure accuracy.

Applications of Predictive Analysis



Telecommunication



Predictive Analysis

$$y = f(X)$$

Observations *or* Examples

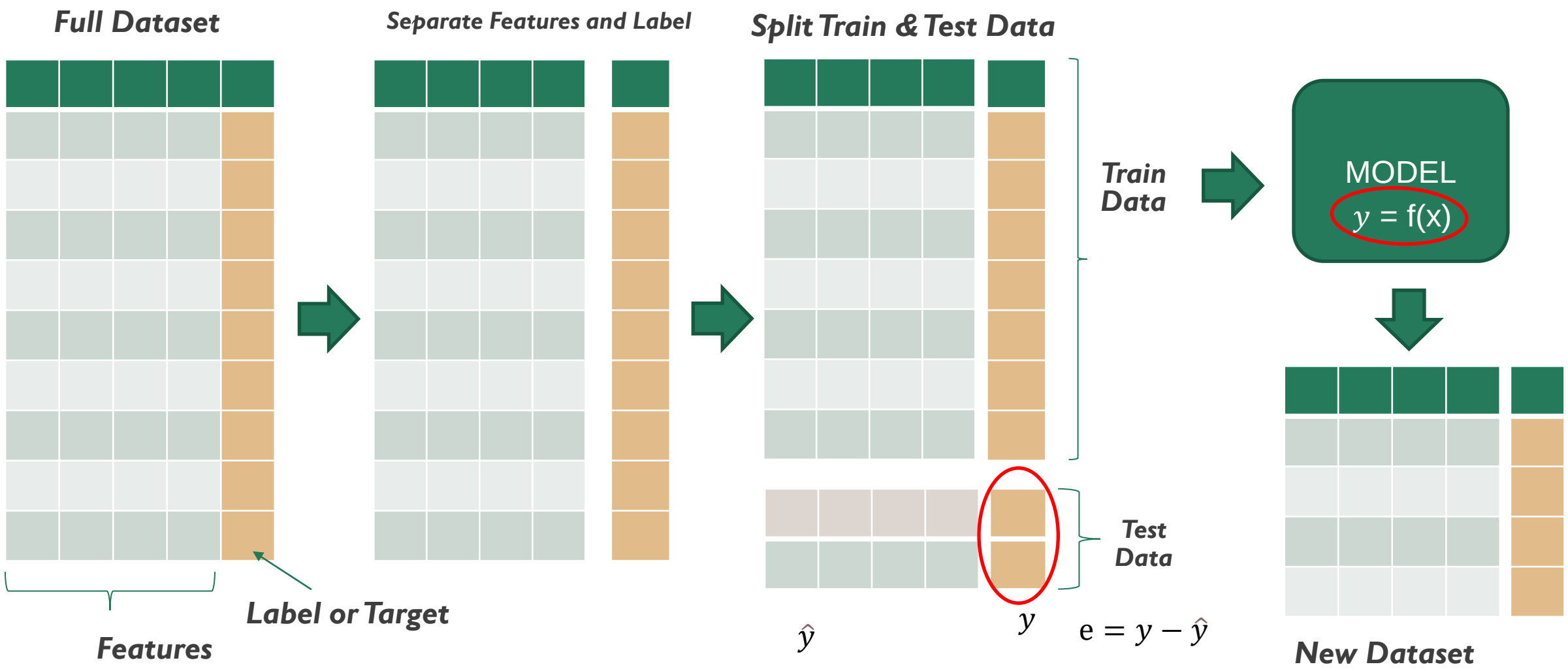
Age	Sex	Cholesterol	Cigarettes per day	Family history of heart disease	Chest pain type	Blood sugar	Heart disease
55	M	221	5	True	typical angina	118	True
50	F	196	0	False	non-anginal pain	98	False
53	F	215	0	True	asymptomatic	110	True
62	M	245	3	False	typical angina	126	True
48	M	190	0	True	non-anginal pain	99	False
70	M	201	0	True	typical angina	105	False

Features X

Label / Target y

The goal is to approximate the mapping function so well that when we have new input data (X) that we can predict the output variables (y) for that data.

Method


Full Dataset
Separate Features and Label
Split Train & Test Data
Train Data
MODEL
$y = f(x)$
Test Data
Label or Target
Features
$\hat{y}$
y
$e = y - \hat{y}$
New Dataset

Regression And Classification Models

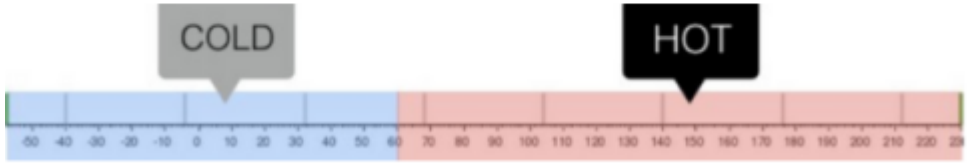
Regression

What will be the temperature tomorrow?

Fahrenheit

Classification

Will it be hot or cold tomorrow?



Fahrenheit

Some Regression Models

Regression Model	Description
Linear Regression	Finds a straight line that best fits the data points.
Ridge Regression	Like Linear Regression, but adds a penalty to avoid memorizing the training data.
Lasso Regression	Similar to Ridge, but can also set some coefficients to zero, effectively choosing only the most important features.
Elastic Net Regression	Combines Ridge and Lasso, balancing their benefits to handle complex data better.

Performance Metrics For Regression

error (or residual) for a data point : $e_i = y_i - \hat{y}_i$

$where\ y_i \text{is}\ observerd\ value\ and\ \hat{y}_i\ is\ predicted\ value$

Sum of Squared Errors : $SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$

Root Mean Square Error

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}$$

high RMSE is “bad” and a low RMSE is “good”

Performance Metrics For Regression

$$SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

$$\text{SST} = \sum_{i=1}^{n} (y_i - \bar{y}_i)^2$$

$$R^2 = 1 - \frac{\text{SS}E}{\text{SS}T}$$

$$\text{Adjusted } R^2 = 1 - \frac{(1 - R^2)(N - 1)}{N - p - 1}$$

Where,

y_i → observed values

$\hat{y}_i$ → predicted values from regression model

n → number of observations

$\bar{y}_i$ → mean of observed values

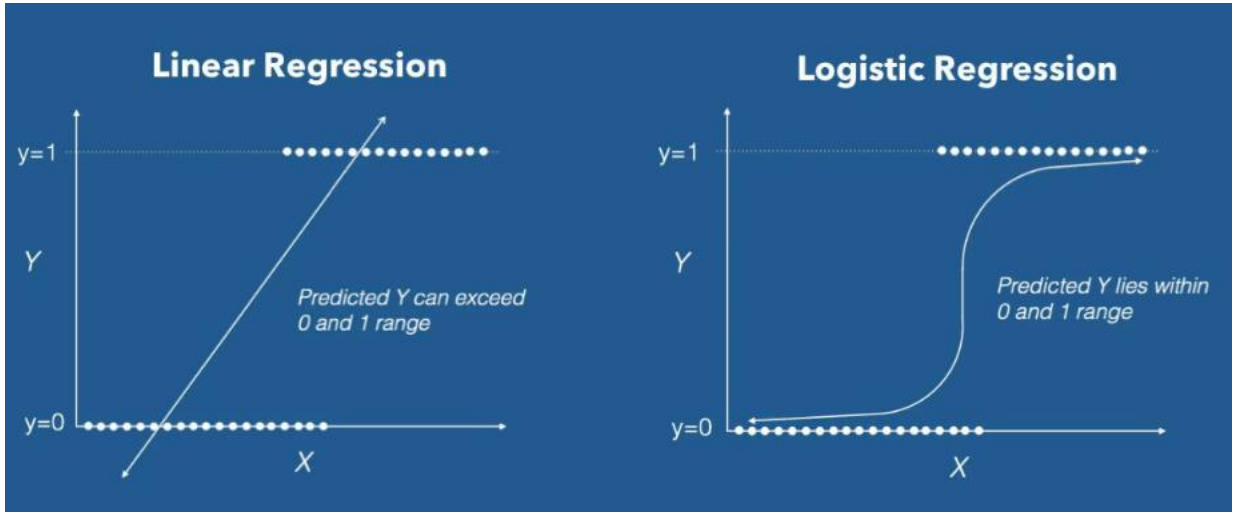
Where,

N – Number of observations or rows in a dataset
p – Number of independent variables in the dataset

High Adjusted R square is good

Logistic Regression

It uses ‘Sigmoid function’ instead of a linear function in regression model.



In logistic regression, we use the concept of the threshold value, which defines the probability of either 0 or 1

Naive Bayes Algorithm

'Bayes Theorem'

$$P(B_k/A) = \frac{P(A/B_k)P(B_k)}{P(A)}$$

where

$$P(A) = \sum_{i=1}^{n} P(A/B)P(B).$$

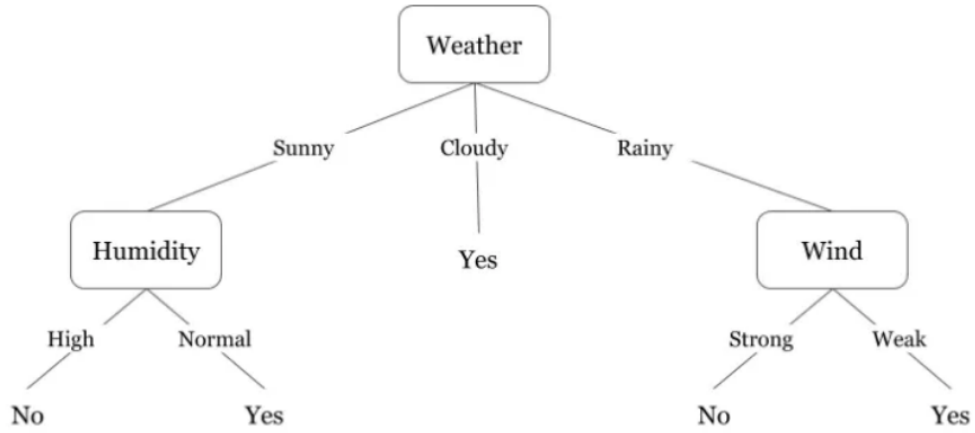
Naive Bayes Classifier formula

$$P(y/x_1x_2....x_j) = \frac{P(x_1x_2....x_j/y)P(y)}{P(x_1x_2....x_j)}$$

Conditional probability of P(y / x) is used for making prediction.

Decision Tree

Day	Weather	Temperature	Humidity	Wind	Play?
1	Cloudy	Hot	High	Weak	Yes
2	Sunny	Hot	High	Weak	No
3	Sunny	Mild	Normal	Strong	Yes
4	Rainy	Mild	High	Strong	No
5	Cloudy	Mild	High	Strong	Yes
6	Rainy	Cool	Normal	Strong	No
7	Rainy	Mild	High	Weak	Yes
8	Sunny	Hot	High	Strong	No
9	Cloudy	Hot	Normal	Weak	Yes
10	Rainy	Mild	High	Strong	No



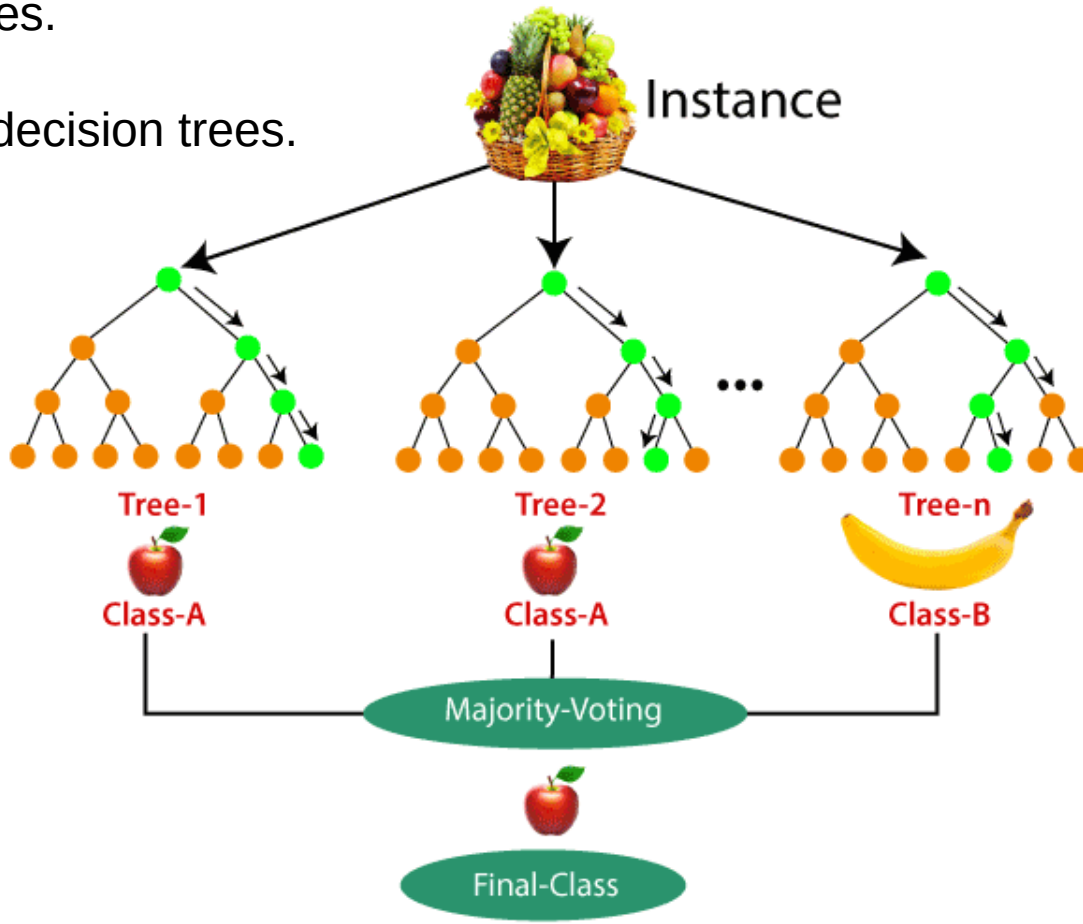
Random Forest

A random forest, is a collection of many decision trees.

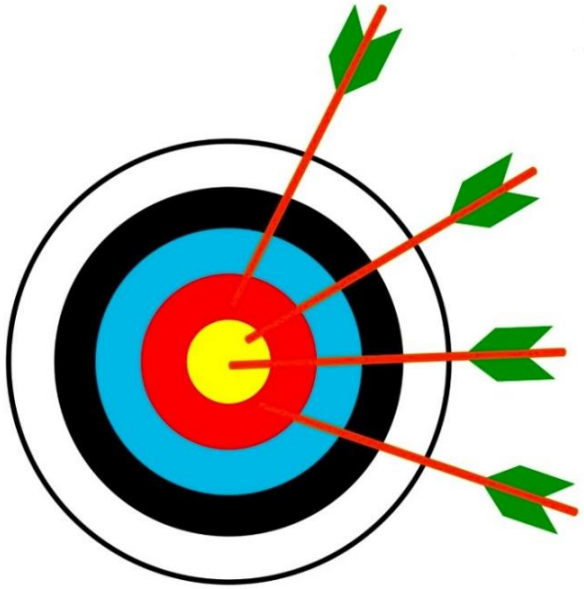
It makes predictions by combining the results of these multiple decision trees.

Final output is considered based on Majority Voting.

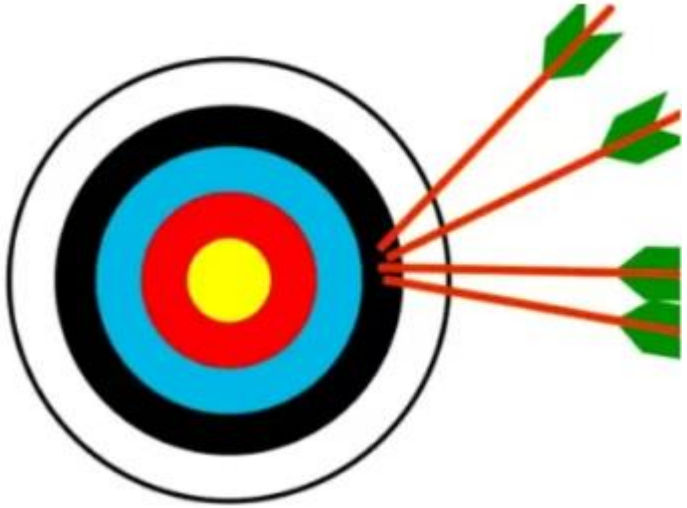
This process helps improve the accuracy and stability of the predictions compared to using just one decision tree.



Performance Metrics In Classification

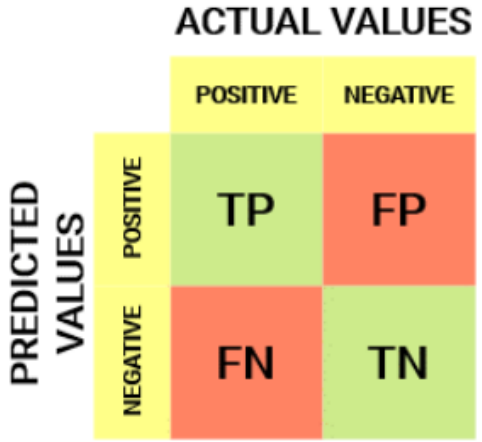


High Accuracy



High Precision

Performance Metrics In Classification



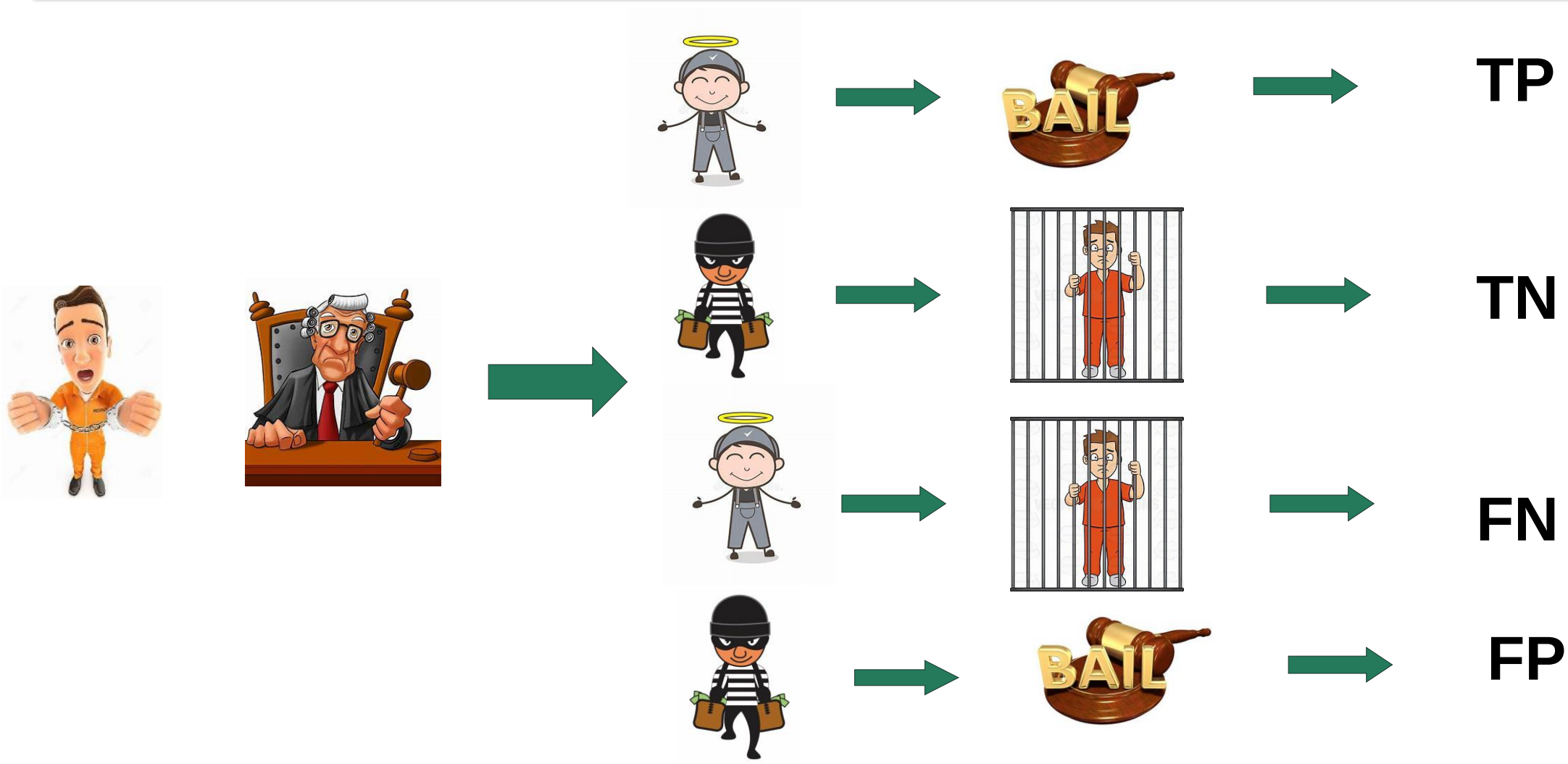
True Positive (TP): The predicted value matches the actual value.

True Negative (TN): The predicted value matches the actual value.

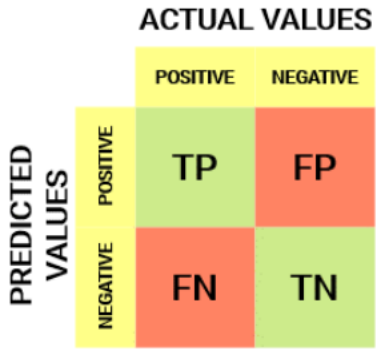
False Positive (FP): The predicted value was falsely predicted.

False Negative (FN): The predicted value was falsely predicted.

Performance Metrics In Classification



Performance Metrics In Classification



Accuracy is a measure of how accurately your model has made predictions for a complete test dataset.
Accuracy = (TP + TN) / (TP + TN + FP + FN)

Precision is a metric for determining how accurate a positive forecast is..
Precision = TP/ (TP + FP)

Recall tells us how many of the actual positive cases we were able to predict correctly with our model.
Recall = TP / (TP + FN)

Precision and recall are combined to determine the F1 score. The more accurate the model is at making predictions, the better the F1 score.
F1Score = 2TP / (2TP + FP + FN)


Thank
You