SUPERVISED AND DEEP LEARNING

UNIT 1

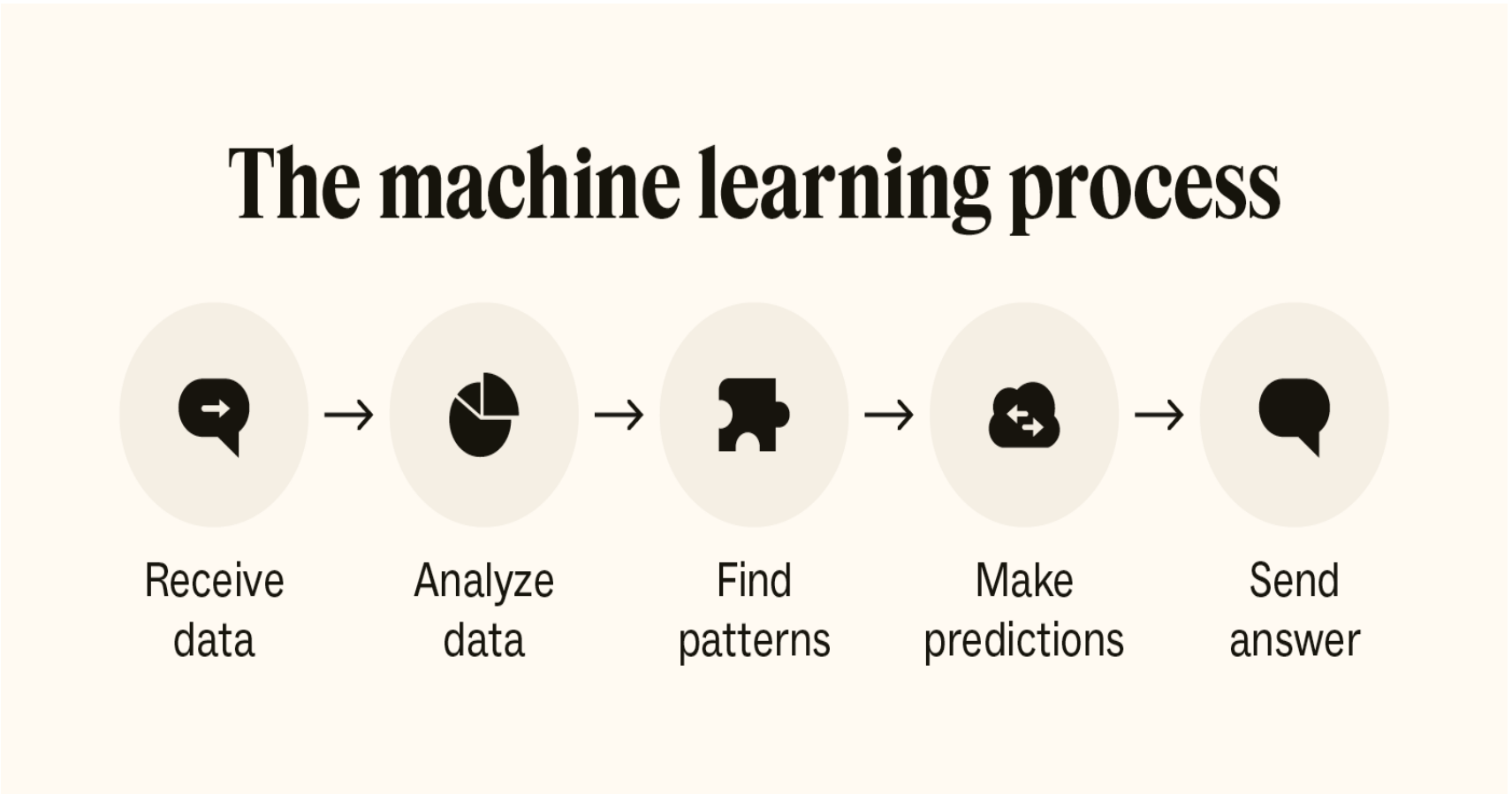
By Ms. Priyanka (Assistant Professor)

INTRODUCTION TO
Machine Learning

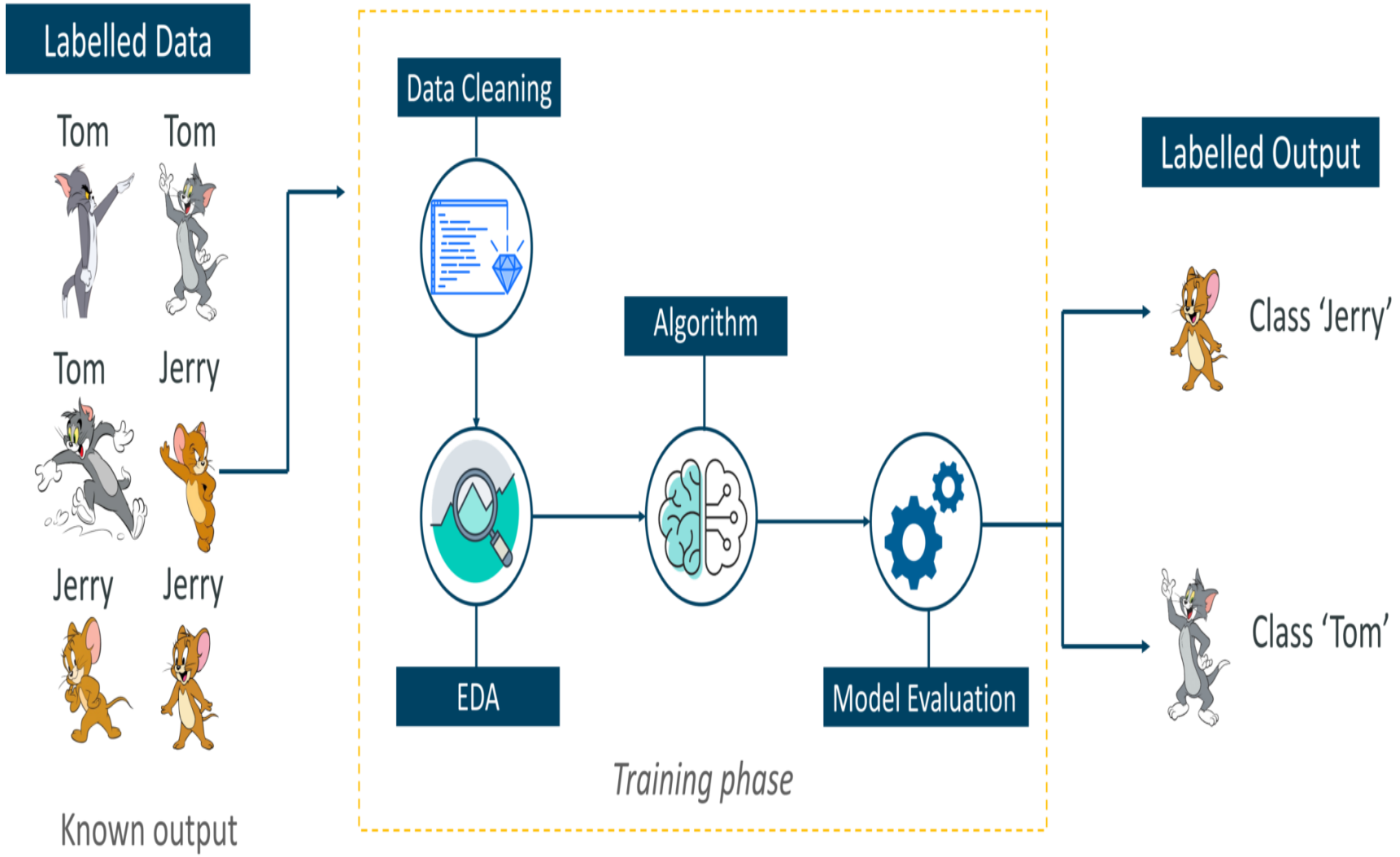
Introduction to Machine learning

- **Machine learning (ML)** allows computers to learn and make decisions without being explicitly programmed. It involves feeding data into algorithms to identify patterns and make predictions on new data.
- It is used in various applications like image recognition, speech processing, language translation, recommender systems, etc. In this article, we will see more about ML and its core concepts.

MACHINE LEARNING PROCESS

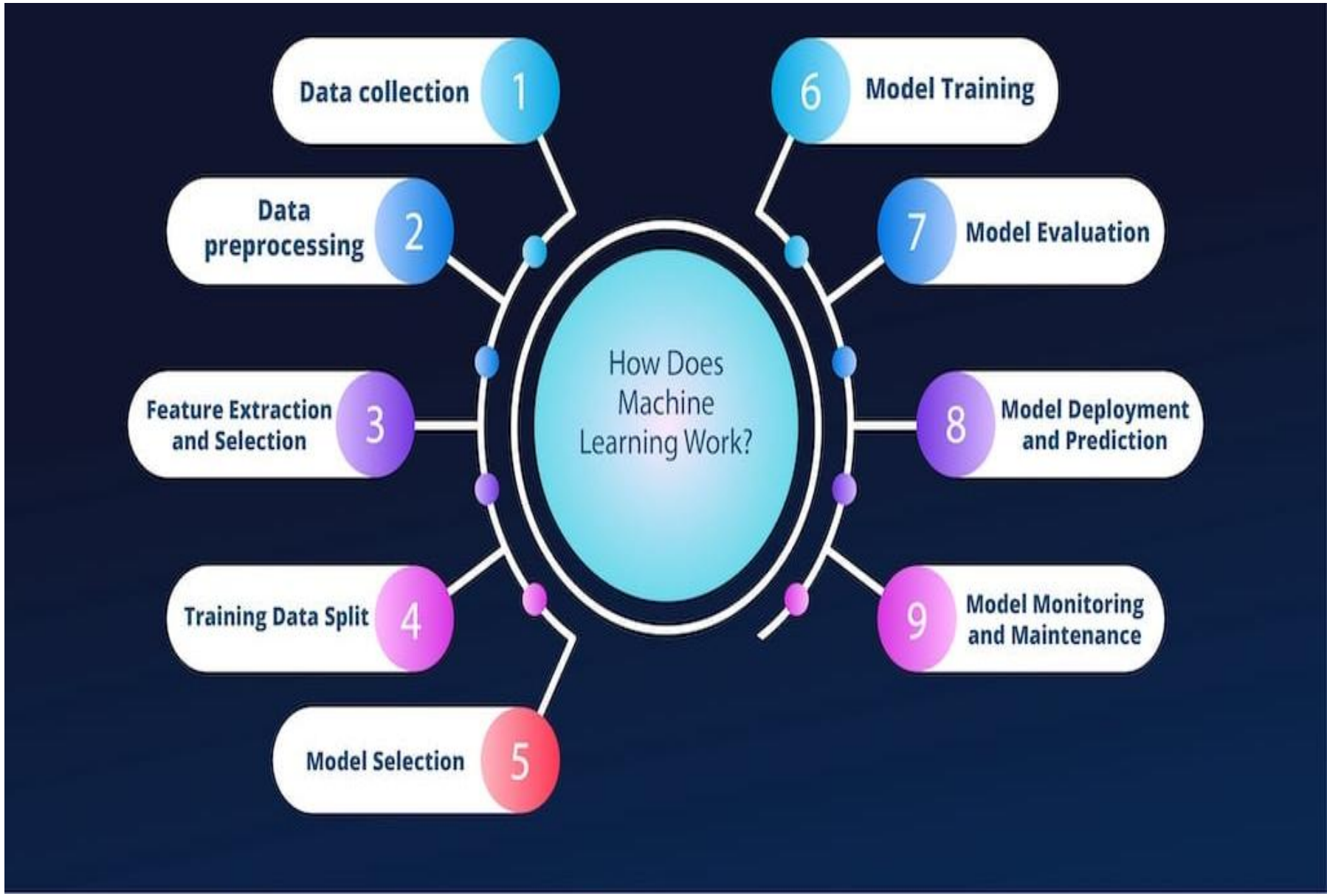


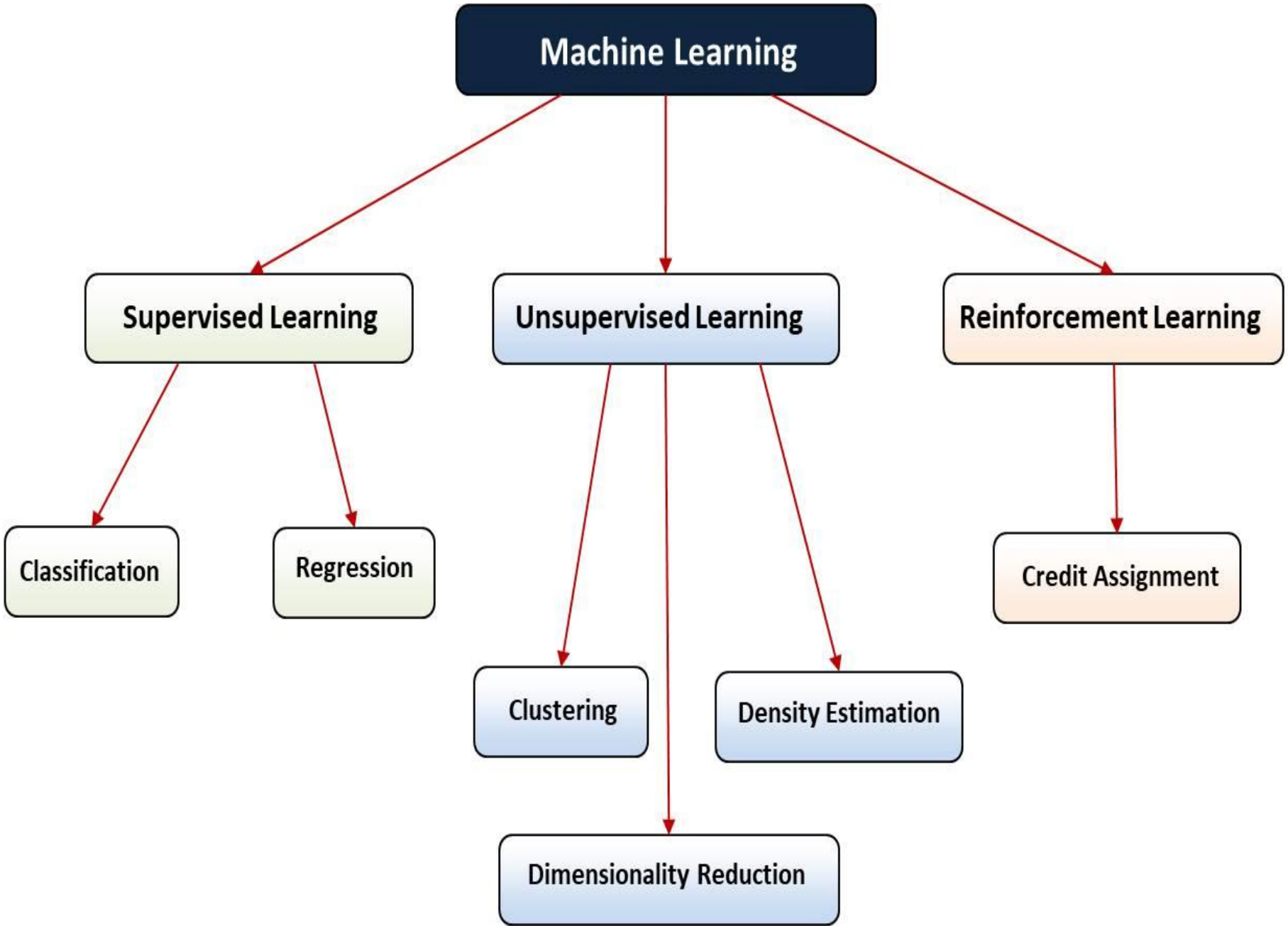
Contd..


Labelled Data
Tom
Tom
Tom
Jerry
Jerry
Jerry
Known output
Data Cleaning
Algorithm
EDA
Model Evaluation
Training phase
Labelled Output
Class 'Jerry'
Class 'Tom'

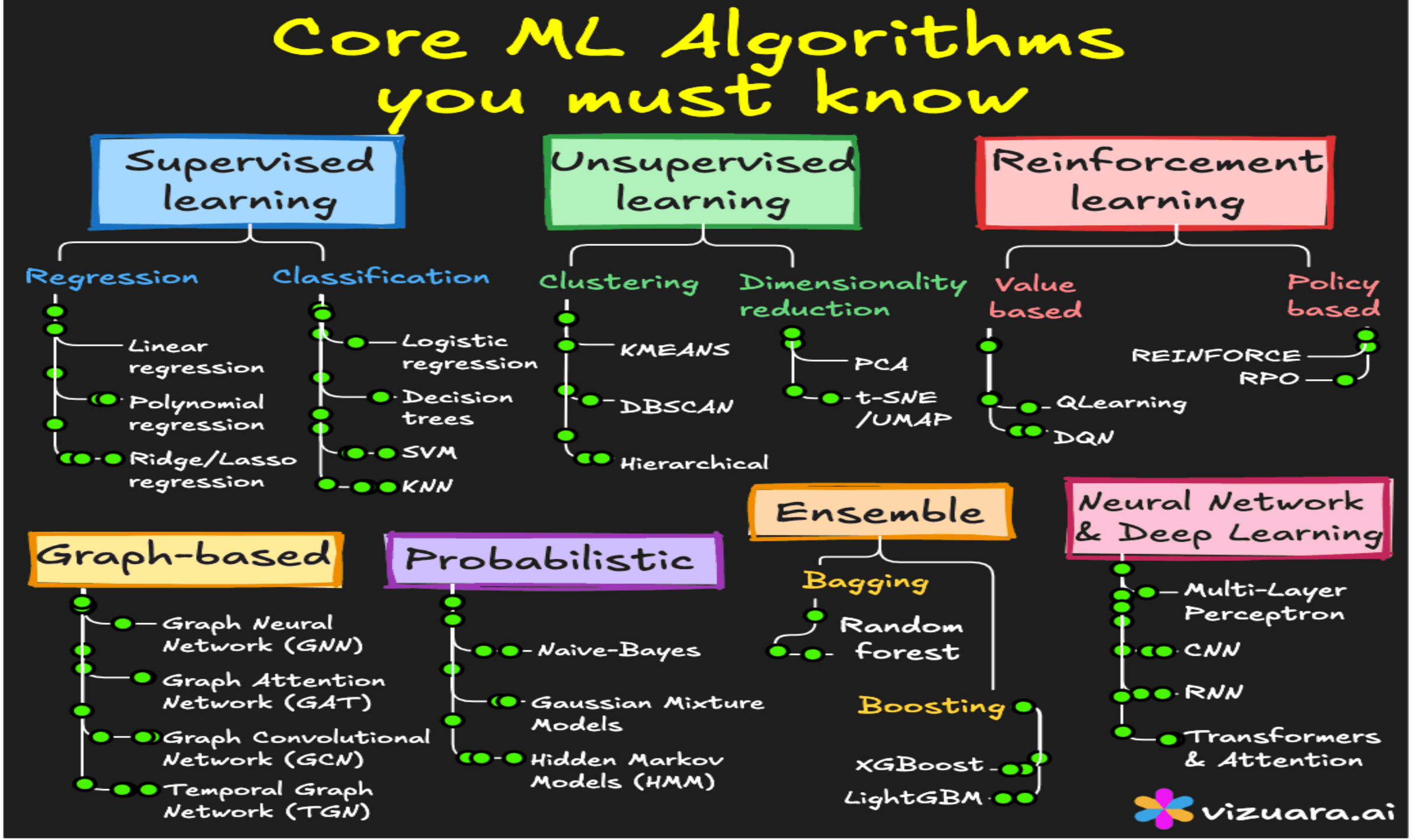
Why do we need Machine Learning?

- Traditional programming requires exact instructions and doesn't handle complex tasks like understanding images or language well. It can't efficiently process large amounts of data.
- Machine Learning solves these problems by learning from examples and making predictions without fixed rules.

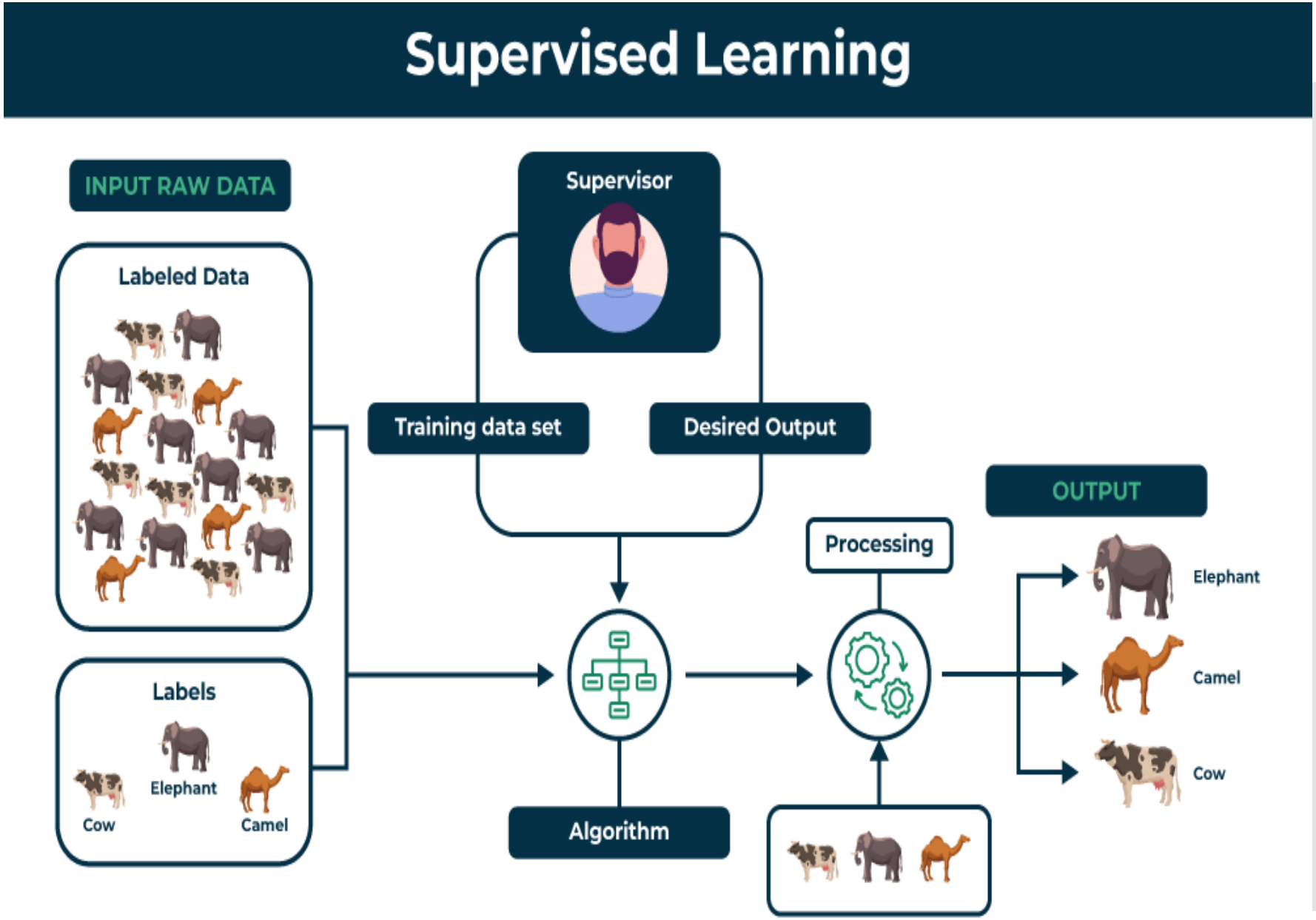

How Does Machine Learning Work?
1 Data collection
2 Data preprocessing
3 Feature Extraction and Selection
4 Training Data Split
5 Model Selection
6 Model Training
7 Model Evaluation
8 Model Deployment and Prediction
9 Model Monitoring and Maintenance


Machine Learning
Supervised Learning
Unsupervised Learning
Reinforcement Learning
Classification
Regression
Clustering
Density Estimation
Dimensionality Reduction
Credit Assignment

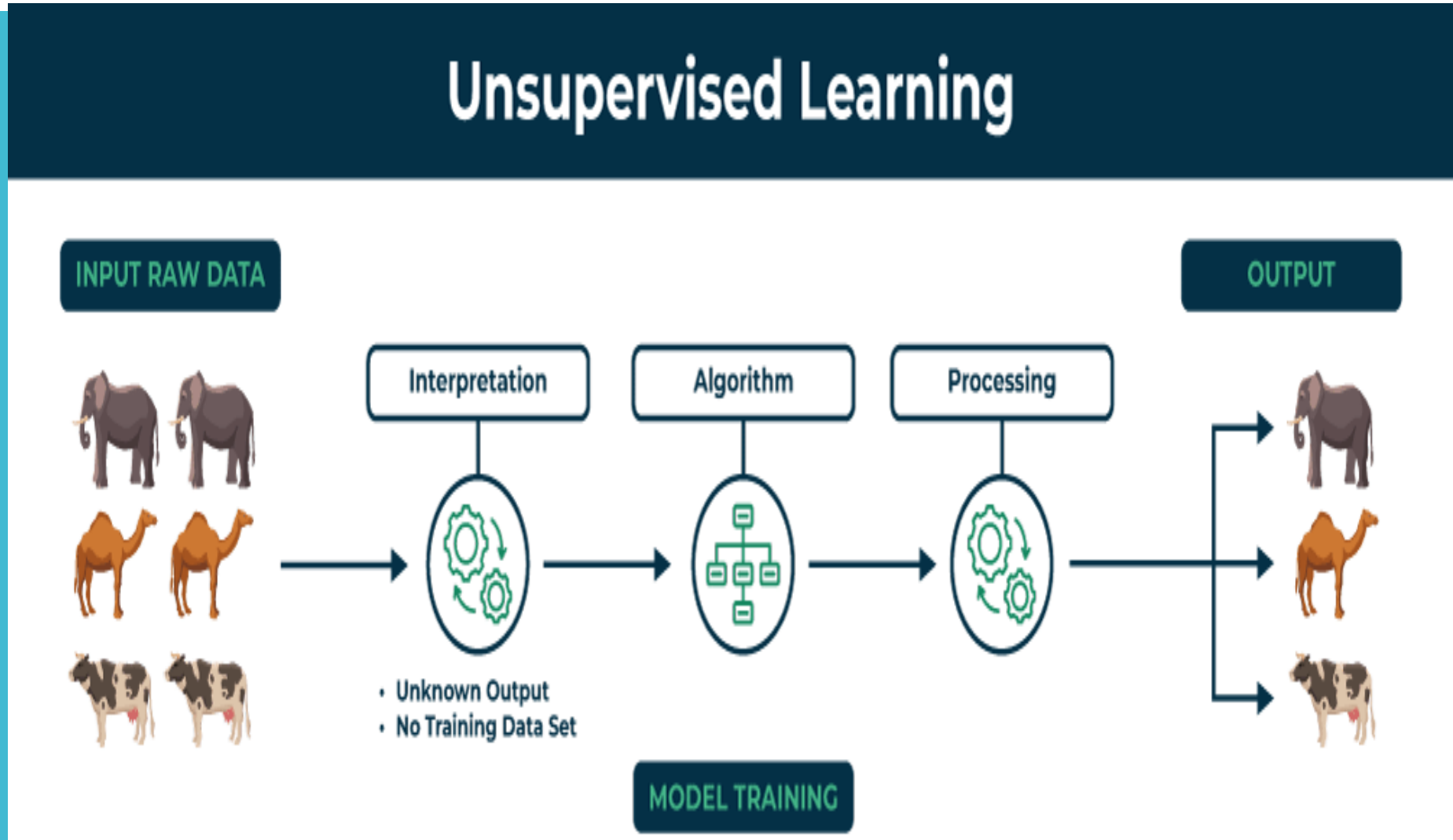
Core ML Algorithms you must know



Supervised learning



Unsupervised learning



SUPERVISED LEARNING

- It involves training a model using labeled data, where each input comes with a corresponding correct output. The process is like a teacher guiding a student—hence the term "supervised" learning.
- **supervised learning** is a type of machine learning where a model is trained on labeled data , meaning each input is paired with the correct output. the model learns by comparing its predictions with the actual answers provided in the training data.
- Over time, it adjusts itself to minimize errors and improve accuracy. The goal of supervised learning is to make accurate predictions when given new, unseen data.

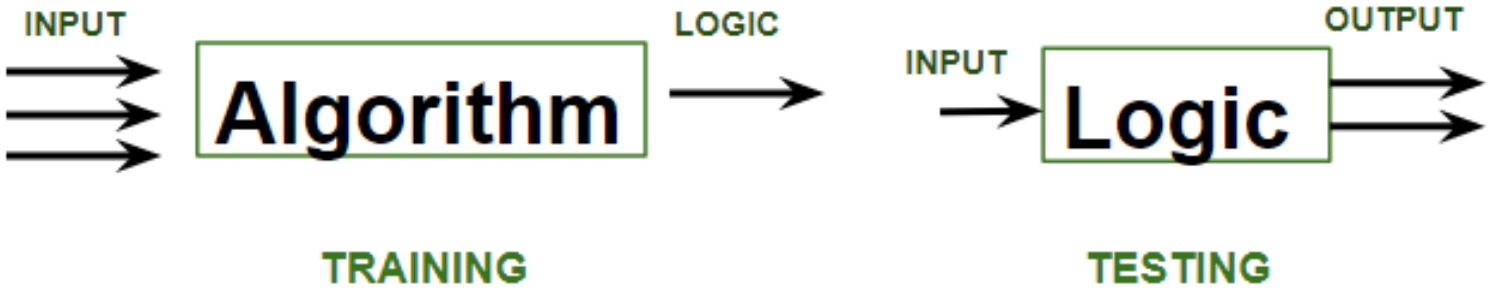
How Supervised Machine Learning Works?

- Where **supervised learning algorithm** consists of input features and corresponding output labels. The process works through:
- **Training Data:** The model is provided with a training dataset that includes input data (features) and corresponding output data (labels or target variables).
- **Learning Process:** The algorithm processes the training data, learning the relationships between the input features and the output labels. This is achieved by adjusting the model's parameters to minimize the difference between its predictions and the actual labels.

How Supervised Machine Learning Works?

- After training, the model is evaluated using a test dataset to measure its accuracy and performance. Then the model's performance is optimized by adjusting parameters and using techniques like cross-validation to balance bias and variance. This ensures the model generalizes well to new, unseen data.

How Supervised Machine Learning Works?



- **Training** phase involves feeding the algorithm labeled data, where each data point is paired with its correct output. The algorithm learns to identify patterns and relationships between the input and output data.

- **Testing** phase involves feeding the algorithm new, unseen data and evaluating its ability to predict the correct output based on the learned patterns.

Regression vs Classification

Feature	Classification	Regression
Output type	In this problem statement, the target variables are discrete. Discrete categories (e.g., "spam" or "not spam")	Continuous numerical value (e.g., price, temperature).
Goal	To predict which category a data point belongs to.	To predict an exact numerical value based on input data.
Example problems	Email spam detection, image recognition, customer sentiment analysis.	House price prediction, stock market forecasting, sales prediction.
Evaluation metrics	Evaluation metrics like Precision, Recall, and F1-Score	Mean Squared Error, R2-Score, , MAPE and RMSE.
Decision boundary	Clearly defined boundaries between different classes.	No distinct boundaries, focuses on finding the best fit line.
Common algorithms	Logistic regression, Decision trees, Support Vector Machines (SVM)	Linear Regression, Polynomial Regression, Decision Trees (with regression objective).

User ID	Gender	Age	Salary	Purchased
15624510	Male	19	19000	0
15810944	Male	35	20000	1
15668575	Female	26	43000	0
15603246	Female	27	57000	0
15804002	Male	19	76000	1
15728773	Male	27	58000	1
15598044	Female	27	84000	0
15694829	Female	32	150000	1
15600575	Male	25	33000	1
15727311	Female	35	65000	0
15570769	Female	26	80000	1
15606274	Female	26	52000	0
15746139	Male	20	86000	1
15704987	Male	32	18000	0
15628972	Male	18	82000	0
15697686	Male	29	80000	0
15733883	Male	47	25000	1

Figure A: CLASSIFICATION

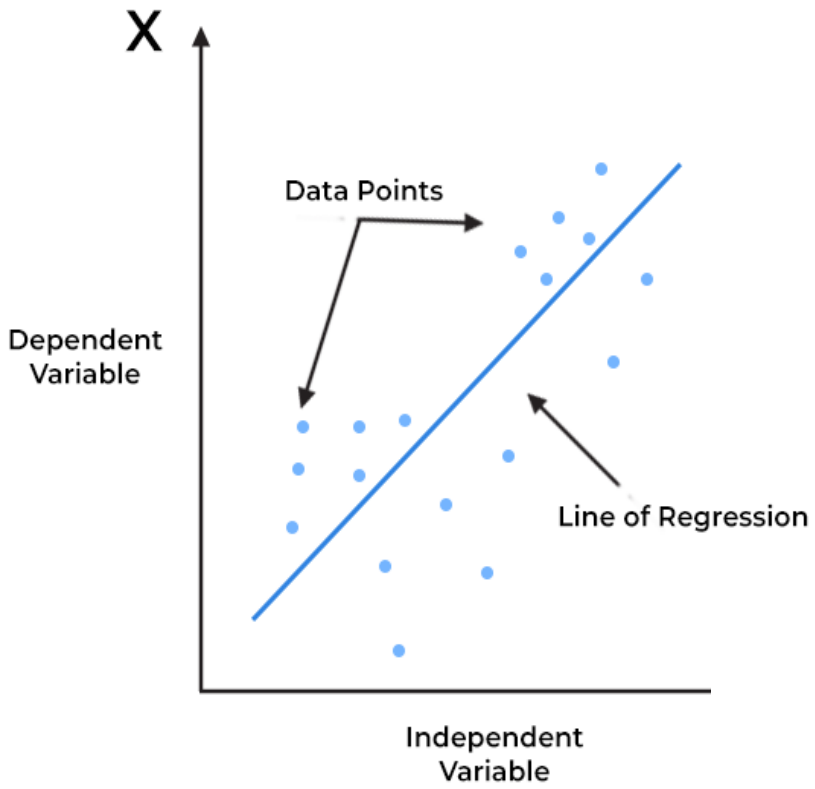
Temperature	Pressure	Relative Humidity	Wind Direction	Wind Speed
10.69261758	986.882019	54.19337313	195.7150879	3.278597116
13.59184184	987.8729248	48.0648859	189.2951202	2.909167767
17.70494885	988.1119385	39.11965597	192.9273834	2.973036289
20.95430404	987.8500366	30.66273218	202.0752869	2.965289593
22.9278274	987.2833862	26.06723423	210.6589203	2.798230886
24.04233986	986.2907104	23.46918024	221.1188507	2.627005816
24.41475295	985.2338867	22.25082295	233.7911987	2.448749781
23.93361956	984.8914795	22.35178837	244.3504333	2.454271793
22.68800023	984.8461304	23.7538641	253.0864716	2.418341875
20.56425726	984.8380737	27.07867944	264.5071106	2.318677425
17.76400389	985.4262085	33.54900114	280.7827454	2.343950987
11.25680746	988.9386597	53.74139903	68.15406036	1.650191426
14.37810685	989.6819458	40.70884681	72.62069702	1.553469896
18.45114201	990.2960205	30.85038484	71.70604706	1.005017161
22.54895853	989.9562988	22.81738811	44.66042709	0.264133632
24.23155922	988.796875	19.74790765	318.3214111	0.329656571

Figure B: REGRESSION

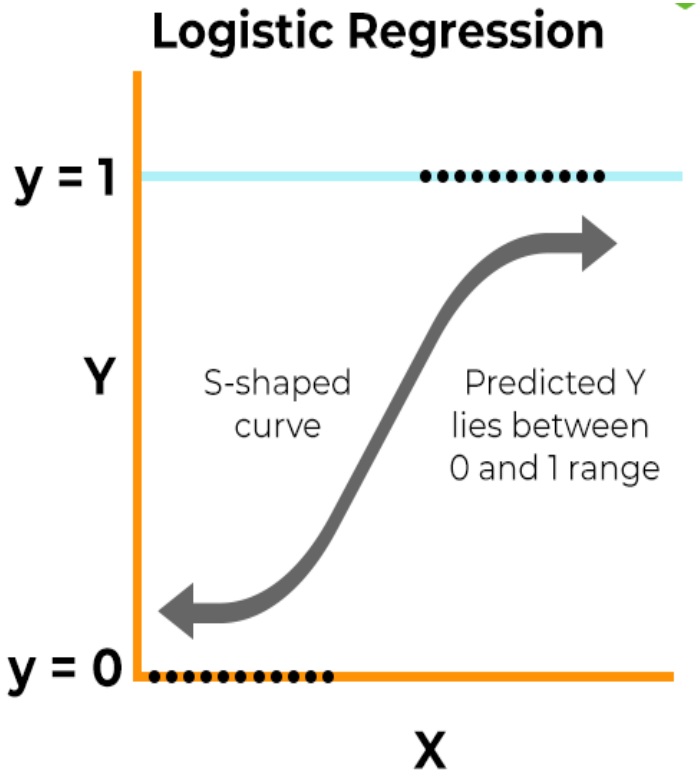
Practical Examples of Supervised learning

- **Fraud Detection in Banking:** Utilizes supervised learning algorithms on historical transaction data, training models with labeled datasets of legitimate and fraudulent transactions to accurately predict fraud patterns.
- **Parkinson Disease Prediction:** Parkinson's disease is a progressive disorder that affects the nervous system and the parts of the body controlled by the nerves.
- **Customer Churn Prediction:** Uses supervised learning techniques to analyze historical customer data, identifying features associated with churn rates to predict customer retention effectively.
- **Cancer cell classification:** Implements supervised learning for cancer cells based on their features, and identifying them if they are 'malignant' or 'benign.
- **Stock Price Prediction:** Applies supervised learning to predict a signal that indicates whether buying a particular stock will be helpful or not.

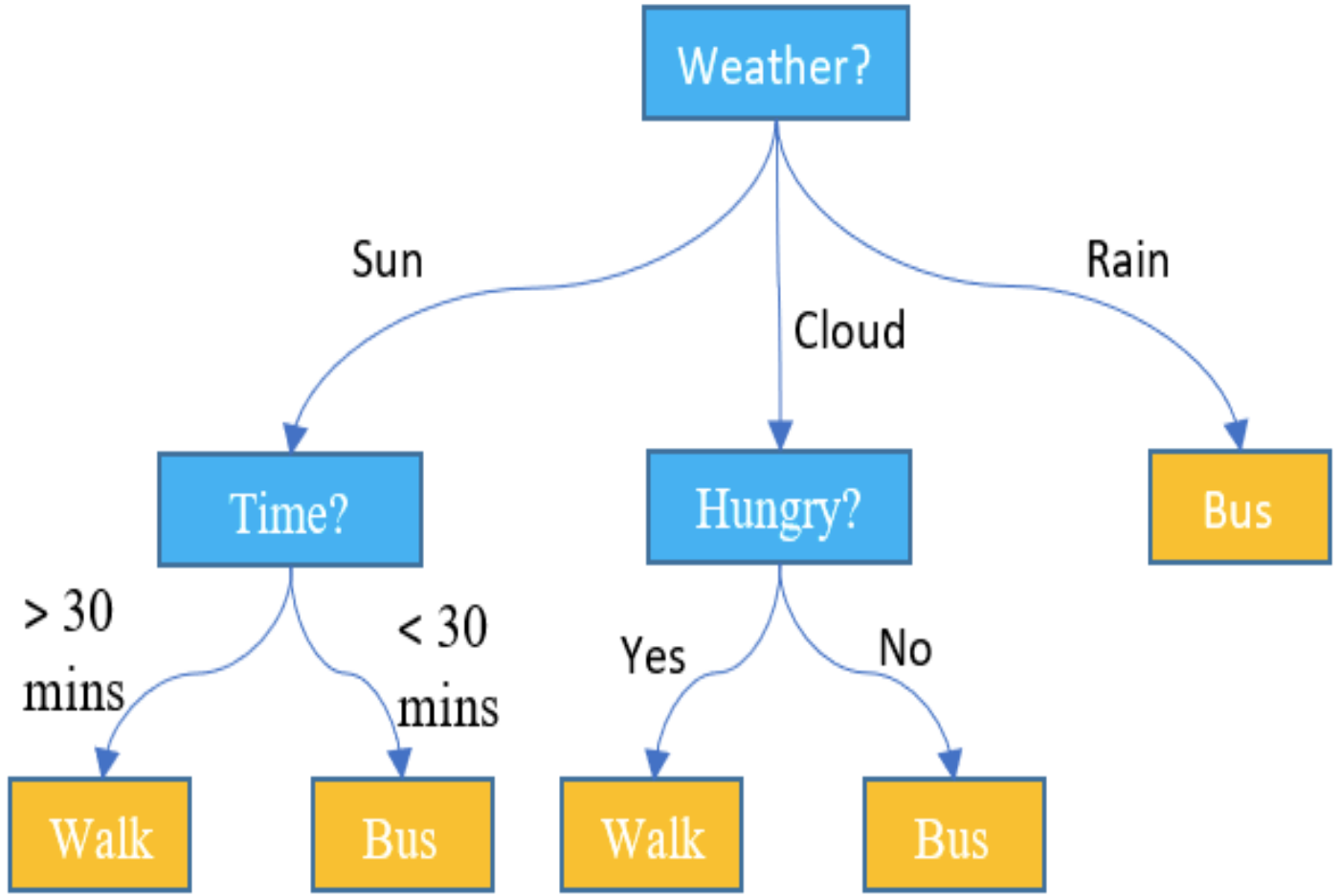
Linear Regression



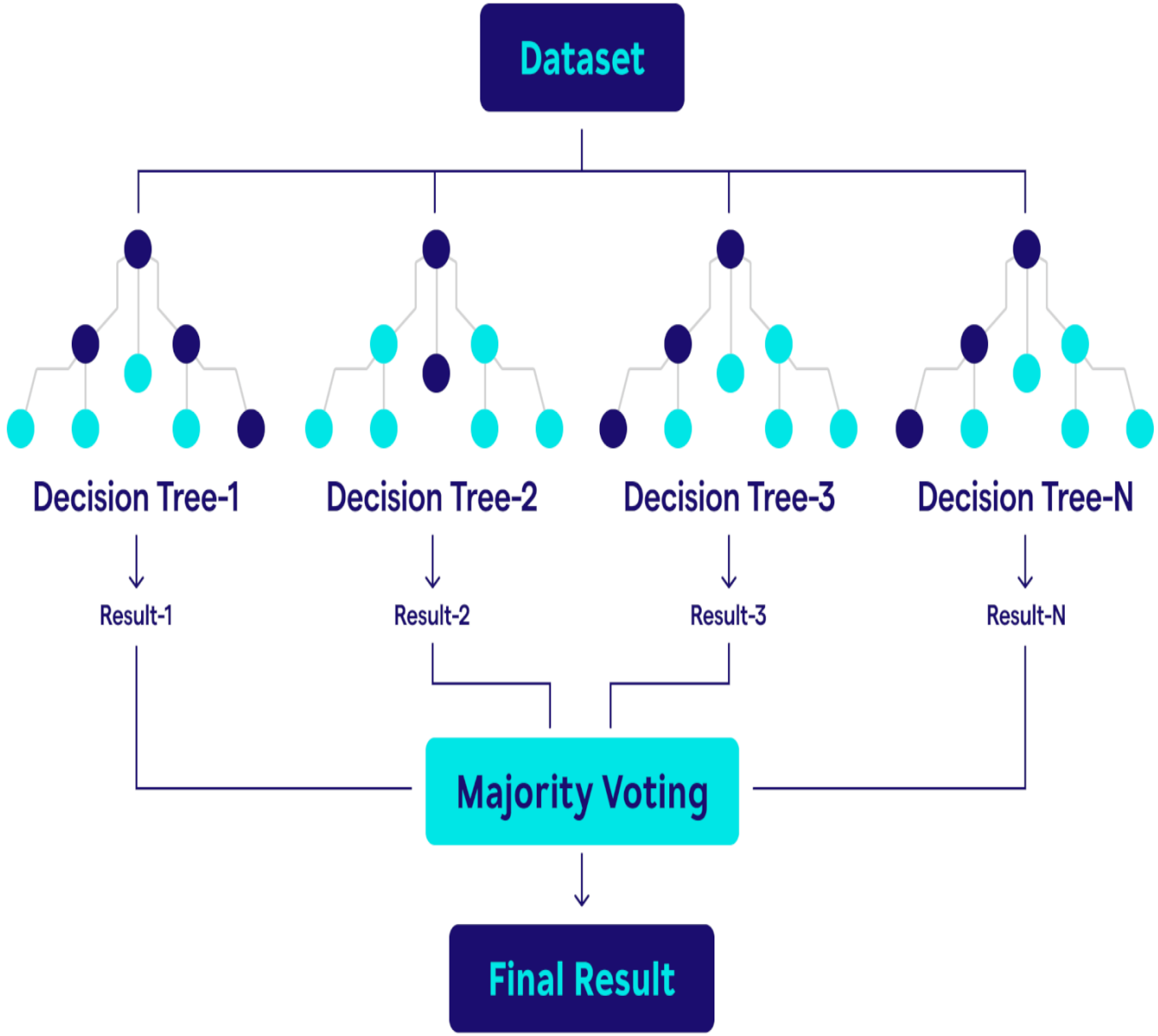
Logistic Regression

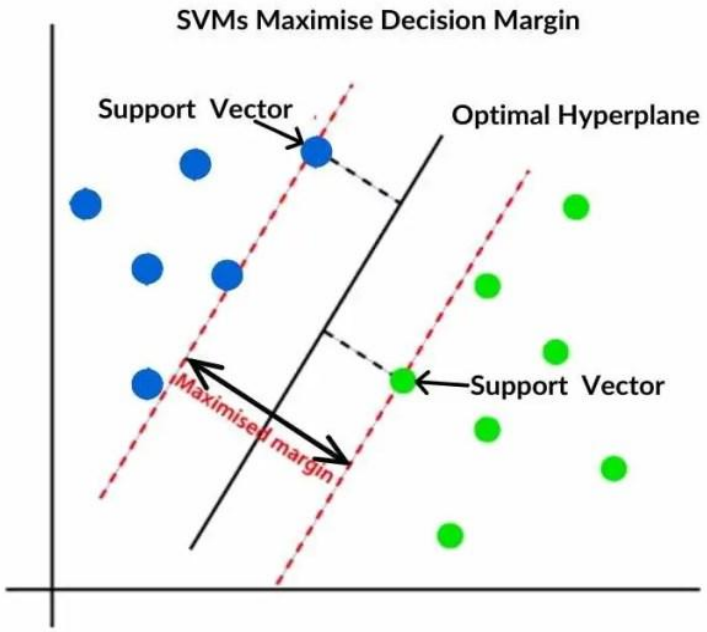


Decision Trees



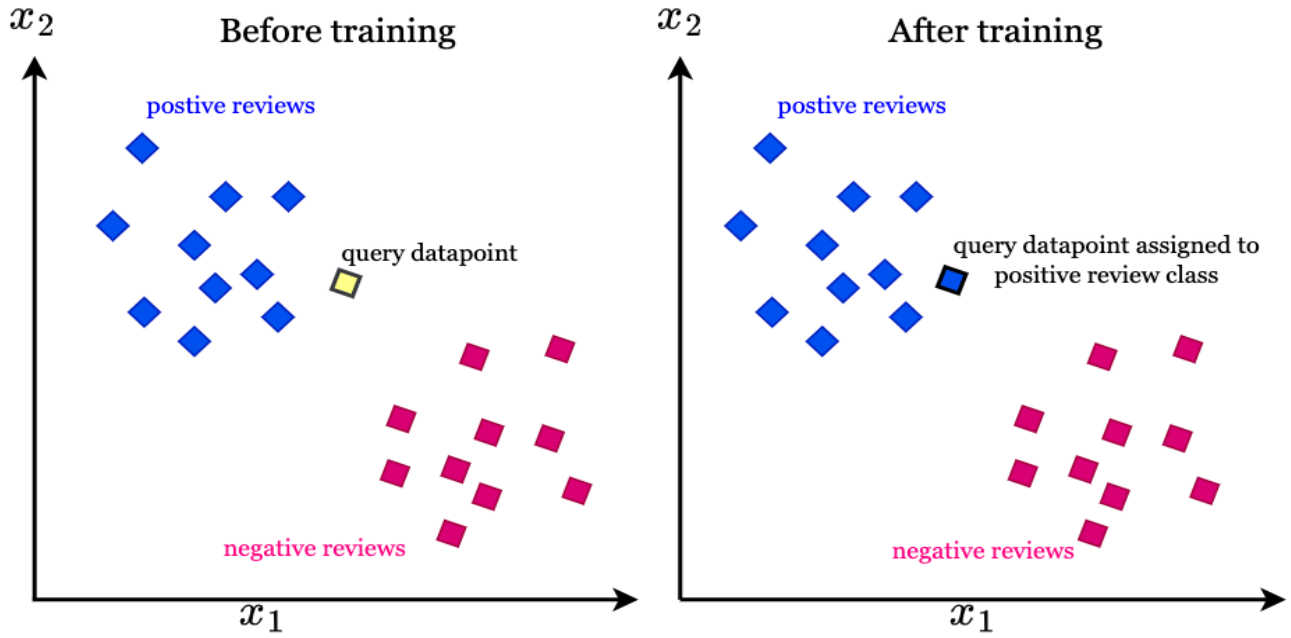
Random forest




SVMs Maximise Decision Margin
Support Vector
Optimal Hyperplane
Support Vector
Maximised margin

SVM

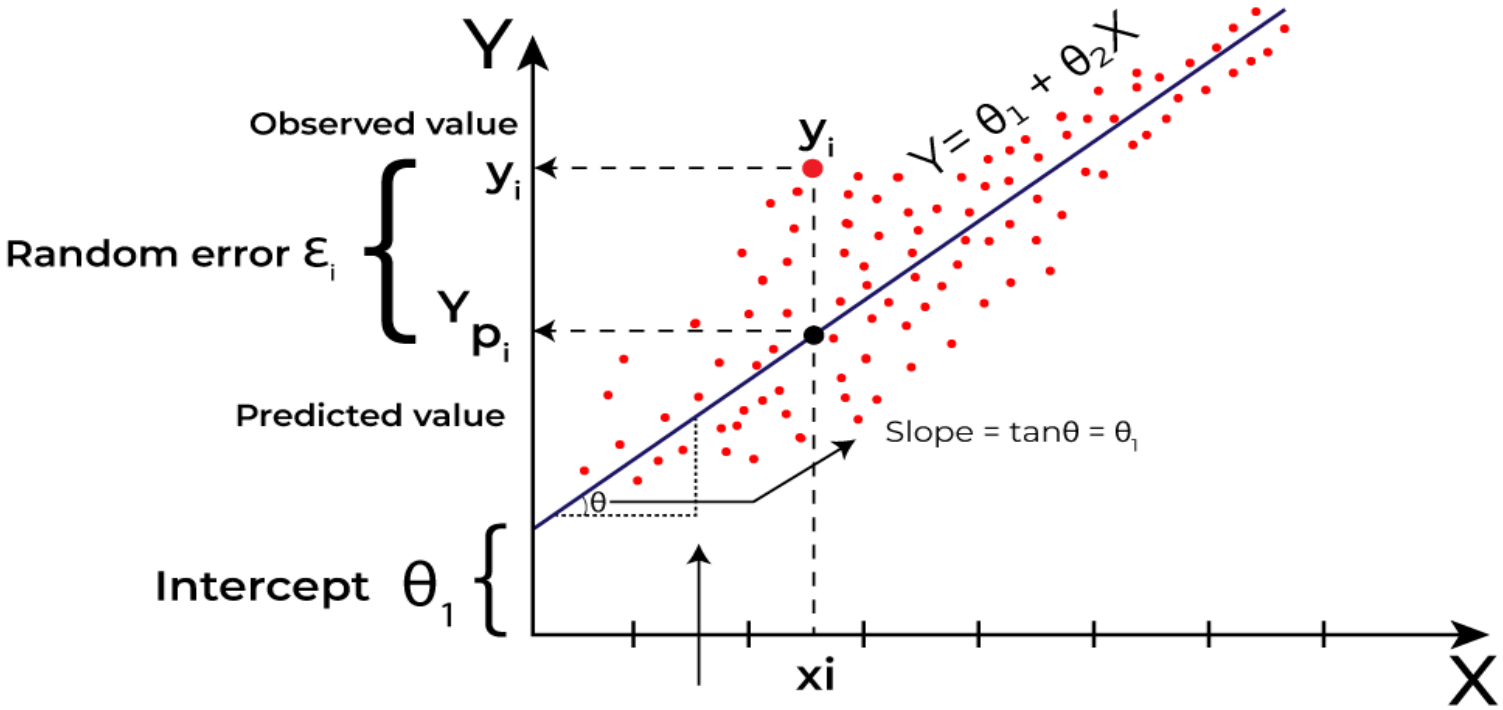
KNN


x_2
Before training
postive reviews
query datapoint
negative reviews
x_1
x_2
After training
postive reviews
query datapoint assigned to positive review class
negative reviews
x_1

Linear Regression

- Linear regression is a type of **supervised machine-learning algorithm** that learns from the labelled datasets and maps the data points with most optimized linear functions which can be used for prediction on new datasets.
- It assumes that there is a linear relationship between the input and output, meaning the output changes at a constant rate as the input changes. This relationship is represented by a straight line.
- **For example,** we want to predict a student's exam score based on how many hours they studied. We observe that as students study more hours, their scores go up. In the example of predicting exam scores based on hours studied. Here
- **Independent variable (input):** Hours studied because it's the factor we control or observe.
- **Dependent variable (output):** Exam score because it depends on how many hours were studied.

Best Fit Line in Linear Regression



- In linear regression, the best-fit line is the straight line that most accurately represents the relationship between the independent variable (input) and the dependent variable (output). It is the line that minimizes the difference between the actual data points and the predicted values from the model.

- The goal of linear regression is to find a straight line that minimizes the error (the difference) between the observed data points and the predicted values. This line helps us predict the dependent variable for new, unseen data.
- Here Y is called a dependent or target variable and X is called an independent variable also known as the predictor of Y.

Equation of the Best-Fit Line

- $y=mx+b$
- **y** is the predicted value (dependent variable)
- **x** is the input (independent variable)
- **m** is the slope of the line (how much y changes when x changes)
- **b** is the intercept (the value of y when x = 0)

- **The best-fit line will be the one that optimizes the values of m (slope) and b (intercept) so that the predicted y values are as close as possible to the actual data points.**

Simple linear regression

- Simple linear regression is used when we want to predict a target value (dependent variable) using only one input feature (independent variable). It assumes a straight-line relationship between the two.

$$\hat{y} = \theta_0 + \theta_1 x$$

Example:

Predicting a person's salary (y) based on their years of experience (x).

2. Multiple Linear Regression

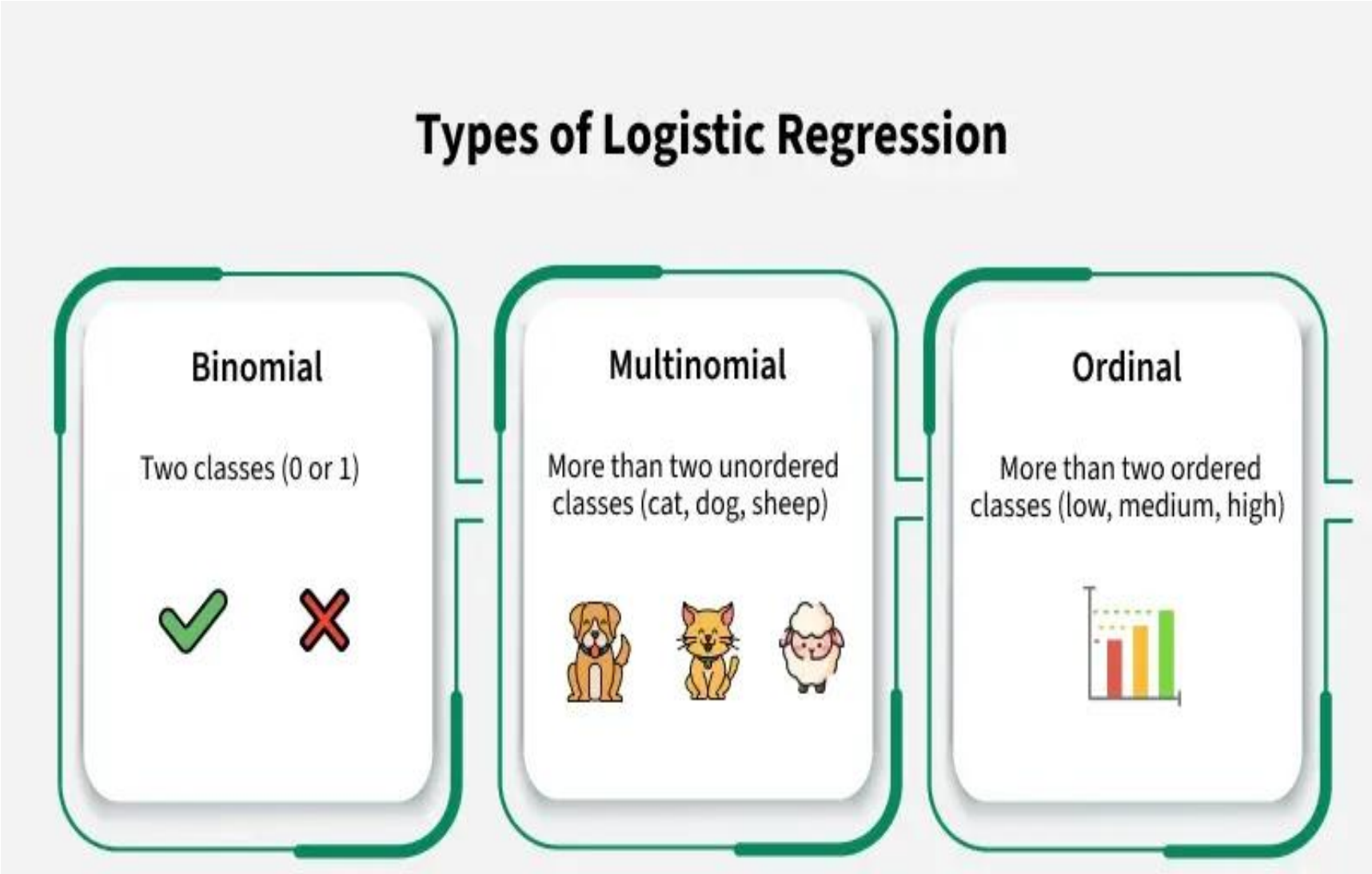
- Multiple linear regression involves more than one independent variable and one dependent variable. The equation for multiple linear regression is:

$$\hat{y} = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$

Logistic Regression

- Logistic Regression is a supervised machine learning algorithm used for classification problems.
- Unlike linear regression which predicts continuous values it predicts the probability that an input belongs to a specific class.
- It is used for binary classification where the output can be one of two possible categories such as Yes/No, True/False or 0/1.
- It uses sigmoid function to convert inputs into a probability value between 0 and 1.

Types of Logistic Regression



Types of Logistic Regression

- **Binomial Logistic Regression**: This type is used when the dependent variable has only two possible categories. Examples include Yes/No, Pass/Fail or 0/1. It is the most common form of logistic regression and is used for binary classification problems.
- **Multinomial Logistic Regression**: This is used when the dependent variable has three or more possible categories that are not ordered. For example, classifying animals into categories like "cat," "dog" or "sheep." It extends the binary logistic regression to handle multiple classes.
- **Ordinal Logistic Regression**: This type applies when the dependent variable has three or more categories with a natural order or ranking. Examples include ratings like "low," "medium" and "high." It takes the order of the categories into account when modeling.

Understanding Sigmoid Function

- 1. The sigmoid function is an important part of logistic regression which is used to convert the raw output of the model into a probability value between 0 and 1.
- 2. This function takes any real number and maps it into the range 0 to 1 forming an "S" shaped curve called the sigmoid curve or logistic curve. Because probabilities must lie between 0 and 1, the sigmoid function is perfect for this purpose.
- 3. In logistic regression, we use a threshold value usually 0.5 to decide the class label.
- If the sigmoid output is same or above the threshold, the input is classified as Class 1.
- If it is below the threshold, the input is classified as Class 0.

How does Logistic Regression work?

- Logistic regression model transforms the linear regression function continuous value output into categorical value output using a sigmoid function which maps any real-valued set of independent variables input into a value between 0 and 1. This function is known as the logistic function.
- Suppose we have input features represented as a matrix:
-

$$X = \begin{bmatrix} x_{11} & \cdots & x_{1m} \\ x_{21} & \cdots & x_{2m} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nm} \end{bmatrix}$$

- and the dependent variable is Y having only binary value i.e 0 or 1.

$$Y = \begin{cases} 0 & \text{if } Class\ 1 \\ 1 & \text{if } Class\ 2 \end{cases}$$

then, apply the multi-linear function to the input variables X.

$$z = (\sum_{i=1}^{n} w_i x_i) + b$$

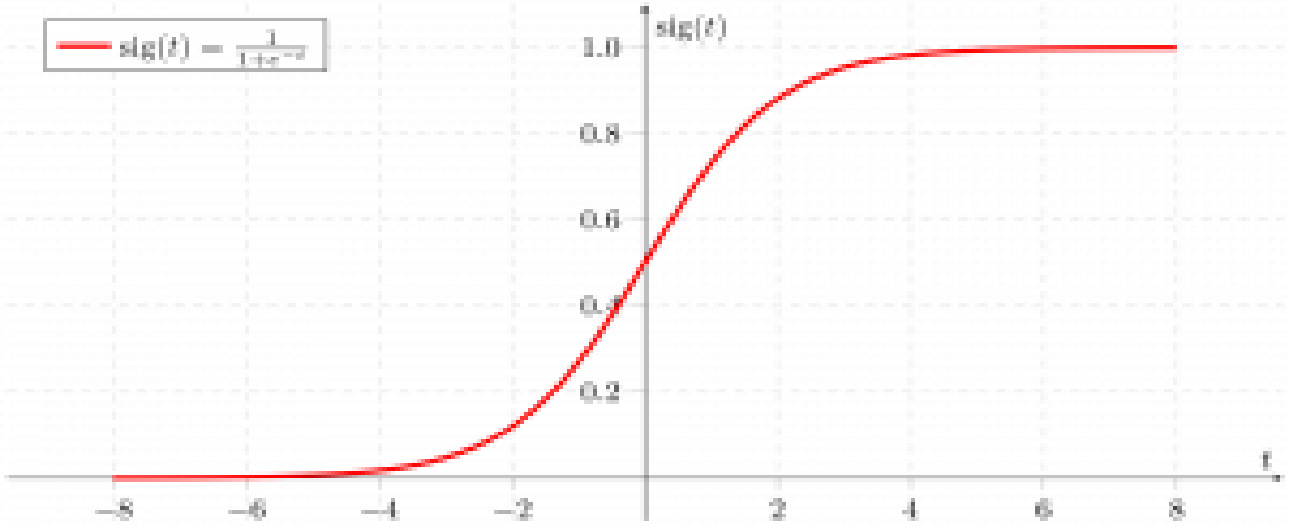
Here x_i is the ith observation of X, $w_i = [w_1, w_2, w_3, \cdots, w_m]$ is the weights or Coefficient and $b$is the bias term also known as intercept. Simply this can be represented as the dot product of weight and bias.

$$z = w \cdot X + b$$

- At this stage, z is a continuous value from the linear regression. Logistic regression then applies the sigmoid function to z to convert it into a probability between 0 and 1 which can be used to predict the class.

- Now we use the sigmoid function where the input will be z and we find the probability between 0 and 1. i.e. predicted y.

$$\sigma(z) = \frac{1}{1+e^{-z}}$$



- As shown above the sigmoid function converts the continuous variable data into the probability i.e between 0 and 1.
- $\sigma(z)$ tends towards 1 as $z \rightarrow \infty$
- $\sigma(z)$ tends towards 0 as $z \rightarrow -\infty$
- $\sigma(z)$ is always bounded between 0 and 1

- where the probability of being a class can be measured as:
- $P(y=1)=\sigma(z)$
- $P(y=0)=1-\sigma(z)$

Logistic Regression Equation and Odds:

- It models the odds of the dependent event occurring which is the ratio of the probability of the event to the probability of it not occurring:

$$\frac{p(x)}{1-p(x)} = e^z$$

Taking the natural logarithm of the odds gives the log-odds or logit:

$$\log\left[\frac{p(x)}{1-p(x)}\right] = z$$

$$\log\left[\frac{p(x)}{1-p(x)}\right] = w \cdot X + b$$

$$\frac{p(x)}{1-p(x)} = e^{w \cdot X + b} \quad \cdots \text{Exponentiate both sides}$$

$$p(x) = e^{w \cdot X + b} \cdot (1 - p(x))$$

$$p(x) = e^{w \cdot X + b} - e^{w \cdot X + b} \cdot p(x))$$

$$p(x) + e^{w \cdot X + b} \cdot p(x)) = e^{w \cdot X + b}$$

$$p(x)(1 + e^{w \cdot X + b}) = e^{w \cdot X + b}$$

$$p(x) = \frac{e^{w \cdot X + b}}{1 + e^{w \cdot X + b}}$$

then the final logistic regression equation will be:

$$p(X; b, w) = \frac{e^{w \cdot X + b}}{1 + e^{w \cdot X + b}} = \frac{1}{1 + e^{-w \cdot X + b}}$$

This formula represents the probability of the input belonging to Class 1.

Likelihood Function for Logistic Regression

- The goal is to find weights w and bias b that maximize the likelihood of observing the data.
- For each data point i
- for y=1, predicted probabilities will be: p(X;b,w) =p(x)
- for y=0,The predicted probabilities will be: 1-p(X;b,w) = 1−p(x)

$$L(b,w) = \prod_{i=1}^{n} p(x_i)^{y_i} (1 - p(x_i))^{1-y_i}$$

Taking natural logs on both sides:

$$
\begin{aligned}
\log(L(b,w)) &= \sum_{i=1}^{n} y_i \log p(x_i) \;+\; (1-y_i)\log(1-p(x_i)) \\
&= \sum_{i=1}^{n} y_i \log p(x_i) + \log(1-p(x_i)) - y_i \log(1-p(x_i)) \\
&= \sum_{i=1}^{n} \log(1-p(x_i)) + \sum_{i=1}^{n} y_i \log \frac{p(x_i)}{1-p(x_i} \\
&= \sum_{i=1}^{n} -\log 1 - e^{-(w\cdot x_i+b)} + \sum_{i=1}^{n} y_i(w\cdot x_i+b) \\
&= \sum_{i=1}^{n} -\log 1 + e^{w\cdot x_i+b} + \sum_{i=1}^{n} y_i(w\cdot x_i+b)
\end{aligned}
$$

This is known as the log-likelihood function.

Gradient of the log-likelihood function

- To find the best w and b we use gradient ascent on the log-likelihood function. The gradient with respect to each weight wj is:

$$\frac{\partial J(l(b,w)}{\partial w_j} = -\sum_{i=n}^{n} \frac{1}{1+e^{w\cdot x_i+b}} e^{w\cdot x_i+b} x_{ij} + \sum_{i=1}^{n} y_i x_{ij}$$

$$= -\sum_{i=n}^{n} p(x_i;b,w) x_{ij} + \sum_{i=1}^{n} y_i x_{ij}$$

$$= \sum_{i=n}^{n} (y_i - p(x_i;b,w)) x_{ij}$$

Linear vs Logistic Regression

Linear Regression	Logistic Regression
Linear regression is used to predict the continuous dependent variable using a given set of independent variables.	Logistic regression is used to predict the categorical dependent variable using a given set of independent variables.
It is used for solving regression problem.	It is used for solving classification problems.
In this we predict the value of continuous variables	In this we predict values of categorical variables
In this we find best fit line.	In this we find S-Curve.
Least square estimation method is used for estimation of accuracy.	Maximum likelihood estimation method is used for Estimation of accuracy.
The output must be continuous value, such as price, age etc.	Output must be categorical value such as 0 or 1, Yes or no etc.
It required linear relationship between dependent and independent variables.	It not required linear relationship.
There may be collinearity between the independent variables.	There should be little to no collinearity between independent variables.

Model Evaluation Metrics

- Model evaluation is a process that uses some metrics which help us to analyze the performance of the model. Think of training a model like teaching a student. **Model evaluation** is like giving them a test to see if they *truly* learned the subject—or just memorized answers.

Cross-Validation: The Ultimate Practice Test

- Cross Validation is a method in which we do not use the whole dataset for training.
- In this technique some part of the dataset is reserved for testing the model.
- There are many types of Cross-Validation out of which K Fold Cross Validation is mostly used. In **K Fold Cross Validation, the original dataset is divided into k subsets. The subsets are known as folds**. This is repeated k times where 1-fold is used for testing purposes, rest k-1 folds are used for training the model. It is seen that this technique generalizes the model well and reduces the error rate.

Holdout

- Holdout is the simplest approach. It is used in neural networks as well as in many classifiers. In this technique the dataset is divided into train and test datasets. The dataset is usually divided into ratios like 70:30 or 80:20. Normally a large percentage of data is used for training the model and a small portion of dataset is used for testing the model.

Evaluation Metrics for Classification Task

- To evaluate the performance of a classification model we commonly use metrics such as **accuracy, precision, recall, F1 score and confusion matrix.** These metrics are useful in assessing how well model distinguishes between classes especially in cases of imbalanced datasets.

1. Accuracy

- Accuracy is defined as the ratio of number of correct predictions to the total number of predictions. This is the most fundamental metric used to evaluate the model. The formula is given by:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

- However , Accuracy has a drawback. It cannot perform well on an imbalanced dataset.
- Suppose a model classifies that the majority of the data belongs to the major class label. It gives higher accuracy, but in general model cannot classify on minor class labels and has poor performance.

2. Precision and Recall

- **Precision** is the ratio of true positives to the summation of true positives and false positives. It basically analyses the positive predictions.

$$\text{Precision} = \frac{TP}{TP+FP}$$

- The drawback of Precision is that it does not consider the True Negatives and False Negatives.
- **Recall** is the ratio of true positives to the summation of true positives and false negatives. It basically analyses the number of correct positive samples.

$$\text{Recall} = \frac{TP}{TP+FN}$$

- The drawback of Recall is that often it leads to a higher false positive rate.

F1 Score

F1 score is the harmonic mean of precision and recall. It is seen that during the precision-recall trade-off if we increase the precision, recall decreases and vice versa. The goal of the F1 score is to combine precision and recall.

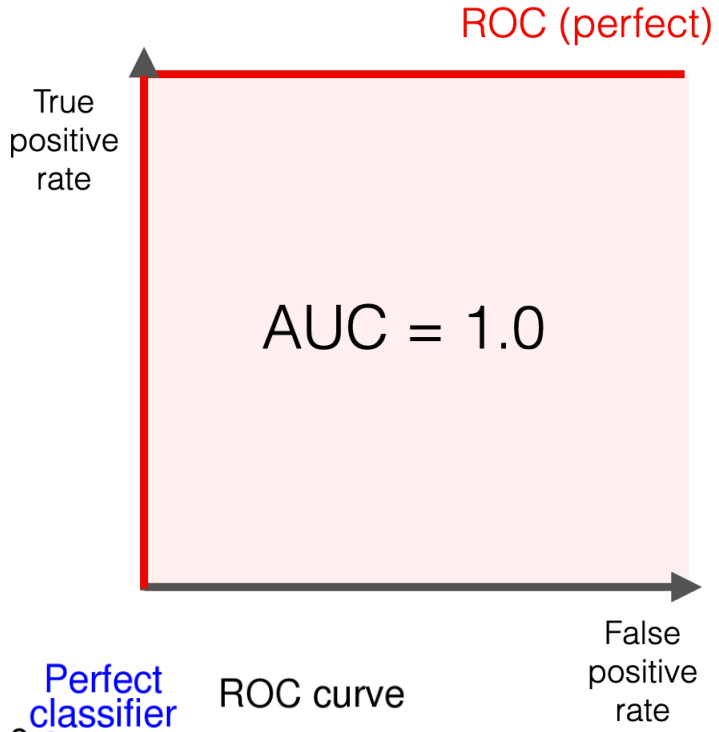
$$\text{F1 Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

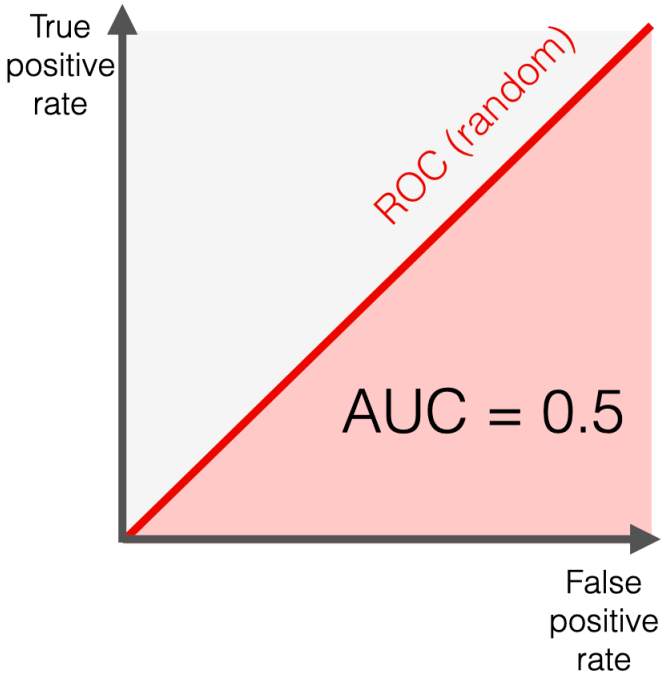
4. Confusion Matrix

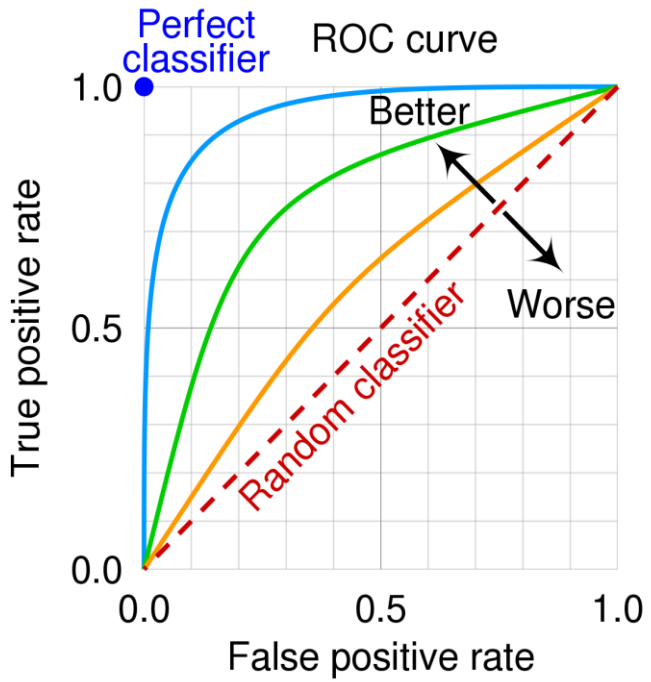
- Confusion matrix is a N x N matrix where N is the number of target classes. It represents number of actual outputs and predicted outputs. Some terminologies in the matrix are as follows:
- **True Positives**: It is also known as TP. It is the output in which the actual and the predicted values are YES.
- **True Negatives:** It is also known as TN. It is the output in which the actual and the predicted values are NO.
- **False Positives:** It is also known as FP. It is the output in which the actual value is NO but the predicted value is YES.
- **False Negatives:** It is also known as FN. It is the output in which the actual value is YES but the predicted value is NO.

5. AUC-ROC Curve

- AUC (Area Under Curve) is an evaluation metric that is used to analyze the classification model at different threshold values. The Receiver Operating Characteristic (ROC) curve is a probabilistic curve used to highlight the model's performance. The curve has two parameters:
- **TPR:** It stands for True positive rate. It basically follows the formula of Recall.
- **FPR:** It stands for False Positive rate. It is defined as the ratio of False positives to the summation of false positives and True negatives.
- **This curve is useful as it helps us to determine the model's capacity to distinguish between different classes.**
- **A model is considered good if the AUC score is greater than 0.5 and approaches 1.**


ROC (perfect)
True positive rate
AUC = 1.0
False positive rate


True positive rate
ROC (random)
AUC = 0.5
False positive rate


Perfect classifier
ROC curve
1.0
Better
Worse
Random classifier
0.5
0.0
True positive rate
0.0
0.5
1.0
False positive rate

Evaluation Metrics for Regression Task

- for regression analysis since we are predicting a numerical value it may differ from the actual output. So, we consider the **error calculation** as it helps to summarize how close the prediction is to the actual value. There are many metrics available for evaluating the regression model.

1. Mean Absolute Error (MAE)

- This is the simplest metric used to analyze the loss over the whole dataset. As we know that error is basically the difference between the predicted and actual values. Therefore, MAE is defined as the average of the errors calculated. Here we calculate the modulus of the error, perform summation and then divide the result by the total number of data points. It is a positive value. The formula of MAE is given by

$$MAE = \frac{\sum_{i=1}^{N} |y_{pred} - y_{actual}|}{N}$$

2. Mean Squared Error(MSE)

- The most commonly used metric is Mean Square error or MSE. It is a function used to calculate the loss. We find the difference between the predicted values and actual variable, square the result and then find the average by all datapoints present in dataset. MSE is always positive as we square the values. Small the value of MSE better is the performance of our model. The formula of MSE is given:

$$MSE = \frac{\sum_{i=1}^{N}(y_{pred} - y_{actual})^2}{N}$$

3. Root Mean Squared Error(RMSE)

- RMSE is a popular method and is the extended version of MSE. It indicates how much the data points are spread around the best line. It is the standard deviation of the MSE. A lower value means that the data point lies closer to the best fit line.

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{N}(y_{pred} - y_{actual})^2}{N}}$$

4. Mean Absolute Percentage Error (MAPE)

- MAPE is used to express the error in terms of percentage. It is defined as the difference between the actual and predicted value. The error is then divided by the actual value. The results are then summed up and finally and we calculate the average. Smaller the percentage better the performance of the model. The formula is given by

$$\text{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{|y_{pred} - y_{actual}|}{|y_{actual}|}\right) \times 100\%$$