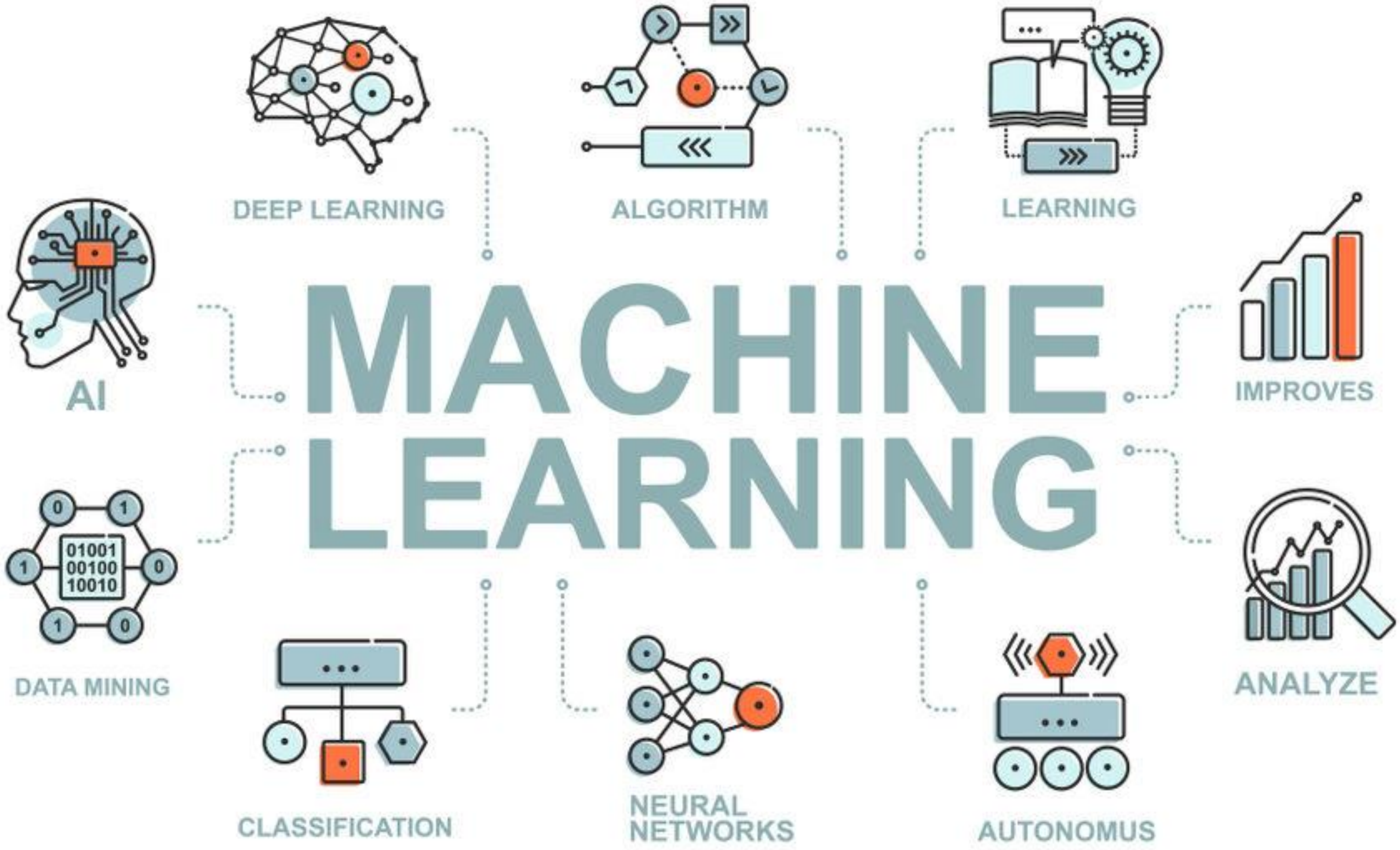


MACHINE LEARNING
DEEP LEARNING
ALGORITHM
LEARNING
AI
IMPROVES
DATA MINING
ANALYZE
CLASSIFICATION
NEURAL NETWORKS
AUTONOMUS
01001
00100
10010

Machine Learning (ML)

ML is an application of AI that provides **Systems or Models** the ability to automatically learn **and improve from experience without being explicitly programmed and** without human intervention .

Machine Learning (ML) is a **subfield** of artificial intelligence (**AI**) that focuses on the development of algorithms and models that enable computers to learn and make predictions or decisions without being explicitly programmed for a specific task.

The primary aim of machine learning is to allow the computers learn automatically without human intervention or assistance and adjust actions accordingly.

Examples:
1. The Self-Driving Car
2. The Spam Filter
3. Music Recommender
4. Personal Assistance (Alexa)

Definition:

According to **Tom Mitchell** Machine Learning **is the study of algorithms that improve** their **performance** P at some task T with experience E. A well-defined learning task for a system is given by <P, T, E>.

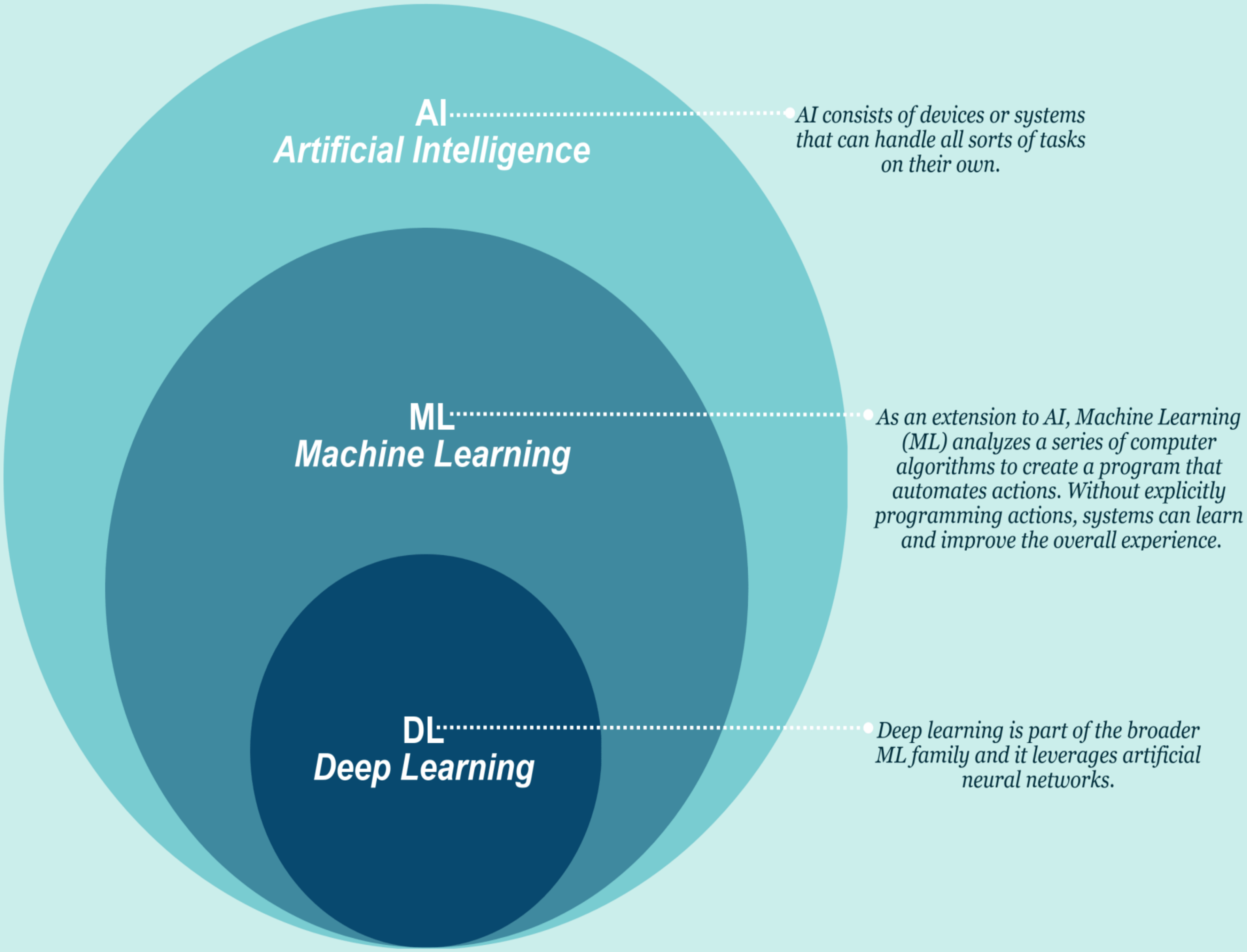
AI, ML and DL:

AI: Creation of Intelligent Machines that **think and act like** a Humans Ex: **Robots**.

ML: Approach for Computers to **learn without being explicitly programmed.**

DL: Training of Machines to think like a **human brains-** Application of Artificial Neural Networks.

AI v/s ML v/s DL


AI
Artificial Intelligence
AI consists of devices or systems that can handle all sorts of tasks on their own.
ML
Machine Learning
As an extension to AI, Machine Learning (ML) analyzes a series of computer algorithms to create a program that automates actions. Without explicitly programming actions, systems can learn and improve the overall experience.
DL
Deep Learning
Deep learning is part of the broader ML family and it leverages artificial neural networks.

Traditional Programming V/S Machine Learning :

Traditional Programming:

In traditional programming, a computer engineer writes a series of directions that instruct a computer how to transform input data into a desired output. Instructions are mostly based on an IF-THEN structure: when certain conditions are met, the program executes a specific action.

Machine Learning:

Machine learning, on the other hand, is an **automated process** that enables machines to solve problems **with little or no human input**, and take actions based on past observations.

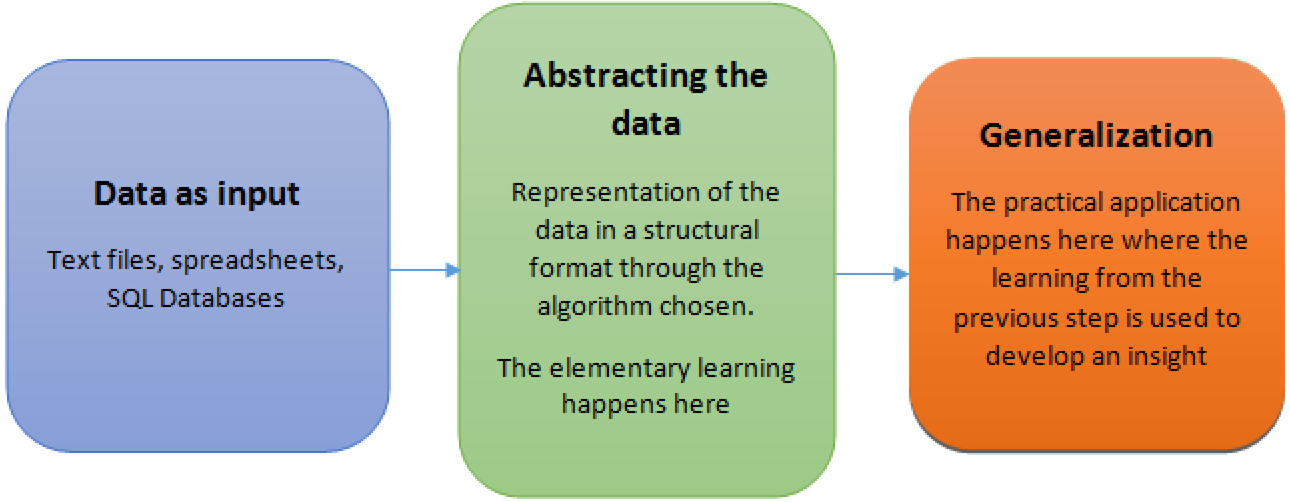
Basic machine learning process:

The basic machine learning process can be divided into **three** parts.

1. **Data Input:** Past data or information is utilized as a basis for future decision-making

2. **Abstraction (**Creating a **Internal representation of the data** by capturing the essential information)**:** The input data is represented in a broader way through the underlying algorithm

3. **Generalization:** Generalization refers to the ability of a **model to perform well on new, unseen data that it hasn't been trained on**.



Data Input:

1. This is the **foundation** of any machine learning task. Data can come from **various sources** like sensors, databases, logs, text files, and images.

2. The data should be **relevant** to the problem you're trying to solve, **diverse enough** to capture different scenarios, and **accurate** to **avoid misleading the model.**

3. Often, data needs **pre-processing** before feeding it into the model. This can involve cleaning, formatting, feature engineering (creating new features), and handling missing values.

Note:
The **quality** and **quantity** of the input data significantly impact the effectiveness of the machine learning model.

Abstraction: (Training the model):

- During the machine learning process, knowledge is fed in the form of input data.
- This is where the magic happens! The chosen machine learning algorithm analyzes the **processed** data, searching for patterns, relationships, and underlying structures.
- The algorithm creates an **internal representation of the data**, capturing the essential information it needs for prediction. This abstraction can be mathematical equations, rules, decision trees, or even complex neural networks.
- Think of it like **summarizing a book** into key themes and plot points; the abstraction captures the **essence** or **core** of the information without needing all the details.

The choice of the model used to solve a specific learning problem **is a human task**. The decision related to the choice of model is taken based on multiple aspects, some of which are listed below:

1. **The type of problem to be solved:** Whether the problem is related to prediction or forecast , analysis of trend, understanding the different segments or groups of objects, etc.

2. **Nature of the input data:** How exhaustive the input data is, whether the data has no values for many fields, the **data types** etc.

3. **Domain of the problem:** If the problem is in a business critical domain with a high rate of data input and need for immediate inference, e.g. fraud detection problem in banking domain.

Generalization:

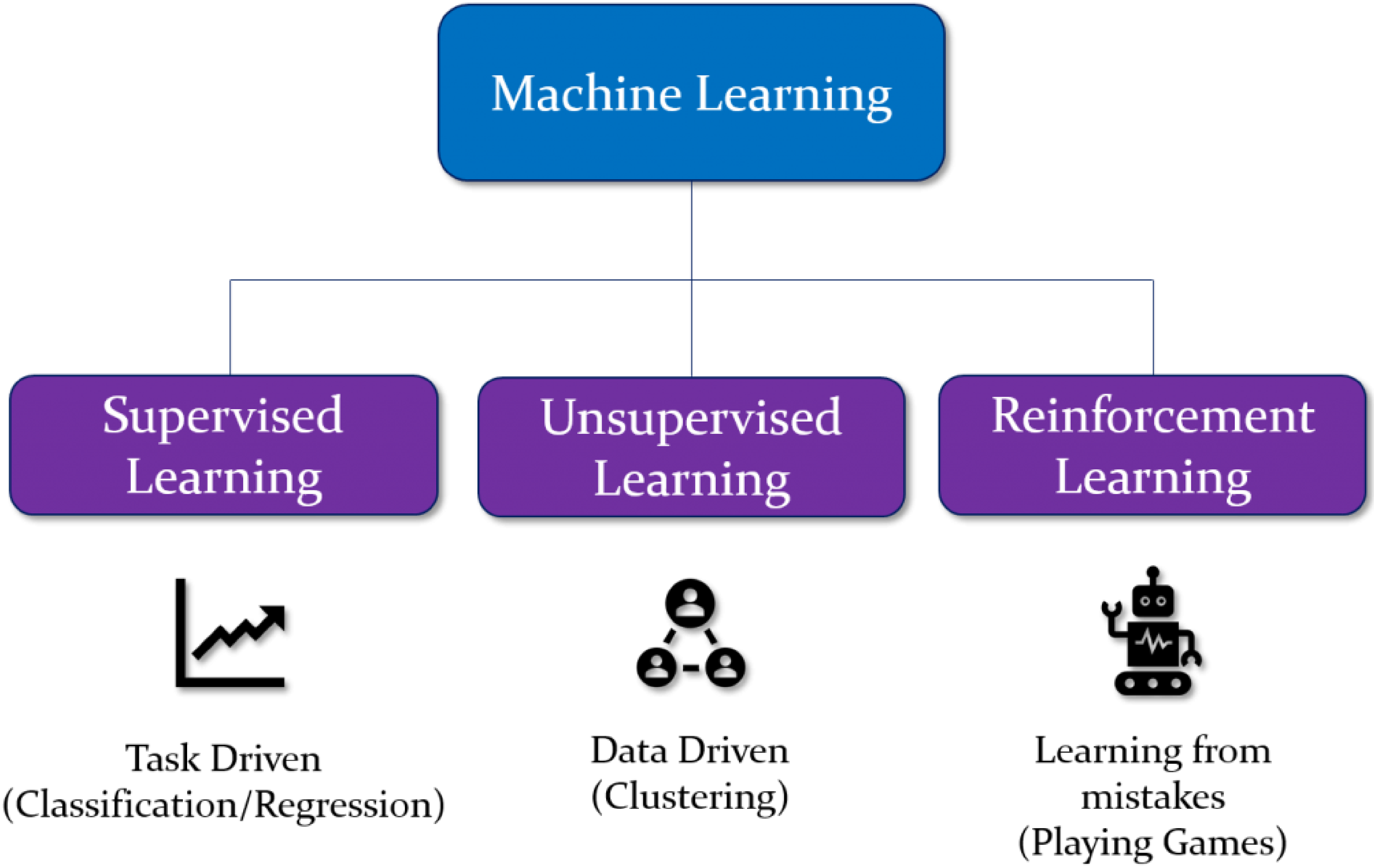
Generalization refers to the ability of a **model to perform well on new, unseen data that it hasn't been trained on**. The ultimate goal of a machine learning model is not just to memorize the training data but to learn underlying patterns and relationships that can be applied to new, unseen data in a meaningful way.

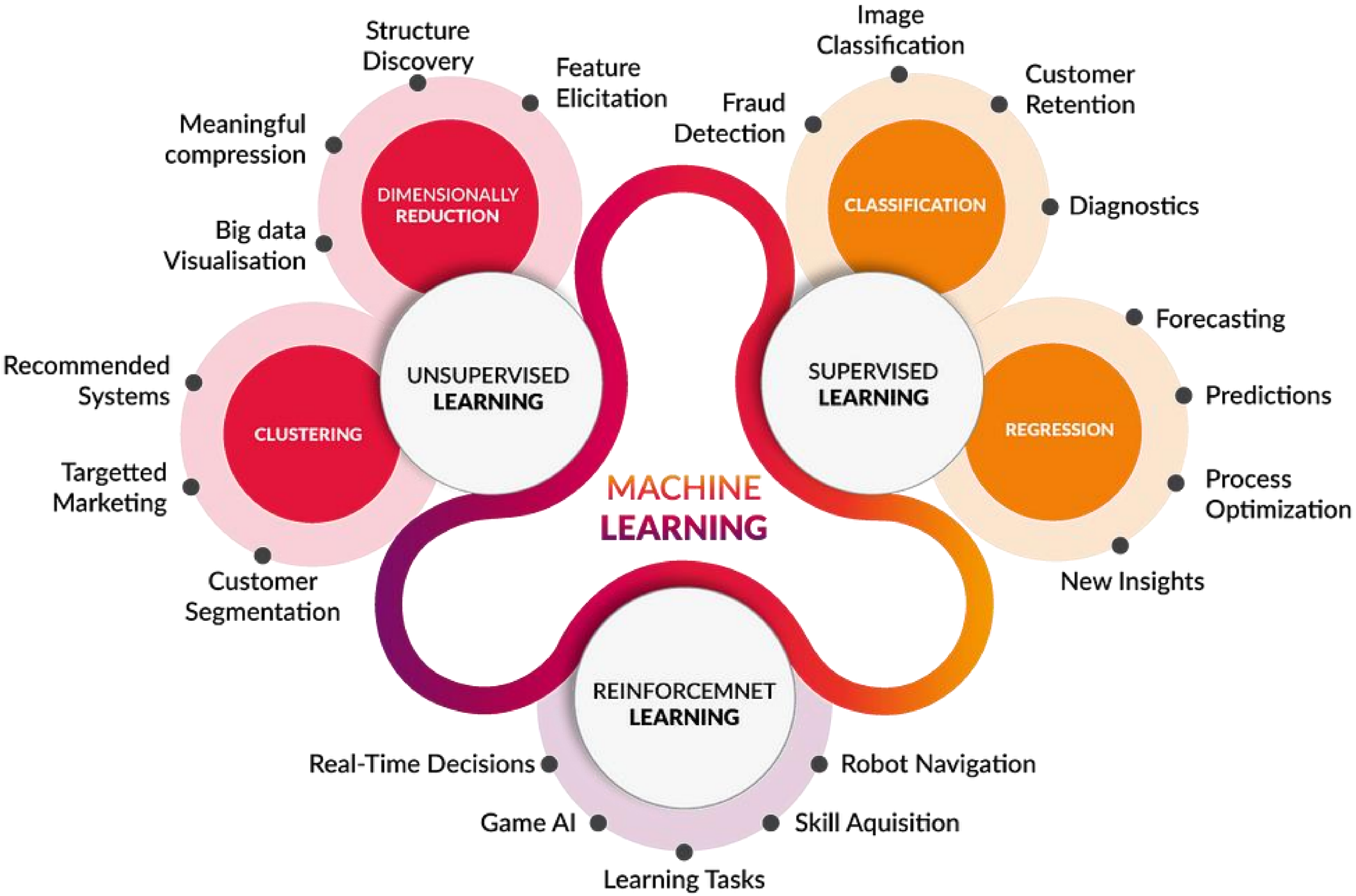
The first part of machine learning process is abstraction i.e. **abstract the knowledge** which comes as input data in the form of a model. However, this abstraction process, or **more popularly training the model**, is just one part of machine learning. The other key part is to **tune up** the abstracted knowledge to a form which can be used to take future decisions. This is achieved as a part of generalization.

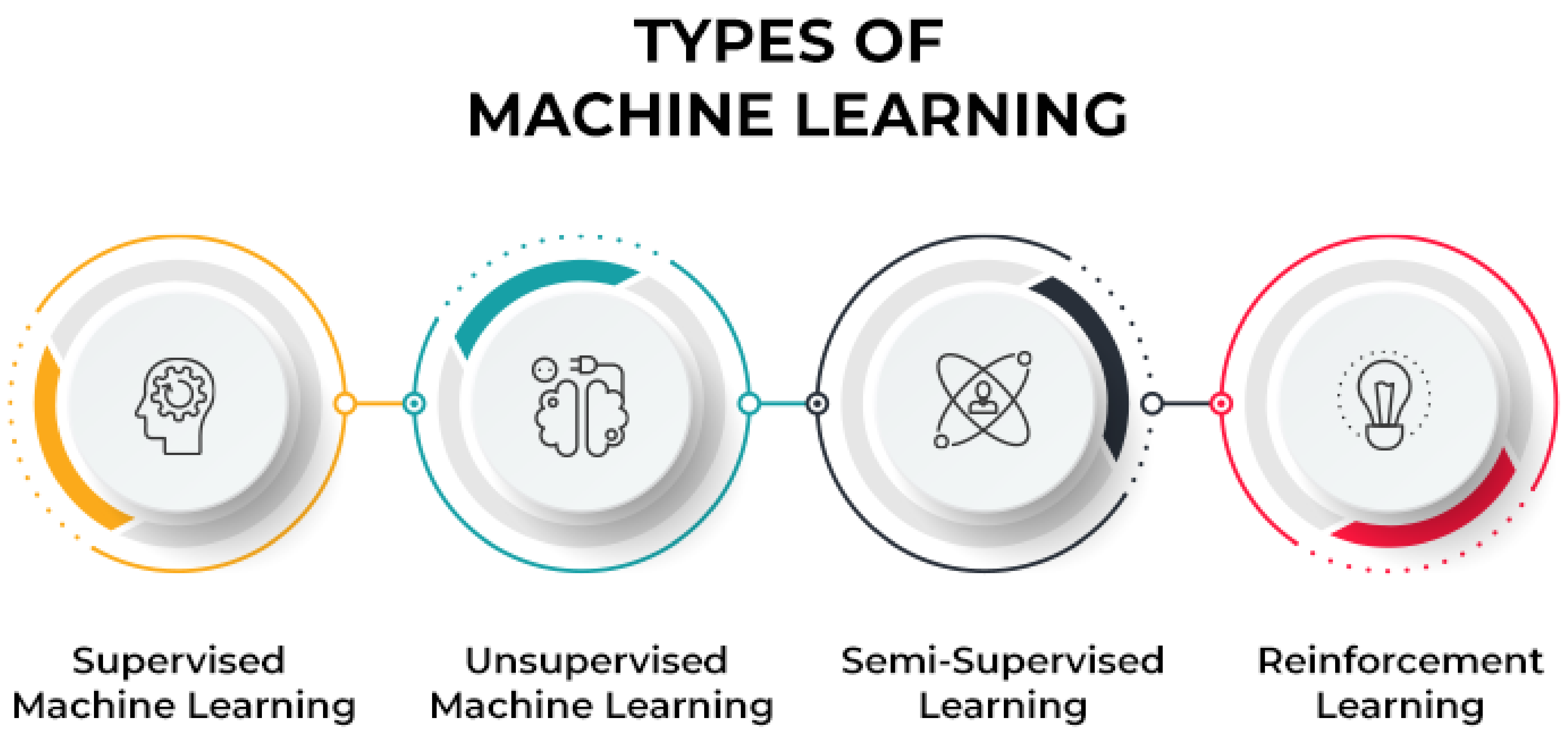
This is where the **model's power lies**. Based on the abstracted representation, the model goes beyond the specific data it was trained on and **learns to make general predictions for unseen data**.

This part is quite difficult to achieve. This is because the model is trained based on a finite set of data, which may possess a limited set of characteristics. But when we want to apply the model to take decision on a set of unknown data, usually termed as **test data**.

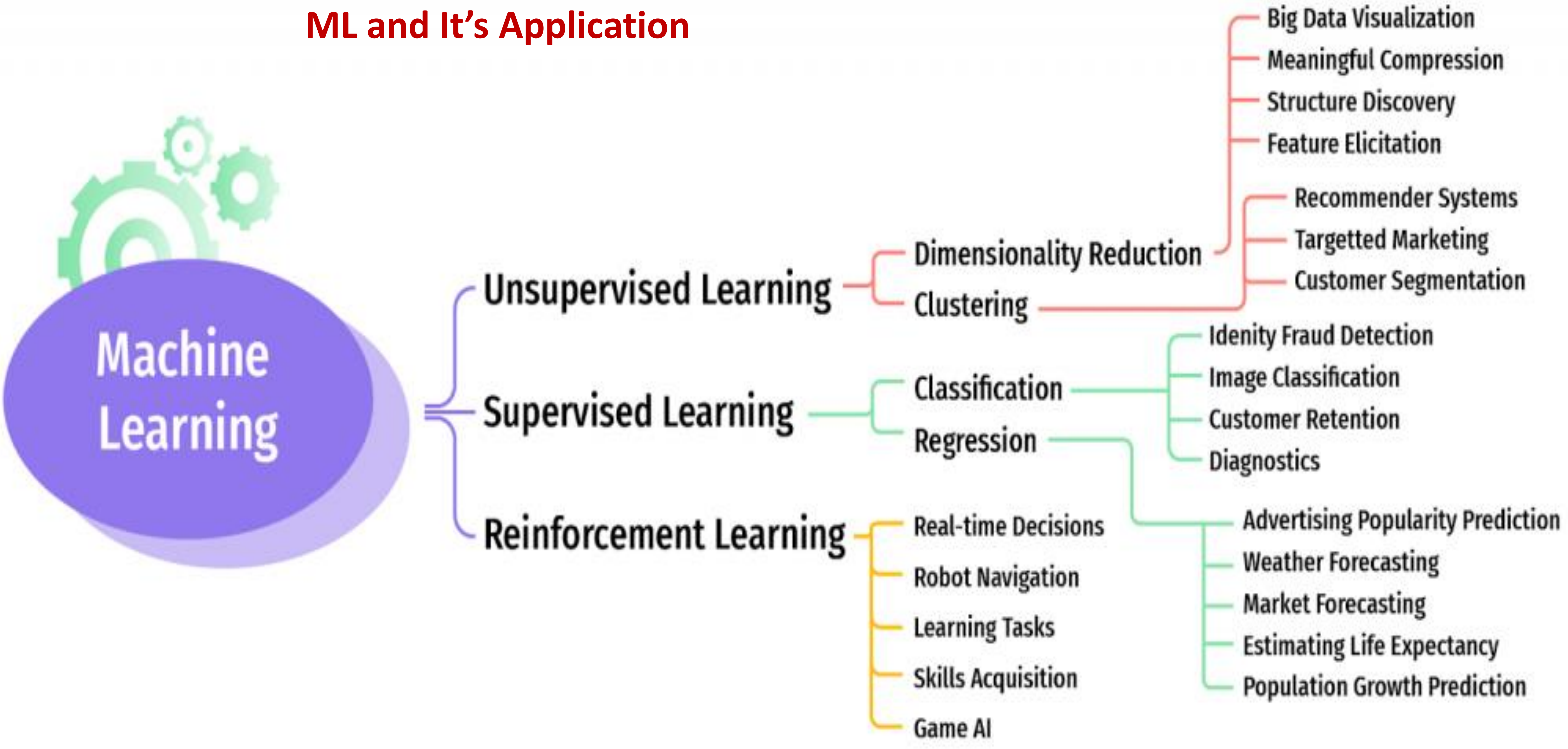
Types of Machine Learning




MACHINE LEARNING
UNSUPERVISED LEARNING
DIMENSIONALLY REDUCTION
Structure Discovery
Feature Elicitation
Meaningful compression
Big data Visualisation
CLUSTERING
Recommended Systems
Targetted Marketing
Customer Segmentation
SUPERVISED LEARNING
CLASSIFICATION
Image Classification
Customer Retention
Fraud Detection
Diagnostics
REGRESSION
Forecasting
Predictions
Process Optimization
New Insights
REINFORCEMNET LEARNING
Real-Time Decisions
Robot Navigation
Game AI
Skill Aquisition
Learning Tasks


TYPES OF
MACHINE LEARNING
Supervised
Machine Learning
Unsupervised
Machine Learning
Semi-Supervised
Learning
Reinforcement
Learning

ML and It's Application



Supervised learning:

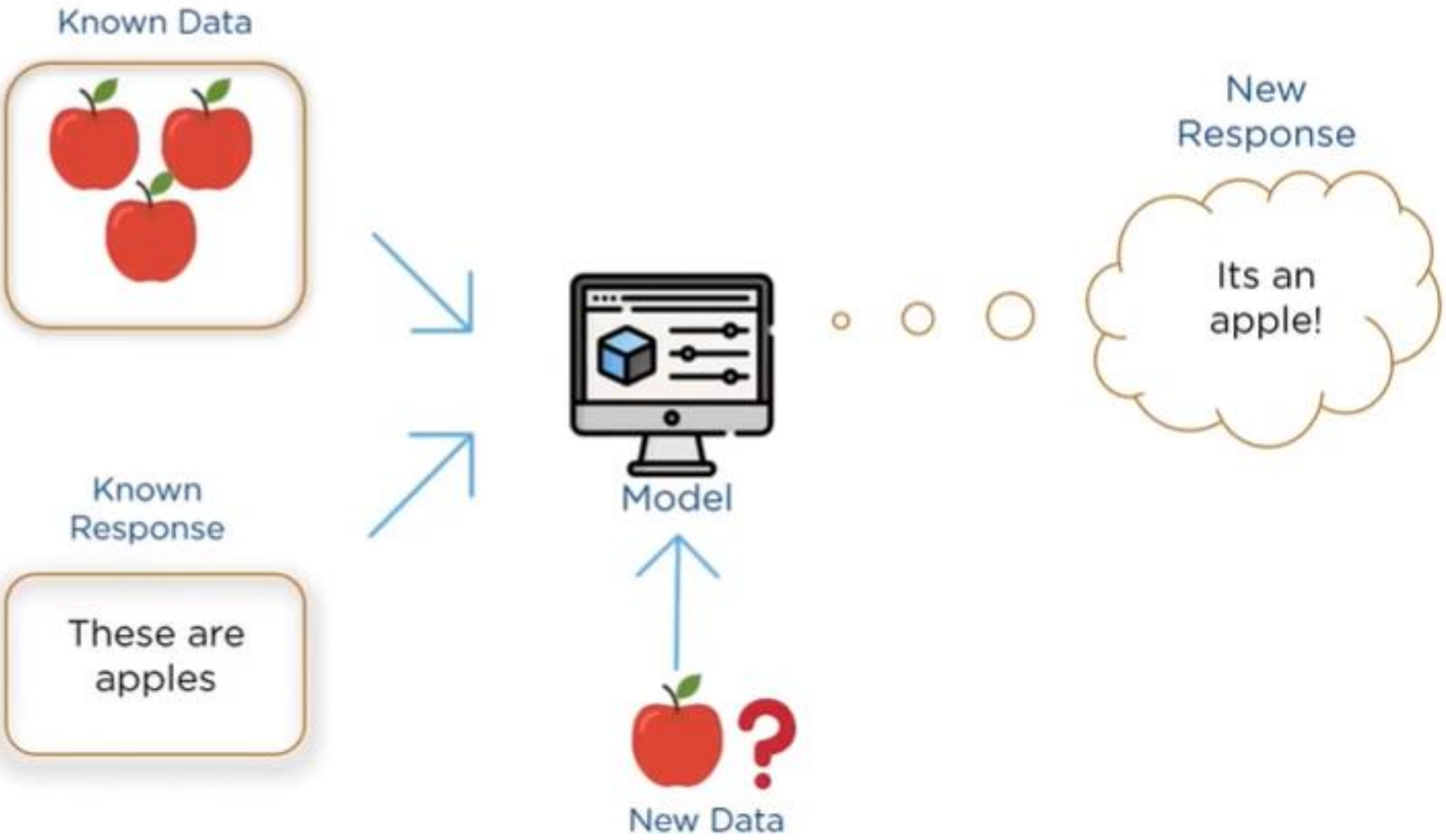
Supervised learning is the types of machine learning in which machines are trained using well "**labelled**" **training data**, and on basis of that data, machines predict the output. The labelled data means some input data is already **tagged** or **labelled** with the **correct output.**

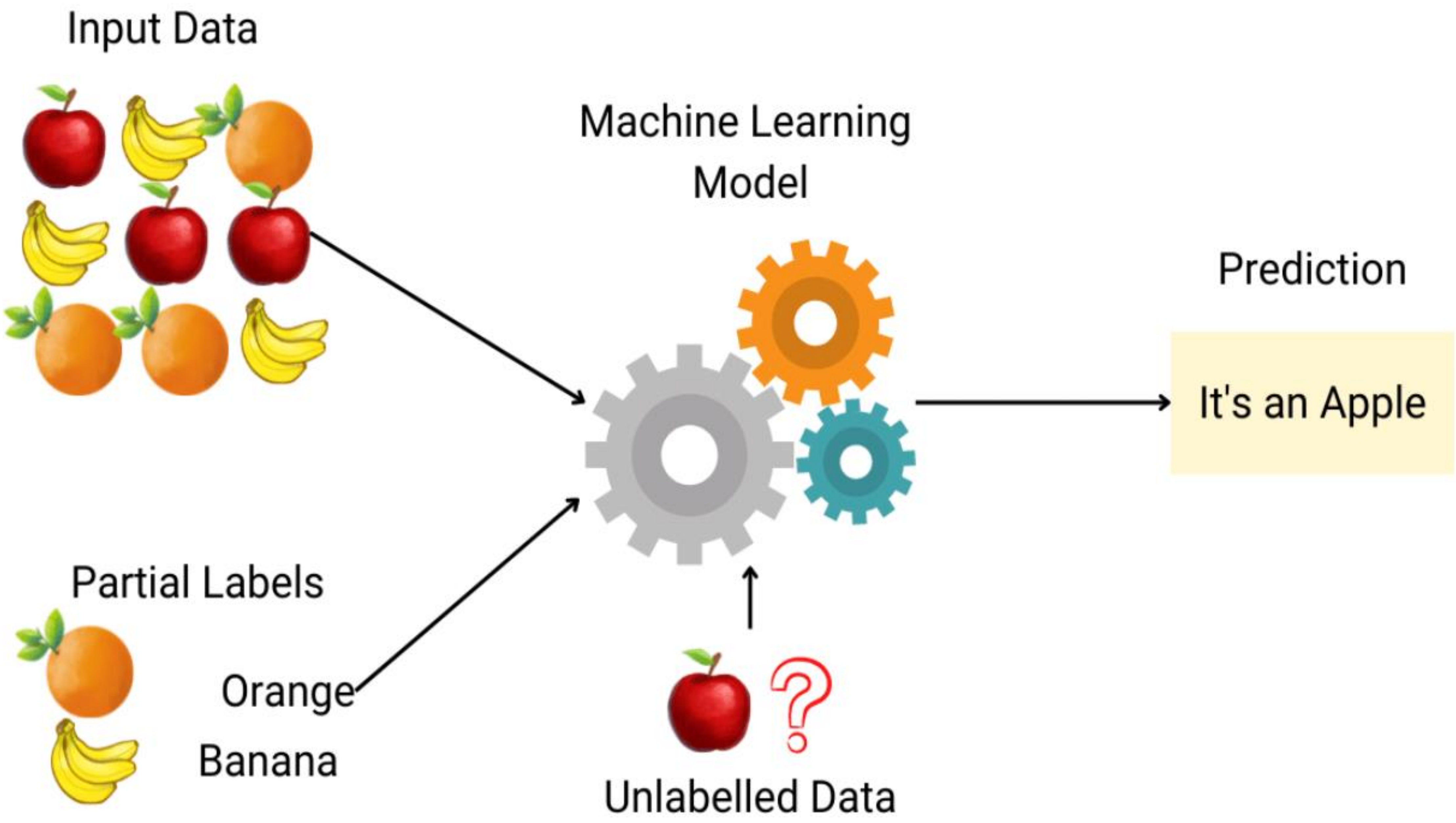
In supervised learning, **the training data provided to the machines work as the supervisor** that teaches the machines to predict the output correctly. It applies the same concept as a student learns in the **supervision of the teacher**. Supervised learning is a process of providing input data as well as correct output data to the machine learning model. The aim of a supervised learning algorithm is to find a mapping function to **map** the **input variable(x)** with the **output variable(y).**

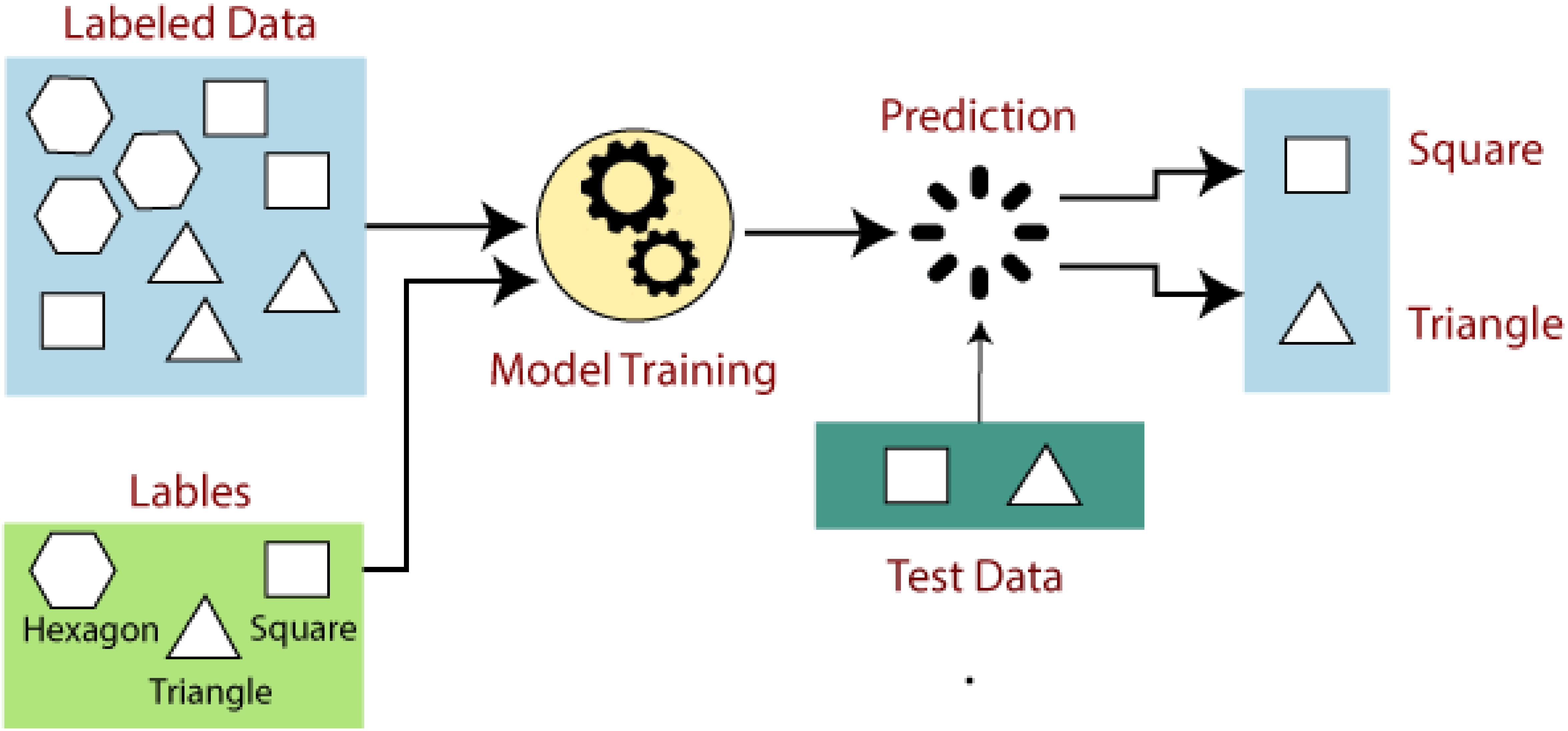
So, the goal of supervised learning is to learn a **mapping from inputs to outputs**, so that the model can make accurate predictions on **new**, **unseen data**.

Note:

In supervised learning, models are trained using labelled dataset, where the model learns about each type of data. Once the **training process** is completed, the model is **tested** on the basis of **test data** (a subset of the training set), and then it predicts the output.


Known Data
Known Response
These are apples
Model
New Data
New Response
Its an apple!


Input Data
Machine Learning
Model
Prediction
It's an Apple
Partial Labels
Orange
Banana
Unlabelled Data


Labeled Data
Lables
Hexagon
Square
Triangle
Model Training
Prediction
Test Data
Square
Triangle

Explanation:

Suppose we have a **dataset** of different types of **shapes** which includes **square, rectangle, triangle, and Polygon**. Now the first step is that we need to train the model for each shape.

1. If the given shape has **four sides**, and all the sides are equal, then it will be labelled as a Square.
2. If the given shape has **three sides**, then it will be labelled as a triangle.
3. If the given shape has **six equal sides** then it will be labelled as hexagon.

Now, after training, we test our model using the test set, and the task of the model is to identify the shape. The machine is already trained on all types of shapes, and when it finds a new shape, it classifies the shape on the bases of a number of sides, and predicts the output

Example-3: Predicting Housing Prices

Let's consider a common example of supervised learning: predicting housing prices based on various features.

1.Input Features:

1. Size of the house (in square feet)
2. Number of bedrooms
3. Number of bathrooms
4. Distance to the city center
5. Presence of nearby amenities (e.g., schools, parks)

2.Labels (Outputs):

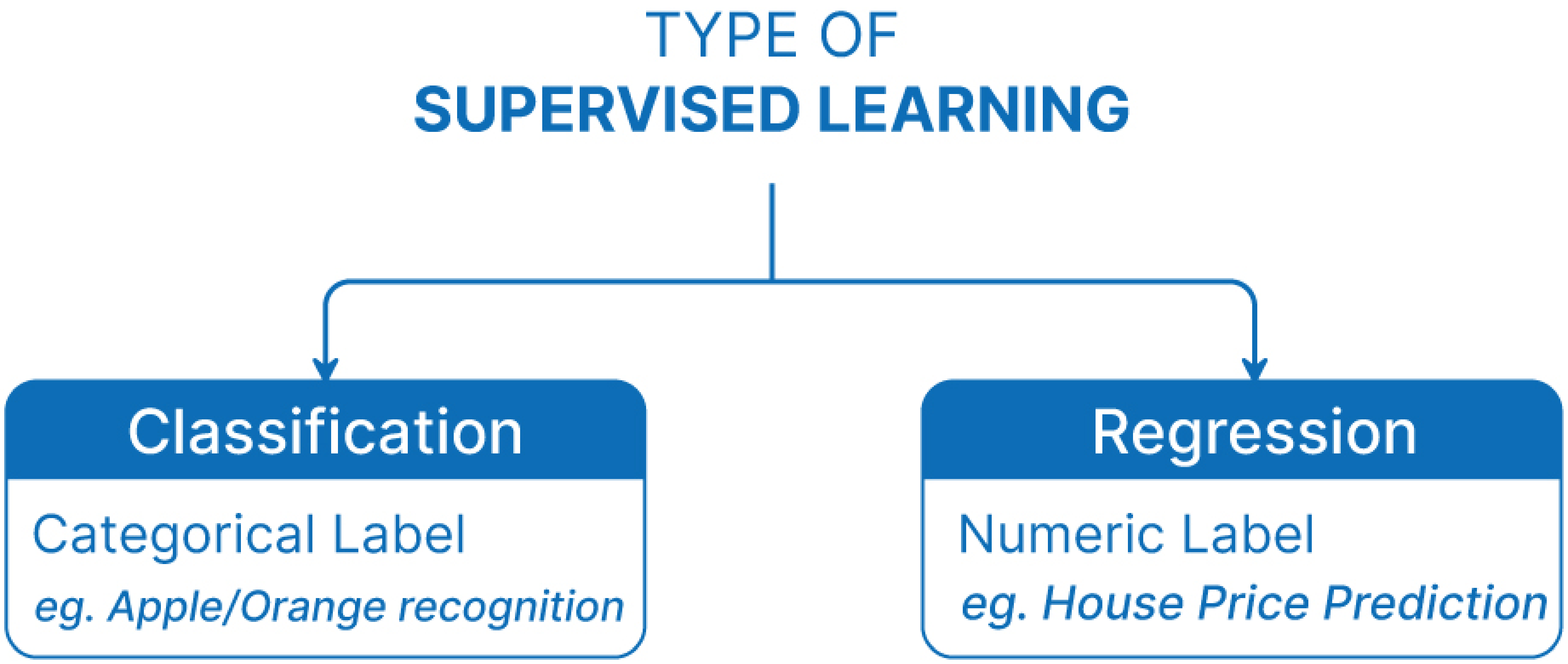
- Sale price of the house

3.Training Dataset:

- A dataset containing historical examples of houses with features and their corresponding sale prices.

4.Model:

- A supervised learning algorithm, such as linear regression, is chosen to learn the relationship between the input features and the sale prices.


TYPE OF
SUPERVISED LEARNING
Classification
Categorical Label
eg. Apple/Orange recognition
Regression
Numeric Label
eg. House Price Prediction

1. Classification:

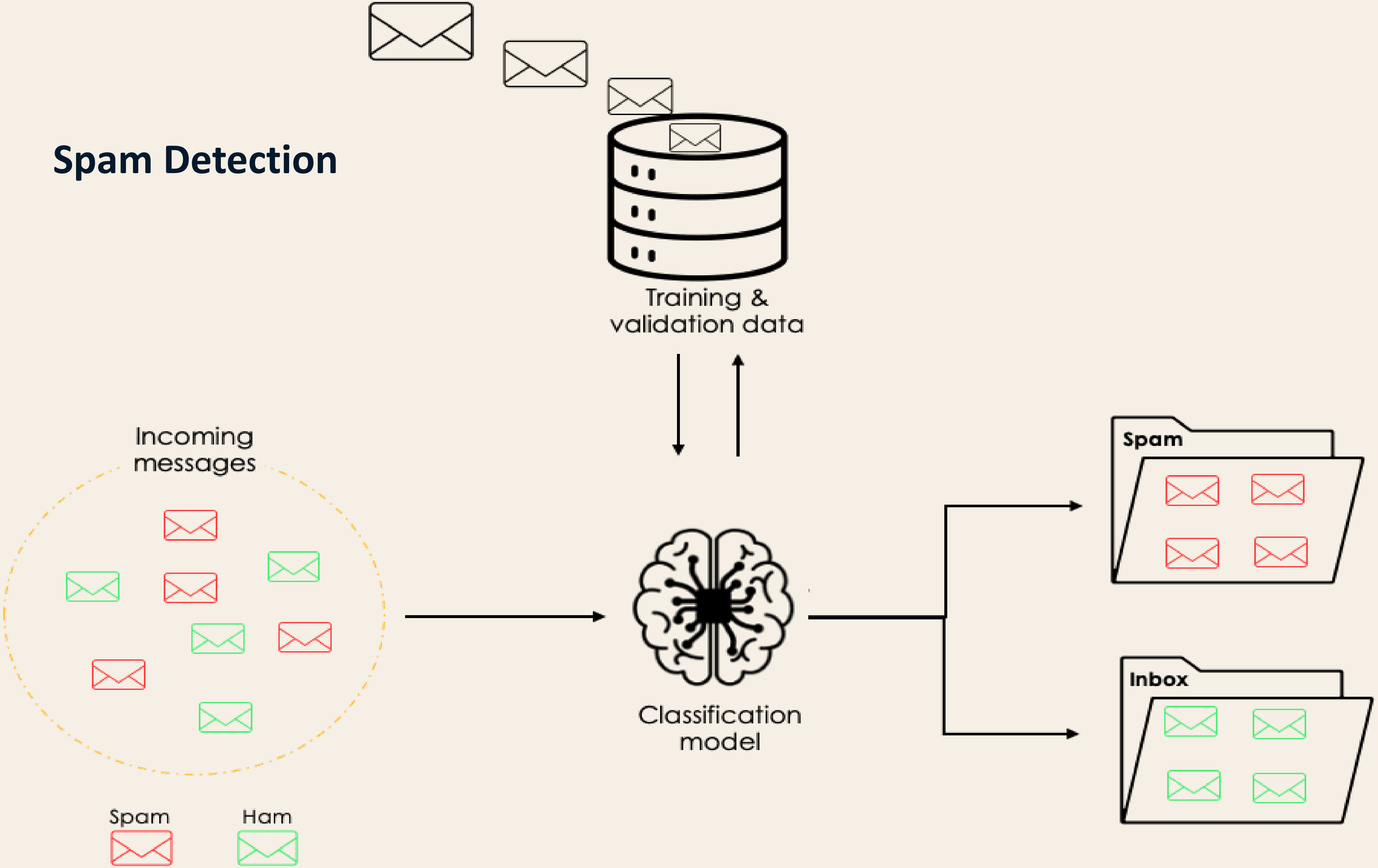
Classification is a type of supervised machine learning task where the goal is to **predict the category or class** that a new instance or observation belongs to, **on the basis of training data**. The output variable in classification is **discrete** and represents different classes or labels.

In Classification, a program learns from the given dataset or observations and then classifies new observation into a number of classes or groups. Such as, Yes or No, 0 or 1, Spam or Not Spam, cat or dog, etc. Classes can be called as targets/labels or categories.

Unlike regression, **the output variable of Classification is a category**, not a value, such as "Green or Blue", "fruit or animal", etc. Since the Classification algorithm is a Supervised learning technique, hence it takes labeled input data, which means it contains input with the corresponding output. The best example of an ML classification algorithm is **Email Spam Detector.**

The main goal of the Classification algorithm is to identify the category of a given dataset, and these algorithms are mainly used to predict the output for the categorical data.

Spam Detection



Zoumana KEITA

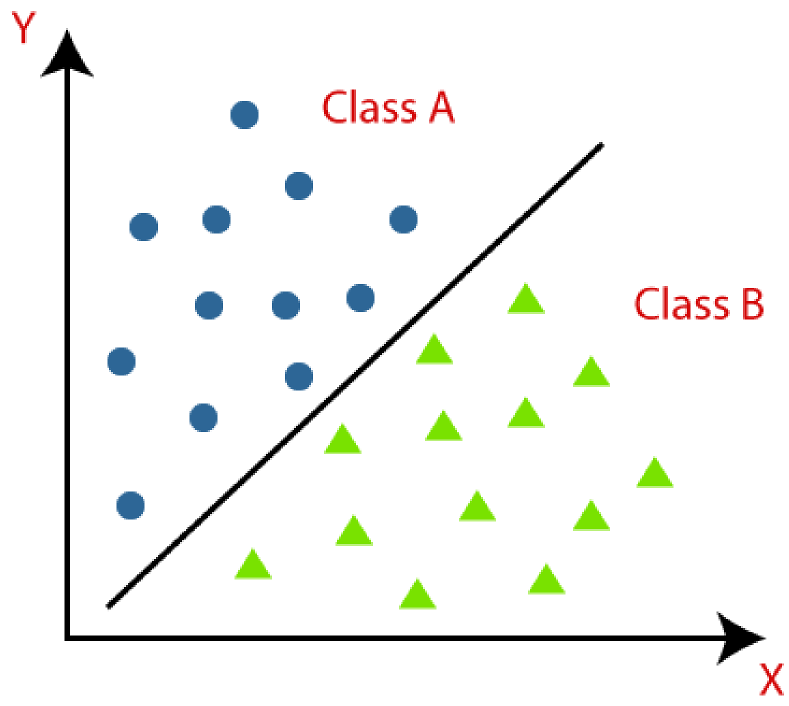
Example-2:

Classification algorithms can be better understood using the below diagram. In the below diagram, there are two classes, class A and Class B. These classes have features that are similar to each other and dissimilar to other classes.

The algorithm which implements the classification on a dataset is known as a **classifier**.

There are two types of Classifications:

1. Binary Classification
2. Multi-class Classifier



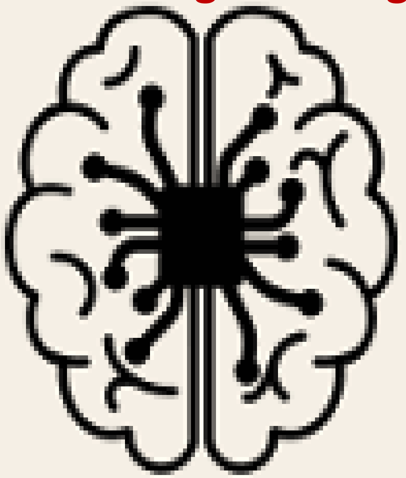
Truck

Binary Classification

In a binary classification task, the goal is to classify the input data **into two mutually exclusive categories**. The training data in such a situation is labeled in a binary format: **true** and **false**; **positive** and **negative**; O and 1; spam and not spam, male or female etc. depending on the problem being tackled.

For instance, we might want to detect whether a **given image is a truck or a boat**.

In Binary classification task, “**Mutually Exclusive**" means that the two categories or classes into which the input data is classified are distinct and cannot occur simultaneously for a single instance of input data. **An email cannot be both "spam" and "not spam" at the same time**.

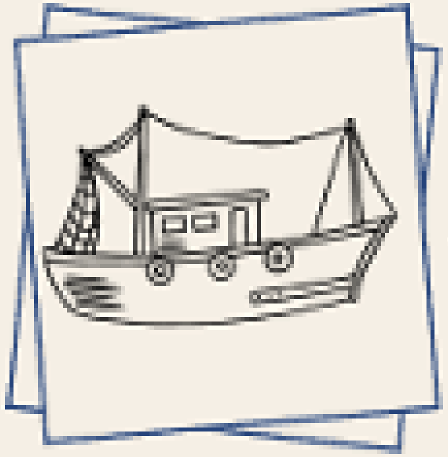


Machine Learning Model

Classification

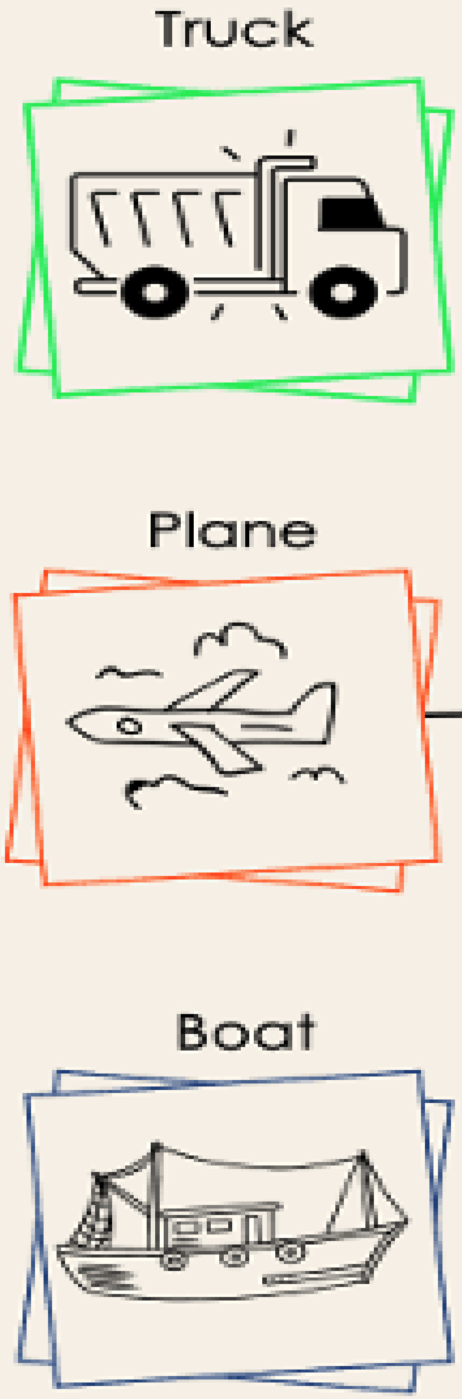
Boat

Boat

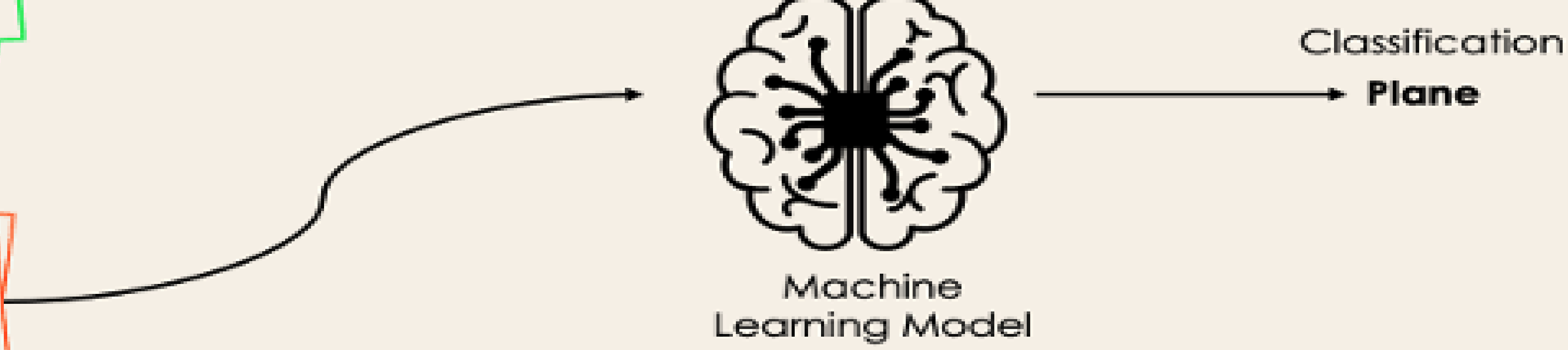


Note: If the classification problem has only two possible outcomes, then it is called as Binary Classifier.

Zoumana KEITA



If a classification problem has more than two outcomes, then it is called as Multi-class Classifier.
Example: Classifications of types of crops, Classification of types of music.



Multi-Class Classification:
The multi-class classification, on the other hand, has at least two mutually exclusive class labels, where the goal is to predict to which class a given input example belongs to. In the following case, the model correctly classified the image to be a plane.

Regression :

Regression is a type of supervised machine learning task where the goal is to predict a **continuous numerical value** or outcome based on input features.

Regression is a type of supervised machine learning where algorithms **learn** from the data to predict continuous values such as sales, salary, weight, or temperature.

Note:
In the context of regression in machine learning, a continuous numerical value refers to an outcome or target variable that can take on an infinite number of values within a specific range. In case of predicting a person's age, age is considered a continuous numerical value because it can theoretically take on any value within a certain range (for example, from 0 to 100+ years). **There are no gaps or intervals between possible ages**, and age can be expressed as a decimal or fraction if necessary (e.g., 25.5 years).

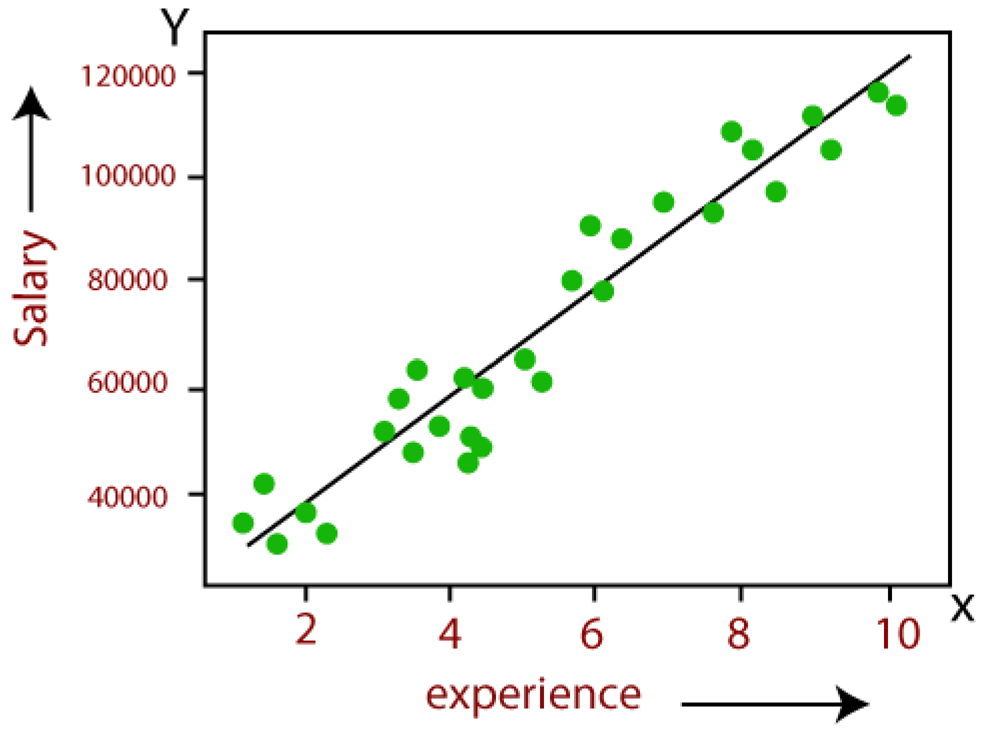
It is a variable that can have any real number value, and there are no distinct categories or classes. The term "continuous" implies that the variable can vary over a continuous range, and there are no gaps or interruptions in the possible values it can take. In contrast, in a classification task, the target variable would be a **discrete set of categories or classes.**

Note:
Here, “discrete set of categories or classes" refers to a set of distinct and separate values that a variable can take

Example:

Predicting the **salary** of an employee on the basis of the **year of experience**.

Linear Regression:



Example:

1. **Predicting age of a person:** Given certain features or attributes of a person, such as **height, weight, gender**, and other relevant factors, the task is to predict the person's age in years.

2. **Predicting the price of houses based on their features:**

In real estate markets, house prices can vary continuously based on factors such as location, size, amenities, market conditions, and other features. The House prices can range from a few thousand dollars for smaller properties in certain areas to millions of dollars for luxury properties in prime locations.

Regression analysis :

Regression analysis is a statistical method to **model the relationship between a dependent (target) and independent (predictor) variables with one or more independent variables**. More specifically, Regression analysis helps us to understand how the value of the dependent variable is changing corresponding to an independent variable when other independent variables are held fixed. It predicts continuous/real values such as temperature, age, salary, price etc. Regression is a supervised learning which helps in finding the correlation between variables and enables us to predict the continuous output variable based on the one or more predictor variables.

It is mainly used for prediction, forecasting, time series modeling, and determining the causal-effect relationship between variables.

Unsupervised Learning:

Unsupervised Machine learning is a type of machine learning where the algorithm is trained on **unlabeled data**, and the **objective is to find patterns, relationships, or structures within the data** without explicit guidance or labeled outcomes. i.e Unsupervised learning is a method we use to **group data** when no **labels are present**.

As the **name suggests**, unsupervised learning is a machine learning technique in which models are **not supervised using training dataset**. Instead, **models itself find** the hidden patterns and insights from the given data. It can be compared to **learning which takes place in the human brain while learning new things**.

It can be defined as: **"**Unsupervised learning is a type of machine learning in which models are trained using unlabeled dataset and are allowed to act on that data without any supervision.**"**

Unsupervised learning cannot be directly applied to a regression or classification problem because unlike supervised learning, we have the input data but no corresponding output data.

Example:

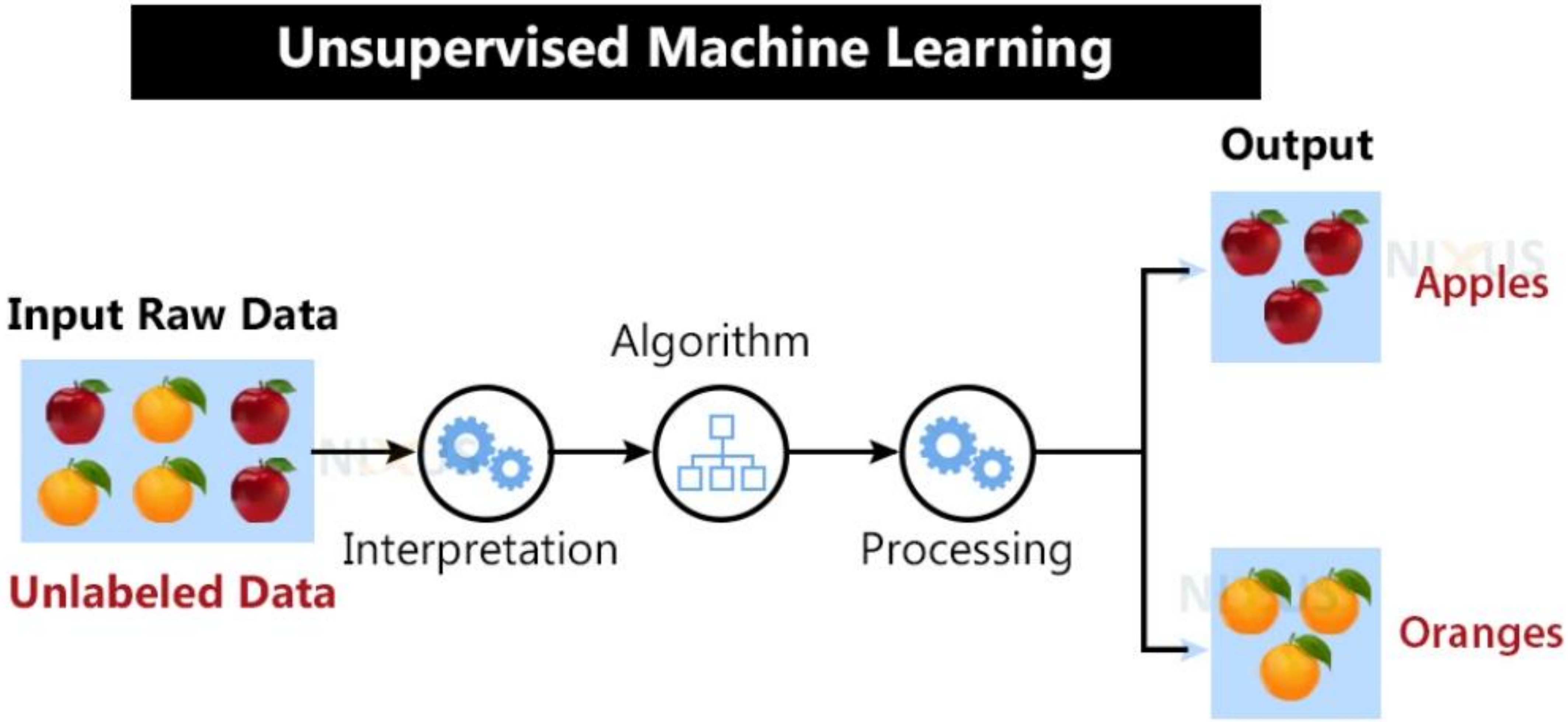
Suppose the unsupervised learning algorithm is given an input dataset containing **images of different types of cats and dogs**. The algorithm is never trained upon the given dataset, which means it does not have any idea about the features of the dataset. The task of the unsupervised learning algorithm is to identify the image features on their own. Unsupervised learning algorithm will perform this task **by clustering the image dataset into the groups according to similarities between images**.

Why use Unsupervised Learning?

1. Unsupervised learning is helpful for finding useful insights from the data.
2. Unsupervised learning is much similar as a **human learns to think by their own experiences**, which makes it closer to the real AI.
3. Unsupervised learning works on unlabeled and uncategorized data which make unsupervised learning more important.
4. In real-world, we do not always have input data with the corresponding output so to solve such cases, we need unsupervised learning.
5. **Semi-supervised Learning:** Unsupervised ML techniques can also be combined with supervised ML in semi-supervised learning approaches. In this paradigm, a small amount of labeled data is augmented with a larger amount of unlabeled data to improve the performance of supervised models.

Note:

Unsupervised machine learning (ML) works differently from supervised ML because **it doesn't rely on labeled data to learn patterns or make predictions**. Instead, unsupervised ML algorithms focus on finding hidden structure or patterns within the data without explicit guidance from predefined labels.


Unsupervised Machine Learning
Output
Apples
Input Raw Data
Algorithm
Interpretation
Processing
Unlabeled Data
Oranges

Here, we have taken an unlabeled input data, which means it is not categorized and corresponding outputs are also not given. Now, this unlabeled input data is fed to the machine learning model in order to train it. Firstly, it will interpret the raw data to find the hidden patterns from the data and then will apply suitable algorithms such as k-means clustering, Decision tree, etc. Once it applies the suitable algorithm, the algorithm divides the data objects into groups according to the similarities and dissimilarities between the objects.

Working of Unsupervised learning models:

1. We feed the model data with no categories or outputs for training
2. Model interprets raw data to identify hidden patterns
3. Depending on data, we use suitable algorithms
4. Algorithm groups data

Disadvantages of Unsupervised Learning:

1. Unsupervised learning is fundamentally **more difficult** than supervised learning since it lacks comparable results.
2. The outcome of an unsupervised learning approach may be **less accurate** since the input data does not have labels and algorithms do not know the precise output in advance.
3. We **need humans to validate the results** of unsupervised learning models which convolutes the process.
4. The operations involved **require a high level of computational power and also take up a lot of time**.

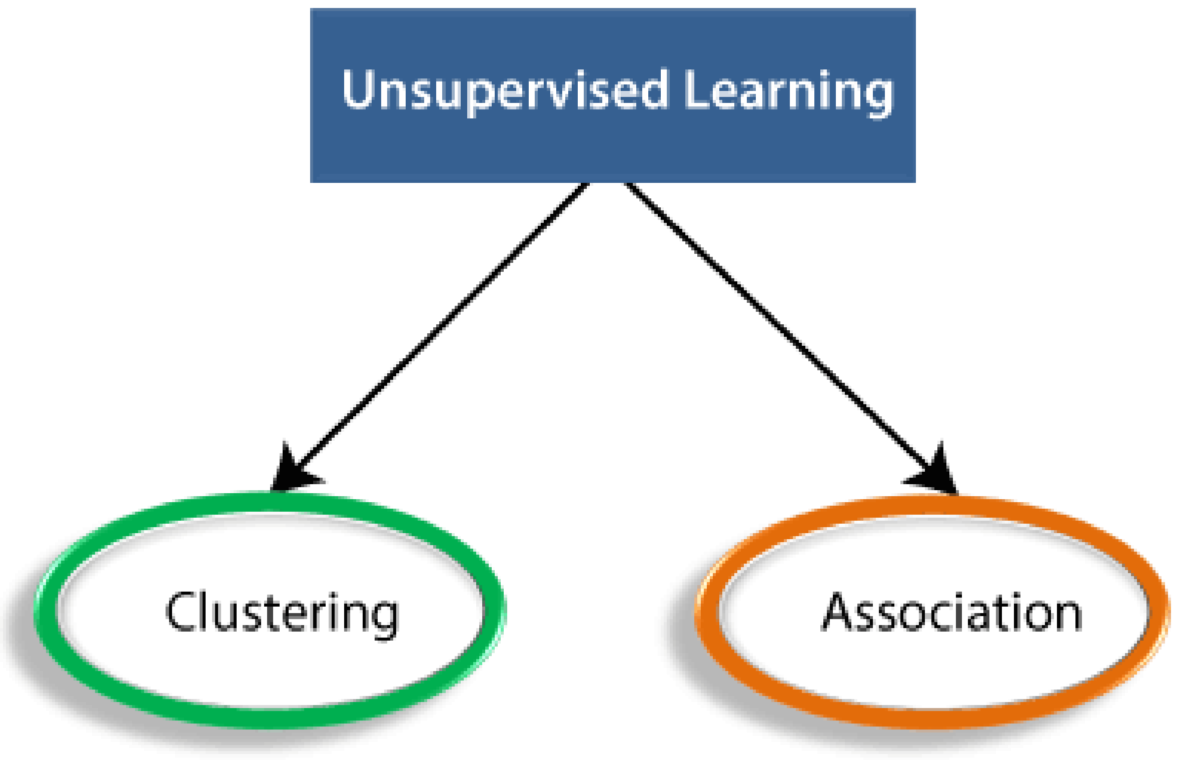
Alternative to Unsupervised Learning:

Semi-supervised learning may be a viable alternative to unsupervised learning. It is a combination or even a min-and-match of both unsupervised and supervised learning methods. The primary benefit of this sort of training would be that it lowers errors.

For example, it will only cluster the unlabeled data that fits the clustering criteria, and it will categorize the output automatically once it has labels. This uses less computer power and takes less time.

Types of Unsupervised Learning:

1. Clustering
2. Association rules
3. Dimensionality reduction.

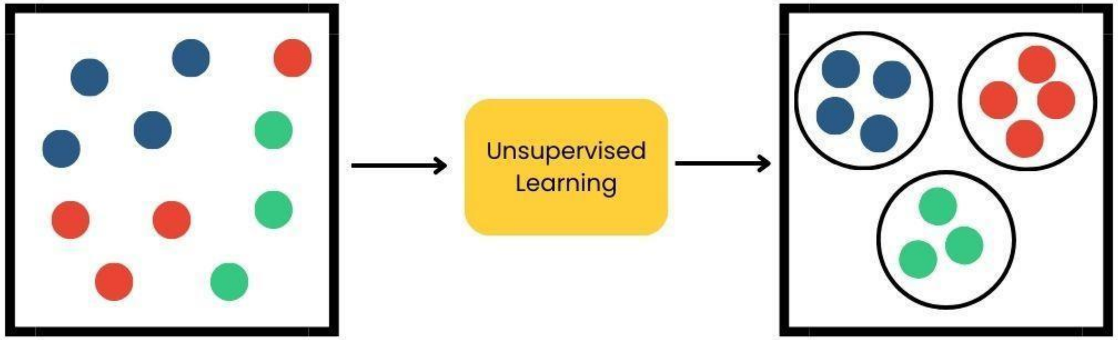


Clustering:

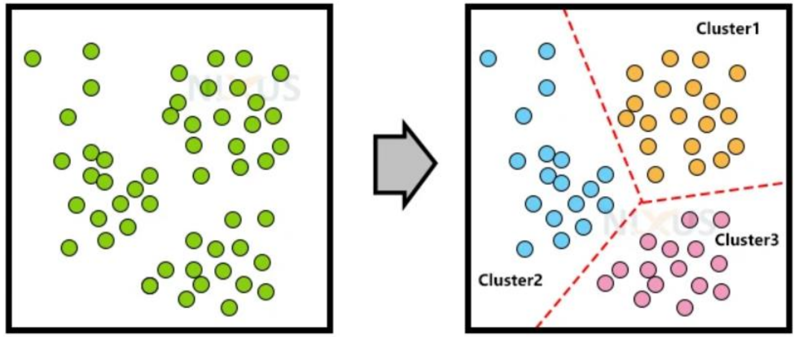
Clustering methods involve **grouping untagged** data based on their similarities and differences. When two instances appear in different groups, we can infer they have dissimilar properties.

Clustering is a type of unsupervised learning, meaning that we do not need labeled data for clustering algorithms; this is one of the biggest advantages of clustering over other supervised learning like Classification.

Clustering is the process of **arranging** a group of objects in such a manner that the objects in the same group (which is referred to as a cluster) are more **similar** to **each other** than to the objects in any **other group**.



OR



Application of Clustering:

1. Customer Segmentation:

- **Objective:** Grouping customers based on similarities in purchasing behavior, demographics, or preferences.
- **Application:** Targeted marketing, personalized recommendations, and tailored customer experiences.

2. Document Clustering:

- **Objective:** Grouping similar documents based on content, topic, or language.
- **Application:** Document categorization, information retrieval, and organizing large document repositories.

3. Anomaly Detection:

- **Objective:** Identifying unusual patterns or outliers in a dataset.
- **Application:** Fraud detection in financial transactions, network intrusion detection, and fault detection in industrial systems.

4. Retail Market Basket Analysis:

- **Objective:** Identifying associations and patterns in the purchasing behavior of customers.
- **Application:** Recommender systems, optimizing product placements, and understanding customer buying habits.

5. E-commerce Product Bundling:

- **Objective:** Grouping products that are often purchased together.
- **Application:** Creating product bundles, improving cross-selling strategies, and enhancing the user shopping experience.

Association Rule:

An association rule is an unsupervised learning method which is used **for finding the relationships between variables in the large database**. It determines the set of items that occurs together in the dataset. Association rule makes marketing strategy more effective. Such as people who buy X item (suppose a bread) are also tend to purchase Y (Butter/Jam) item. A typical example of Association rule is **Market Basket Analysis(MBA)**.

We typically see association rule mining used for market basket analysis: this is a data mining technique retailers use to gain a better understanding of customer purchasing patterns based on the relationships between various products.

So Association is the process of discovering interesting relationships, associations, or patterns within a dataset. This type of analysis is often applied to **transactional data**, where the goal is to identify associations between items or events that frequently co-occur. Association rules are used to express these relationships, and they help reveal hidden connections in the data.

For example, if a customer buys bread, he most likely can also buy butter, eggs, or milk, so these products are stored within a shelf or mostly nearby.

3. Dimensionality Reduction:

Dimensionality reduction is a technique used in unsupervised machine learning **to reduce the number of input features or variables in a dataset while preserving as much of its essential information**. In other words, it is a process of **transforming high-dimensional data into a lower-dimensional space** that still preserves the essence of the original data. The primary goal is to overcome the **curse of dimensionality**, improve computational efficiency, and sometimes enhance the interpretability of the data. Two common methods for dimensionality reduction are **Principal Component Analysis (PCA) and t-Distributed Stochastic Neighbor Embedding (t-SNE).**

In machine learning, high-dimensional data refers to data with a large number of features or variables. The curse of dimensionality is a common problem in machine learning, where the performance of the model deteriorates or drops as the number of features increases. This is because the complexity of the model increases with the number of features, and it becomes more difficult to find a good solution. In addition, high-dimensional data can also lead to overfitting, where the model fits the training data too closely and does not generalize well to new data.

It's helpful to reduce the dimensionality of a dataset during EDA (Exploratory Data Analysis) to help visualize data: this is **because visualizing data in more than three dimensions is difficult**. From a data processing perspective, reducing the dimensionality of the data simplifies the modeling problem.

When more input features are being fed into the model, the model must learn a more complex approximation function. This phenomenon can be summed up by a saying called the "curse of dimensionality."

Dimensionality reduction can help to mitigate these problems by reducing the complexity of the model and improving its generalization performance. There are two main approaches to dimensionality reduction: feature selection and feature extraction.

Feature Engineering:

1. Feature Selection:

Feature selection involves **selecting a subset of the original features** that are most relevant to the problem at hand. The goal is to reduce the dimensionality of the dataset while retaining the most important features. There are several methods for feature selection, including filter methods, wrapper methods, and embedded methods. Filter methods rank the features based on their relevance to the target variable, wrapper methods use the model performance as the criteria for selecting features, and embedded methods combine feature selection with the model training process.

2. Feature Extraction:

Feature extraction involves **creating new features by combining or transforming the original features**. The goal is to create a set of features that captures the essence of the original data in a lower-dimensional space. There are several methods for feature extraction, including principal component analysis (PCA), linear discriminant analysis (LDA), and t-distributed stochastic neighbor embedding (t-SNE). PCA is a popular technique that projects the original features onto a lower-dimensional space while preserving as much of the variance as possible.

Reinforcement Learning:

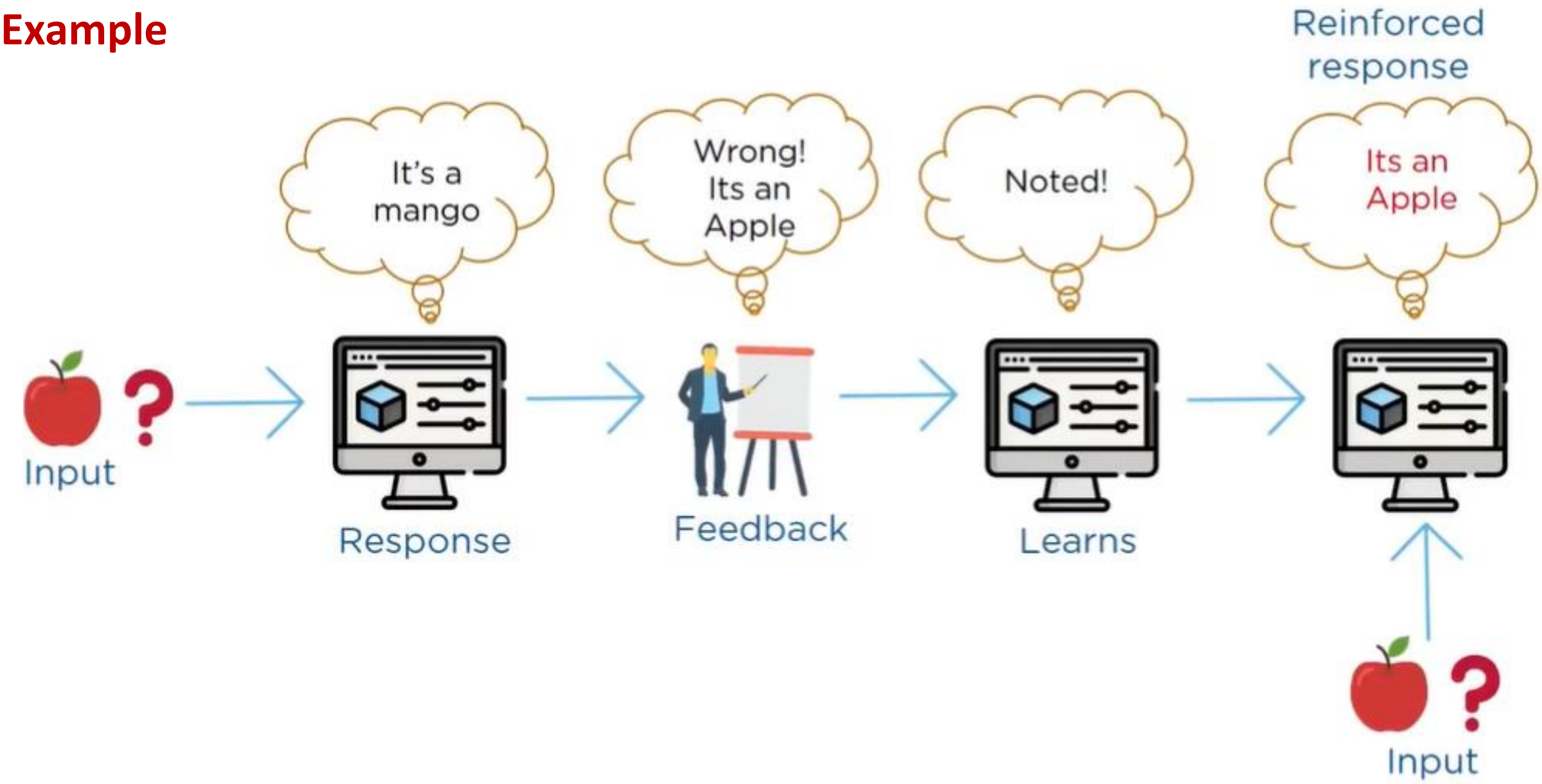
Reinforcement Learning (RL) is a type of machine learning paradigm in which an **agent learns to make decisions by interacting with an environment**. The agent takes actions in the environment, and in **return, it receives feedback in the form of rewards or punishments**.

Reinforcement Learning is a **feedback-based Machine learning technique** in which an agent learns to behave in an environment by performing the actions and seeing the results of actions. For each good action, the agent gets **positive feedback**, and for each bad action, the agent gets **negative feedback or penalty**. In Reinforcement Learning, the agent learns automatically using feedbacks without any labeled data, unlike supervised learning Since there is no labeled data, so the agent is bound to learn by its experience only. RL solves a specific type of problem where decision making is sequential, and the goal is long-term, such as game-playing, robotics, etc. The agent interacts with the environment and explores it by itself. The primary goal of an agent in reinforcement learning is to **improve the performance by getting the maximum positive rewards.**

The agent learns with the process of **hit and trial**, and based on the experience, it learns to perform the task in a better way. Hence, we can say that "Reinforcement learning is a type of machine learning method where an intelligent agent (computer program) interacts with the environment and learns to act within that." How a **Robotic dog learns** the movement of his arms is an example of Reinforcement learning.

It is a **core part of Artificial Intelligence**, and all AI agent works on the concept of reinforcement learning. Here we do not need to pre-program the agent, as it learns from its own experience without any human intervention

Example


Reinforced response
It's a mango
Wrong! Its an Apple
Noted!
Its an Apple
Input
Response
Feedback
Learns
Input

Class-2

MACHINE LEARNING ACTIVITIES:

The first step in machine learning activity starts with **data**. In case of supervised learning, it is the labelled training data set followed by **test data which is not labelled**. In case of unsupervised learning, there is no question of labelled data but the task is to find patterns in the input data.

A thorough review and exploration of the data is needed to understand the type of the data, the quality of the data and relationship between the different data elements. Based on that, multiple pre-processing activities may need to be done on the input data before we can go ahead with core machine learning activities. Following are the typical preparation activities done once the input data comes into the machine learning system:

- Understand the type of data in the given input data set (For example Numerical Data).
- Explore the data to understand the nature and quality.
- Explore the relationships amongst the data elements, e.g. inter-feature relationship.
- Find potential issues in data. (you might find missing values, outliers, duplicate entries, or data entry errors)
- Do the necessary remediation, e.g. impute missing data values, etc., if needed. Once issues are identified, you take steps to address them.
- Apply the following pre-processing steps, as necessary.
    1. Dimensionality Reduction
    2. Feature sub-set selection

Once the data is prepared for modelling, then the learning tasks start off.

Once the data is prepared for modelling, then the learning tasks start off. As a part of it, do the following activities:

1. “ The input data is **first divided into parts** — the **training data** and the **test data** (called holdout). This step is applicable for supervised learning only.
2. Consider different **models** or **learning algorithms** for selection. “Train the model based on the training data for supervised learning problem and apply to unknown data. Directly apply the chosen unsupervised model on the input data for unsupervised learning problem.
3. After the model is selected, trained (for supervised learning), and applied on input data, the performance of the model is evaluated. Based on options available, specific actions can be taken to improve the performance of the model, if possible.

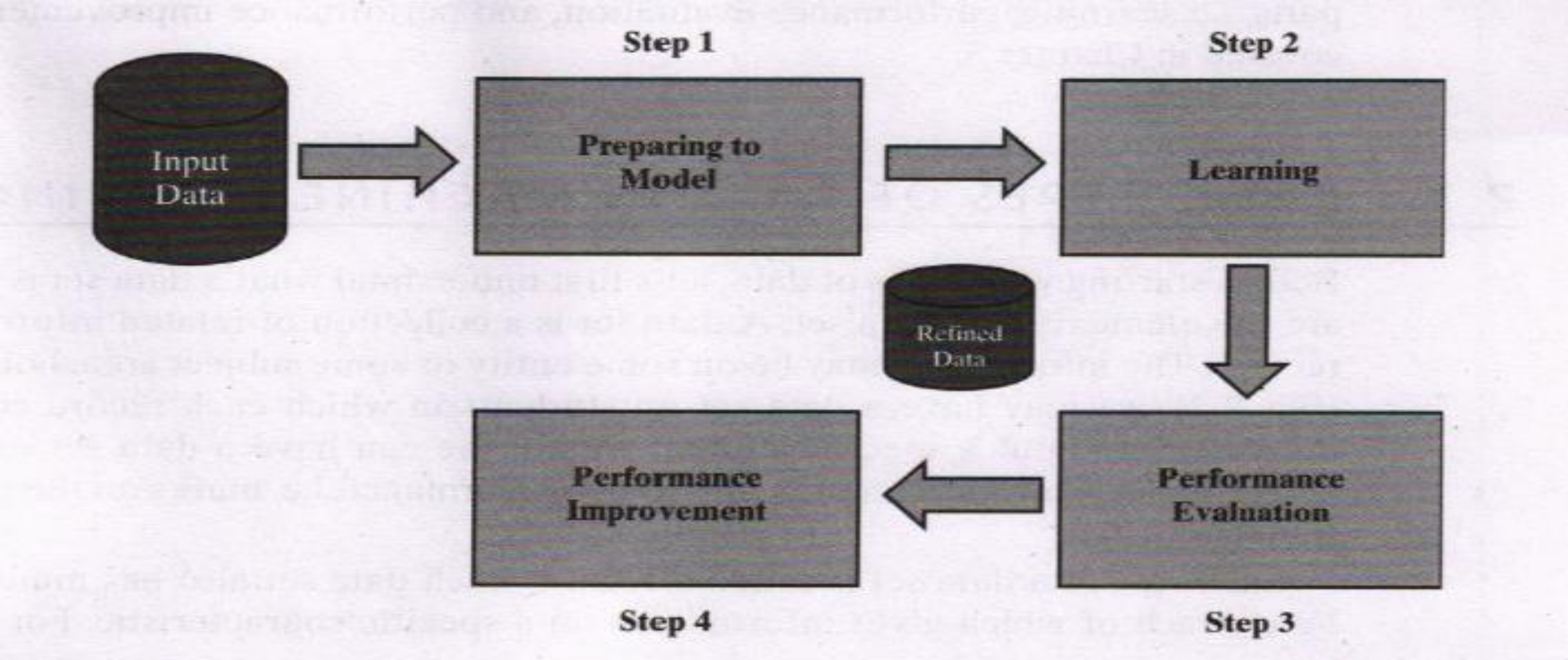
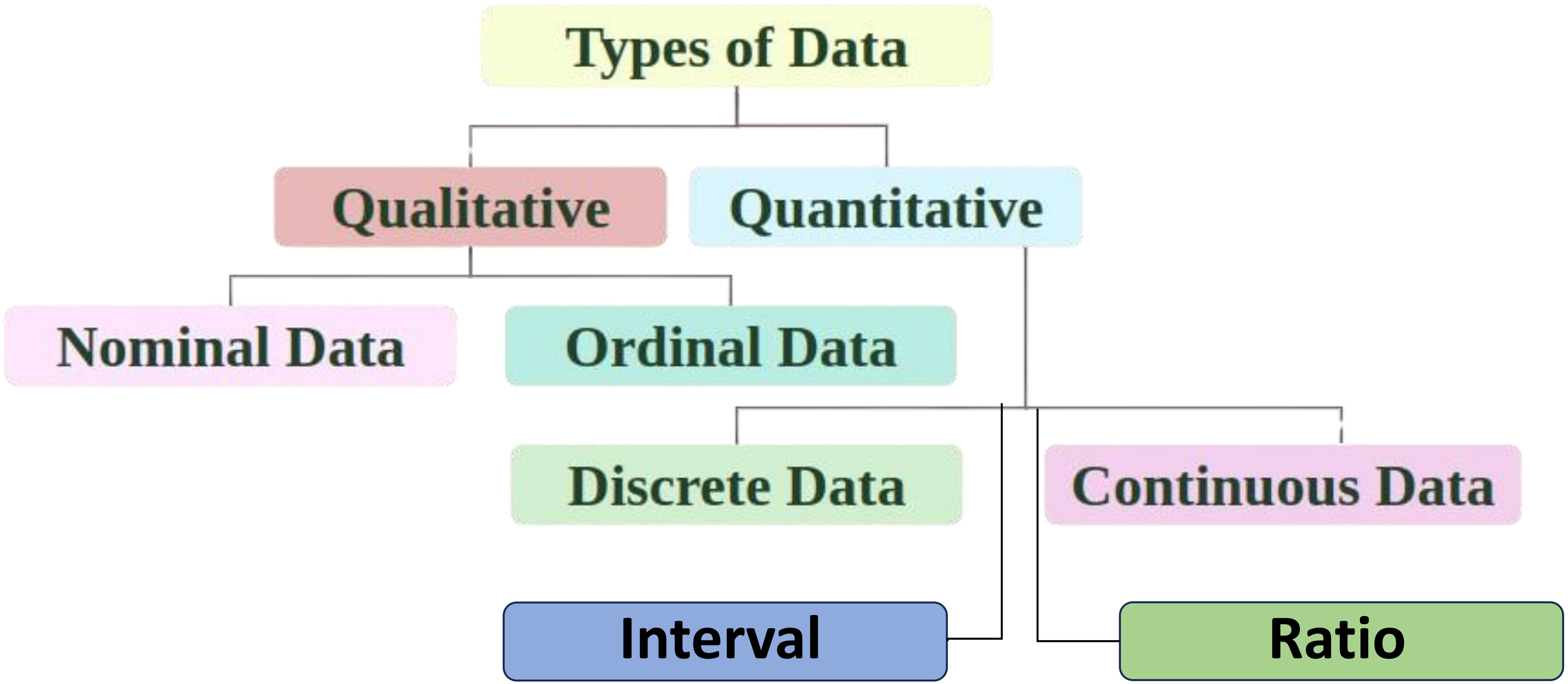


FIG. 2.1
Detailed process of machine learning



Note:
Data can be classified into different types based on its **nature and characteristics**. Two primary classifications of data are qualitative (categorical) and quantitative (numerical).

1. Qualitative Data: (Non- Measurable One)

Definition: Qualitative data, also known as **categorical data**, represents **characteristics** or **attributes** that are not measured on a numerical scale. Instead, they are categorical in nature and represent different categories or groups.

For example, if we consider the quality of performance of students in terms of ‘Good’ ‘Average’, and ‘Poor’ it falls under the category of qualitative data.

Examples:

- Gender (Male, Female)
- Color (Red, Blue, Green)
- Marital Status (Married, Single, Divorced)
- Type of Vehicle (Car, Truck, Motorcycle)

Characteristics:

- Non-numeric in nature.
- Represents categories or groups.
- Can be nominal (unordered) or ordinal (ordered)

Qualitative data can be further subdivided into two types as follows: (Also Called a Categorical Level of Measurement):

1. **Nominal data (Unordered data)**
2. **Ordinal data (Ordered)**

1. Nominal data :

Nominal data represents **categories or groups with no inherent order or ranking**. Nominal data does not follow any hierarchy. The categories are **distinct** and, **mutually exclusive** (**separate**) but there is no meaningful numerical value associated with them.

Examples of nominal data are :

1. Blood group: A, B, O, AB, etc.
2. Nationality: Indian, American, British, etc.
3. Gender: Male, Female, Other.

EXAMPLE-1: Let us understand nominal data with an example. For instance, the **color of a car can be black, red, or orange.** Here we need to realize that no color is greater than the other. It just represents a particular color of a car.

EXAMPLE-2: Mutually exclusive (separate) means - consider a categorical variable representing different types of fruits: Apple, Banana, and Orange. If a fruit is categorized as an "Apple," it cannot simultaneously be categorized as a "Banana" or "Orange." Each fruit type is distinct, and there is no overlap or sharing of categories among observations.

Note:

A special case of nominal data is when only two labels are possible, e.g. pass/fail as a result of an examination. This sub-type of nominal data is called 'dichotomous' It is obvious, mathematical operations such as addition, subtraction, multiplication, etc. cannot be performed on nominal data. For that reason, statistical functions such as mean, variance, etc. can also not be applied on nominal data. However, a **basic count is possible**. So, mode, i.e. most frequently occurring value, can be identified for nominal data.

That is, Statistical functions that rely on **numerical calculations**, such as mean and variance, cannot be applied directly to nominal data. However, basic operations like **counting the frequency** of each category and identifying the mode (most frequently occurring value) are still possible and meaningful.

2. Ordinal:

Ordinal data represents categories or groups with a **specific order or ranking**. While **the categories themselves are distinct** and **mutually exclusive**, they also have a meaningful sequence or hierarchy.

In addition to possessing the properties of nominal data, can also be **naturally ordered**. This means ordinal data also assigns named values to attributes but unlike nominal data, they can be **arranged in a sequence of increasing or decreasing value** so that we can say whether a value is better than or greater than another value.

Examples of ordinal data are:

1. Customer satisfaction: ‘Very Happy’ ‘Happy’ ‘Unhappy; etc.
2. Grades: A, B, C, etc.
3. Hardness of Metal: ‘Very Hard’, ‘Hard’, ‘Soft’, etc.
4. Likert scale responses :Strongly Disagree, Disagree, Neutral, Agree, Strongly Agree
5. Educational levels :High School, Bachelor's, Master's, Ph.D.
6. Performance ratings :Poor, Fair, Good, Excellent

Like nominal data, basic **counting** is possible for ordinal data. Hence, the **mode** can be identified. Since ordering is possible in case of ordinal data, median, and quartiles can be identified in addition.

Note:

Categories are distinct and mutually exclusive: means that each category represents a unique and separate level of the variable, and no observation can belong to more than one category simultaneously.

2. Quantitative Data: (Measurable One)

Definition: Quantitative data, also known as numerical data, consists of numerical measurements or quantities that can be expressed as numbers and subjected to mathematical operations. It involves values that can be expressed as numbers and subjected to mathematical operations.

For example, if we consider the attribute 'marks' it can be measured using a scale of measurement. Quantitative data is also termed as numeric data.

Examples:

- Height (in centimeters or inches)
- Age (in years)
- Income (in dollars)
- Temperature (in Celsius or Fahrenheit)
- Number of products sold.
- Test scores.

Characteristics:

- Numeric in nature.
- Can be **discrete or continuous.**
- Allows for mathematical operations such as addition, subtraction, multiplication, and division.

There are Four types of quantitative data:

1. **Discrete**
2. **Continues**
3. **Interval data**
4. **Ratio data**

1. **Discrete Data: (Whole Number or a Number without Fractional Part)**

Data that can only take certain values is called discrete data or discrete values. This is data that can be counted and has a limited number of values. It usually comes in the form of **whole numbers or integers.**

Examples:

•Number of siblings.

•Number of goals scored in a soccer match.

•Number of defects in a manufacturing process.

•Number of customers in a store at a given time.

•Age of a Person

•Number of cars in a parking lot

2. Continues Data: (Number with Fractional Part)

Continuous data is data **that can take any value.** Height, weight, temperature and length are all examples of continuous data. Some continuous data will change over time; the weight of a baby in its first year or the temperature in a room throughout the day.

Continuous data represents measurements that can take on any value within a certain range. These values are not restricted to whole numbers and can include decimals or fractions.

Example:

- Height of individuals.
- Weight of objects.
- Temperature readings.
- Time taken to complete a task.
- Distance traveled by a vehicle.

3. Interval data : (Numerical Level of Measurements)

Interval data is a type of **quantitative or Numerical** data that represents numerical values where the difference between any two adjacent points is meaningful and consistent, but there is **no true zero point**. In other words, interval data has equal intervals between consecutive points on the scale, but the absence of a zero point does not imply the absence of the attribute being measured.

Specifically Interval level of data has Order like ordinal data, but in addition to this the spaces between the measurement points are equal, unlike ordinal data.

Examples:

1. Credit Scores which ranges from 300 – 850

2. Temperature:

Temperature measured in degrees Celsius (°C) or Fahrenheit (°F) is an example of interval data. The difference between any two adjacent points on the temperature scale represents a consistent amount of change, but there is **no true zero point**. For example, the difference between 20°C and 30°C is the same as the difference between 30°C and 40°C which is 10°C, but a temperature of 0°C does not signify the absence of temperature i.e The temperature of 0°C does not mean that there is no temperature. So the zero is arbitrary. Therefore, the absence of a zero point does not imply the absence of the attribute being measured.

4. Ratio: is more sophisticated level of measurement.

Ratio data is a type of quantitative data that shares characteristics with interval data **but includes a true zero** point, allowing for meaningful ratios and comparisons.

Ratio data represents numeric data for which exact value can be measured. Absolute zero is available for ratio data. Also, these variables can be added, subtracted, multiplied, or divided. The central tendency can be measured by mean, median, or mode and methods of dispersion such as standard deviation. Examples of ratio data include **height, weight, age, salary etc.**

So what makes the ratio the king of measurement level is that the zero point reflects an absolute meaningful zero, unlike interval level data.

For Example:

1. Height and weight measurements: Height and weight are examples of ratio data. **A height of 0 centimeters or a weight of 0 kilograms** represents the absence of height or weight, making ratios and comparisons meaningful.

2. Age: Age, when measured in years, is ratio data because it includes a true zero point (birth). A person who is 30 years old is twice as old as a person who is 15 years old.

3. Income: Income levels, when measured in currency (e.g., dollars), are ratio data. A person with an income of $0 has no income, and ratios can be calculated to compare incomes between individuals.

Why Diff Levels of Measurement

	Nominal	Ordinal	Interval	Ratio
Categories	✓	✓	✓	✓
Order/rank		✓	✓	✓
Equal spacing			✓	✓
True absolute zero				✓
Can add and subtract			✓	✓
Can multiply and divide				✓
Can calculate mode	✓	✓	✓	✓
Can calculate median		✓	✓	✓
Can calculate arithmetic mean			✓	✓
Can calculate geometric mean				✓

Why Diff Levels of Measurement

Because It is very important to define types of data , is because the level of measurement for any set of data will directly impact **which statistical test we can and cannot use on it**.

For example we can only calculate the mean, the average, for certain levels of data as shown in the diagram. So, if you try to run a statistical test on a unsuitable data set , data that is at the wrong level of measurement we will end up with meaningless result.

So we need to understand what we are dealing with before we really work with the numbers.

The Level of Measurement impacts which statistical test you can use.

CHAPTER-2 FEATURE ENGINEERING & BAYESIAN CONCEPT

Session 1: Introduction to Feature Engineering, Feature transformation

Session 2: Feature subset selection

Session 3: Importance of Bayesian methods

Session 4: Bayes Theorem, Concept learning through Bayes' theorem

Session 5: Bayesian Belief Network

Session 6: Applications of Bayes Theorem

Session 7: Applications Bayesian Belief Network

Session 8: Revision and Assignments

Introduction:

Feature Engineering plays a **vital role in solving any machine learning problem**. This area deals with **features** of the data set, which form an important input of any machine learning problem — be supervised or unsupervised learning. Feature engineering is a critical **preparatory process** in machine learning. It is responsible for taking raw input data and **converting that to well aligned features** which are ready to be used by the machine learning models.

What is Feature:

A feature is an **attribute** of a data set that is used in a machine learning process.
In machine learning, features are **individual independent variables** that act like a input in our system. Actually, while making the predictions, models use such features to make the predictions. And using the feature engineering process, new features can also be obtained from old features in machine learning.

The features in a data set are also called **its dimensions**. Hence, a data set having ‘n’ features is called an **n-dimensional** data set.

Examples of features:

In a **document**, the word count, character count and the number of pages could be the features. Hence, it can be called as 3 dimensional d**image**ata set.

In a **black and white**, the length, width, number of black pixels and number of white pixels could be its features. Hence, it is known as 4-dimensional data set

FEATURE ENGINEERING:

Feature engineering refers to the process of **translating** a **data set into features** such that these features are able to **represent the data set more effectively** and result in a **better learning performance**. Feature engineering is an important pre-processing step for machine learning. It has two major elements:

1. **Feature transformation**
2. **Feature subset selection**

Feature engineering is the process of **creating new features** or **modifying existing ones** from the raw data to improve the performance of machine learning models. It involves **selecting**, **transforming**, and **creating features** that are relevant and informative for the task at hand. Effective feature engineering can lead to better model performance and **generalization**.

Feature transformation:

Feature transformation **transforms the data** — structured or unstructured, into a **new set of features** which can represent the underlying problem which machine learning is trying to solve.

Feature transformation involves **changing the representation** of the features in the dataset to make them more suitable for the machine learning algorithm.

Engineering a good feature space is a crucial prerequisite for the **success** of any machine learning model. However, **often it is not clear which feature is more important**. For that reason, all available attributes of the data set are used as features and the problem of identifying the important features is left to the learning model. This is definitely not a feasible approach, particularly for certain domains e.g. medical image classification, text categorization, etc. In case a model has to be trained to classify a document as spam or non-spam, we can represent a document as a bag of words. Then the feature space will contain all unique words occurring across all documents. This will easily be a feature space of a few hundred thousand features. If we start including bigrams or trigrams along with words, the count of features will run in millions. To deal with this problem, feature transformation comes into play. **Feature transformation is used as an effective tool for dimensionality reduction** and hence **for boosting** learning model performance. **Broadly, there are two distinct goals of feature transformation:**

1. Achieving best reconstruction of the original features in the data set
2. Achieving highest efficiency in the learning task

There are two variants of feature transformation:

1. **Feature construction (or Generation):**
2. **Feature extraction:**

1. **Feature construction (or Generation):**

Feature construction involves **creating new features by combining or transforming the existing features** in the dataset. This process discovers missing information about the relationships between features and increases the feature space by creating additional features. Hence, if there are 'n' features or dimensions in a data set, after feature construction 'm' more features or dimensions may get added. So at the end, the data set will become '**n + m**' dimensional.

Feature construction involves transforming a given set of input features to generate a new set of **more powerful features**. To understand more clearly, let's take the example of a **real estate data set having details of all apartments sold in a specific region**. The data set has three features — **apartment length, apartment breadth, and price of the apartment**. If it is used as an input to a regression problem, such data can be training data for the regression model. So given the training data, the model should be able to predict the price of an apartment whose price is not known or which has just come up for sale. However, instead of using length and breadth of the apartment as a predictor, it is much convenient and makes more sense to use the **area of the apartment**, which is not an existing feature of the data set. Hence, such a feature, namely apartment area, can be added to the data set. In other words, we transform **the three-dimensional data set** to a **four-dimensional data set**, with the newly 'discovered' feature apartment area being added to the original data set.

2. Feature extraction:

Feature extraction involves **reducing the dimensionality** of the data **by selecting or extracting a subset of relevant features** from the original feature set. This process aims to retain as much relevant information as possible **while discarding redundant or irrelevant features.**
Unlike feature construction, which creates entirely new features, feature extraction aims to capture the essence of the original features in a more compact or meaningful representation.

Feature subset selection: (or simply feature selection)

In Feature subset selection **no new feature is generated**. Feature subset selection is a technique in feature engineering that involves choosing a subset of most relevant features from the original set of features in a dataset. The goal is to improve the **model's performance** **by reducing the dimensionality** of the data and eliminating irrelevant or redundant features. This process can lead to more efficient and interpretable models, as well as potentially faster training times.

Certainly! Feature selection is a **critical process** in machine learning aimed at identifying the most relevant subset of features from the original set of features in a dataset. Initially, we have a set of features $F=\{F1,F2,...,Fn\}$, representing various attributes of the data. The goal is to derive a subset $F'=\{Fj,Fo,...,Fm\}$ from F, where $m<n$ and $Fj,Fo,...,Fm$ are the selected features. Among these selected features, Fy represents the subset deemed most meaningful and relevant for the machine learning task at hand.

Feature Extraction Algorithms:

The most popular Feature extraction algorithms in Machine Learning are:

1. Principal Component Analysis
2. Singular value decomposition
3. Linear Discriminant Analysis

Introduction to Bayesian Learning:

Bayes theorem is given by an English statistician, philosopher, and Presbyterian minister named **Mr. Thomas Bayes** in 17^{th} century. Bayes theorem is also known with some other name such as **Bayes rule or Bayes Law.**

Bayes theorem helps to determine the probability of an event with random knowledge. It is used to **calculate the probability of occurring one event while other one already occurred.** It is a best method to relate the condition probability and marginal probability.

In simple words, we can say that Bayes theorem **helps to contribute more accurate results**.

Bayesian Learning: A Probabilistic Approach to Machine Learning

Bayesian learning is a powerful statistical approach that utilizes **probability** to create and update models based on data. Unlike traditional machine learning algorithms that **focus on minimizing error**, Bayesian learning works by estimating **the posterior probability** of a model being correct given the observed data.

In the context of Bayesian learning, "**observed data**" refers to the **data that is available for analysis and model training**. It consists of the input features and corresponding labels or outcomes that are used to build and update the model.

Key concepts:

1. **Probabilistic Framework:** Everything in Bayesian learning is **expressed in terms of probabilities**. This allows for uncertainty quantification and flexible model updates as new data arrives.

2. Prior Probability:

Before seeing any data, you express your initial belief about the model parameters using a **prior distribution**. This prior reflects your existing knowledge or assumptions about the problem.

3. Likelihood: (Something Which is likely to occur)

This represents the probability of observing the data given a specific set of model parameters. It tells you how well the model explains the data.

4. Posterior Probability:

Using Bayes' theorem, you combine the prior and likelihood to calculate the **posterior probability**, which reflects your updated belief about the model parameters after seeing the data.

5. Continuous Learning:

As new data becomes available, the posterior probability is recalculated, effectively updating the model and incorporating the new information.

Example for Bayesian Learning: Spam Filtering

Imagine you're training a **filter** to classify emails as spam or not spam.

- **Prior:** You might start with a prior belief that 5% of emails are spam (**prior probability**).
- **Likelihood:** If you have features like "words in subject line" and "sender domain," you can calculate the likelihood that an email is spam based on these features.
- **Posterior:** Using Bayes' theorem, you update your belief about whether an email is spam based on the actual words in the email and the sender domain.
- **Continual Learning:** As you receive more emails with classifications (spam or not spam), you can keep updating your posterior probabilities and priors for better filtering.

Features of Bayesian learning :

The features of Bayesian learning methods that have made them popular are as follows:

1. Prior knowledge of the candidate hypothesis is combined with the observed data for arriving at the final probability of a hypothesis. So, two important components are the prior probability of each candidate hypothesis and the probability distribution over the observed data set for each possible hypothesis.

2. The Bayesian approach to learning is more flexible than the other approaches because each observed training pattern can influence the outcome of the hypothesis by increasing or decreasing the estimated probability about the hypothesis, whereas most of the other algorithms tend to eliminate a hypothesis if that is inconsistent with the single training pattern.

3. Bayesian methods can perform better than the other methods while validating the hypotheses that make probabilistic predictions. For example, when starting a new software project, on the basis of the demographics of the project, we can predict the probability of encountering challenges during execution of the project.

4. Through the easy approach of Bayesian methods, it is possible to classify new instances by combining the predictions of multiple hypotheses, weighted by their respective probabilities.

5. In some cases, when Bayesian methods cannot compute the outcome deterministically, they can be used to create a standard for the optimal decision against which the performance of other methods can be measured.

Bayes Theorem: (Conditional Probability)

Bayes theorem is one of the most popular machine learning concepts that **helps to calculate the probability of occurring one event with uncertain knowledge while other one has already occurred.**

In ML, Bayes Theorem is used to calculate the probability of a **Hypothesis (theory)** or an **event** based on prior knowledge or evidence.

Bayes' Theorem is a fundamental concept in probability theory and statistics, and it plays a crucial role in various machine learning algorithms, particularly those based on Bayesian inference. In machine learning, Bayes' Theorem is **used to calculate the posterior probability of a hypothesis given observed data.**

Here "observed data" refers to the dataset or information that is available for analysis. It consists of the input features and corresponding outcomes or labels that are used to make inferences or predictions.

Definition:

Bayes theorem One of the most well-known theories in machine learning, the Bayes theorem helps determine the **likelihood that one event will occur with unclear information while another has already happened.** The mathematical formulation of the Bayes theorem is:

Conditional Probability: (Pre requisite for Bayes Theorem)

Conditional probability is a measure of the probability of an event occurring given that another event has already occurred. It is denoted as $P(A|B)$ and is read as "**the probability of event A given event B**.“

Formula for Conditional Probability for Event A given that (|) event B already occurred:

Probability of *A* and *B*

$$P(A\mid B)=\frac{P(A\cap B)}{P(B)}$$

Probability of *A* given *B*

Probability of *B*

Here:

- $P(A|B)$ is the conditional probability of event A given event B. So A and B are 2 events.
- $P(A\cap B)$ is the joint probability of events A and B occurring together.
- $P(B)$ is the probability of event B.

Formula for Conditional Probability for Event B given that (|) event A already occurred:

$$P(B|A)=\frac{P(A\ and\ B)}{P(A)}$$

Derivation of **Bayes Theorem:**

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$$

It states that the probability of an event A given that the event B has already occurred is equal to the probability of an event B given that the event A has occurred, Multiplied by the probability of event A, divided by probability of event B.

Here: Consider A is our Hypothesis and B is the Evidence or data.

1. In P(A|B) : In this A is a Hypothesis and B is a evidence or data. So we have to find the probability of Hypothesis A given the Evidence or Data B.

2. To Compute the Posterior Probability P(A|B) we can't find directly instead have to take the help of 3 other probability which is P(B|A), P(A) and P(B). So after combining Prior, Likelyhood, and Marginal probability we will get Posterior.

3. In P(B|A): Here the probability of Evidence B given that our Hypothesis A is True. So here Our Hypothesis is True and based on this we have to find the Probability of the Evidence.

4. P(A) : is Probability of our Hypothesis A before considering the Evidence or Data B.

5. P(B): Probability of the Evidence or given Data B

LIKELIHOOD

The probability of "B" being True, given "A" is True

PRIOR

The probability "A" being True. This is the knowledge.

$$P(A|B) = \frac{P(B|A).P(A)}{P(B)}$$

POSTERIOR

The probability of "A" being True, given "B" is True

MARGINALIZATION

The probability "B" being True.

Class-3