# MACHINE LEARNING — MODULE 1

**Simple + Understandable + Exam Ready Notes**

---

# 1. What is Machine Learning?

## Simple meaning

**Machine Learning (ML)** means teaching a computer to **learn from data and experience** so that it can make predictions or decisions.

Normally, we give a computer exact instructions.

But in ML:

**Data → Computer learns patterns → Makes prediction**

The machine is not explicitly programmed for every possible situation.

## 🧠 Real-world example

Think about **YouTube recommendations**.

You watch:

- 5 programming videos
- 3 AI videos
- 2 Python videos

YouTube studies your previous activity and notices your interests.

Then it recommends more programming and AI videos.

That's Machine Learning.

## Exam definition

> **Machine Learning is a branch of Artificial Intelligence that enables computers to learn patterns from data and use them to make predictions or decisions without being explicitly programmed for every task.**

Arthur Samuel introduced the term **Machine Learning in 1959**.

### Remember

**ML = Learn from data + Find patterns + Predict**

---

# 2. How Does Machine Learning Work?

The basic idea is very simple:

1. Give data to the computer.
2. The computer studies the data.
3. It finds patterns.
4. It creates a model.
5. Give new data.
6. The model predicts the answer.

### Example

Suppose we give a computer:

| Hours studied | Exam result |
|---|---|
| 2 | Fail |
| 4 | Pass |
| 6 | Pass |
| 8 | Pass |

The computer finds a relationship between **study hours and results**.

Now give:

> 5 hours

The model may predict:

> **Pass**

---

# 3. Machine Learning Life Cycle

The **Machine Learning Life Cycle** is the series of steps followed to create a machine learning system.

There are **7 major steps**:

1. Gathering Data
2. Data Preparation
3. Data Wrangling
4. Data Analysis
5. Train the Model
6. Test the Model
7. Deployment

### Easy way to remember

**G → P → W → A → T → T → D**

**Gather → Prepare → Wrangle → Analyse → Train → Test → Deploy**

---

## 3.1 Gathering Data

This is the **first step**.

We collect the data needed for our ML problem.

Data can come from:

- Files
- Databases
- Internet
- Mobile devices

The quality and quantity of data affect the final result.

### Example

If we want to build an ML model that predicts house prices, we collect:

- House size
- Location
- Number of rooms

- Previous prices

---

## 3.2 Data Preparation

After collecting data, we prepare it for ML.

This includes:

- Putting data together
- Organizing it
- Randomizing the order
- Understanding the data

Data exploration helps us find:

- Relationships
- Trends
- Outliers

### Example

Suppose we have student marks:

```
80, 75, 90, 85, 999
```

We can identify **999** as an unusual value.

---

# 3.3 Data Wrangling

**Data wrangling** means **cleaning and converting raw data into a useful format**.

Real-world data can contain:

- Missing values
- Duplicate data
- Invalid data
- Noise

We clean these problems before using the data.

### Example

Suppose a student database contains:

| Name | Age |
|---|---|
| Rahul | 20 |
| Rahul | 20 |
| Anu | — |
| John | -50 |

Problems:

- Rahul appears twice → duplicate
- Anu's age is missing
- John has invalid age

These problems need to be handled.

### Remember

**Data Wrangling = Clean the dirty data**

---

# 3.4 Data Analysis

After cleaning the data, we analyse it.

Main activities:

1. Select analytical techniques
2. Build models
3. Review the results

We decide what type of ML problem we have, such as:

- Classification
- Regression
- Clustering
- Association

---

# 3.5 Train the Model

Now we **teach the model** using training data.

The model learns:

- Patterns
- Rules
- Features

### Example

Give the model thousands of cat and dog images.

It learns things like:

- Shape
- Colour
- Eyes
- Ears
- Tail

---

# 3.6 Test the Model

After training, we test the model using **test data**.

The purpose is to check how accurately the model works.

### Simple example

If we test 100 images and the model correctly identifies 90:

**Accuracy = 90%**

---

# 3.7 Deployment

If the model performs well enough, we put it into a **real-world system**.

This is called **deployment**.

### Example

A trained spam detection model is added to Gmail.

Now it can automatically detect spam emails.

---

# 4. Features of Machine Learning

Important features:

- Uses data to detect patterns
- Learns from past data
- Can improve automatically
- Data-driven technology
- Can handle large amounts of data

---

# 5. Need for Machine Learning

Why do we need ML?

Because humans cannot manually analyse **huge amounts of data**.

Machine Learning can:

- Handle large amounts of data
- Solve complex problems
- Save time
- Save money
- Help make decisions
- Find hidden patterns

### Real-world examples

ML is used in:

- Self-driving cars
- Fraud detection
- Face recognition
- Netflix recommendations
- Amazon recommendations

### Simple idea

**More data + ML → Faster analysis + Better predictions**

---

# 6. Machine Learning vs Traditional Programming vs AI

This is an important exam topic.

| Machine Learning | Traditional Programming | Artificial Intelligence |
|---|---|---|
| Learns from data | Uses rules written by programmer | Broad field of intelligent machines |
| Data-driven | Rule-based | Can use ML, rules and other techniques |
| Learns patterns | Programmer provides instructions | Attempts human-like intelligent tasks |
| Can adapt to new data | Usually requires code changes | Includes ML and Deep Learning |

### Easy example

### Traditional Programming

You tell the computer:

> **If marks >= 40 → Pass**
> **Otherwise → Fail**

The programmer writes the rule.

### Machine Learning

Give the computer lots of previous student data.

The computer learns patterns and creates a model to predict results.

### Main difference

**Traditional programming:**

```
Rules + Data → Output
```

**Machine Learning:**

```
Data + Output examples → Model
```

Then:

```
New Data + Model → Prediction
```

ML is therefore more flexible and adaptive compared with traditional programming.

---

# 7. Applications of Machine Learning

ML is used in many areas.

## 7.1 Image Recognition

ML can identify:

- People
- Objects
- Places
- Faces

### Example

Facebook can suggest names when you upload a photo containing your friends.

---

## 7.2 Speech Recognition

Converts **voice into text**.

Examples:

- Google Assistant
- Siri
- Alexa

---

## 7.3 Traffic Prediction

Google Maps can predict traffic using:

- Current vehicle locations
- Sensors
- Historical travel information

---

## 7.4 Product Recommendations

Amazon and Netflix use ML to recommend products, movies and shows based on user activity and interests.

### Example

You watch many action movies.

Netflix may recommend more action movies.

---

## 7.5 Self-driving Cars

ML is used in self-driving technology to help detect:

- People
- Objects
- Road conditions

---

## 7.6 Spam Detection

ML helps classify emails as:

- Important
- Normal
- Spam

Algorithms mentioned in the module include:

- Multi-Layer Perceptron
- Decision Tree
- Naive Bayes

---

## 7.7 Virtual Personal Assistants

Examples:

- Google Assistant
- Siri
- Alexa
- Cortana

They understand voice commands and perform tasks.

---

## 7.8 Online Fraud Detection

ML can identify whether an online transaction is:

**Genuine or Fraudulent**

The module mentions Feed Forward Neural Networks for this purpose.

---

## 7.9 Stock Market Trading

ML can be used to predict stock market trends.

The module mentions **Long Short-Term Memory (LSTM)** networks.

---

## 7.10 Medical Diagnosis

ML is used in medical diagnosis and can help identify conditions such as brain tumors using medical data and images.

---

## 7.11 Language Translation

ML can translate one language into another.

Example:

**English → Malayalam**

The module mentions Google's **GNMT** and sequence-to-sequence learning.

---

# 8. Data in Machine Learning

**Data is the most important part of ML.**

Data means observations or measurements used to train an ML model.

Examples:

- Age
- Salary
- Marks
- Images
- Sound
- Text
- Time-series values

### Why is data important?

The quality and quantity of data affect the performance of the model.

**Good data → Better learning**

---

# 9. Labeled and Unlabeled Data

## Labeled Data

Data that already has the **correct answer/target**.

Example:

| Image | Label |
|---|---|
| 🐱 | Cat |
| 🐶 | Dog |

The answer is already provided.

Used mainly in **Supervised Learning**.

## Unlabeled Data

Data without a target/answer.

Example:

| Image |
|---|
| 🐱 |
| 🐶 |
| 🍎 |
| 🍊 |

The machine has to find patterns itself.

Used mainly in **Unsupervised Learning**.

---

# 10. Data, Information and Knowledge

### Data

Raw facts that have not been interpreted.

Example:

```
85, 90, 78
```

### Information

Data that has been processed and given meaning.

Example:

> Bob scored 90 marks.

### Knowledge

Understanding gained from information, experience and learning.

Example:

> Bob is performing well in the subject.

### Easy memory

**Data → Information → Knowledge**

---

# 11. Forms of Data

There are three important forms mentioned in the module.

## 11.1 Numeric Data

Data represented using numbers.

Examples:

- Age = 21
- Salary = ₹30,000
- Height = 170 cm

---

## 11.2 Categorical Data

Data representing categories.

Examples:

- Male/Female
- Red/Blue
- Apple/Banana/Orange

---

## 11.3 Ordinal Data

Categorical data where the categories have an **order**.

Example:

**Small < Medium < Large**

or

**Not Happy → Neutral → Happy → Very Happy**

---

# 12. Mean, Median and Mode

These are basic statistical values.

### Mean

**Mean = Average**

Example:

```
10, 20, 30
```

Mean:

```
(10 + 20 + 30) / 3 = 20
```

---

## Median

**Median = Middle value**

First arrange the values in order.

Example:

```
10, 20, 30
```

Median = **20**

If there are two middle values:

```
10, 20, 30, 40
```

Median:

```
(20 + 30) / 2 = 25
```

---

## Mode

**Mode = Most frequently occurring value**

Example:

```
10, 20, 20, 30
```

Mode = **20**

## Memory trick

**Mean → Average**
**Median → Middle**
**Mode → Most**

---

# 13. Types of Machine Learning

There are **4 main types**:

1. Supervised Learning
2. Unsupervised Learning
3. Semi-Supervised Learning
4. Reinforcement Learning

---

# 14. Supervised Learning

## Simple meaning

In **Supervised Learning**, the machine learns using **labeled data**.

That means:

> **Input + Correct Answer**

are given during training.

## Real-world example

We give the computer:

- Cat image → Cat
- Dog image → Dog
- Cat image → Cat
- Dog image → Dog

The machine learns the patterns.

Then we give a new image.

It predicts:

> **Cat**

## Easy analogy

👨‍🏫 Teacher teaches student.

The teacher provides the correct answers.

That's why it is called **Supervised**.

---

# 15. Types of Supervised Learning

There are two main types:

1. Classification
2. Regression

---

## 15.1 Classification

Classification predicts a **category**.

Examples:

- Yes / No
- Spam / Not Spam
- Cat / Dog
- Red / Blue

### Example

Email:

> "Congratulations! You won ₹10 lakh!"

Model:

**Spam**

Popular algorithms mentioned:

- Random Forest
- Decision Tree
- Logistic Regression
- Support Vector Machine (SVM)

---

## 15.2 Regression

Regression predicts a **continuous numerical value**.

Examples:

- House price
- Temperature
- Stock trend
- Sales amount

### Example

House size:

```
1000 sq.ft → ₹40 lakh
```

```
1500 sq.ft → ₹60 lakh
```

The model can predict the price of a new house.

Algorithms mentioned:

- Simple Linear Regression
- Multivariate Regression
- Decision Tree
- Lasso Regression

### Easy difference

**Classification → Category**

**Regression → Number**

---

# 16. Advantages and Disadvantages of Supervised Learning

### Advantages

- Uses labeled data.

- We know the classes/categories.
- Can predict outputs based on previous experience.

### Disadvantages

- May not work well with very complex tasks.
- Can give wrong predictions when new data is very different from training data.
- Training can require significant computational time.

### Applications

- Medical diagnosis
- Fraud detection
- Spam detection
- Speech recognition
- Image segmentation

---

# 17. Unsupervised Learning

### Simple meaning

In **Unsupervised Learning**, the machine learns from **unlabeled data**.

There is no teacher telling it the correct answer.

The machine tries to find:

- Patterns
- Similarities
- Differences
- Groups

### Real-world example

Imagine giving a supermarket customer database to a computer.

You don't tell it:

> "This is a student customer."
>
> "This is a rich customer."

Instead, the computer looks at purchasing patterns and may automatically create groups.

### Easy analogy

**👦 Student studying by himself**

No teacher gives the answer.

That's similar to **Unsupervised Learning**.

---

# 18. Types of Unsupervised Learning

Two important types:

1. Clustering
2. Association

---

## 18.1 Clustering

Clustering means **putting similar things into groups**.

### Example

Customers:

- Customer A buys expensive products.
- Customer B buys expensive products.
- Customer C buys cheap products.
- Customer D buys cheap products.

The model may create:

**Group 1 → Expensive-product customers**

**Group 2 → Cheap-product customers**

A popular algorithm is:

**K-Means**

Other algorithms mentioned:

- Mean-shift
- DBSCAN
- PCA
- ICA

---

# 19. Association

Association finds **relationships between things**.

### Real-world example

A supermarket notices:

> People who buy bread often buy butter.

So the supermarket can place bread and butter near each other or recommend butter when someone buys bread.

This is called **Association**.

It is used in:

- Market Basket Analysis
- Web usage mining
- Continuous production

Algorithms include:

- Apriori
- Eclat
- FP-growth

### Remember

**Clustering → Groups**

**Association → Relationships**

---

# 20. Advantages and Disadvantages of Unsupervised Learning

### Advantages

- Works with unlabeled data.
- Unlabeled data is easier to obtain.
- Can be useful for complex tasks.

### Disadvantages

- Results may be less accurate.
- There is no predefined correct output.
- It can be difficult to interpret the results.

---

# 21. Semi-Supervised Learning

Semi-supervised learning is **between supervised and unsupervised learning**.

It uses:

**A small amount of labeled data + a large amount of unlabeled data**

### Example

Suppose we have:

- 1000 images
- Only 100 images are labeled
- 900 images are unlabeled

Instead of throwing away the 900 images, we use both types of data.

### Easy student example

- Teacher teaches you → **Supervised**
- You study completely alone → **Unsupervised**
- Teacher teaches you and then you revise by yourself → **Semi-supervised**

This exact student analogy is given in your module.

# 22. Reinforcement Learning

## Simple meaning

Reinforcement Learning (RL) is learning through **trial and error**.

An agent:

1. Takes an action.
2. Gets feedback.
3. Gets a reward for a good action.
4. Gets punishment for a bad action.
5. Learns from the experience.
6. Tries to maximize rewards.

## 🎮 Real-world example

Imagine a game-playing AI.

If it makes a good move:

🏆 **Reward +10**

If it makes a bad move:

❌ **Penalty -10**

After playing many times, it learns which actions give better results.

## Easy analogy

A child learning to ride a bicycle:

- Falls → learns what went wrong
- Balances → gets better
- Repeats → improves

That's the basic idea of reinforcement learning.

# 23. Positive and Negative Reinforcement

### Positive Reinforcement

A good action is encouraged by **adding something positive**.

### Negative Reinforcement

A behaviour is encouraged by **avoiding a negative condition**.

### Applications

- Video games
- Robotics
- Resource management
- Text mining

### Advantages

- Useful for complex real-world problems
- Similar to learning through experience
- Can achieve long-term results

### Disadvantages

- Not ideal for simple problems
- Requires large amounts of data and computation

---

# 24. Major Challenges of Machine Learning

Machine Learning has several challenges.

## 1. Inadequate Training Data

Not enough data or poor-quality data can cause poor results.

---

## 2. Poor Quality Data

Data may contain:

- Noise
- Missing information
- Incorrect information
- Unclean data

This can reduce accuracy.

---

## 3. Non-representative Training Data

If the training data does not properly represent the real-world situation, the model may perform poorly on new data.

---

# 25. Overfitting

### Simple meaning

**Overfitting means the model learns the training data too much but performs poorly on new data.**

### Example

Imagine a student who memorizes the exact questions from previous exams.

Previous paper:

> 2 + 2 = ?

Student memorizes:

> 4

But when the teacher asks:

> 3 + 2 = ?

The student cannot answer.

The student **memorized instead of learning the concept**.

That's similar to overfitting.

### In ML

**Training performance → Very good**

**New/unseen data → Poor**

### Ways to reduce overfitting

- Increase training data
- Reduce model complexity
- Use Ridge/Lasso regularization
- Early stopping
- Reduce noise
- Reduce unnecessary attributes
- Constrain the model

---

# 26. Underfitting

Underfitting is basically the opposite of overfitting.

The model **does not learn enough**.

It performs poorly on:

- Training data
- New data

### Example

A student studies only one chapter for an exam.

The student doesn't know enough to answer either old or new questions.

That's similar to **underfitting**.

### How to reduce underfitting

- Increase model complexity
- Remove noise
- Use better/more features

---

# 27. Overfitting vs Underfitting

| Overfitting | Underfitting |
|---|---|
| Learns too much | Learns too little |
| Very good on training data | Poor on training data |
| Poor on new data | Poor on new data |
| Model is too complex | Model is too simple |

### Memory trick

**Over → Overlearns**

**Under → Underlearns**

---

# 28. Monitoring and Maintenance

An ML model cannot simply be created once and forgotten.

The model needs **regular monitoring and maintenance**.

Why?

Because real-world data can change.

---

# 29. Data Drift / Bad Recommendations

Suppose Netflix learns that you love action movies.

After some time, you stop watching action movies and start watching comedy.

But the system continues recommending action movies.

The user's interests have changed, but the model is still using old patterns.

This is related to **data drift**.

Regularly updating and monitoring data can help.

---

# 30. Process Complexity

Machine Learning projects can be complicated because they involve:

- Data collection
- Cleaning
- Analysis
- Training
- Testing
- Deployment
- Maintenance

The module identifies process complexity as a major challenge.

---

# 31. Data Bias

**Data bias** occurs when some parts of the dataset are given too much importance or are not represented fairly.

Biased data can produce inaccurate results.

### Simple example

Suppose an image recognition system is trained mostly using one type of image and has very little variety.

It may perform poorly on images that are different from its training data.

---

# 32. NumPy

**NumPy = Numerical Python**

It is a Python library used mainly for:

- Numerical calculations
- Arrays
- Mathematical operations
- Statistics
- Matrix operations

It provides efficient multidimensional arrays.

### Simple example

```
import numpy as np

arr = np.array([10, 20, 30, 40])

print(arr)
print(arr + 5)
print(np.mean(arr))
```

### Remember

**NumPy → Numbers + Arrays**

---

# 33. Pandas

**Pandas** is a Python library used for **data manipulation and analysis**.

Think of Pandas as a tool that helps us work with data in **table form**, similar to Excel.

### Two main data structures

**Series → 1D**

**DataFrame → 2D table**

### Example

```
import pandas as pd

data = {
```

```
    "Name": ["Alice", "Bob", "Charlie"],
    "Marks": [85, 90, 78]
}

df = pd.DataFrame(data)

print(df)
print(df["Marks"].mean())
```

### Remember

**Pandas → Tables + Data**

---

# 34. Scikit-learn

**Scikit-learn** is a Python library used to build and work with Machine Learning models.

It provides tools for:

- Training models
- Testing models
- Evaluating models
- Data preprocessing
- Supervised learning
- Unsupervised learning

### Algorithms mentioned

**Supervised:**

- Linear Regression
- Logistic Regression
- Decision Tree
- Random Forest
- SVM
- KNN
- Naive Bayes

**Unsupervised:**

- K-Means
- Hierarchical Clustering

- PCA

### Simple example

from sklearn.linear_model import LinearRegression

model = LinearRegression()

print("Linear Regression Model Created")

### Remember

**Scikit-learn → Machine Learning models**

---

# 35. NumPy vs Pandas vs Scikit-learn

| Library | Main purpose | Easy way to remember |
|---|---|---|
| **NumPy** | Numerical computing | Numbers |
| **Pandas** | Data analysis | Tables |
| **Scikit-learn** | Machine Learning | Models |

### Together

**NumPy → Calculate**

**Pandas → Organize**

**Scikit-learn → Learn**

---

# 36. NumPy Array Arithmetic

Example:

import numpy as np

a = np.array([1, 2, 3])

```
b = np.array([4, 5, 6])

print(a + b)
print(b - a)
print(a * b)
print(b / a)
```

Output:

```
Addition: [5 7 9]
Subtraction: [3 3 3]
Multiplication: [ 4 10 18]
Division: [4.  2.5 2.]
```

---

# 37. Matrix Operations

Example:

```
import numpy as np

A = np.array([[1,2],[3,4]])
B = np.array([[5,6],[7,8]])

print(A + B)
print(np.dot(A,B))
```

Matrix addition:

```
[[ 6  8]
 [10 12]]
```

Matrix multiplication:

```
[[19 22]
 [43 50]]
```

---

# ⭐ MODULE 1 — SUPER IMPORTANT REVISION

If you have very little time before the exam, remember these:

## Machine Learning

**ML = Computer learns from data and makes predictions.**

## ML Life Cycle

**Gather → Prepare → Wrangle → Analyse → Train → Test → Deploy**

## Four Types

**Supervised → Unsupervised → Semi-supervised → Reinforcement**

## Supervised

**Labeled data**

Two types:

**Classification → Category**

**Regression → Number**

## Unsupervised

**Unlabeled data**

Two types:

**Clustering → Groups**

**Association → Relationships**

## Semi-supervised

**Small labeled + large unlabeled data**

## Reinforcement

**Action → Feedback → Reward/Punishment → Learn**

## Statistics

**Mean → Average**

**Median → Middle**

**Mode → Most common**

### Problems

**Overfitting → Learns too much**

**Underfitting → Learns too little**

### Python Libraries

**NumPy → Numerical calculations**

**Pandas → Data/table handling**

**Scikit-learn → ML models**

---

# 🎯 Most Important Exam Questions

Based directly on the topics covered in your Module 1:

### 2/3 Mark Questions

1. What is Machine Learning?
2. Define supervised learning.
3. What is unsupervised learning?
4. What is reinforcement learning?
5. What is classification?
6. What is regression?
7. Define clustering.
8. What is data wrangling?
9. What is overfitting?
10. What is underfitting?
11. What is data bias?
12. What is NumPy?
13. What is Pandas?
14. What is Scikit-learn?
15. What is labeled data?
16. What is unlabeled data?
17. Define mean, median and mode.

## 5 Mark Questions

1. Explain the Machine Learning life cycle.
2. Explain supervised learning with an example.
3. Explain unsupervised learning with an example.
4. Explain classification and regression.
5. Explain clustering and association.
6. Explain reinforcement learning.
7. Explain the applications of Machine Learning.
8. Explain the challenges of Machine Learning.
9. Explain NumPy, Pandas and Scikit-learn.
10. Differentiate Machine Learning and traditional programming.

## 10 Mark Questions

1. **Explain the complete Machine Learning life cycle.**
2. **Explain the different types of Machine Learning with examples.**
3. **Explain supervised and unsupervised learning in detail.**
4. **Explain the major applications of Machine Learning.**
5. **Explain the major challenges faced by Machine Learning.**
6. **Explain NumPy, Pandas and Scikit-learn with examples.**

---