Introduction to Machine Learning

Unit 5: Learning Paradigms, Classification & Clustering, Algorithms & Applications

Dr. Gopal Chandra Jana

Dept. of CSE, Sharda University
Fundamentals of Computing and AI (CSAI1101) — Unit 5

September 3, 2026

Course Page:
https://www.gcjana.in/courses/shardauniversity/2601/fundamentals-of-computing-and-artificial-intelligence/

Outline

Learning Outcomes of Unit 5

After this unit, a student will be able to...

1. **Explain** what Machine Learning is and its role within AI.
2. **Distinguish** the three learning paradigms — supervised, unsupervised and reinforcement learning.
3. **Describe** classification and clustering, and tell them apart.
4. **Explain** the working principles of basic supervised algorithms (KNN, decision trees, and others).
5. **Identify** real-world applications of ML across many domains and outline the steps of a simple ML workflow.

Mapping to the Syllabus

A. Role of ML in AI; supervised, unsupervised & reinforcement learning. **B.** Classification & clustering; supervised algorithms. **C.** Applications of ML across domains.

From Unit 4 to Unit 5

Where we are

In **Unit 4** we saw that *Machine Learning* is the engine behind most modern AI: instead of hand-coding rules, the machine **learns from data**. Now we look inside that engine.

- **How** does a machine "learn"?
- What **kinds** of learning are there?
- How do the basic **algorithms** work?
- Where is ML **used** in the real world?

This is the **core skill** of a CSE (AI & ML) engineer.

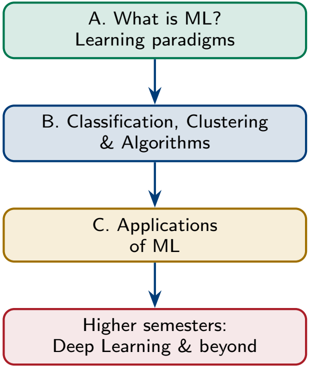


Figure: The road through Unit 5.

What is Machine Learning?

Definition

Machine Learning (ML) is a branch of AI in which computers **learn patterns from data** and improve at a task with experience — *without* being explicitly programmed with fixed rules.

- Arthur Samuel (1959): ML gives computers the ability to learn *without being explicitly programmed*.
- The machine finds the **rules** itself, from examples.
- More **data** + **experience** $\Rightarrow$ better performance.

The big shift

Traditional: *we* write the rules. ML: the machine **learns** the rules from data.

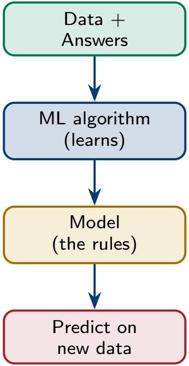


Figure: Data in, a trained model out.

The Role & Significance of ML within AI

Why ML is at the heart of AI

- Most modern AI **is** machine learning.
- It handles problems with **no clear formula** — vision, speech, language.
- It **improves** automatically as more data arrives.
- It **scales** to huge datasets a human could never process.
- It **adapts** to new situations.

Why ML *now*?

Three things came together:

- **Big data**: the internet, phones and sensors.
- **Computing power**: fast GPUs.
- **Better algorithms**: especially deep learning.

This is why ML exploded after **2012**.

Recall: nested circles

Deep Learning $\subset$ **Machine Learning** $\subset$ **AI**. ML is the practical engine that makes AI work today.

How Does a Machine "Learn"?

Learning = improving with data

A machine "learns" by adjusting its internal settings (**parameters**) to reduce its **errors** on the data.

1. Make a prediction.
2. Compare with the correct answer → measure the **error**.
3. **Adjust** to reduce the error.
4. Repeat many times → the model gets better.

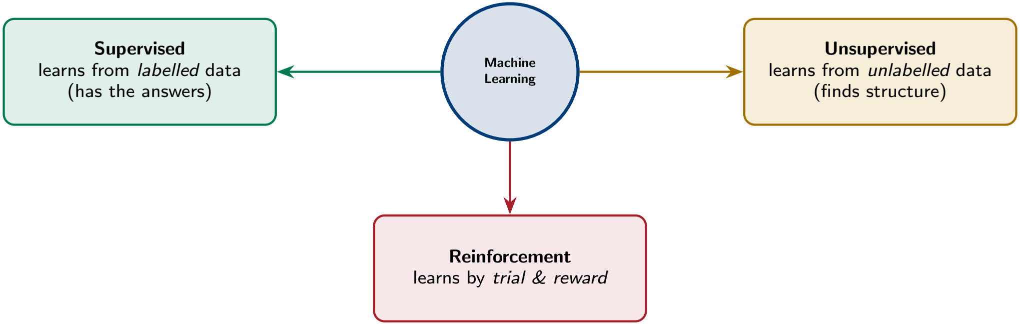
Like a student

A student practises problems, checks answers, learns from mistakes, and improves. ML does the same with data.

Key vocabulary

- **Feature**: an input attribute (e.g. height, colour).
- **Label**: the correct answer (e.g. "cat").
- **Training**: the learning phase, using known data.
- **Model**: what the machine learns.
- **Prediction**: the model's answer on new data.

The Three Learning Paradigms — Overview



The paradigm is chosen by *what kind of data* you have and *what* you want to achieve.

Paradigm 1 — Supervised Learning

Learning with a “teacher”

In **supervised learning**, the data is **labelled** — every example comes with the correct answer. The model learns the mapping from *input* to *output*.

- Data: (features, **label**) pairs.
- Goal: predict the label for new, unseen inputs.
- Two kinds: **classification** and **regression**.

Example

Show the machine 10,000 emails, each labelled *spam* or *not spam*. It learns to label a new email.

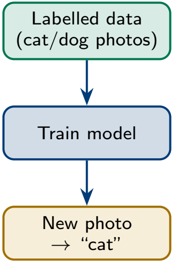


Figure: The teacher provides the answers.

Classification vs Regression (within Supervised)

Classification

Predicts a **category** (a discrete label).

- "Is this email spam or not?"
- "Is this a cat, dog or bird?"
- Output is one of a few *classes*.

Regression

Predicts a **number** (a continuous value).

- "What will this house cost?"
- "What will tomorrow's temperature be?"
- Output is a *quantity*.

Quick test: if the answer is a *label* $\rightarrow$ classification; if a *number* $\rightarrow$ regression.

Paradigm 2 — Unsupervised Learning

Learning *without* a teacher

In **unsupervised learning**, the data has **no labels**. The machine finds *hidden structure* or *groups* on its own.

- Data: only features — *no* answers given.
- Goal: discover patterns, groups or structure.
- Main kinds: **clustering** and **association**.

Example

Give a shop's customer data (no labels). The machine *groups* similar customers together — useful for targeted offers.

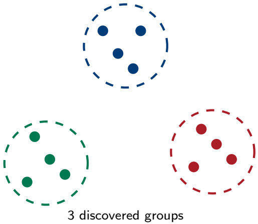


Figure: Clustering finds groups with no labels.

Paradigm 3 — Reinforcement Learning

Learning by trial and reward

In **reinforcement learning (RL)**, an **agent** interacts with an **environment**. It takes actions, receives a **reward** or **penalty**, and learns a strategy to maximise total reward.

- No labelled dataset — it learns from *experience*.
- Good action → reward; bad action → penalty.
- Over time it learns the best **policy** (what to do).

Examples

Training a dog with treats; game-playing AI (AlphaGo); self-driving cars; robots learning to walk.

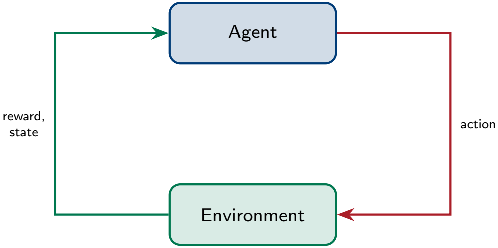


Figure: The RL loop of action and reward.

The Three Paradigms — Side by Side

Aspect	**Supervised**	**Unsupervised**	**Reinforcement**
Data	Labelled	Unlabelled	No dataset; experience
Goal	Predict a label/number	Find groups/structure	Maximise reward
Feedback	Correct answers	None	Rewards & penalties
Examples	Spam filter, prices	Customer grouping	Game AI, robotics
Key tasks	Classification, regression	Clustering, association	Policy learning

Memory hook: Supervised = *learn with answers*; Unsupervised = *find patterns*; Reinforcement = *learn by doing*.

Quick Check — Round 1 (Paradigms)

[Q1] In one line, what is Machine Learning?

[Q2] Which paradigm uses *labelled* data?

[Q3] Predicting a house price is classification or regression? Why?

[Q4] Which paradigm learns by rewards and penalties?

Think for 60 seconds — answers on the next slide.

Answers — Round 1

[A1]

ML is teaching computers to **learn patterns from data** and improve with experience, instead of following hand-coded rules.

[A2]

Supervised learning — every example comes with its correct answer (label).

[A3]

Regression — the output is a *number* (a price), not a category.

[A4]

Reinforcement learning — the agent learns by trial, guided by rewards and penalties.

Classification — Sorting into Classes

The core idea

Classification is a *supervised* task: given labelled examples, learn to place a new item into one of a fixed set of **classes**.

- **Binary**: two classes (spam / not spam).
- **Multi-class**: many classes (cat / dog / bird).
- The model draws a **decision boundary** between classes.

Everyday examples

Spam detection, disease positive/negative, handwriting digit 0–9, sentiment positive/negative.

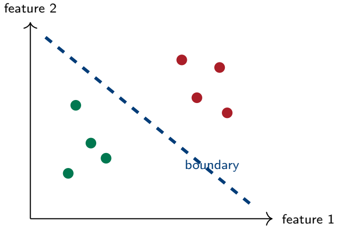


Figure: A boundary separates two classes.

Clustering — Finding Natural Groups

The core idea

Clustering is an *unsupervised* task: with **no labels**, group data so that items in the same cluster are *similar* and items in different clusters are *different*.

- The machine *discovers* the groups.
- Similarity is measured by **distance**.
- Popular method: **K-Means**.

Everyday examples

Customer segmentation, grouping news articles by topic, organising photos by similarity.

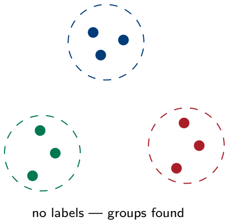


Figure: Clustering discovers groups by similarity.

Classification vs Clustering

Aspect	**Classification**	**Clustering**
Learning type	Supervised	Unsupervised
Data	Labelled	Unlabelled
Goal	Assign to *known* classes	Discover *unknown* groups
Classes known?	Yes, defined in advance	No, found by the algorithm
Example	Spam vs not spam	Group similar customers
Example algorithm	KNN, Decision Tree	K-Means

Key difference: classification sorts into *known* boxes; clustering *creates* the boxes itself.

A Simple ML Workflow

1. Collect data → **2. Prepare** clean & split → **3. Train** the model → **4. Test** & evaluate → **5. Predict** & deploy

Train / test split

We split data into a **training set** (to learn) and a **test set** (to check on unseen data) — usually about 80% / 20%.

Golden rule

Never test on the training data! A model must be judged on data it has *never seen*, just like a fair exam.

Supervised Algorithm 1 — K-Nearest Neighbours (KNN)

"You are like your neighbours"

To classify a new point, **KNN** looks at its **K** closest labelled neighbours and takes a **majority vote**.

1. Choose K (e.g. $K = 3$).
2. Measure **distance** to all labelled points.
3. Find the K nearest.
4. The most common class among them wins.

Choosing K matters

Small $K \rightarrow$ sensitive to noise. Large $K \rightarrow$ smoother but may blur boundaries. Often K is odd, to avoid ties.

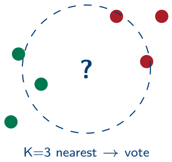


Figure: The new point (?) is classed by its 3 neighbours.

KNN — A Tiny Worked Example

The data (fruit by weight & colour)

Weight	Colour score	Fruit
150	8	Apple
170	7	Apple
130	6	Orange
140	5	Orange

New fruit: (155, 7)

Find its nearest neighbours by distance.

Reasoning ($K = 3$)

- Closest points: (150,8), (170,7) $\rightarrow$ Apple; (140,5) $\rightarrow$ Orange.
- Votes: **Apple 2**, Orange 1.
- Majority $\Rightarrow$ predict **Apple**.

Note

KNN is a **lazy learner**: it does no training — it just stores the data and computes distances at prediction time.

Supervised Algorithm 2 — Decision Trees

A tree of yes/no questions

A **decision tree** asks a series of simple questions, splitting the data until it reaches a decision at a **leaf**.

- Easy to read and **explain** (white box).
- Each split uses the most **informative** feature.
- Works for classification *and* regression.

Watch out

A very deep tree can **overfit** — memorise the training data and fail on new data.

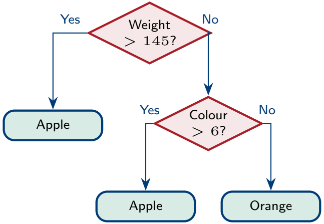


Figure: A decision tree for classifying fruit.

More Supervised Algorithms (At a Glance)

Algorithm	Idea in one line	Task
Linear Regression	Fit a straight line to predict a number	Regression
Logistic Regression	Predict a yes/no probability	Classification
KNN	Vote of the nearest neighbours	Classification
Decision Tree	Yes/no questions to a decision	Both
Naïve Bayes	Use probability & Bayes' rule	Classification
Support Vector Machine	Find the best separating boundary	Classification
Random Forest	Many trees vote together	Both

No single best algorithm — the right choice depends on the data and the problem.

Linear Regression — Fitting a Line

Predicting a number

Linear regression finds the best straight line through the data: $y = mx + c$. It then predicts y for a new x.

- x: input (e.g. house size).
- y: output (e.g. price).
- The line minimises the total **error**.

Example

Bigger house $\rightarrow$ higher price. The line captures that trend and predicts a price for a new size.

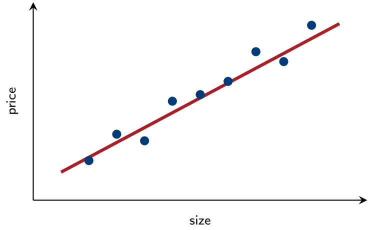


Figure: A line fits the trend in the data.

K-Means — How Clustering Works

The K-Means steps

1. Choose **K** (number of clusters).
2. Place K **centres** randomly.
3. Assign each point to its *nearest* centre.
4. Move each centre to the *average* of its points.
5. Repeat steps 3–4 until centres stop moving.

Choosing K

You must pick K in advance. Tools like the *elbow method* help find a good value.

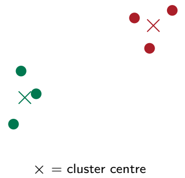


Figure: Points snap to the nearest centre ($\times$).

Overfitting, Underfitting & Good Fit

Getting the balance right

- **Underfitting**: model too simple — misses the pattern (poor on training *and* test).
- **Good fit**: captures the real pattern — works well on new data.
- **Overfitting**: model too complex — memorises noise (great on training, poor on test).

Analogy

A student who *memorises* answers (overfits) fails a new question; one who *understands* (good fit) succeeds.

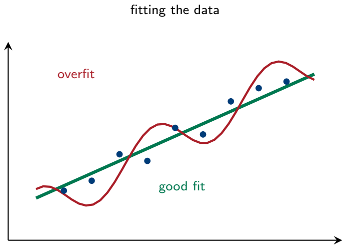


Figure: Green generalises; red chases every wiggle.

Quick Check — Round 2 (Classification, Clustering & Algorithms)

[Q1] Is clustering supervised or unsupervised? Why?

[Q2] In KNN with $K = 3$, a point's neighbours are Apple, Apple, Orange. What is predicted?

[Q3] What does a decision tree use at each split?

[Q4] What is overfitting, in one line?

Answers — Round 2

[A1]

Unsupervised — it works on *unlabelled* data and discovers the groups itself.

[A2]

Apple — by majority vote (Apple 2, Orange 1).

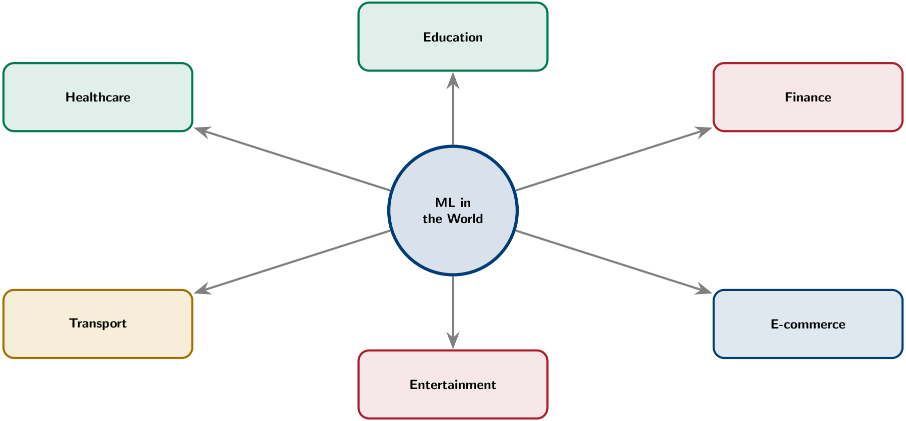
[A3]

A **yes/no question** on the most *informative* feature, to best split the data.

[A4]

Overfitting is when a model memorises the training data (including noise) and performs poorly on new, unseen data.

ML Applications — The Big Picture



ML is a *general-purpose* technology — it improves almost every field.

Applications — Healthcare & Finance

Healthcare

- **Disease detection** from X-rays, MRI, CT scans.
- **Diabetic-retinopathy** screening from eye images.
- **Drug discovery** and protein-structure prediction.
- **Patient risk** prediction and early warnings.
- **Personalised** treatment recommendations.

Finance

- **Fraud detection** in card & UPI transactions.
- **Credit scoring** for loan approval.
- **Algorithmic trading** of stocks.
- **Customer support** chatbots.
- **Risk** assessment and forecasting.

Both fields have *lots of data* and *no simple formula* — ideal for ML.

Applications — Transport, Retail & Daily Life

Transport & industry

- **Self-driving** cars (vision + RL).
- **Traffic prediction** and smart routing.
- **Predictive maintenance** of machines.
- **Quality inspection** on factory lines.

E-commerce & retail

- **Recommendations** ("you may also like").
- **Demand forecasting** & smart pricing.
- **Customer segmentation** (clustering).

In your daily life

- **Voice assistants** & speech recognition.
- **Face unlock** & photo tagging.
- **Spam filtering** in email.
- **Translation** between languages.
- **Streaming** recommendations (Netflix, Spotify, YouTube).
- **Maps** & live navigation.

Applications — Matching the Paradigm to the Task

Application	**Paradigm**	**Task type**
Spam / not-spam email	Supervised	Classification
House-price prediction	Supervised	Regression
Customer segmentation	Unsupervised	Clustering
Market-basket ("bought together")	Unsupervised	Association
Game-playing AI (AlphaGo)	Reinforcement	Policy learning
Self-driving control	Reinforcement	Policy learning

Skill to build: given a problem, name the right *paradigm* and *task type*.

Limitations & Responsible ML

Limitations of ML

- **Data hungry**: needs lots of good data.
- **Garbage in, garbage out**: bad data → bad model.
- **Bias**: unfair data → unfair predictions.
- **Black box**: some models are hard to explain.
- No true **understanding** or common sense.

Building ML responsibly

- Use **fair**, representative data.
- Protect **privacy**.
- Prefer **explainable** models where decisions matter.
- Keep **humans** in charge of important choices.
- Test carefully before deploying.

Remember

A model is only as good — and as fair — as the *data* it learns from.

A Glimpse Ahead — Deep Learning

Where ML is heading

Deep Learning uses **neural networks** with many layers to learn very complex patterns directly from raw data.

- Powers modern **vision**, **speech** and **language** AI.
- Learns its own features — less manual work.
- Needs **lots** of data and computing power.
- Behind ChatGPT, image generators, translators.

Your next steps

In later semesters you will study neural networks and deep learning in depth — this unit is your foundation.

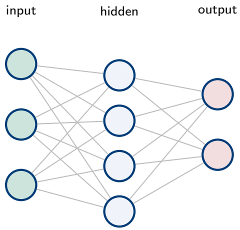


Figure: A simple neural network.

Quick Check — Round 3 (Applications)

[Q1] Which paradigm and task fits "group similar customers"?

[Q2] Give one ML application in healthcare and one in finance.

[Q3] Why can an ML model be "unfair"?

[Q4] What is deep learning based on?

Think for 60 seconds — answers on the next slide.

Answers — Round 3

[A1]

Unsupervised learning, **clustering** task (e.g. K-Means).

[A2]

Healthcare: **disease detection from scans**; Finance: **fraud detection** (any valid example).

[A3]

Because it learns from **biased data** — if the training data is unfair, the model's predictions will be unfair too.

[A4]

Neural networks with many layers, learning complex patterns from large amounts of data.

Summary of Unit 5

What we covered

1. **What ML is** — machines that learn patterns from data; the engine of modern AI.
2. **Three paradigms** — supervised (labelled), unsupervised (unlabelled), reinforcement (reward-based).
3. **Classification vs regression** and **classification vs clustering**.
4. **Supervised algorithms** — KNN (neighbour vote), decision trees (yes/no questions), plus regression, SVM, Naïve Bayes, random forest.
5. **K-Means** clustering and the ML workflow (collect, prepare, train, test, predict), with over/underfitting.
6. **Applications** — healthcare, finance, transport, retail, daily life — and responsible ML.

Machine Learning turns data into decisions — the core skill of an AI & ML engineer.

Key Terms to Remember

Paradigms & tasks

- **Supervised / Unsupervised / Reinforcement**
- **Classification** = predict a class
- **Regression** = predict a number
- **Clustering** = find groups
- **Feature / Label / Model**

Algorithms & pitfalls

- **KNN** = neighbour majority vote
- **Decision tree** = yes/no questions
- **K-Means** = cluster by centres
- **Train / test split**
- **Overfitting** vs **underfitting**

Practice Questions (Exam Oriented)

Short answer (2–5 marks)

1. Define Machine Learning and state its role within AI.
2. Differentiate supervised, unsupervised and reinforcement learning.
3. Differentiate classification and regression with one example each.
4. Explain how the KNN algorithm classifies a new point.
5. What is overfitting? How can we detect it?

Long answer (8–10 marks)

6. Explain the three learning paradigms with a diagram and examples.
7. Compare classification and clustering on at least five points.
8. Explain the working of a decision tree with a small example.
9. Describe the steps of the K-Means clustering algorithm.
10. Discuss five real-world applications of ML, naming the paradigm and task type for each.

Think & Explore (Take-Home)

Mini tasks

- For 5 real problems, label each as supervised / unsupervised / reinforcement, and classification / regression / clustering.
- Do a KNN dry run: classify (160, 6) using the fruit table from class.
- Draw a decision tree to decide "Should I carry an umbrella?"
- List 10 apps you use and the ML feature in each.

Reading & reflection

- Explore a free, visual ML demo (e.g. "Teachable Machine").
- Read about one ML success story in a field you like.
- Start learning *Python* basics — your main ML tool.

Reminder

Post any doubts on the course page. You have now completed the foundation of Computing and AI — well done!

References

Textbooks (as per the course page)

1. S. Russell & P. Norvig, *Artificial Intelligence: A Modern Approach*, Pearson. **[Main text]**
2. Tom M. Mitchell, *Machine Learning*, McGraw Hill.
3. Ethem Alpaydin, *Introduction to Machine Learning*, MIT Press / PHI.
4. Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn & TensorFlow*, O'Reilly.

Further reading

5. C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer.
6. A. Samuel, "Some Studies in Machine Learning Using the Game of Checkers," IBM Journal, 1959.

Course page: https://www.gcjana.in/courses/shardauniversity/2601/fundamentals-of-computing-and-artificial-intelligence/

Thank You

Any Questions?

"A breakthrough in machine learning would be worth ten Microsofts." — Bill Gates

End of Unit 5 — and of the Fundamentals of Computing and AI course.

Post your doubts on the course page →

`https://www.gcjana.in/courses/shardauniversity/2601/fundamentals-of-computing-and-artificial-intelligence/#Post-doubts`