Core Concepts of Artificial Intelligence, Machine Learning, and Deep Learning

October, 2025

Byung-Gul(BG) Lee

Learning Objectives

- Explain AI, machine learning and deep learning in plain language
- Distinguish the three main types of machine learning
- Understand the basic workflow:

  data → model → evaluation → deployment
- Ask the right questions when reviewing AI projects

Agenda

- Why AI matters for government
- AI, ML, DL: definitions and relationships
- How machine learning works
- Deep learning and neural networks
- Key takeaways and Q&A

Why AI Matters for Government

•Data-Driven Scientific Policymaking and Decision-Making

- Enables evidence-based policy formulation by analyzing and predicting massive administrative records, environmental and meteorological information, and satellite datasets in real time, moving beyond intuition and fragmented statistics.
- Facilitates scenario-based advance simulations for complex urban challenges (traffic congestion, energy consumption) and environmental issues (sea level rise, climate change impacts).

•Strengthening Disaster Response and Predictive Public Safety Capabilities

- Secures the critical golden hour by detecting and monitoring national hazards early, including floods, cyclone, sea-level rise, wildfires, and air pollution.
- Transitions governance from reactive post-disaster recovery to proactive, predictive public safety nets.

•Maximizing Administrative Efficiency and Elevating Public Service Quality

- Automates repetitive civil petitions, document reviews, and data compilation, reducing administrative burdens on public officials and cutting operational costs.
- Substantially enhances public accessibility and satisfaction through 24/7 personalized administrative consultations and proactive identification of welfare blind spots.

•Establishing National AI Governance and Ethical Standards

- Governments are not merely passive consumers of AI technology; they must serve as stewards establishing guidelines and regulatory frameworks for **responsible AI deployment (AI Ethics & Governance)**, ensuring fairness, data privacy, algorithmic transparency, and cybersecurity.

Why Public Officials Need to Understand the Core Concepts of AI, ML, and DL

Public officials do not need to learn AI, ML, and DL to become coders, but to serve as **informed policymakers, competent project commissioners, and responsible public administrators**.

1. Filtering Out Hype and Procuring Fit-for-Purpose Solutions

AI, ML, and DL differ significantly in complexity, cost, required data volume, and computing resources. Recognizing these boundaries prevents costly procurement failures—such as overspending on deep learning when basic statistical tools suffice, or procuring overly simplistic models that cannot handle complex sensor data.

2. Enhancing Procurement Oversight and Technical Communication

Because most public AI initiatives are outsourced, officials must understand core training principles, data bias, and performance metrics (e.g., accuracy, precision, recall).

This baseline knowledge enables officials to critically evaluate vendor proposals, track project milestones, and conduct rigorous quality audits.

3. Recognizing the Critical Role of Data Governance and Quality

Model performance hinges directly on data quality and integrity ("Garbage In, Garbage Out").

Essential data hygiene—standardization, missing-value imputation, and privacy de-identification—serves as the indispensable foundation for effective AI deployment.

4. Ensuring Explainability (XAI), Algorithmic Fairness, and Accountability

Despite superior pattern recognition, deep learning models often operate as opaque "black boxes".

Because administrative actions directly affect citizens' rights and public benefits, decisions cannot rely solely on algorithmic outputs. Officials must understand these limitations to enforce Explainable AI (XAI), mitigate bias, and uphold constitutional accountability.

A Brief History of AI

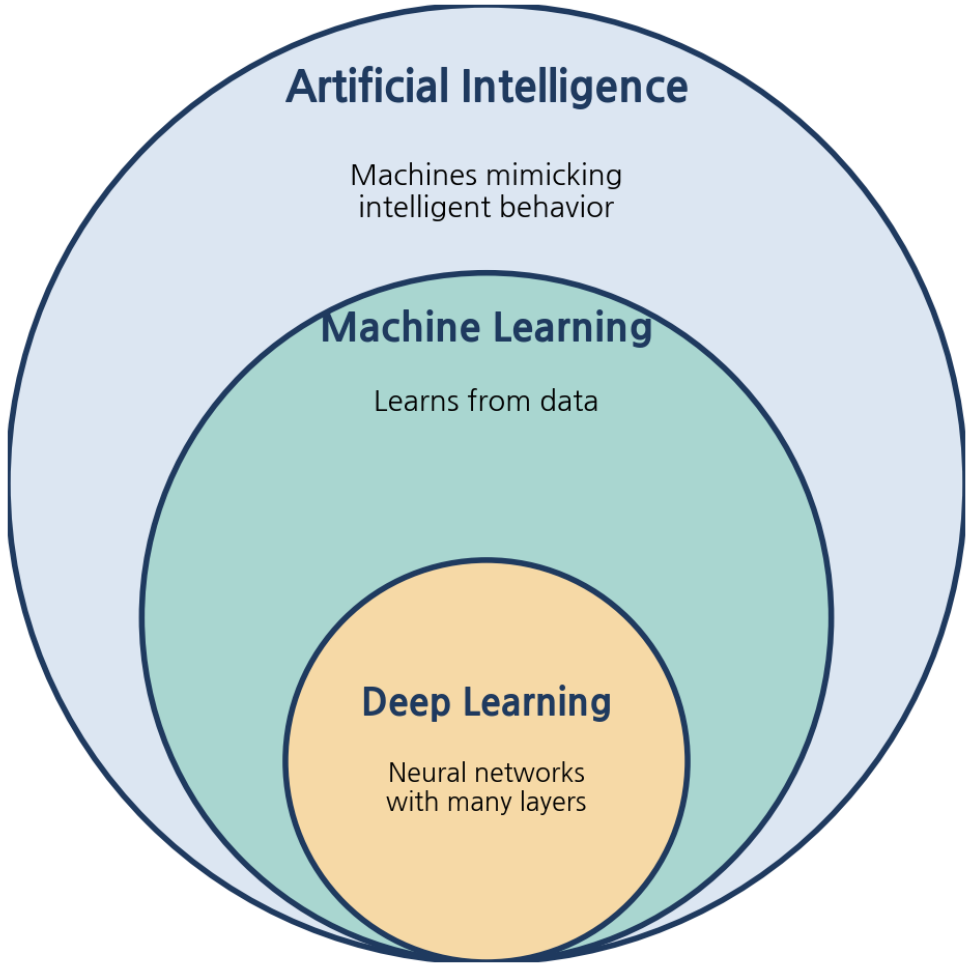


Key Milestones (Chronological Order)

- **1950 — Turing Test**: Establishment of the foundational benchmark for assessing machine intelligence.
- **1956 — Dartmouth Conference**: The birth of artificial intelligence as an official academic discipline.
- **1997 — Deep Blue**: The first computer chess system to defeat a reigning world champion in tournament conditions.
- **2012 — AlexNet**: Demonstration of Convolutional Neural Networks (CNN) outperforming traditional computer vision algorithms.
- **2016 — AlphaGo**: Breakthrough achievement in complex heuristic evaluation and reinforcement learning applied to Go.
- **2017 — Transformer**: Introduction of the self-attention mechanism, replacing recurrent network limitations in sequence modeling.
- **2022 — Generative AI**: Transition into commercialized large language models and public multimodal AI systems.

AI vs. Machine Learning vs. Deep Learning

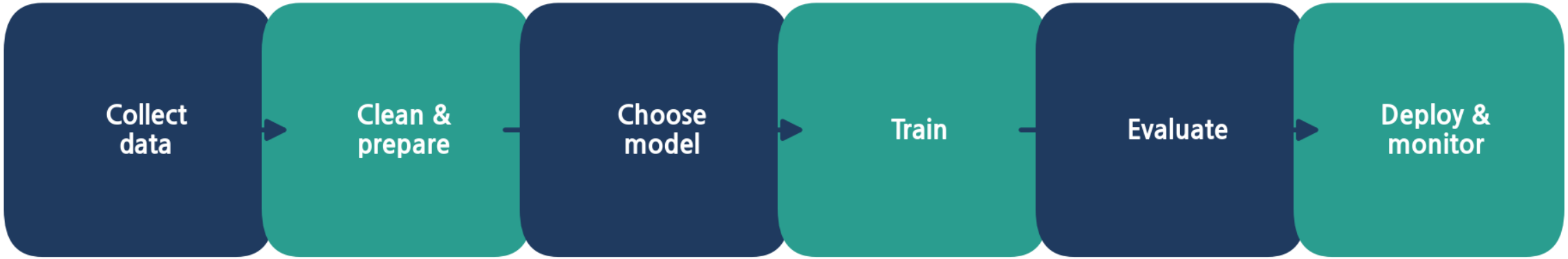
- Artificial Intelligence (AI): machines performing tasks that normally require human intelligence
- Machine Learning (ML): a subset of AI; systems learn patterns from data instead of following hand-written rules
- Deep Learning (DL): a subset of ML using multi-layer neural networks
- Each is contained within the previous one



Traditional Programming vs. Machine Learning

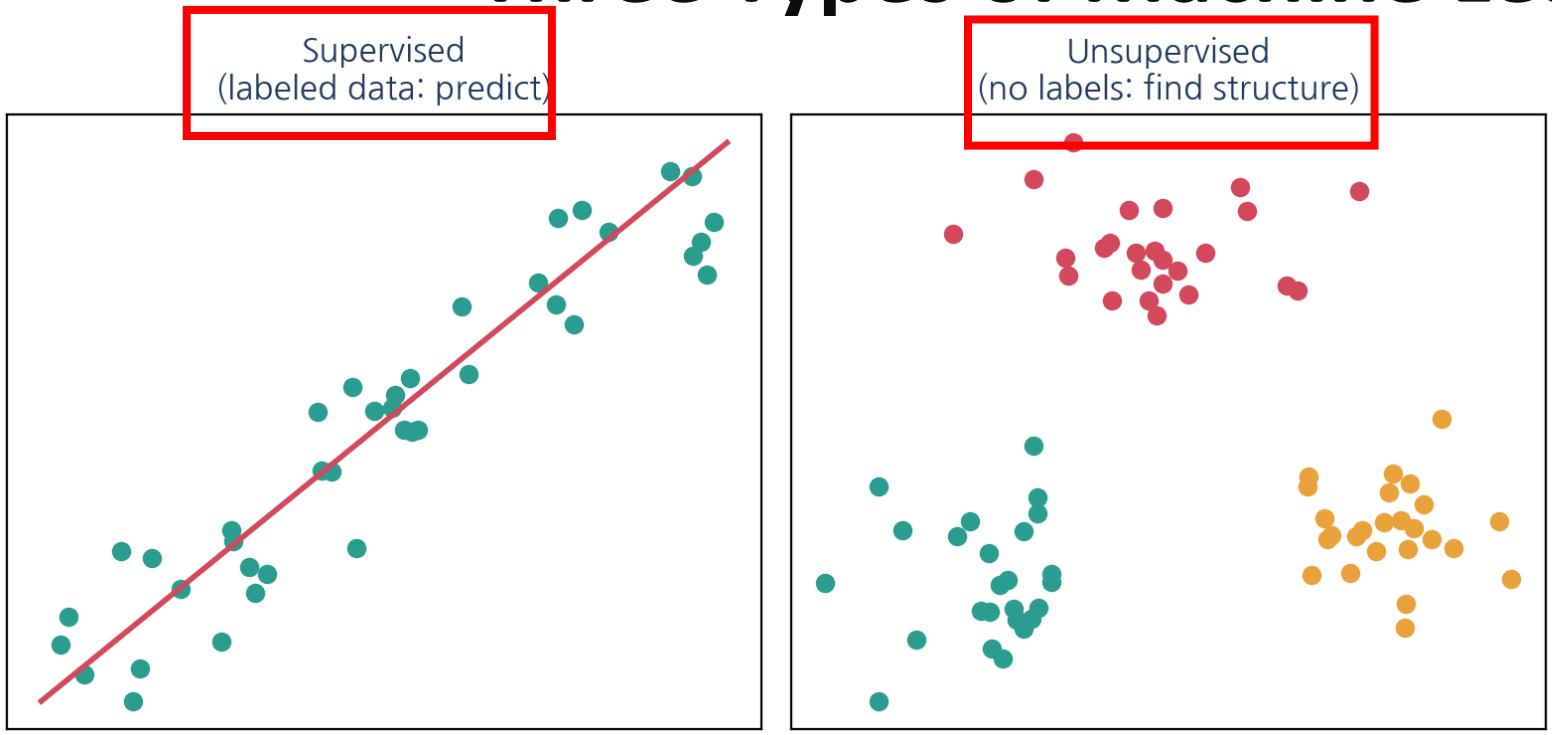
- Traditional programming: Data + Rules → Answers
  - Example: 'If the email contains "free money", mark as spam'
- Machine learning: Data + Answers → Rules (a model)
  - Example: show 100,000 emails labeled spam/not spam; the algorithm discovers the patterns
- Advantage: handles problems where rules are too complex to write by hand (images, speech, language)

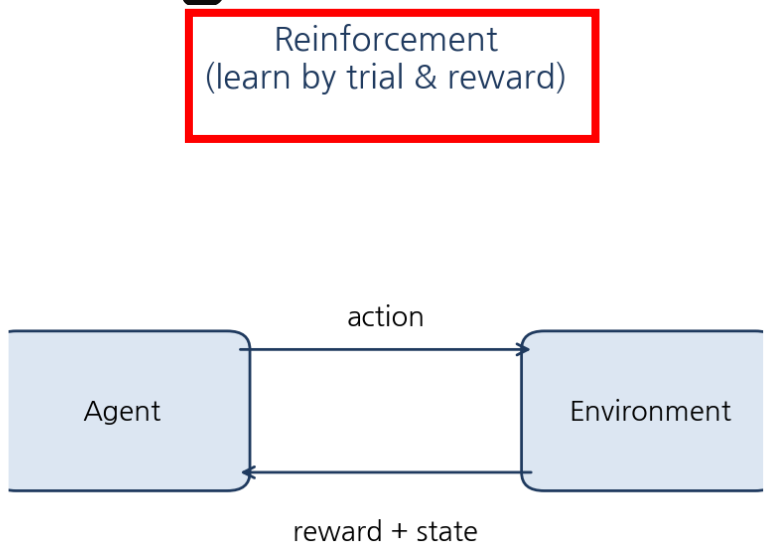
The Machine Learning Workflow



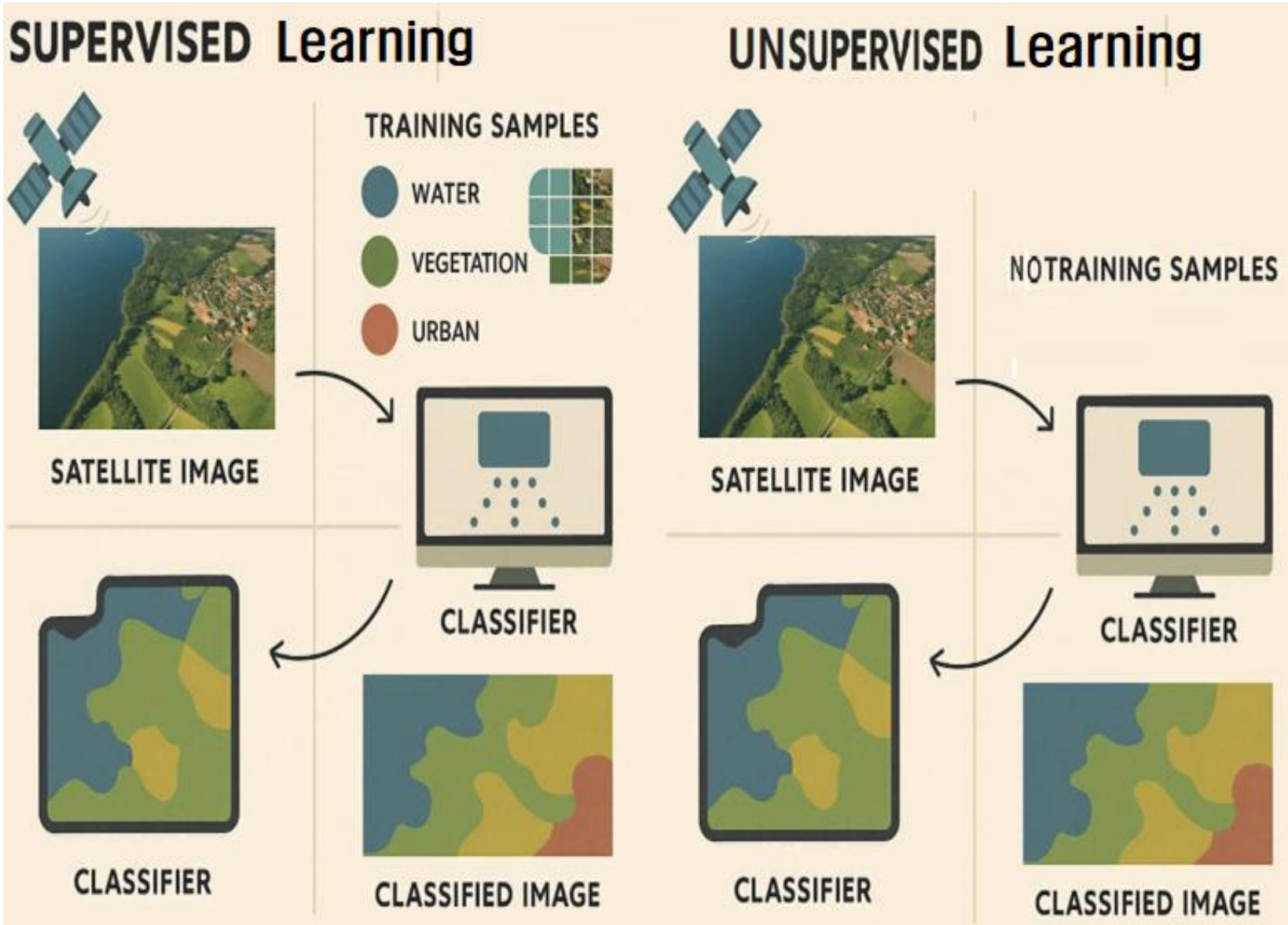
- Data quality determines model quality: 'garbage in, garbage out'
- Most project effort goes to data collection and cleaning
- Models must be monitored after deployment because the world changes (data drift)

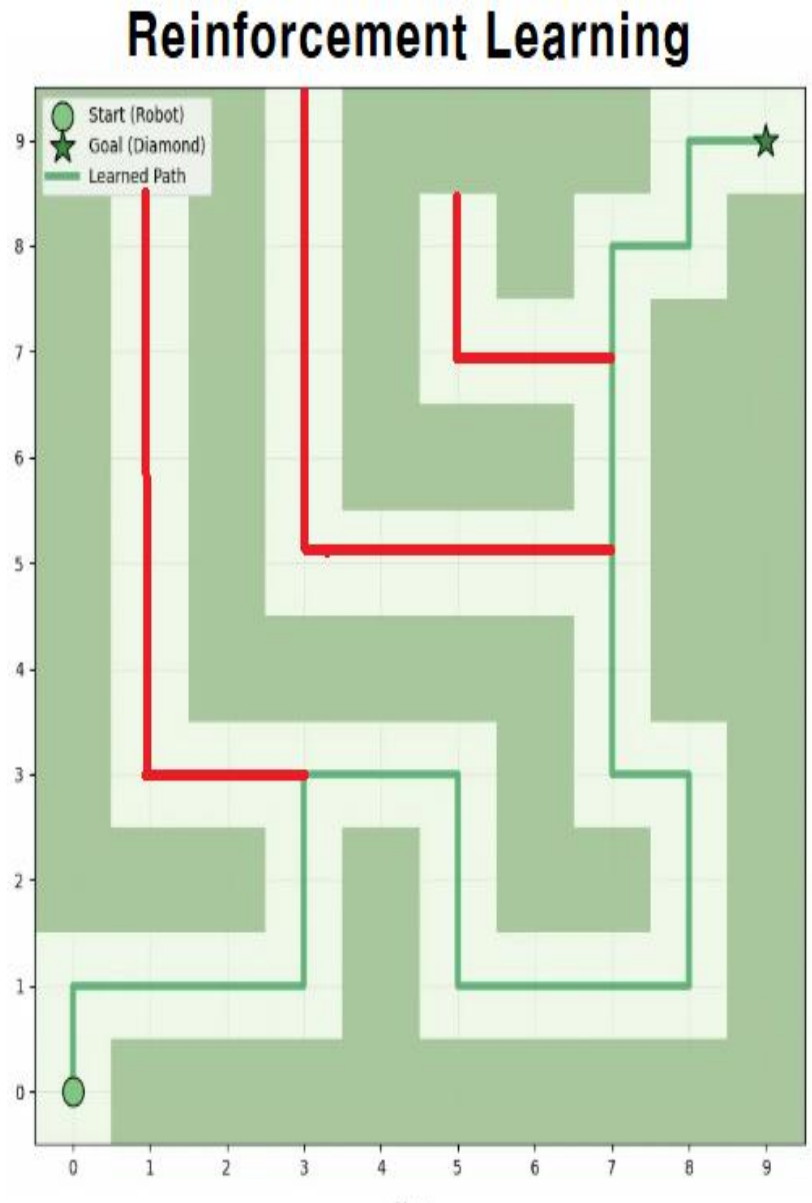
Three Types of Machine Learning





- Supervised: learns from labeled examples (e.g., predict house prices, classify documents)
- Unsupervised: finds hidden groups without labels (e.g., citizen segmentation, anomaly detection)
- Reinforcement: learns by trial and reward (e.g., robot control, game-playing AI)


SUPERVISED Learning
TRAINING SAMPLES
WATER
VEGETATION
URBAN
SATELLITE IMAGE
CLASSIFIER
CLASSIFIED IMAGE
CLASSIFIER
UNSUPERVISED Learning
NOTRAINING SAMPLES
SATELLITE IMAGE
CLASSIFIER
CLASSIFIED IMAGE
CLASSIFIER


Reinforcement Learning
Start (Robot)
Goal (Diamond)
Learned Path
0
1
2
3
4
5
6
7
8
9

Examples of Supervised and Unsupervised Learning

Supervised Learning

Training data

House size (m^2)	Actual price ($1,000s)
50	100
75	150
100	200
125	250
150	300

Question 1: What price would the model predict for a house of 110 m^2, assuming a linear relationship?

Unsupervised Learning — Group Citizens by Characteristics

Scenario: A local government wants to understand different groups of citizens to improve public services. It collects age and annual income data but does not provide predefined group labels.

Citizen data

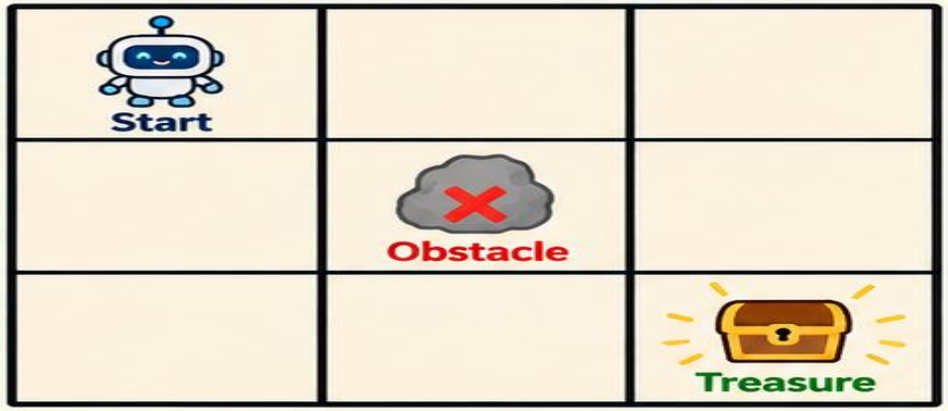
Citizen	Age	Annual income ($1,000s)
A	25	30
B	28	35
C	27	32
D	55	80
E	58	85
F	60	82

Question 1: Which approach is most appropriate for identifying groups of similar citizens?

Examples of Reinforcement

Reinforcement Learning Example: Find the Treasure in the Maze!

The robot learns by trying actions and receiving rewards.



Maze Map

- = Robot (Agent)
- = Treasure (Goal)
- = Obstacle
- = Empty cell

Available Actions

- = Move Up
- = Move Down
- = Move Left
- = Move Right

Reward Rules

+10	Reach the treasure
-5	Hit the obstacle or go out of bounds
-1	Move to an empty cell

1 Which action is best now?

The robot is at the Start position.
What is the best action to take?



Action	Result	Reward
Up	Cannot move	-5
Down	Empty cell	-1
Left	Cannot move	-5
Right	Empty cell	-1

Q: What is the best action?

① Up ② Down ③ Left ④ Right

2 What happens next?

Suppose the robot moves right.
Now it is at the new position.



If the robot moves down from here, it will hit the obstacle.

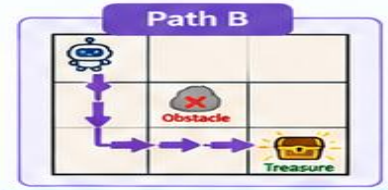
Q: How many points will the robot get if it moves down?

3 Which path is better?

There are two possible paths to reach the treasure.



Right → Right → Down → Down
Rewards: -1 → -1 → -1 → +10
Total Reward = +7



Down → Down → Right → Right
Rewards: -1 → -1 → -1 → +10
Total Reward = +7

Q: Which path is better?

4 Why is this "learning"?

At first, the robot does not know which action is good.
It tries actions, gets rewards, and learns from experience.

State (current position) → Action (up / down / left / right) → Environment (the maze) → Reward (+10 / +2 / -3 / -5) → Learning (update knowledge)

Key Concepts

- **Agent:** The robot
- **Environment:** The maze
- **State:** The robot's current position
- **Action:** Up / Down / Left / Right
- **Reward:** +10, +2, -3, -5
- **Goal:** Find the treasure and get the highest score

In short...
Reinforcement learning is a way for AI to learn the best actions by trying and receiving rewards.

Artificial Intelligence (AI)

A broad field of technology that enables machines to perform tasks that normally require human intelligence.

Machine Learning (ML)

A subset of AI that learns patterns from data without being explicitly programmed.

Supervised Learning
(Learning with Labeled Data)

Learns from data with correct answers (labels).

- **Classification**
  e.g., cat vs. dog, spam vs. non-spam, pipe vs. no pipe
- **Regression**
  e.g., house price, temperature, depth

Unsupervised Learning
(Learning without Labels)

Finds hidden patterns or groups in data without correct answers.

- **Clustering**
  e.g., K-means, hierarchical clustering, DBSCAN
- **Dimensionality Reduction (Feature Learning)**
  e.g., PCA, t-SNE, autoencoder

Reinforcement Learning
(Learning from Rewards)

Learns by interacting with an environment and receiving rewards or penalties.

e.g., robot control, game playing, resource allocation, decision making

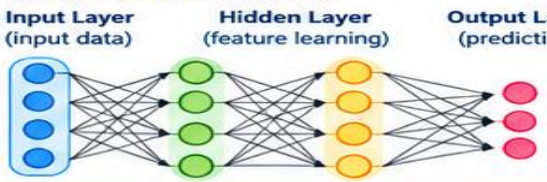
Models

Various machine learning models can be used in each learning paradigm.

- **Decision Tree** (interpretable)
- **Random Forest** (ensemble)
- **SVM** (kernel-based)
- **Regression** (linear / nonlinear)
- **Neural Network** (deep learning model)
  - **Deep Learning** (many layers)
    e.g., CNN, RNN, LSTM, Transformer, GNN

Neural Network (Deep Learning Model)

A model inspired by the human brain, consisting of connected layers of artificial neurons.

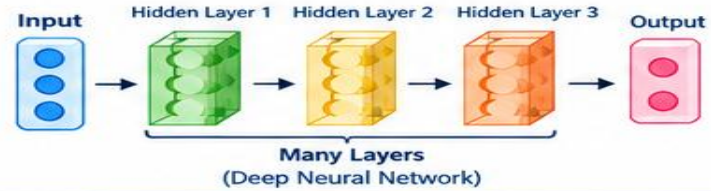


Key Features

- Learns features from data
- Adjusts weights (learning)
- Can be used for various tasks

Deep Learning (Deep Neural Network)

A neural network with many layers, capable of learning complex patterns.



Common Architectures

- CNN (image data)
- RNN / LSTM (sequence data)
- Transformer (language data)
- GNN (graph data)

Examples of Learning Combinations

1. Supervised Learning + Neural Network

Learn from labeled data to predict the target.



e.g., pipe detection (Classification / Regression)

2. Unsupervised Learning + Neural Network

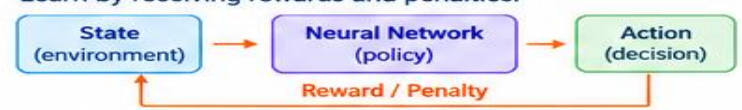
Find patterns or groups in unlabeled data.



e.g., GPR data clustering, anomaly detection

3. Reinforcement Learning + Neural Network

Learn by receiving rewards and penalties.



e.g., robot control, game playing, resource management

Key Takeaways

- ✓ Supervised / Unsupervised / Reinforcement Learning are different learning methods.
- ✓ Neural Network is a model that can be used in all three learning methods.
- ✓ Classification is a task (problem type), not a learning method.

In short,
Neural Network = a tool
used in various learning methods
to make machines learn from data.

Neural Network Algorithm

A neural network is an algorithm that mimics the way nerve cells (neurons) in the human brain exchange signals and learn. To put it simply, it works like a **"panel of judges" that holds step-by-step deliberations to reach a conclusion, improving its accuracy over time by reflecting on its mistakes.**

1. The Three-Tier Evaluation Structure

A neural network is typically divided into three types of layers. For instance, imagine a task designed to **look at a picture and determine whether it is an apple or a pear**:

1.Input Layer (Receiving the Data):

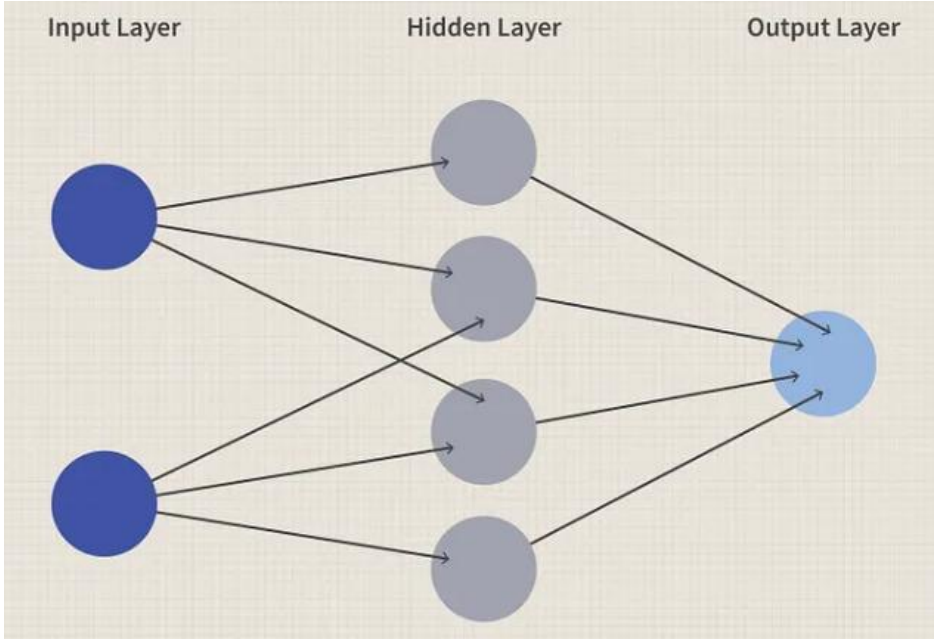
1. It takes raw data—such as colors, roundness, and stem shape—and converts them into numerical values that a computer understands.

2.Hidden Layers (Intermediate Judges):

1. The first group of judges checks simple lines and edges.
2. The next group combines colors and textures.
3. Subsequent judges assess the overall silhouette of the fruit.
4. As the layers run deeper, they synthesize increasingly complex and critical features.

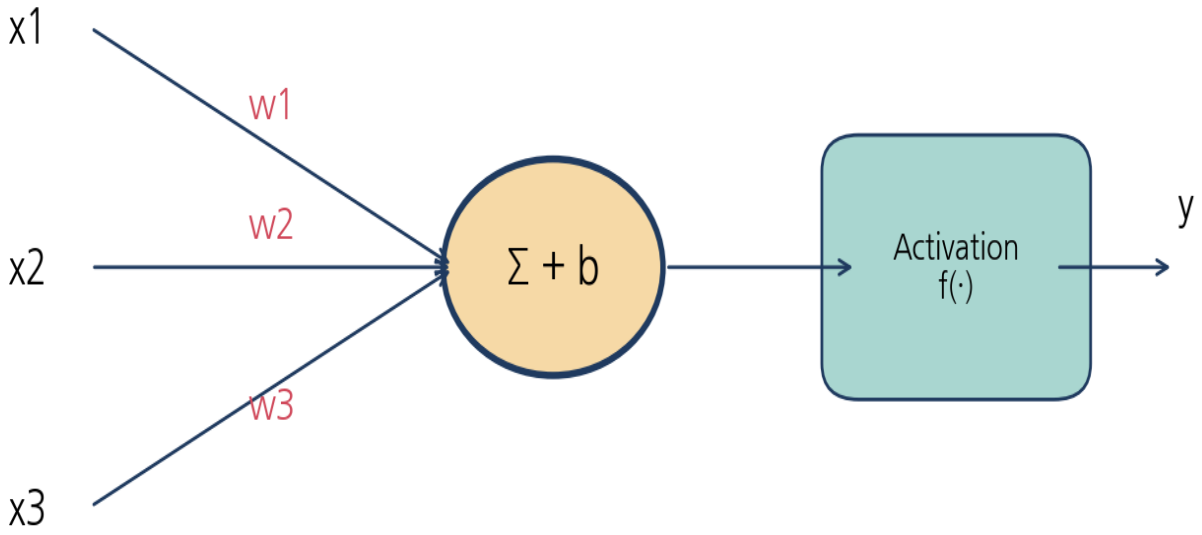
3.Output Layer (Final Verdict):

1. The head judge calculates the overall score and delivers the final conclusion: "90% probability it is an apple, 10% probability it is a pear."

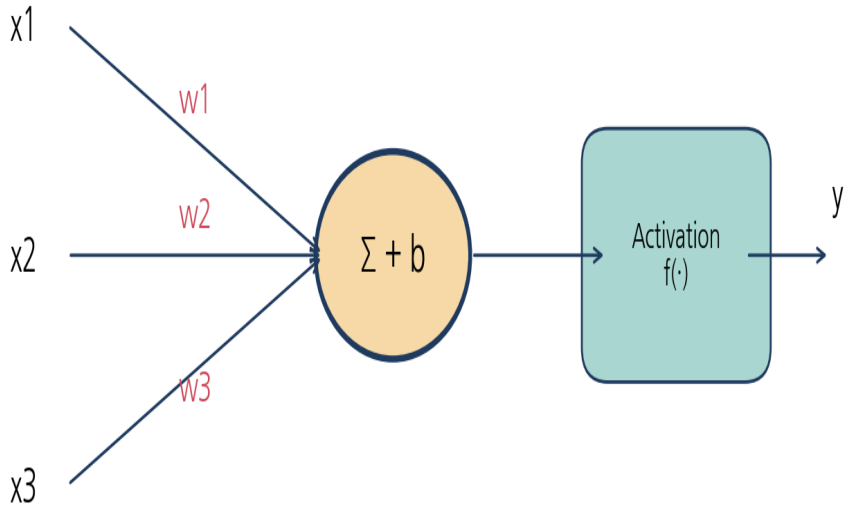


Artificial Neural Network(A Single Artificial Neuron)

- Inputs are multiplied by weights (w) and summed with a bias (b)
- An activation function (e.g., ReLU, sigmoid) decides the output
- Training = adjusting millions or billions of weights to reduce error
- Method: gradient descent with backpropagation



A Single Artificial Neuron



$$Z = (x_1 \times w_1) + (x_2 \times w_2) + (x_3 \times w_3) + b$$

[Problem Description]

Using a single artificial neuron, we aim to predict the final apartment price (y) by combining apartment features (inputs x), weights (w), and a bias (b), and then passing the result through an activation function. PNG

[Given Conditions]

- **Input Values (x):**
  - x_1 (Area): **30** (m^2)
  - x_2 (Distance to subway station): **5** (minutes)
  - x_3 (Building age): **2** (years)
- **Weights (w):**
  - $w_1 = 0.3$ (Area weight)
  - $w_2 = -0.5$ (Station distance weight - negative effect since closer is better)
  - $w_3 = -1.0$ (Building age weight - negative effect for newer buildings)
- **Bias (b):**
  - $b = 5.0$ (Base price constant)
- **Activation Function $f(\cdot)$:**
  - Assumed to use an **Identity Function ($f(z) = z$)** which outputs the neuron's value as is.

A Single Artificial Neuron

[Problem Description]

Using a single artificial neuron, we aim to predict the final apartment price (y) by combining apartment features (inputs x), weights (w), and a bias (b), and then passing the result through an activation function.

assume 30m^2= 1 hundred million$

[Given Conditions]

- **Input Values (x):**
  - x_1 (Area): **30** (m^2)
  - x_2 (Distance to subway station): **5** (minutes)
  - x_3 (Building age): **2** (years)
- **Weights (w):**
  - $w_1 = 0.3$ (Area weight)
  - $w_2 = -0.5$ (Station distance weight - negative effect since closer is better)
  - $w_3 = -1.0$ (Building age weight - negative effect for newer buildings)
- **Bias (b):**
  - $b = 5.0$ (Base price constant)
- **Activation Function $f(\cdot)$:**
  - Assumed to use an **Identity Function ($f(z) = z$)** which outputs the neuron's value as is.

[Solution Process]

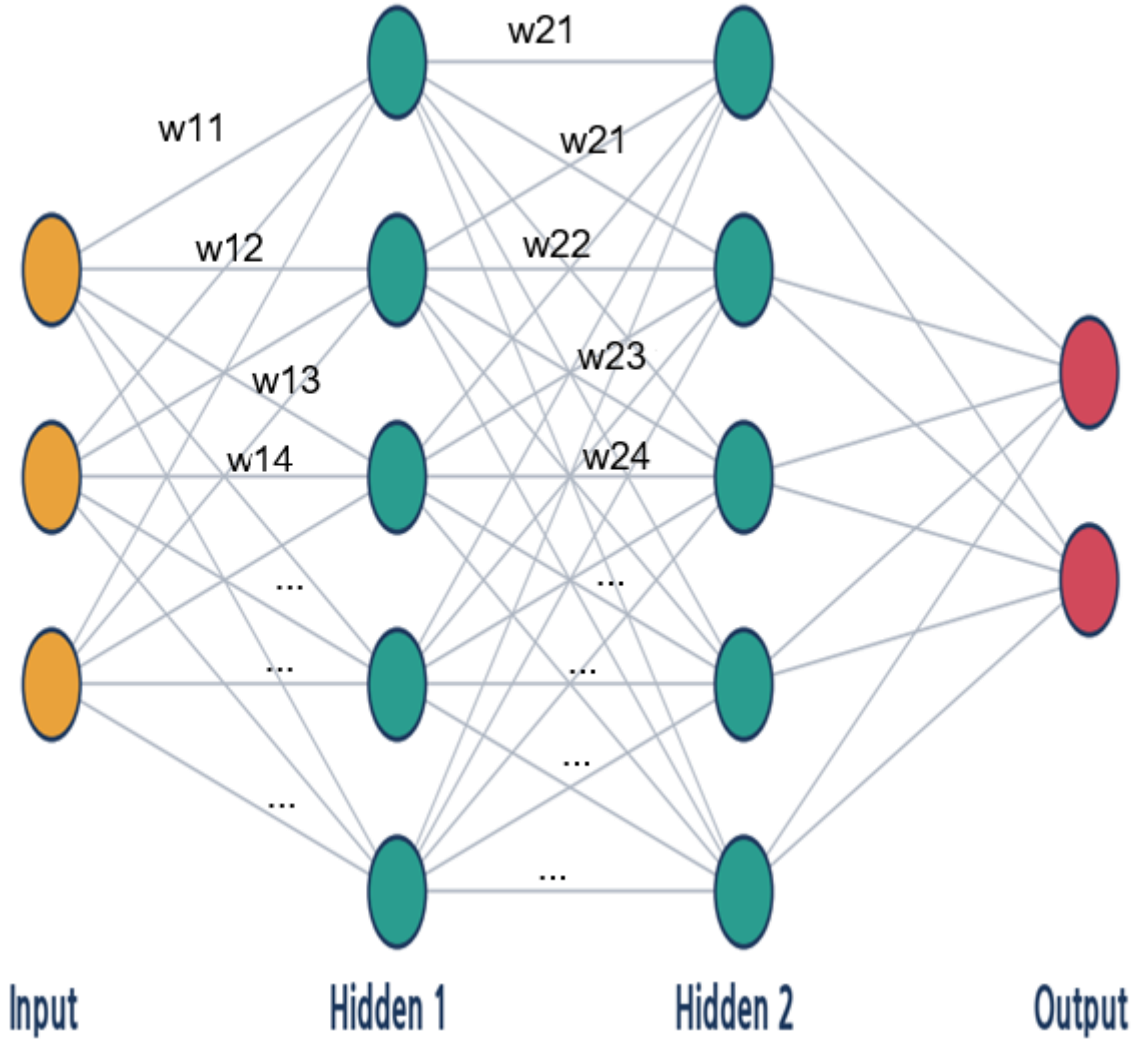
1. **Calculate the Weighted Sum ($\Sigma + b$)**
   - Inside the neuron, each input value is multiplied by its weight, summed together, and the bias is added. PNG
   - $Z = (x_1 \times w_1) + (x_2 \times w_2) + (x_3 \times w_3) + b$
   - $Z = (30 \times 0.3) + (5 \times -0.5) + (2 \times -1.0) + 5.0$
   - $Z = 9.0 - 2.5 - 2.0 + 5.0 = 9.5$
2. **Pass Through the Activation Function (y)**
   - The calculated value Z passes through the activation function. PNG
   - $y = f(9.5) = 9.5$

[Answer]

- **Predicted Apartment Price (y): 9.5 (hundred million $)**

Deep Learning: Artificial Neural Networks

- Inspired (loosely) by the brain: layers of connected 'neurons'
- Input layer receives data; hidden layers extract increasingly abstract features; output layer gives the result
- 'Deep' = many hidden layers
- Learns features automatically—no manual feature design

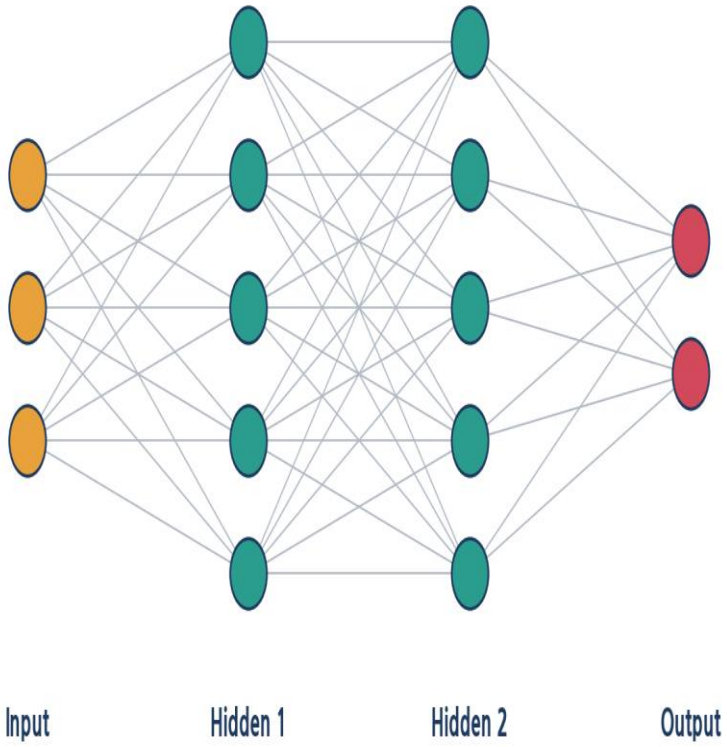


Deep Learning: Artificial Neural Networks

1. Apartment Price Evaluation Problem

[Given Conditions]

1. **Input Data (Features)**
   - X_1 (Standardized area value): **0.6** (30 pyeong)
   - X_2 (Standardized distance to subway station): **0.8** (5 minutes on foot / higher score for closer distance)
   - X_3 (Standardized building age): **0.9** (2 years old / higher score for newer buildings)
2. **Trained Hidden Layer Calculation Rules (Weights & Biases)**
   - **Hidden Layer 1 (1st Feature Extraction):**
     - $H_{1,1}$ (Location score) = $(X_1 \times 0.5) + (X_2 \times 0.5)$
     - $H_{1,2}$ (Building condition score) = $(X_1 \times 0.2) + (X_3 \times 0.8)$
   - **Hidden Layer 2 (Comprehensive Premium Score):**
     - $H_{2,1} = (H_{1,1} \times 0.6) + (H_{1,2} \times 0.4)$
3. **Final Output Layer Calculation Rule**
   - **Apartment Price Formula:** $(H_{2,1} \times \text{Maximum Market Value 1.0 billion KRW}) + \text{Base Minimum Price 0.2 billion KRW}$



Deep Learning: Artificial Neural Networks

1. Apartment Price Evaluation Problem

[Given Conditions]

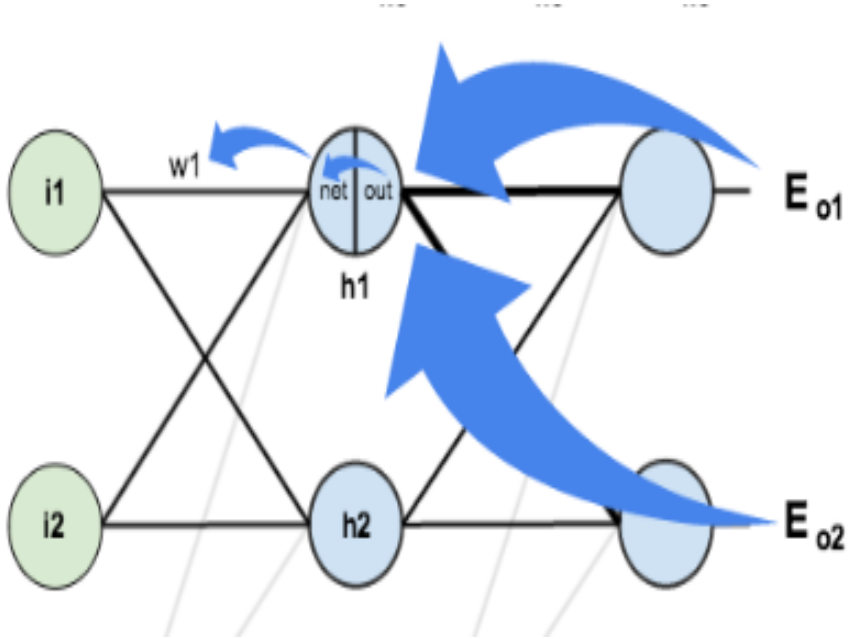
1. Input Data (Features)
   - X_1 (Standardized area value): **0.6** (30 pyeong)
   - X_2 (Standardized distance to subway station): **0.8** (5 minutes on foot / higher score for closer distance)
   - X_3 (Standardized building age): **0.9** (2 years old / higher score for newer buildings)
2. Trained Hidden Layer Calculation Rules (Weights & Biases)
   - Hidden Layer 1 (1st Feature Extraction):
     - $H_{1,1}$ (Location score) = $(X_1 \times 0.5) + (X_2 \times 0.5)$
     - $H_{1,2}$ (Building condition score) = $(X_1 \times 0.2) + (X_3 \times 0.8)$
   - Hidden Layer 2 (Comprehensive Premium Score):
     - $H_{2,1} = (H_{1,1} \times 0.6) + (H_{1,2} \times 0.4)$
3. Final Output Layer Calculation Rule
   - Apartment Price Formula: $(H_{2,1} \times$ Maximum Market Value 1.0 billion KRW$) +$ Base Minimum Price 0.2 billion KRW

2. Evaluation and Solution

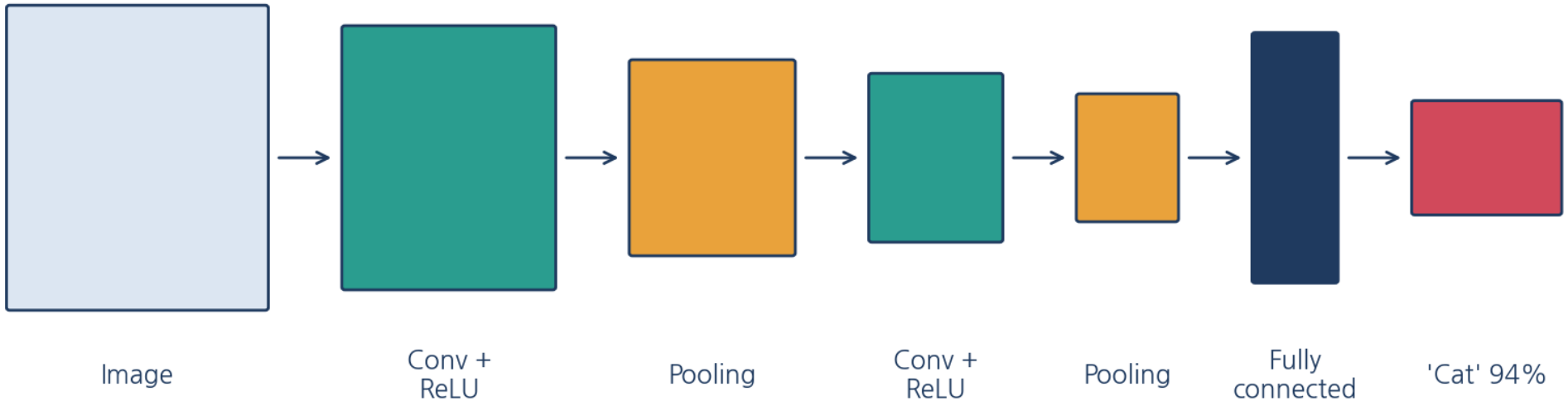
1. Hidden Layer 1 Calculation:
   - $H_{1,1} = (0.6 \times 0.5) + (0.8 \times 0.5) = 0.3 + 0.4 = 0.7$
   - $H_{1,2} = (0.6 \times 0.2) + (0.9 \times 0.8) = 0.12 + 0.72 = 0.84$
2. Hidden Layer 2 Calculation:
   - $H_{2,1} = (0.7 \times 0.6) + (0.84 \times 0.4) = 0.42 + 0.336 = 0.756$
3. Final Output Layer (Price) Calculation:
   - Price = $(0.756 \times 1.0$ billion $\$) + 0.2$ billion $\$ =$ 0.756 billion $\$ + 0.2$ billion $\$ = 956$ million $\$$

How a Network Learns

1. Forward pass: input flows through the network to make a prediction
2. Loss: measure how far the prediction is from the correct answer
3. Backpropagation: compute how each weight contributed to the error
4. Update: adjust the weights slightly (gradient descent)
5. Repeat over many examples and many epochs



Common Deep Learning Architectures



- CNN (Convolutional Neural Network): images, video, satellite and medical imaging
- RNN / LSTM: sequences such as time series and speech
- Transformer: language and increasingly images; basis of large language models and generative AI

Machine Learning vs. Deep Learning

	Classical ML	Deep Learning
Data needed	Small to medium	Very large
Features	Designed by humans	Learned automatically
Compute	Modest (CPU)	Heavy (GPU)
Interpretability	Often easier to explain	Often a 'black box'
Best for	Tabular/structured data	Images, speech, language

Questions to Ask Before Adopting AI

- What problem are we solving, and is AI truly needed?
- Is there enough high-quality, legally usable data?
- How will accuracy be measured, and who tested it?
- Can decisions be explained and appealed?
- Who is accountable if the system errors? Is there human review?
- What are the costs of maintenance and monitoring?

Why a question is important to AI

Q1. Calculate SST(the Sea Surface Temperature) data to understand yearly trend of the temperature
Q2. Calculate SST(the Sea Surface Temperature) data to understand yearly trend of the temperature using the machine learning algorithm.

- **Interpretation of Q1 (Statistical & Traditional Time Series Analysis)**
- **Approach**: Assumes conventional statistical methods and standard data processing pipelines.
- **Key Tasks**: Calculates descriptive statistics and trendlines (typically using Pandas, SciPy, or xarray), such as annual means, linear regression, moving averages, seasonal decomposition (STL), and the Mann-Kendall trend test.
- **Output**: Descriptive metrics and trend extraction based on existing observational or gridded data.

- **Interpretation of Q2 (Machine Learning-Based Modeling / Trend Prediction)**
- **Approach**: Includes an explicit constraint to apply machine learning algorithms.
- **Key Tasks**: Establishes a full ML pipeline, including feature engineering (lagged variables, temporal features), train/test data splitting, model selection (such as Random Forest, XGBoost, LightGBM, or SVR), hyperparameter tuning, and model evaluation (RMSE). It is treated as a request for non-linear pattern modeling or forecasting rather than a simple annual aggregation.
- **Output**: A model training and inference pipeline learning from feature representations.

Key Questions to Ask When Reviewing AI Projects

A checklist to evaluate the feasibility and risks of an AI initiative:

1. **Problem Definition:** "Does this problem truly require AI? Can it be solved using simple rule-based automation or standard statistical analysis instead?"
2. **Data Readiness:** "Is there sufficient high-quality data available for training? Are there concerns regarding data bias, copyright, or data privacy infringement?"
3. **Performance Baseline:** "Is there a clearly defined performance target (e.g., precision of 95% or higher) measured against human workers or existing legacy systems?"
4. **Cost of Failure:** "Can the organization absorb the legal, financial, or safety costs when the model makes an error (False Positive / False Negative)?"
5. **Maintenance Plan:** "Is there an established monitoring system and retraining pipeline (MLOps) in place to handle data shifts post-launch?

Summary of AI

AI Basic Workflow: Data → Model → Evaluation → Deployment

Building an AI system is not a one-and-done linear process; it is a continuously iterative pipeline.

1.Data (Data Collection and Preprocessing): Poor-quality data creates a failed model. Collect the necessary structured and unstructured data, and perform missing value imputation, noise removal, normalization, and labeling. Split the dataset into training, validation, and test sets.
2.Model (Model Selection and Training): Select an algorithm (such as XGBoost, Transformer, CNN, etc.) tailored to the problem type (classification, regression, clustering, etc.), feed the data into the model, and optimize parameters (weights).
3.Evaluation (Model Performance Validation): Rigorously inspect for overfitting. Evaluate performance using independent test data unseen during training, measuring objective metrics aligned with the problem (such as Accuracy, Precision, Recall, F1-Score, RMSE, etc.).
4.Deployment (Operational Deployment and Monitoring): Real-time data drift tracking is essential. Deploy the validated model to an API or serving infrastructure to connect it with real users. Continuously monitor performance degradation and real-world data shifts (Data Drift), and conduct periodic retraining.

Questions & Discussion