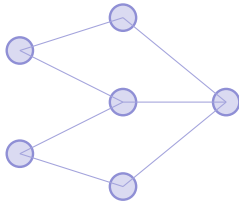


Artificial Intelligence

A-Level Computer Science ■ Topic 18

A-Level Computer Science



What we will cover

- 1 What AI is
- 2 Graphs in AI
- 3 Neural networks
- 4 ML, DL, RL, regression
- 5 Recap



Three nested ideas

Artificial intelligence builds systems that do tasks normally needing human intelligence – speech, images, translation, games, driving, generating text.

Artificial intelligence

Machine learning – learn patterns from data

Deep learning – many-layer neural nets

AI, machine learning, deep learning

Each is a subset of the one before – deep learning is one way of doing machine learning, which is one way of doing AI.

Artificial Intelligence (AI)

machines do tasks that
normally need human intelligence

Machine Learning (ML)

systems learn patterns
from data, not by fixed rules

Deep Learning (DL)

many-layer
neural networks



Many AI problems sit on a graph

Nodes are states or places; **edges** are moves or relationships.

pathfinding

roads; shortest route – Dijkstra, A*

game playing

board positions; **minimax** with alpha-beta

state space

apply operators until you reach a goal

knowledge

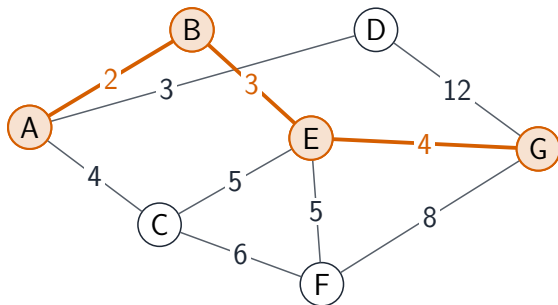
a **semantic network** : dog IS-A animal

Standard

tools: **BFS** and **DFS** .

Shortest path on a graph

Weights are costs; Dijkstra keeps the cheapest known route to each node and improves it.



— shortest path $A \rightarrow G$ (length 9)

Searching a graph, and when to reach for ML

breadth-first search

explores level by level – finds the **fewest-edge** path first

depth-first search

follows one branch to the end before backtracking – less memory, no shortest guarantee

Dijkstra's algorithm

the shortest path when edges carry **weights**

A*

Dijkstra plus a heuristic estimate of the remaining distance, so it explores fewer nodes

machine learning

the umbrella term: **any algorithm that learns from data** rather than following written rules

when to use it

where explicit rules would be **impractical** – spam filters, recommendations, fraud detection

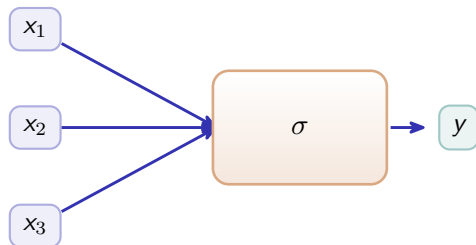
Three paradigms: **supervised** (labelled examples), **unsupervised** (structure found in unlabelled data), **reinforcement** (learning from reward).



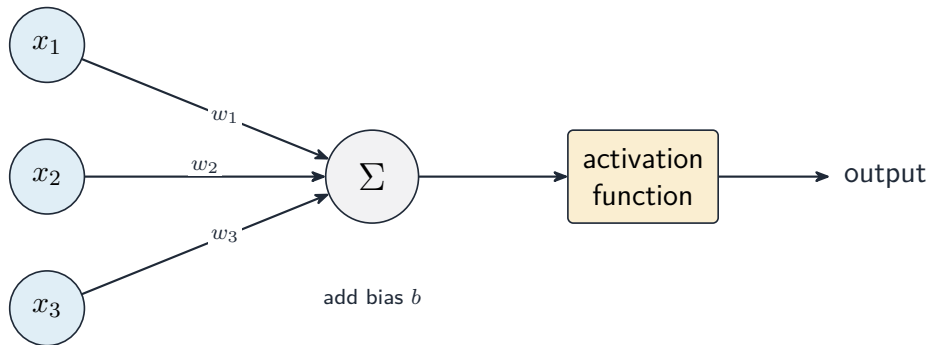
One artificial neuron

An **artificial neuron** :

- 1 multiply each **input** by a **weight**
- 2 add them up with a **bias**
- 3 apply an **activation function** (e.g. ReLU)
- 4 the output feeds the next layer



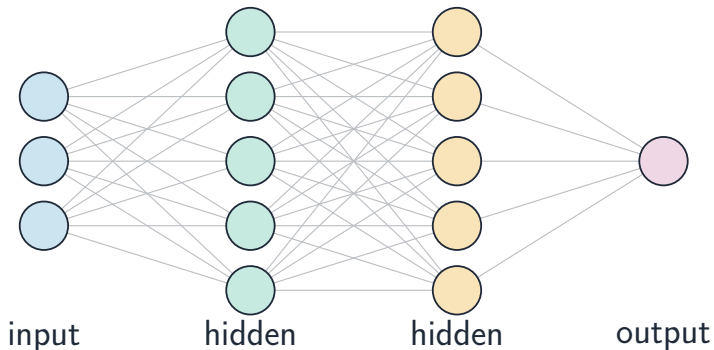
A single neuron



A single neuron: each input times its weight, summed with a bias, then an activation function

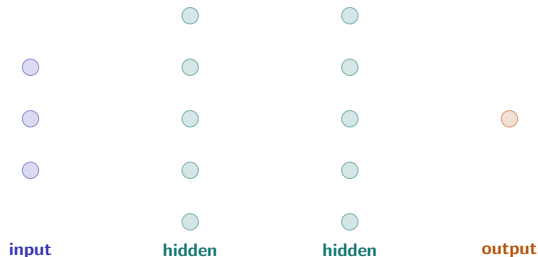
A layered network

Inputs feed hidden layers feed outputs; “deep” just means more than one hidden layer.



Layers: input, hidden, output

Neurons sit in layers. With many **hidden layers** it is a **deep neural network**.



ANNs learn patterns straight from raw data – pixels, audio, text – without hand-designed features.

Three kinds of learning

supervised

data has **labels** .
Learns input →
label. Classification
or regression.

unsupervised

no labels. Finds struc-
ture – e.g. a **cluster**
of similar customers.

reinforcement

an **agent** acts,
gets a **reward** ,
learns a **policy**
by trial and error.

How a model learns, and where it is used

Supervised learning

Supervised learning : labelled examples, so the error against the known answer drives the training.

Unsupervised learning

Unsupervised learning : no labels – the model finds clusters and structure by itself.

The bias term

Each neuron computes $\sum w_i x_i + b$: the weights scale the inputs, and the **bias term** shifts the threshold at which it fires.

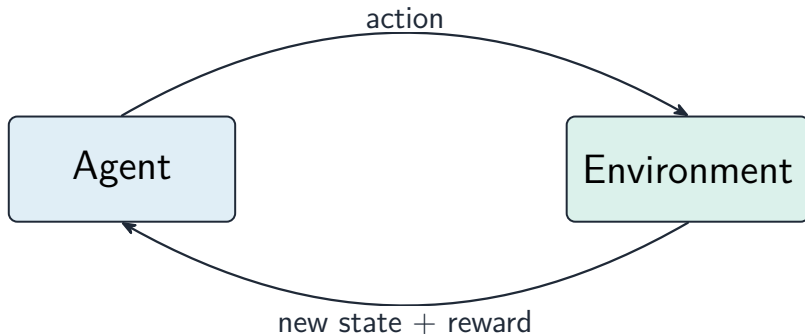
Where it shows up

Speech recognition and **text-to-speech** are the two directions of the same problem; a **knowledge graph** stores facts as entities and relations for a system to reason over.

Ask what the training signal is. Labels, no labels, or a reward – that single question separates the three kinds.

Reinforcement learning

The agent acts, the environment returns a state and a reward, and the policy is updated to earn more reward.



Industrial robot arms on a production line



Industrial robot arms on a production line: reinforcement learning can teach a robot to control its movements

Deep learning vs reinforcement learning

Deep learning is ML with deep nets. Lower layers learn edges and phonemes; higher layers combine them. Needs **lots of data** and **lots of compute** (GPUs). For small datasets, simpler ML often wins.

Reinforcement learning: the agent sends an action; the environment returns a new state and a reward. Used for games, robot control, driving – sequential decisions, no labels up front.

Where it ends up



Perception, planning and control – each a learned model, all running together.

Training: backpropagation

Training adjusts the **weights** so outputs match the targets – **backpropagation** with **gradient descent**.

- 1 forward pass – input through to the output
- 2 compute the error with a **loss function**
- 3 backward pass – each weight's gradient, via the chain rule
- 4 update the weights by a small **learning rate**

Repeat over many examples and many **epochs**. After training, a new input needs only one forward pass.

Training an ANN: backpropagation

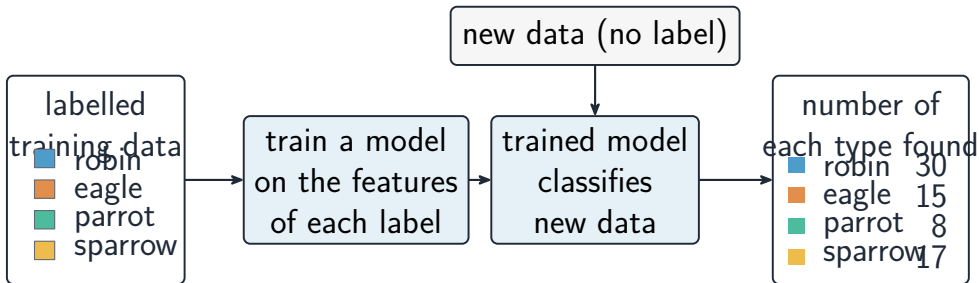


gradient descent: update w opposite the slope of the error surface

Training adjusts the weights to reach the minimum error

The training loop

Forward pass, compare with the label, propagate the error back, adjust the weights – repeated over the whole training set.



Regression predicts a number

Regression predicts a number (a house price, tomorrow's temperature).

Classification predicts a category. **Linear regression** fits a line that **minimises the sum of squared errors**:

$$y = m_1x_1 + \dots + m_nx_n + c$$

Temperature or spam?

(a) tomorrow's temperature – a number $\Rightarrow$ **regression**, one output neuron.

(b) spam or not – a category $\Rightarrow$ **classification**, one probability per class.

Both are supervised.

Reinforcement learning, and regression

reinforcement learning

an agent acts, receives a **reward or penalty**, and adjusts to maximise reward over time

a real example

a **self-driving car** : **lidar** and camera sensors feed the model, which learns which actions are rewarded

the difference

no labelled right answer is supplied – only a signal about how well the action turned out

regression

predicts a **number** rather than a class

fitting it

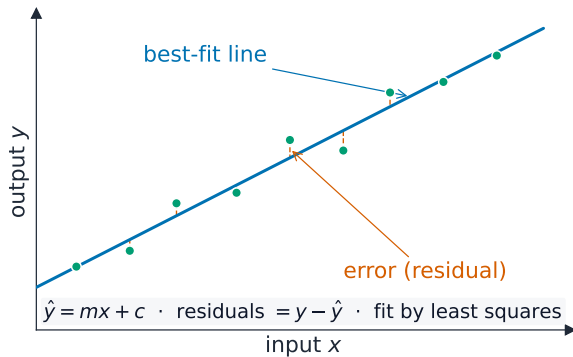
choose the coefficients to **minimise the sum of squared errors** against the training data

when to use it

where the relationship looks roughly linear and you need a quantity, not a label

Classification answers *which*; regression answers *how much*. Choosing the wrong one is a modelling error no amount of data will fix.

Regression



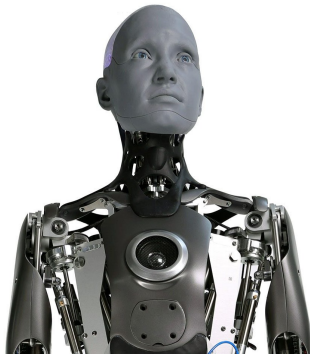
Linear regression fits the line that makes the total squared error (the dashed gaps) as small as possible

How AI is used in a real scenario

Many exam scenarios use the same pattern – a **deep-learning model trained on labelled data**, often several combined into a pipeline:

- **customer identification** at an automated shop: the system is trained on labelled face images; a camera captures a face; image recognition extracts a representation.
- **reading text from images**: image recognition finds text regions; **optical character recognition** extracts the characters; machine translation converts them.
- **checkout item-detection**: object-detection AI, trained on labelled product images, sees which items go into a basket and charges the account.

A humanoid robot uses AI to see



Every one of those is a **separate** trained model – vision, speech, planning – stitched together. Nothing here understands the room the way a person does.

Exam tips

- Distinguish **machine learning, deep learning and reinforcement learning** with an example of each.
- Describe an **ANN** (input, hidden and output layers; weighted connections) and how **backpropagation** adjusts weights to cut error.
- Distinguish **supervised vs unsupervised** learning; regression predicts a continuous value.



Topic 18 – key takeaways

1 Nesting

$DL \subset ML \subset AI$

2 Neuron

weighted sum + bias + activation

3 Learning

supervised, unsupervised, reinforcement

4 Train

backprop adjusts weights to cut the loss

Check yourself

- 1 Is deep learning a kind of machine learning, or the other way round?
- 2 What does an activation function do?
- 3 An agent learns from reward, with no labels – which paradigm?
- 4 Predicting a house price: regression or classification?

Answers 1. DL is a kind of ML 2. adds non-linearity after the weighted sum 3. reinforcement learning 4. regression