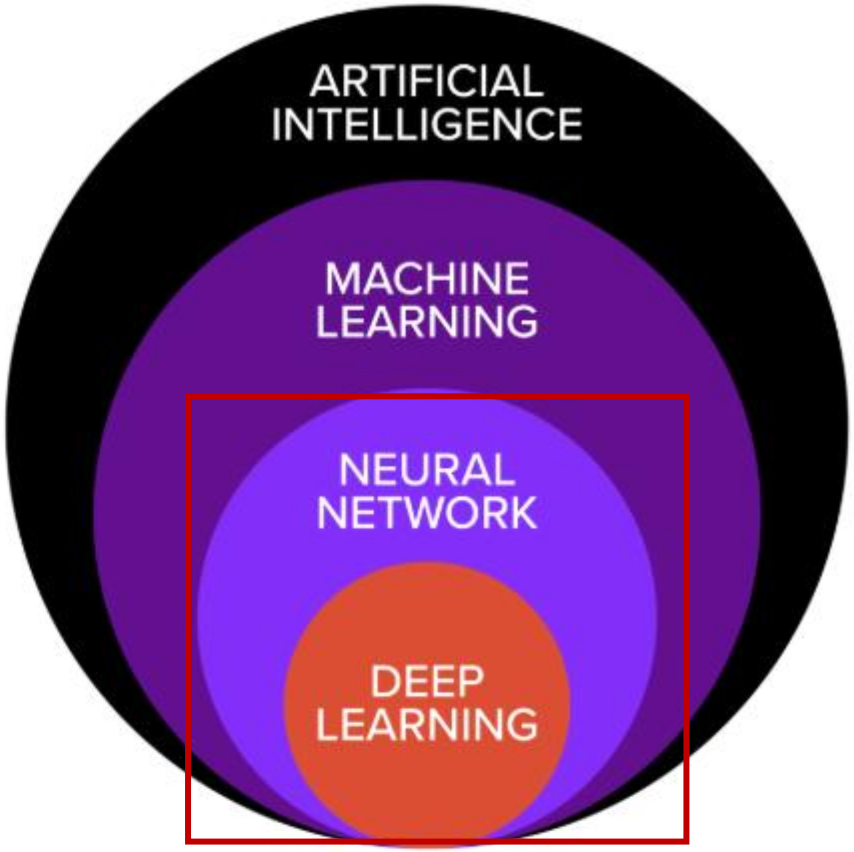
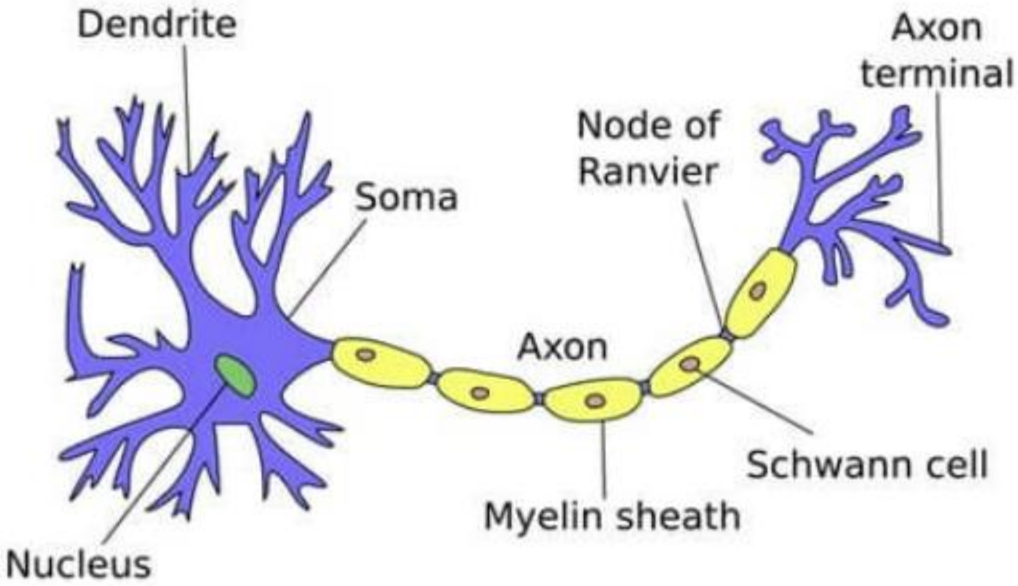
# Introduction to Neural Networks and Deep Learning

Onoja Anthony, Ph.D.

# What Are Neural Networks?

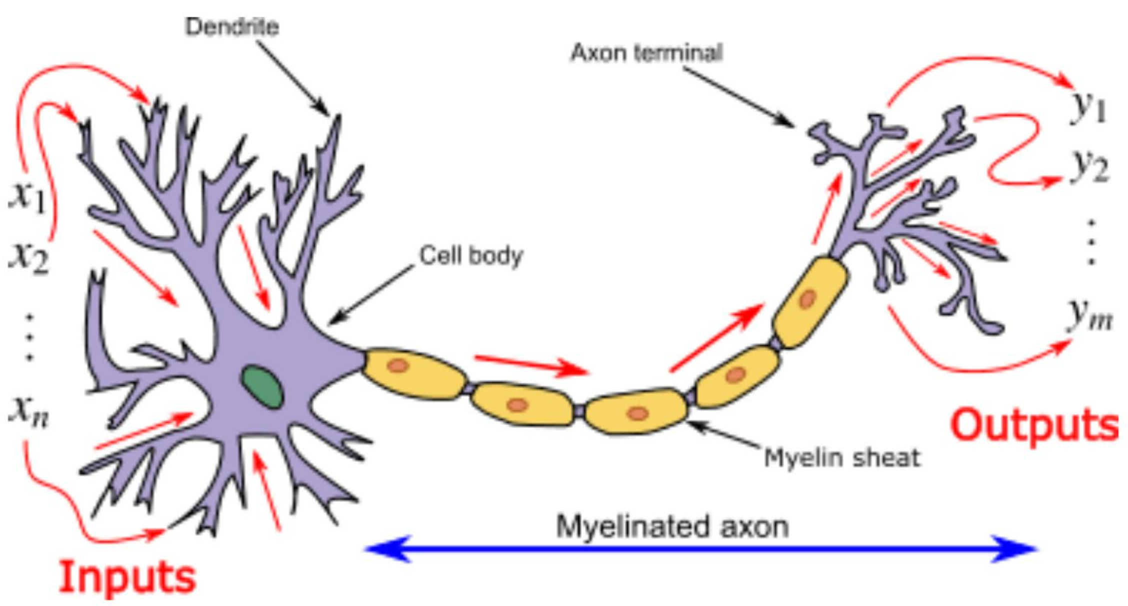



A neural network is a computational model inspired by the human brain.

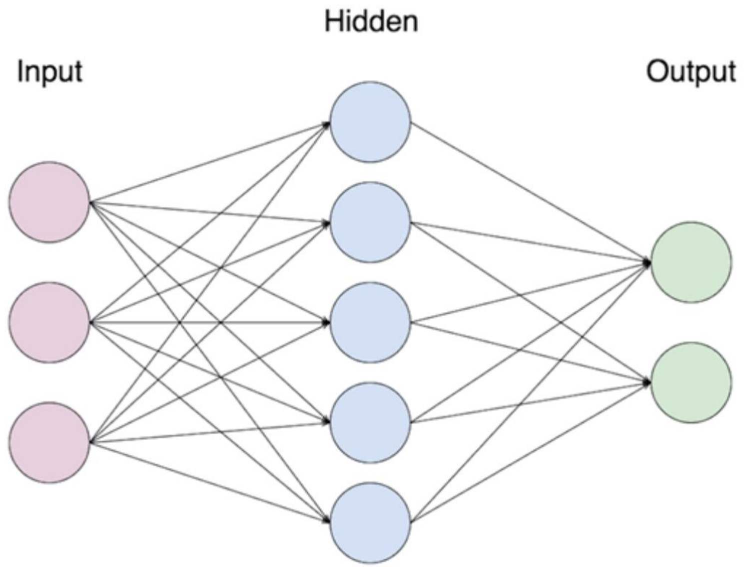
# What is Neural Networks

- ✓ They consist of interconnected nodes (neurons) organised in layers, processing input data to produce outputs for tasks like classification, regression, or generation.

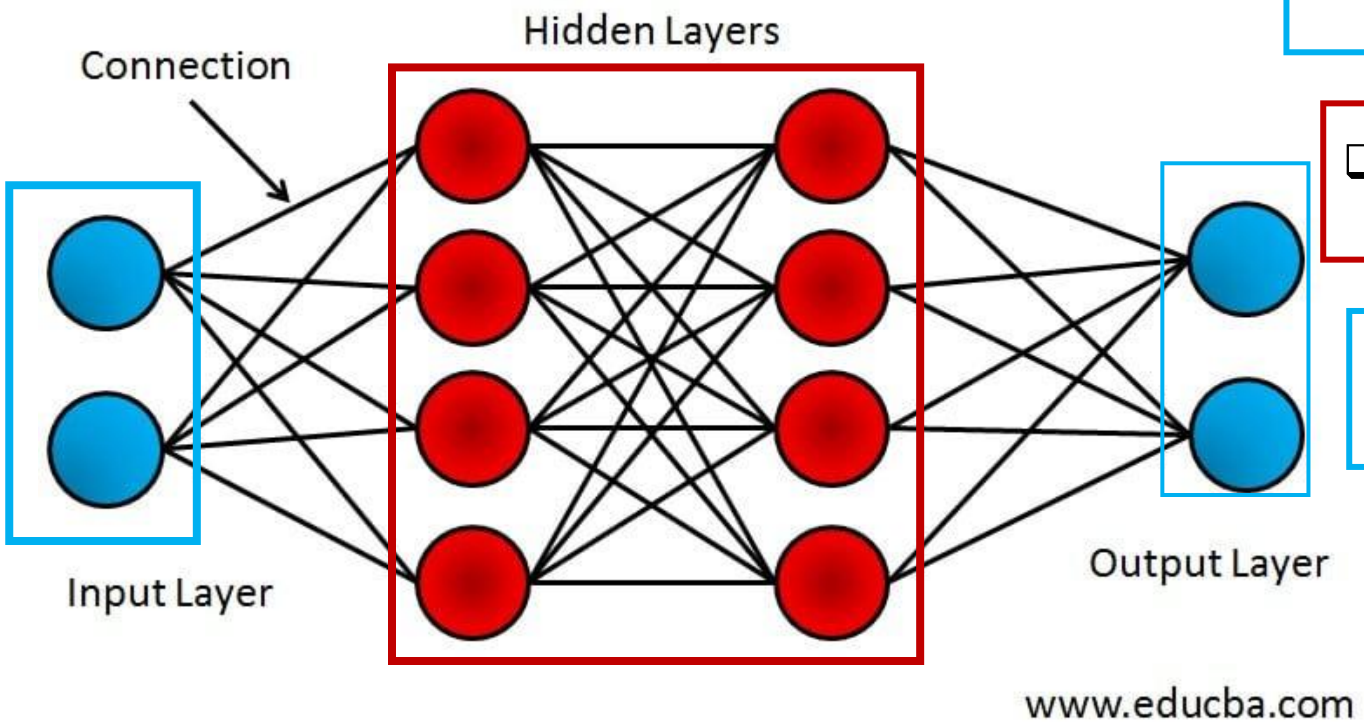



Key Components:

- ❑ Neurons
- ❑ Layers: Input, Hidden, Output
- ❑ Weights & Biases
- ❑ Activation Functions: Sigmoid, ReLU, Tanh



# Structure of a Neural Network

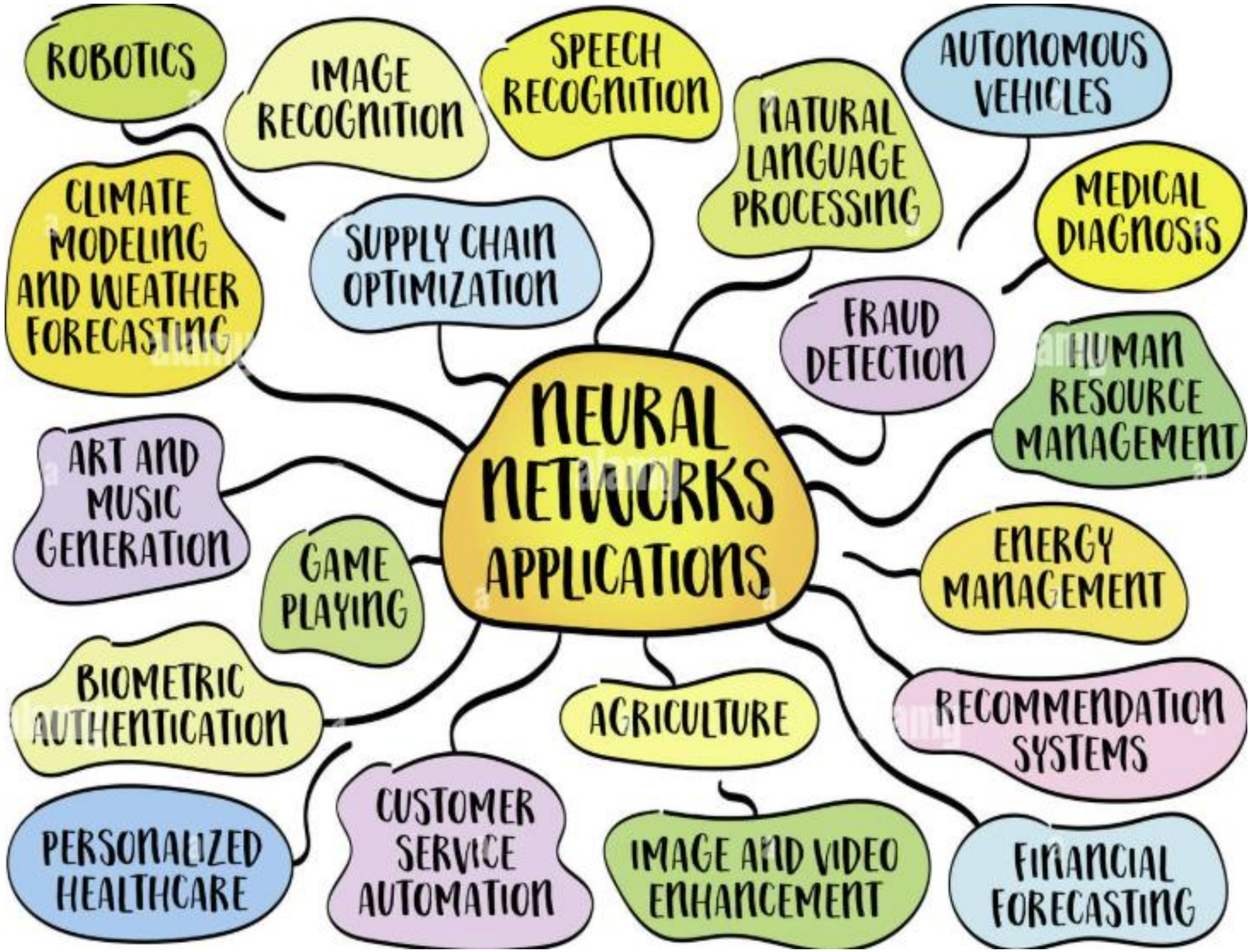



- Input Layer: Receives raw data (e.g., pixel values of an image).
- Hidden Layers: Process data through weighted connections and activation functions.
- Output Layer: Produces the final prediction or classification.

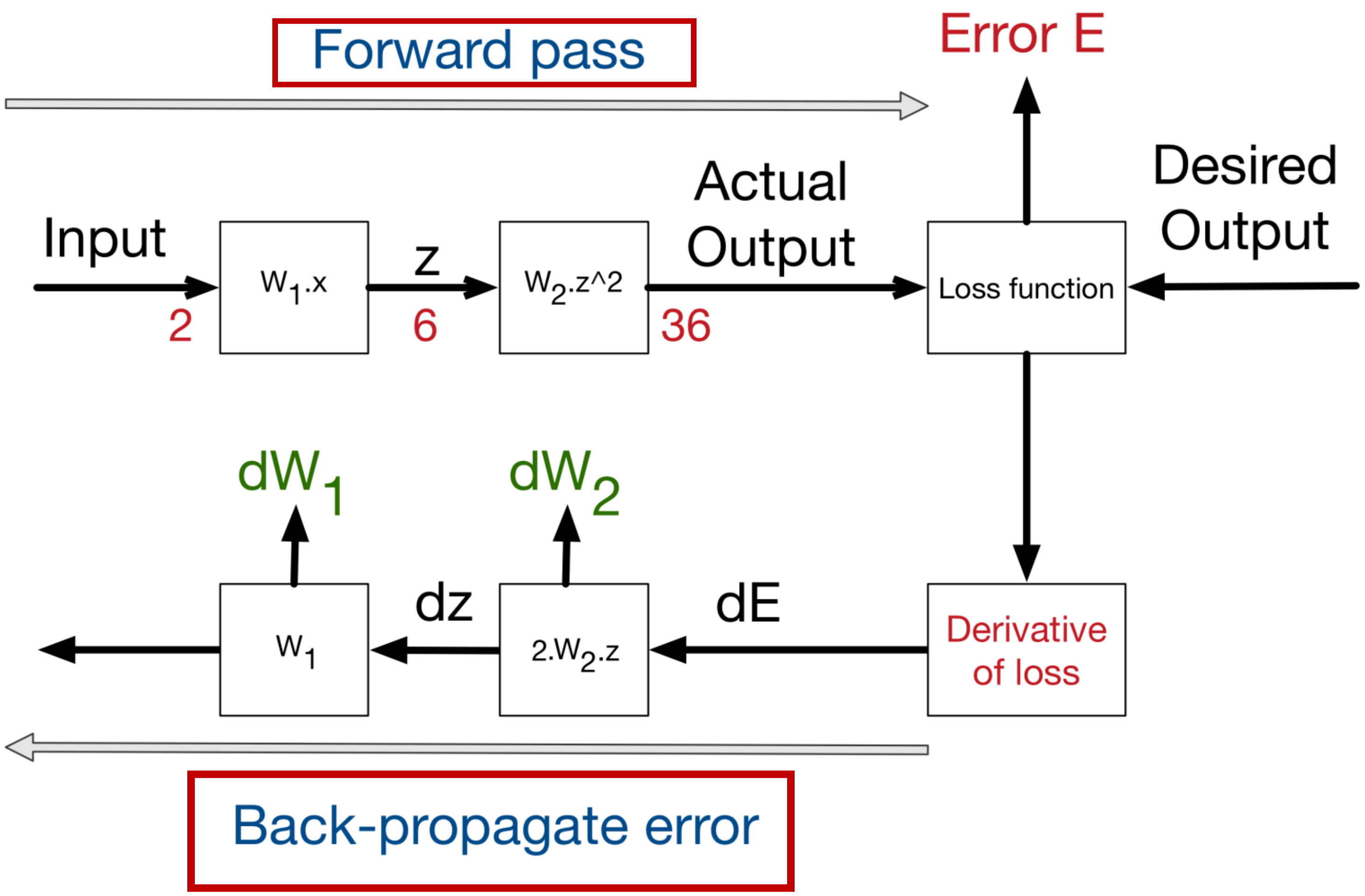
Each neuron computes a weighted sum of inputs, applies an activation function, and passes the result to the next layer.

# Applications

- Image recognition (e.g., identifying objects in photos).
- Natural language processing (e.g., sentiment analysis).
- Speech recognition (e.g., voice assistants).



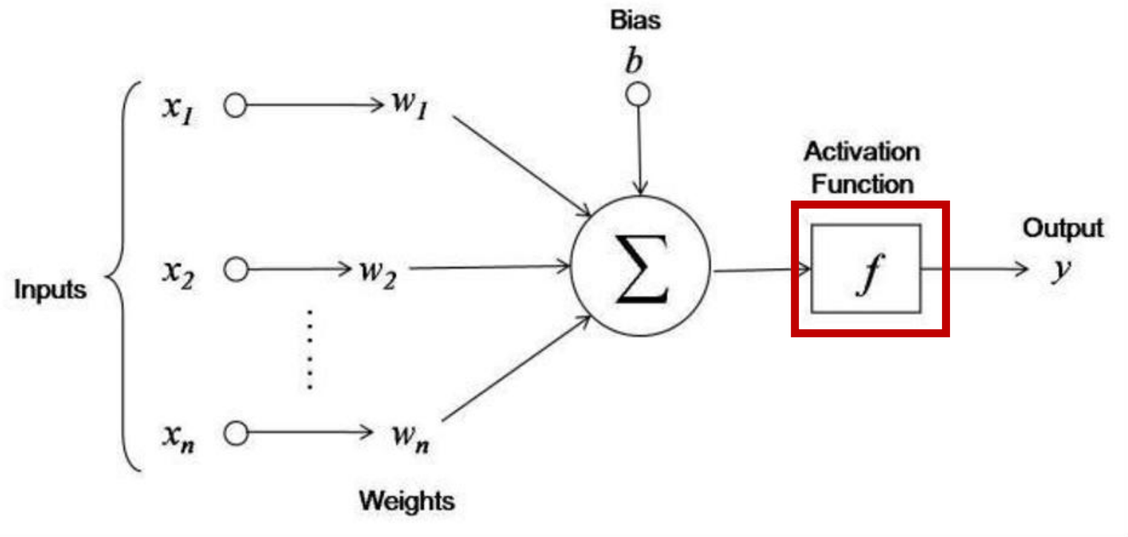
# How Neural Networks Learn

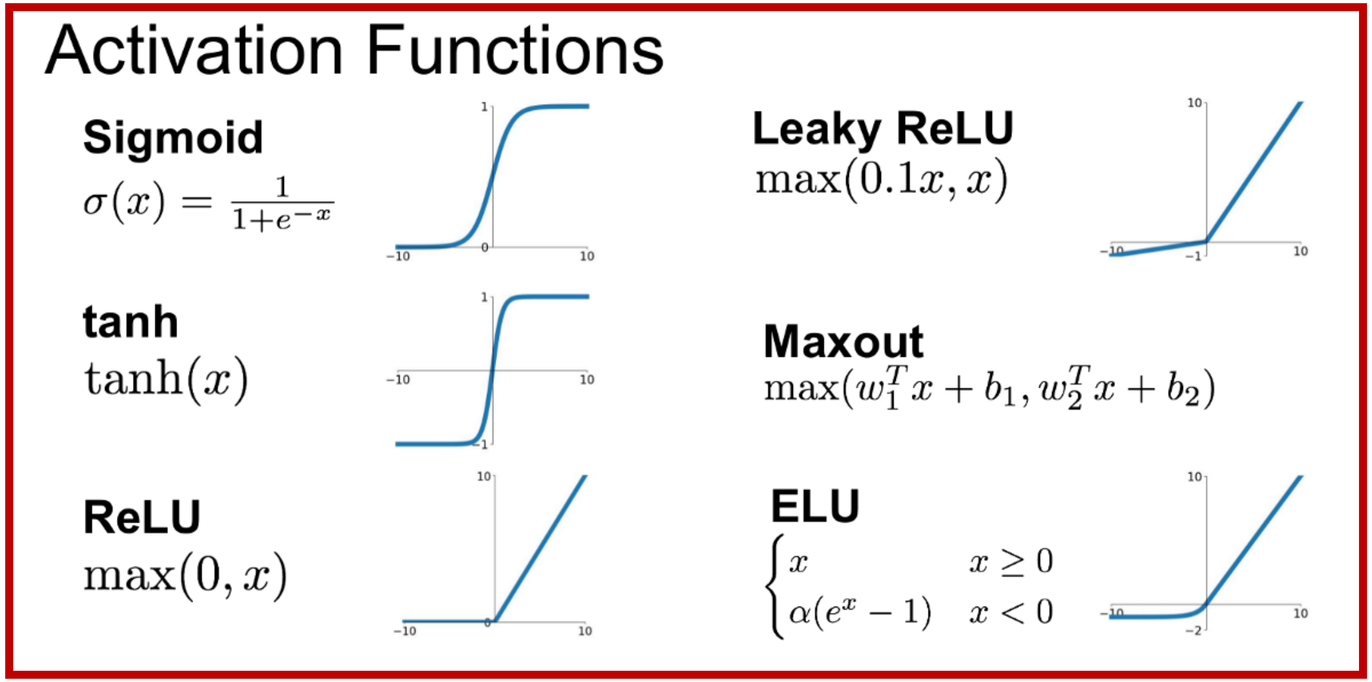



- ❑ Forward Propagation: Flow of data through the network.
- ❑ Loss Functions: MSE, Cross-Entropy
- ❑ Backpropagation:
  - Gradient Descent
  - Chain Rule
  - Updating weights

# How Neural Networks Learn: Activation Functions

Activation functions introduce non-linearity to neural networks, enabling them to model complex patterns. They determine whether a neuron fires and how strongly



# How Neural Networks Learn: Activation Functions

- Sigmoid: Binary classification but suffers from vanishing gradients. Range: (0, 1).

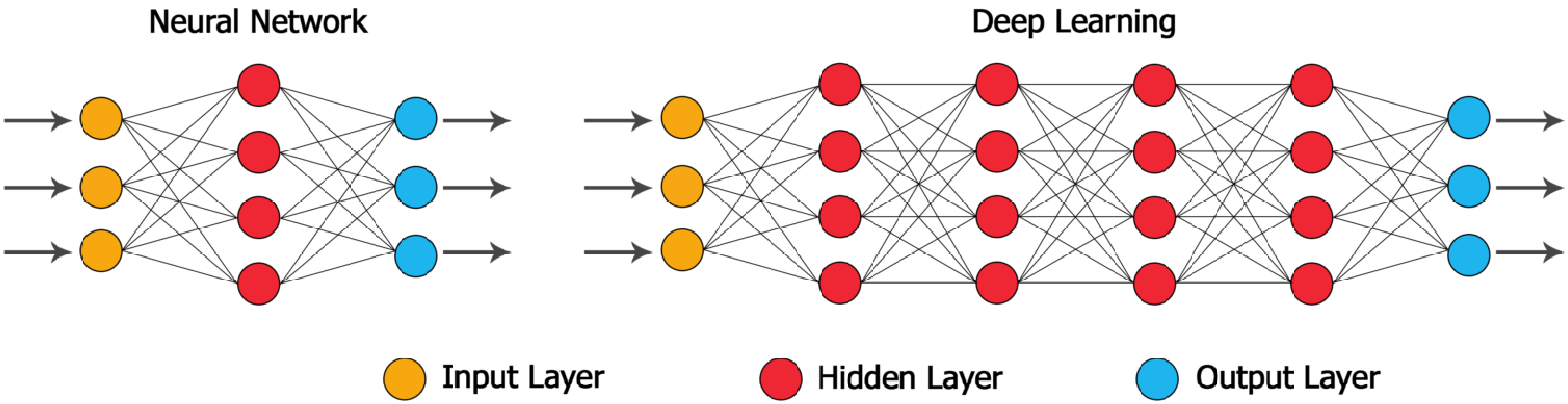
- ReLU (Rectified Linear Unit): Default for hidden layers, fast convergence, avoids vanishing gradients. Range: [0, ∞).

- Tanh: Centered outputs, better for some optimisation tasks. Range (-1, 1).

- Softmax: Multi-class classification (output layer). Range (0, 1).

# What Is Deep Learning?

Neural networks with many hidden layers.

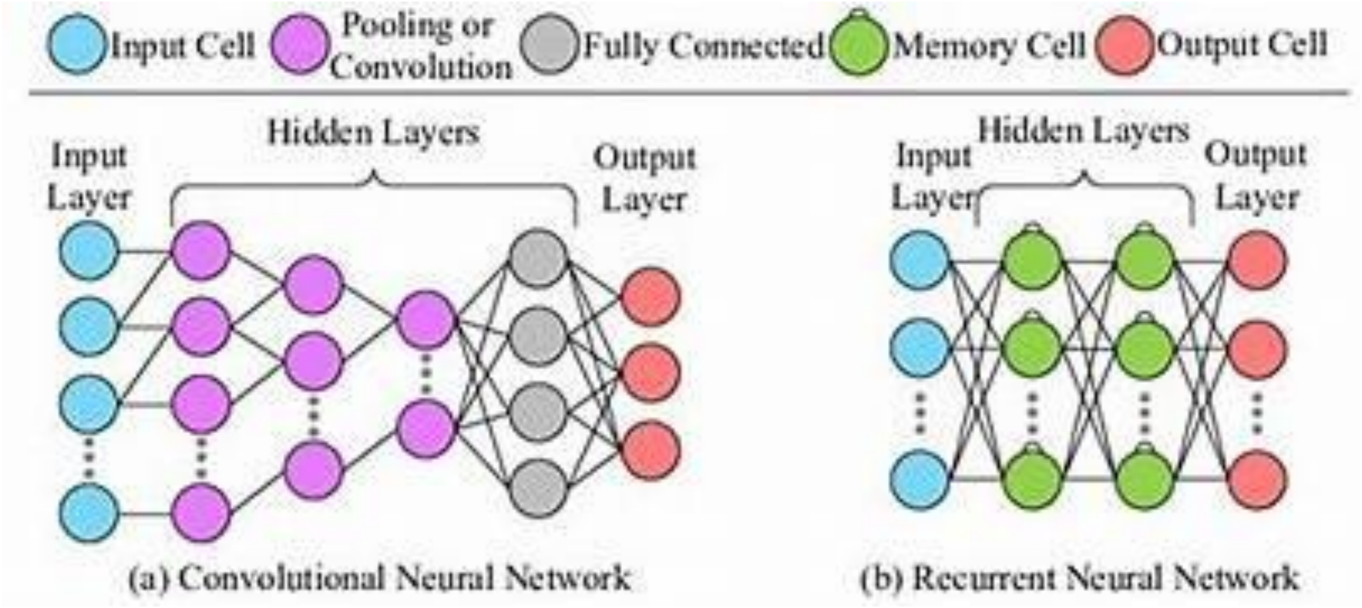



Why Deep? Captures hierarchical features (e.g., edges → shapes → objects in images)

Examples: Image classification (CNNs), Text generation (RNNs, Transformers), Recommendation engines.

# Popular Neural Network Architectures

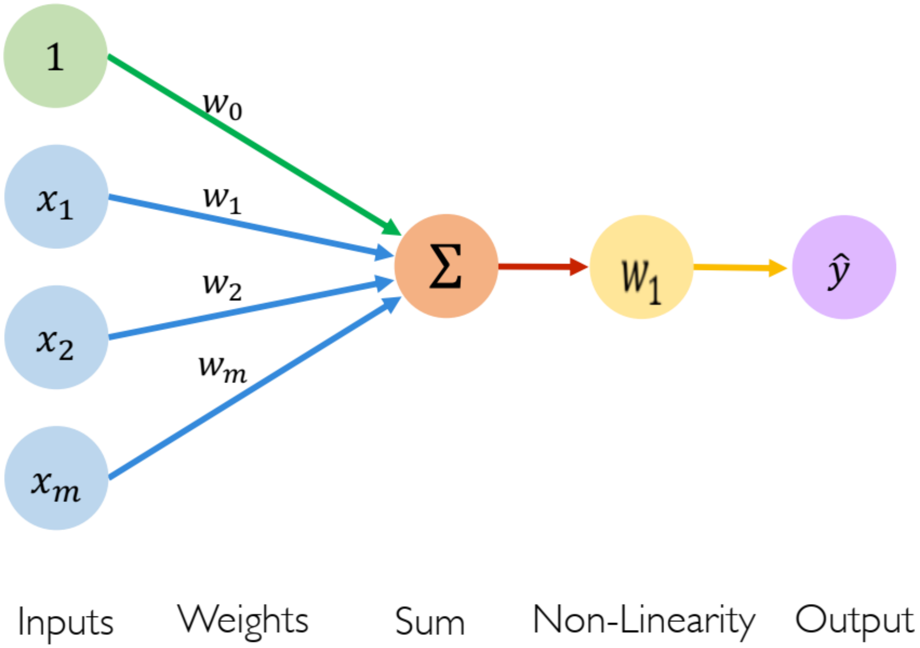
- ❑ Feedforward Neural Networks (FNNs)
- ❑ Convolutional Neural Networks (CNNs) – for image data
- ❑ Recurrent Neural Networks (RNNs) – for sequential data
- ❑ Transformers & BERT



# Concept of Feedforward propagation

Feedforward is the process of passing input data through the network to produce an output:

- ✓ Input data is fed into the input layer.
- ✓ Each layer computes weighted sums and applies activation functions.
- ✓ Output layer produces predictions (e.g., class probabilities).




Output

Linear combination of inputs

$$\hat{y} = g\left(w_0 + \sum_{i=1}^{m} x_i \, w_i\right)$$

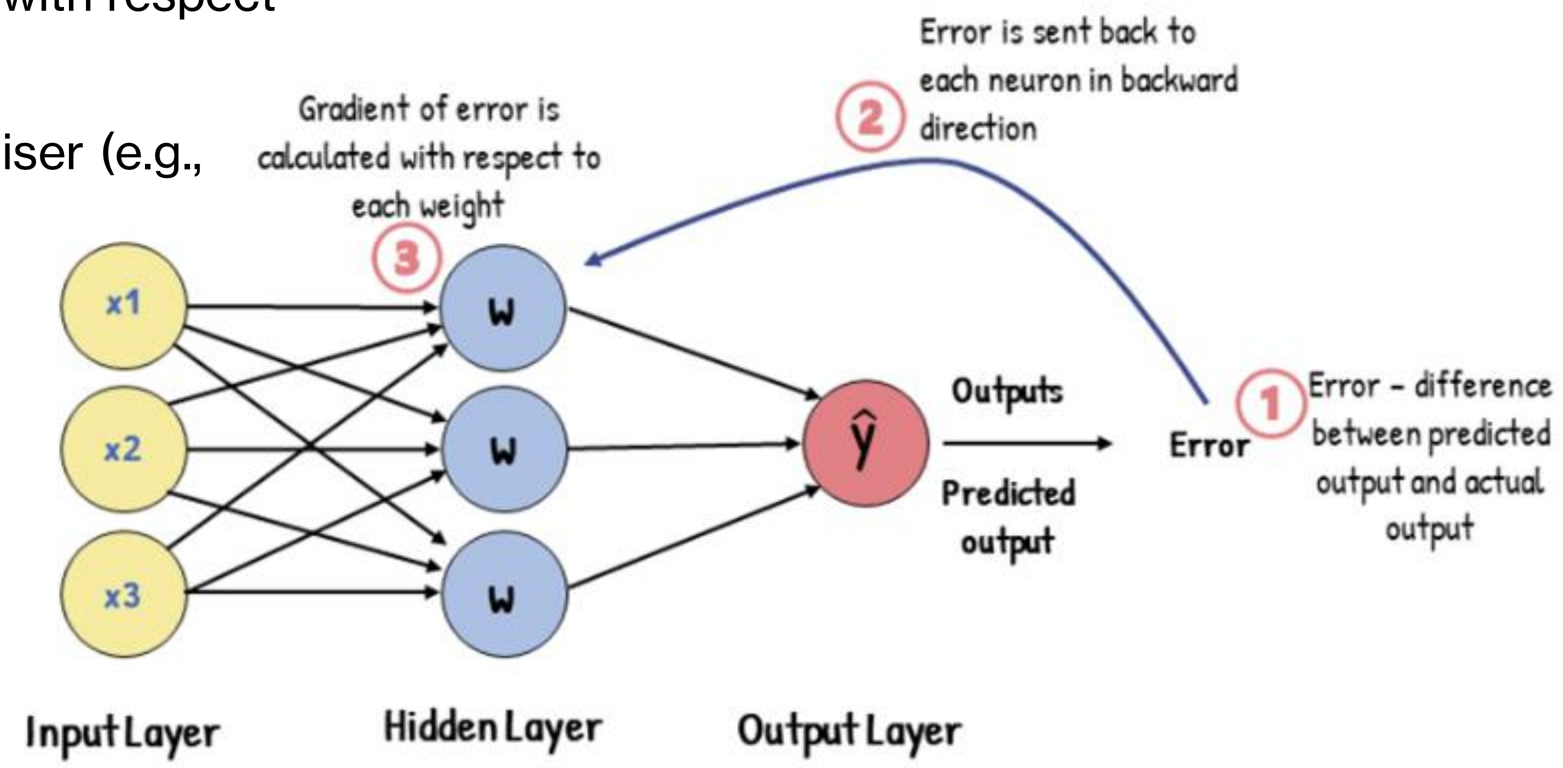
Non-linear activation function

Bias

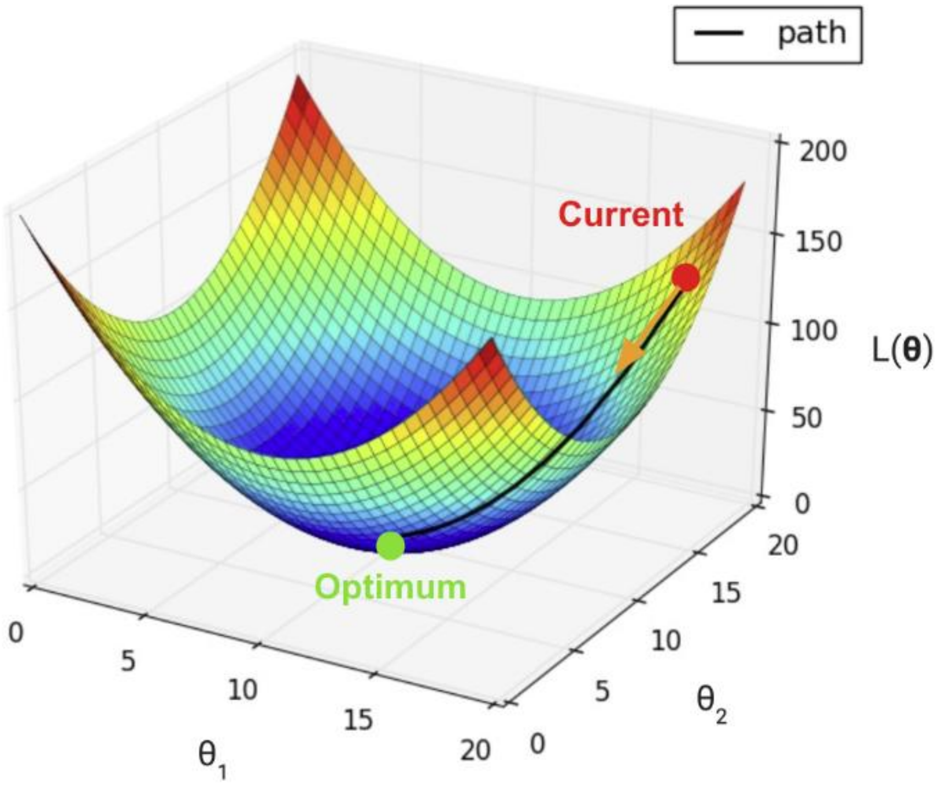
# Concept of Backpropagation

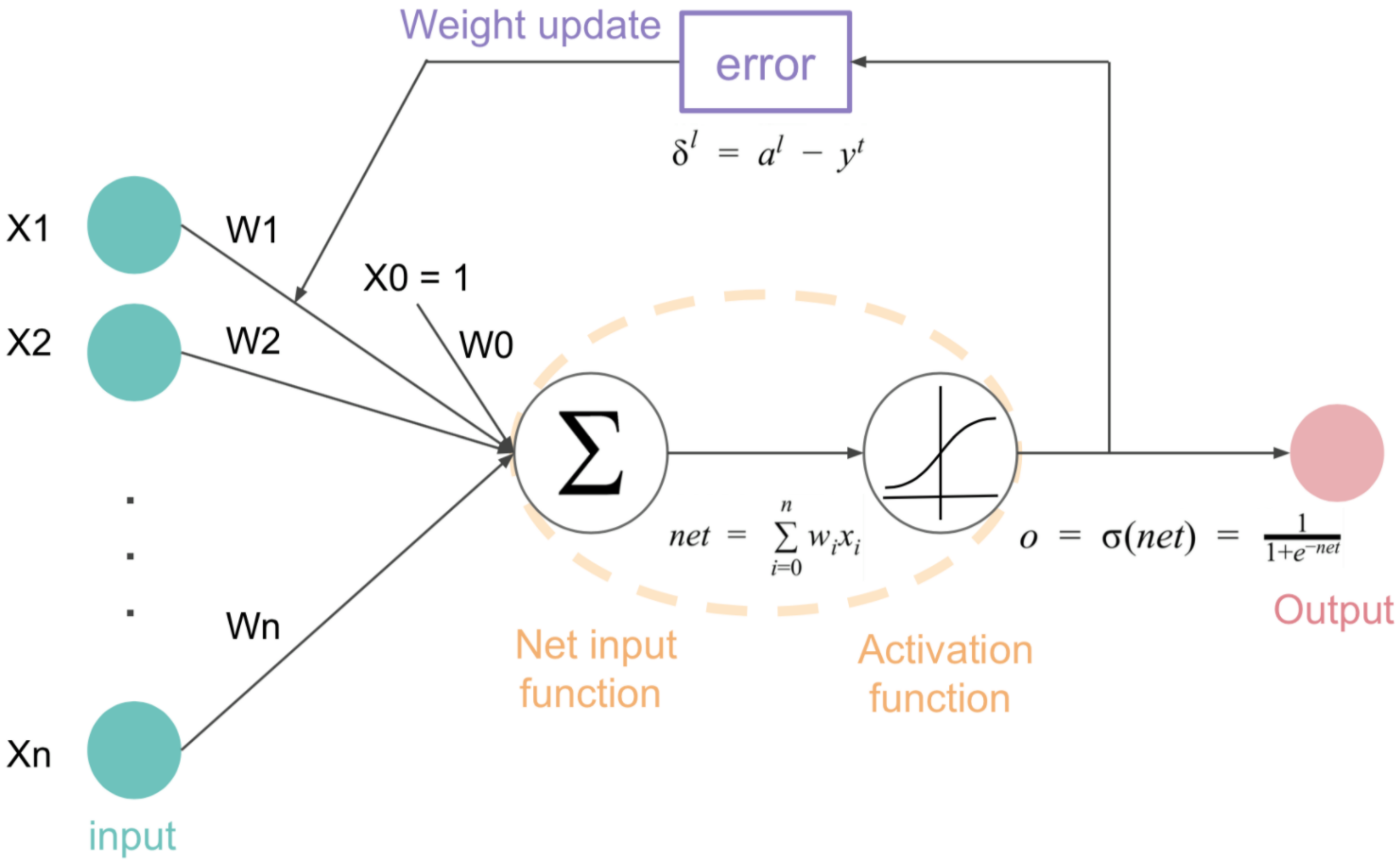
Backpropagation (backward propagation of errors) updates weights to minimise the loss function:

- ✓ Compute the loss (e.g., mean squared error or cross-entropy).
- ✓ Calculate gradients of the loss with respect to weights using the chain rule.
- ✓ Update weights using an optimiser (e.g., gradient descent).



# Loss Functions




Mean Squared Error (MSE): $L = n_1 \sum n_i = 1(y_i - y\hat{}_i)*_2$ (regression).

Cross-Entropy Loss: $L = -\sum_i y_i \log(y\hat{}_i)$ (classification).

# Challenges in Training Deep Networks

- ❑ Overfitting
- ❑ Vanishing/Exploding gradients
- ❑ Need for large data and compute
- ❑ Regularization: Dropout, L2
- ❑ Batch Normalization