University Of Canberra

WEEK10

Pattern Recognition and Machine Learning

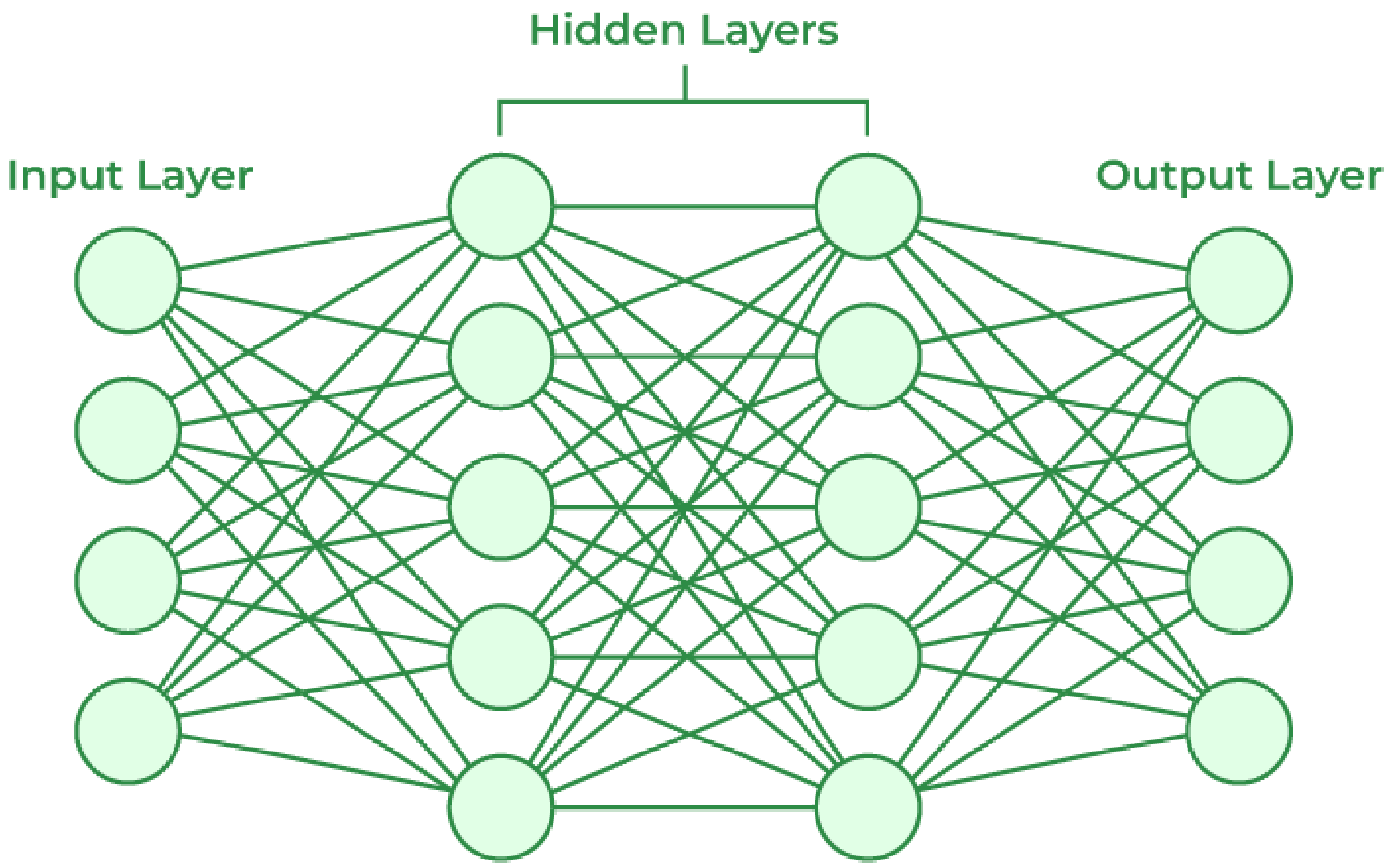
Created by Elane Peng

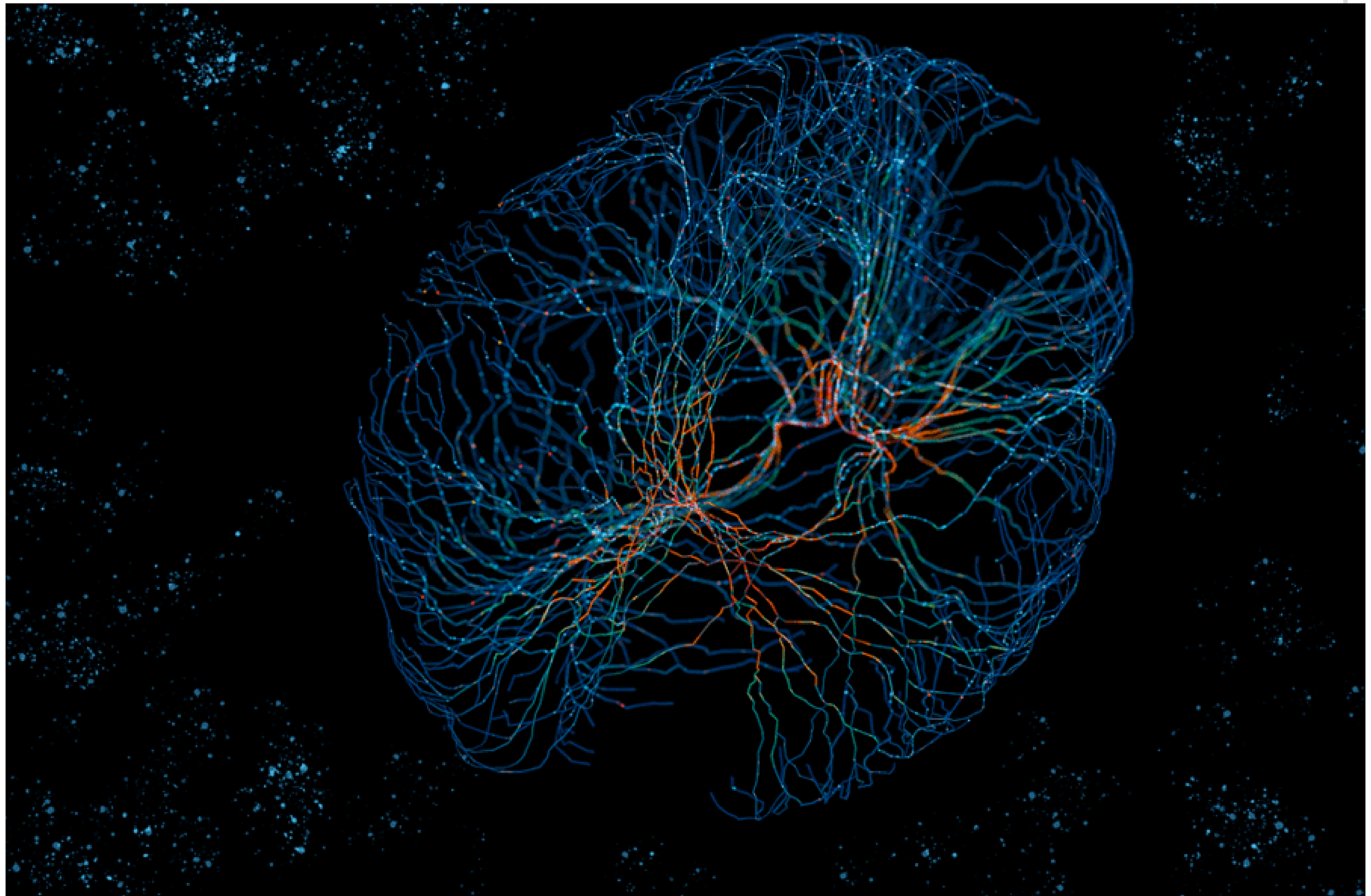
Objectives

- Understand the fundamentals of neural networks and deep learning, including their structure, components, and how they are inspired by the human brain.
- Explain how neural networks learn from data, covering key concepts such as weights, biases, activation functions, forward and backward propagation, epochs, learning rate, and loss functions.
- Apply knowledge of neural networks to practical problems and real-world applications.

Neural Network

1. What are the similarities between neural networks and the way the human brain learns patterns and knowledge?





Neural Network

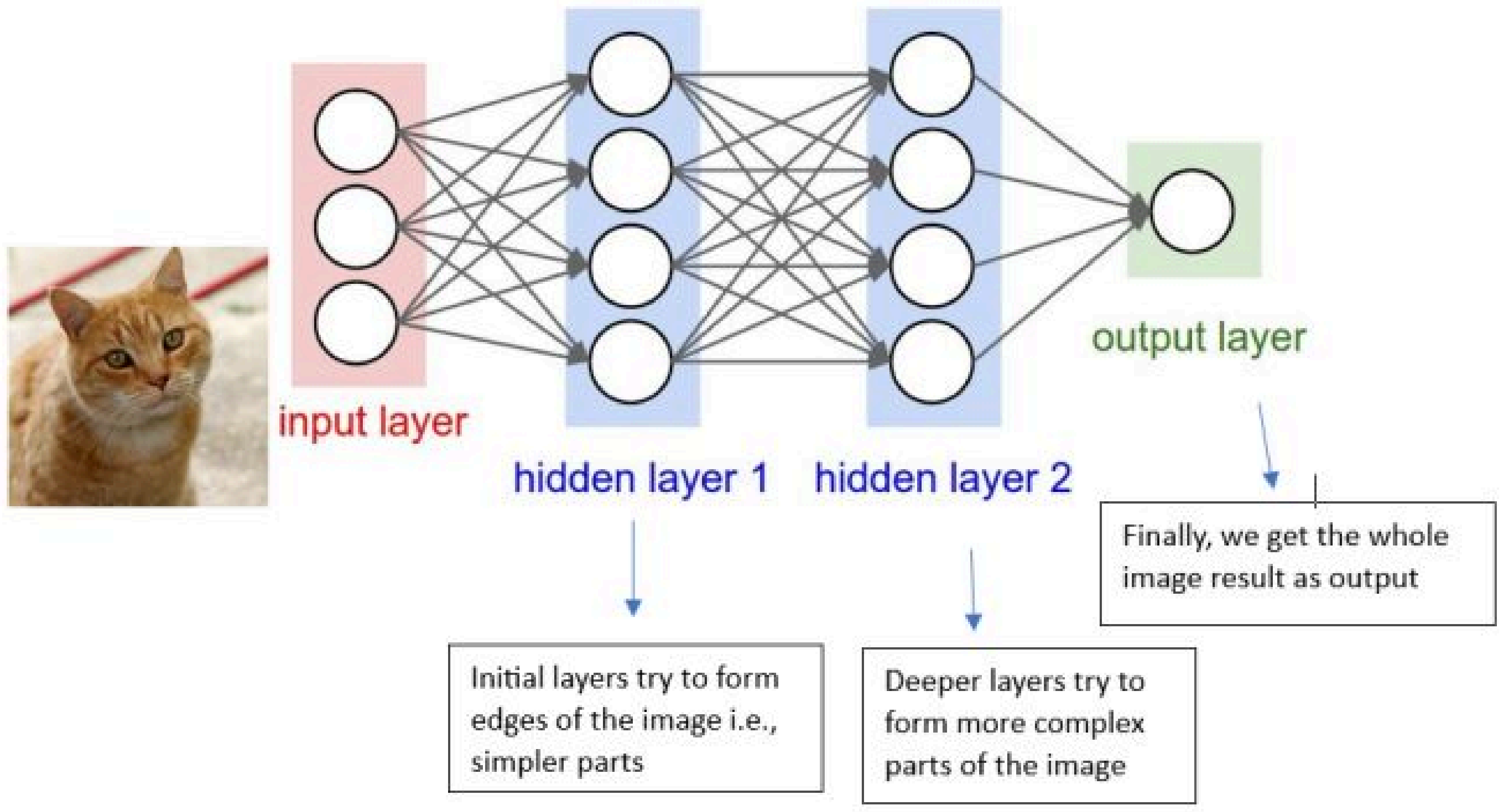
1. What are the similarities between neural networks and the way the human brain learns patterns and knowledge?

• Both systems have neurons that process inputs and produce outputs. • Learning occurs through the strength of connections—synapses in the brain and weights in neural networks. • Information is processed hierarchically, with each layer extracting higher-level features. • Both can adapt to new data and generalize to unseen patterns. • Neural networks are simplified models and do not capture the full complexity of the brain.

Neural Network

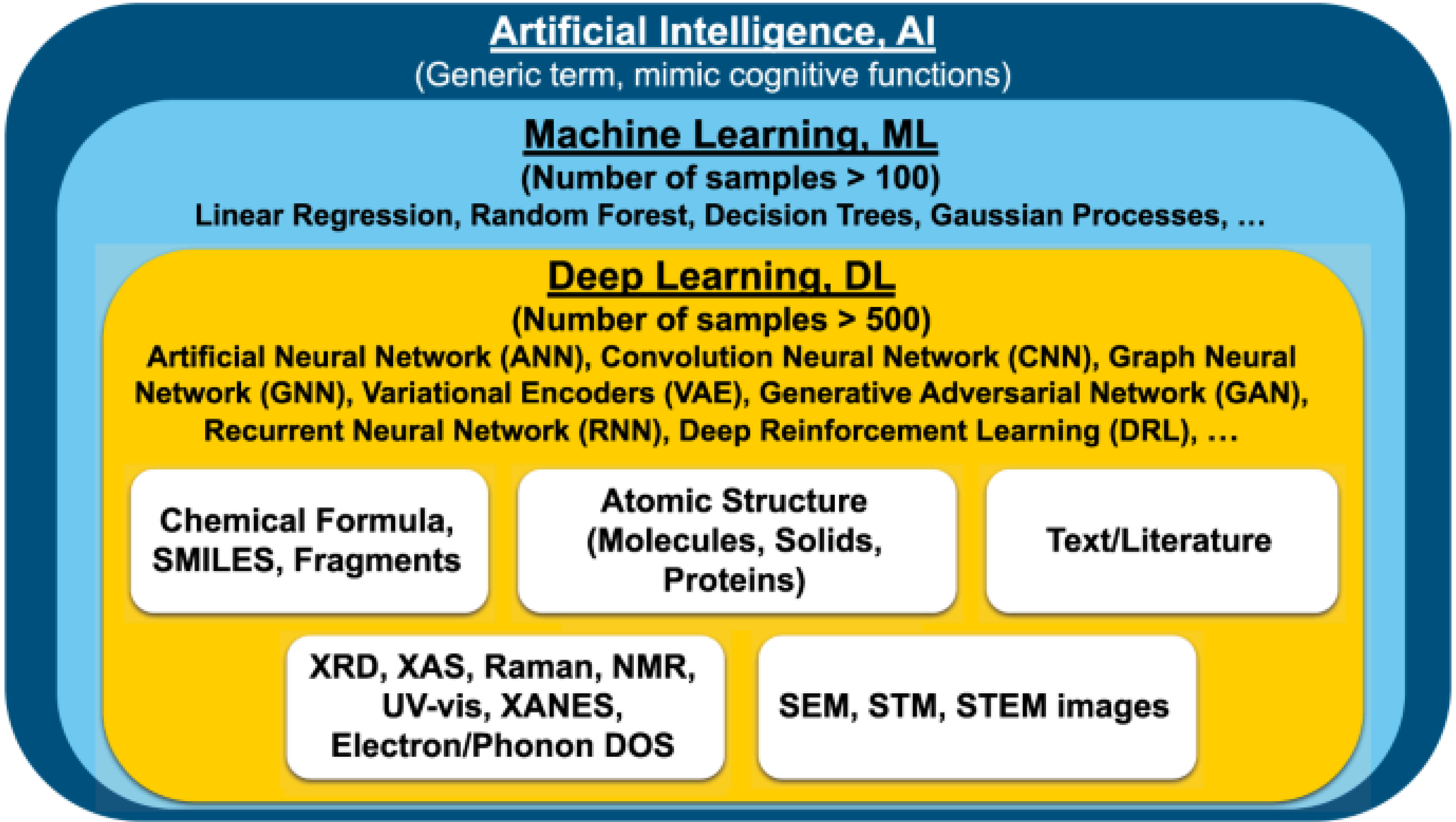
Child's brain → learns cats via synapses adjusting through repeated exposure.

Neural network → learns cats via weights adjusting through repeated training data.



Neural Network

2. What is deep learning, and how is it related to neural networks?



Neural Network

2. What is deep learning, and how is it related to neural networks?

- Deep learning is a subset of machine learning that uses neural networks with multiple hidden layers.
- These deep networks automatically learn hierarchical features from data.
- Deep learning relies on neural networks as its core architecture, making them suitable for tasks such as image recognition and natural language processing.

Neural Network

2. What is deep learning, and how is it related to neural networks?

Deep learning power comes from how those layers discover features on their own. In older machine learning, people had to hand-engineer features (like edges in images or grammar rules in text). Deep learning lets the network learn those features directly from raw data. That’s why it can handle complex tasks like translating languages or generating art—it learns structure humans might not even see.

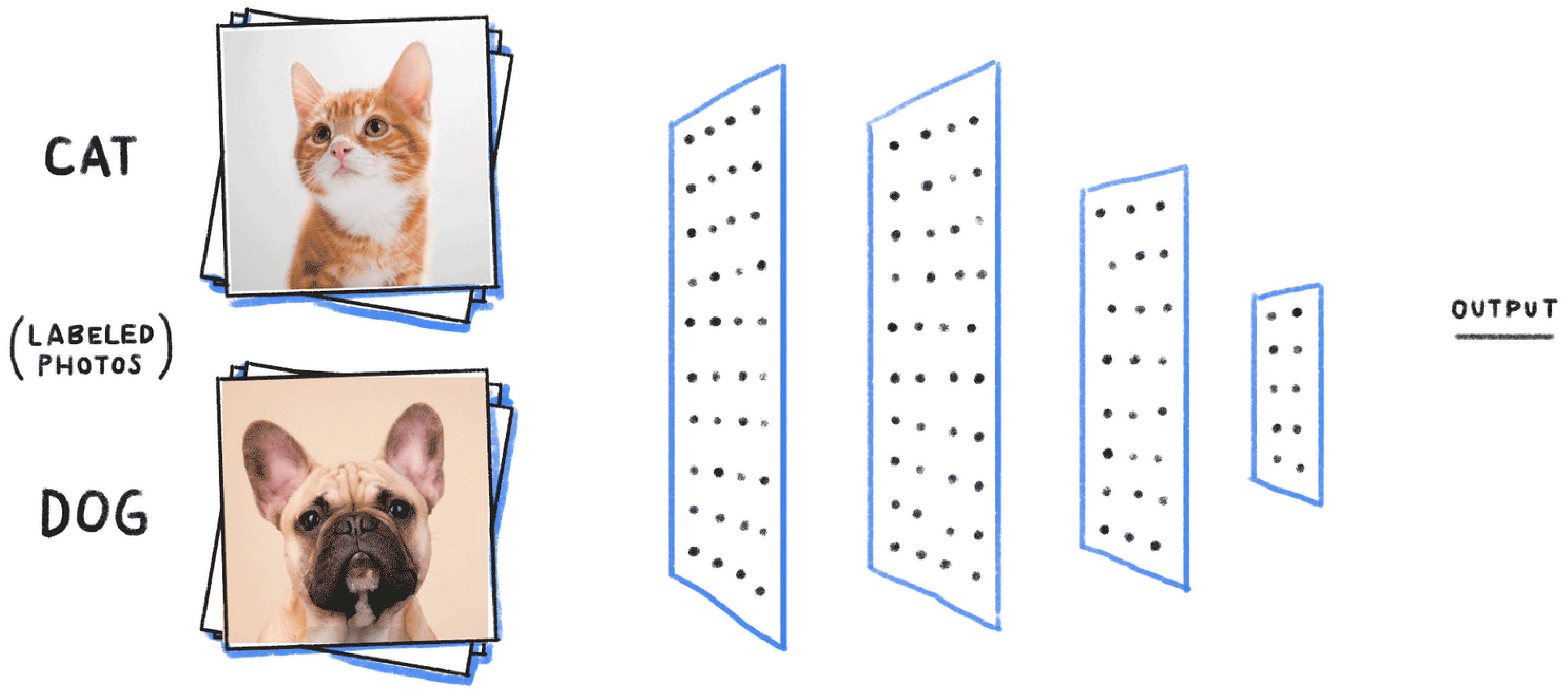
Neural Network

2. What is deep learning, and how is it related to neural networks?

deep learning is data-hungry and computationally expensive. It shines when there’s massive data and strong hardware (GPUs, TPUs)

Neural Network

learn from mistakes, refine patterns, and generalize.



Neural Network

Feature	Traditional Machine Learning	Shallow Neural Networks	Deep Learning
Feature Extraction	Manually engineered by experts	Some manual feature design needed	Learns multi-level features automatically
Model Complexity	Simple algorithms (e.g., decision trees, SVM)	1–2 hidden layers	Many layers (10, 100, or more)
Data Requirement	Works with small datasets	Needs moderate data	Requires massive datasets
Computational Resources	Runs on a regular computer	Higher requirements	Heavy reliance on GPUs/TPUs
Typical Tasks	Tabular data, basic classification	Basic pattern recognition	Complex tasks: image recognition, speech, NLP,
Strengths	Interpretable, fast to train	Handles non-linear patterns	Extremely powerful, learns complex patterns
Weaknesses	Limited performance, feature design is costly	Limited expressive power	“Black box,” expensive to train, data-hungry

Neural Network

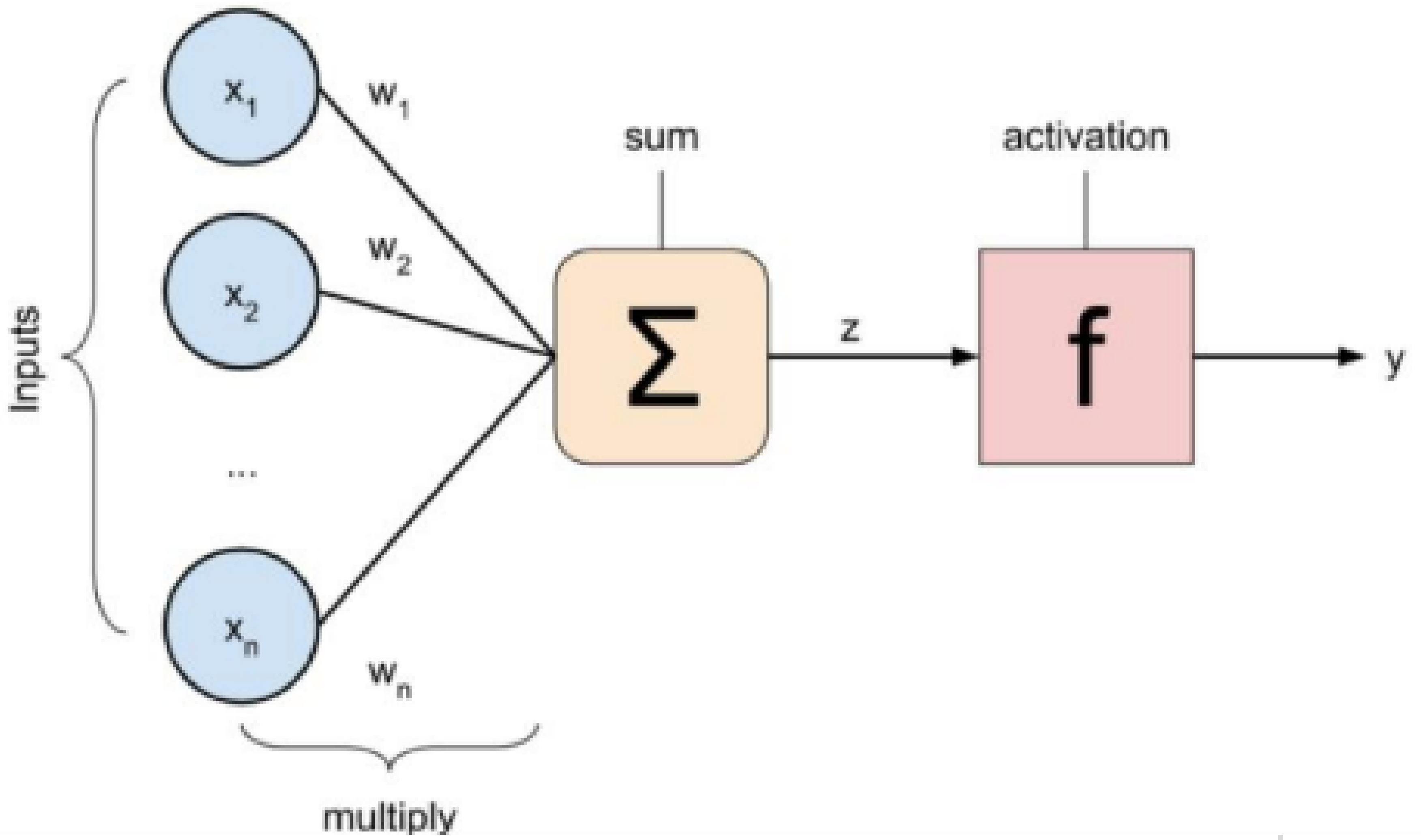
3. How do deep learning algorithms perform compared to traditional machine learning algorithms when the size of the data increases?

- Deep learning algorithms can learn hierarchical representations automatically.
- They capture complex non-linear patterns that traditional ML may miss.
- They scale efficiently with large datasets and generalize better with more data.
- Traditional ML often relies on handcrafted features and may struggle with large or complex datasets.

Neural Network

4. What are the main components of a neural network, and how do they function? Explain the computation process of a neuron.

- **Input Layer:** Receives features from the dataset.
- **Hidden Layers:** Process data and extract patterns through computations.
- **Output Layer:** Produces predictions or classifications.
- **Weights and Biases:** Parameters that control how inputs are transformed and adjusted during training.
- **Activation Functions:** Introduce non-linearity to enable the network to learn complex patterns.



Neural Network

4. What are the main components of a neural network, and how do they function? Explain the computation process of a neuron.

Computation:

• Weighted sum of its inputs, b is the bias term, which allows the neuron to shift the activation function and improves flexibility.

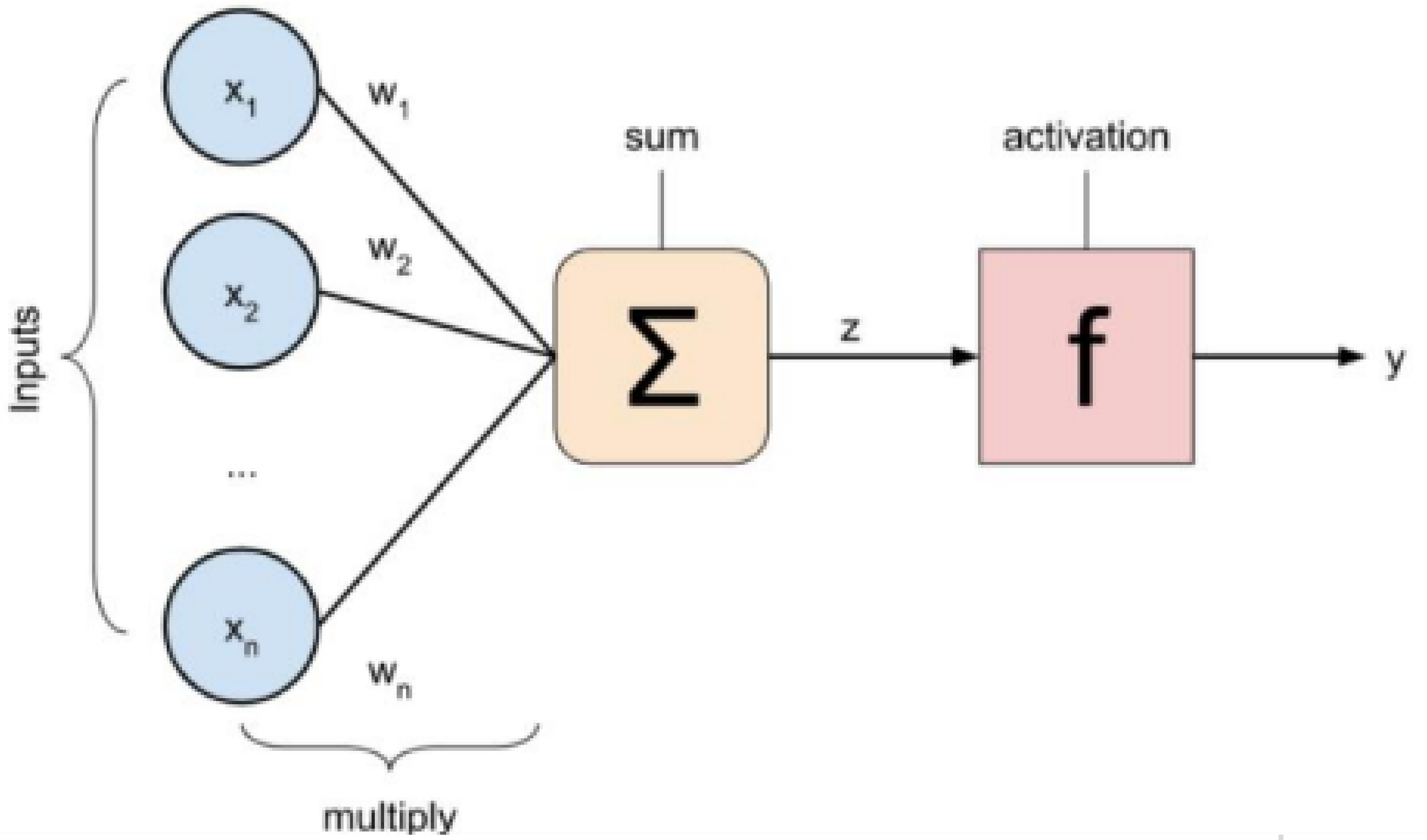
$$z = \sum_{i=1}^{n} w_i x_i + b$$

• Activation function, introduces non-linearity, allowing the network to learn complex patterns.

Sigmoid: $f(z) = \frac{1}{1+e^{-z}}$ → outputs between 0 and 1

Tanh: $f(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$ → outputs between -1 and 1

ReLU: $f(z) = \max(0, z)$ → outputs z if positive, else 0



Neural Network

5. What are the key reasons behind the huge success of deep learning algorithms?

- Availability of large datasets for training.
- Increased computational power, such as GPUs and TPUs.
- Deep architectures that learn hierarchical features automatically.

Neural Network

5. What are the key reasons behind the huge success of deep learning algorithms?

- Regularization techniques like dropout and batch normalization to prevent overfitting.
- Advanced activation functions such as ReLU that allow deeper networks to train effectively.
- Transfer learning with pretrained models for faster and more efficient training.
- Improved optimization algorithms such as Adam and RMSprop.
- Support for parallel processing in modern frameworks.

Neural Network

6. Which component is most important for enabling neural networks to learn nonlinear patterns? What are the different types of activation functions, and how does the Rectified Linear Unit (ReLU) work?

Activation functions are the important component for enabling neural networks to learn nonlinear patterns.

- Sigmoid: Maps input to [0,1], commonly used in binary classification.
- Tanh: Maps input to [-1,1], zero-centered.
- ReLU: Outputs the input if positive, otherwise 0.

Neural Network

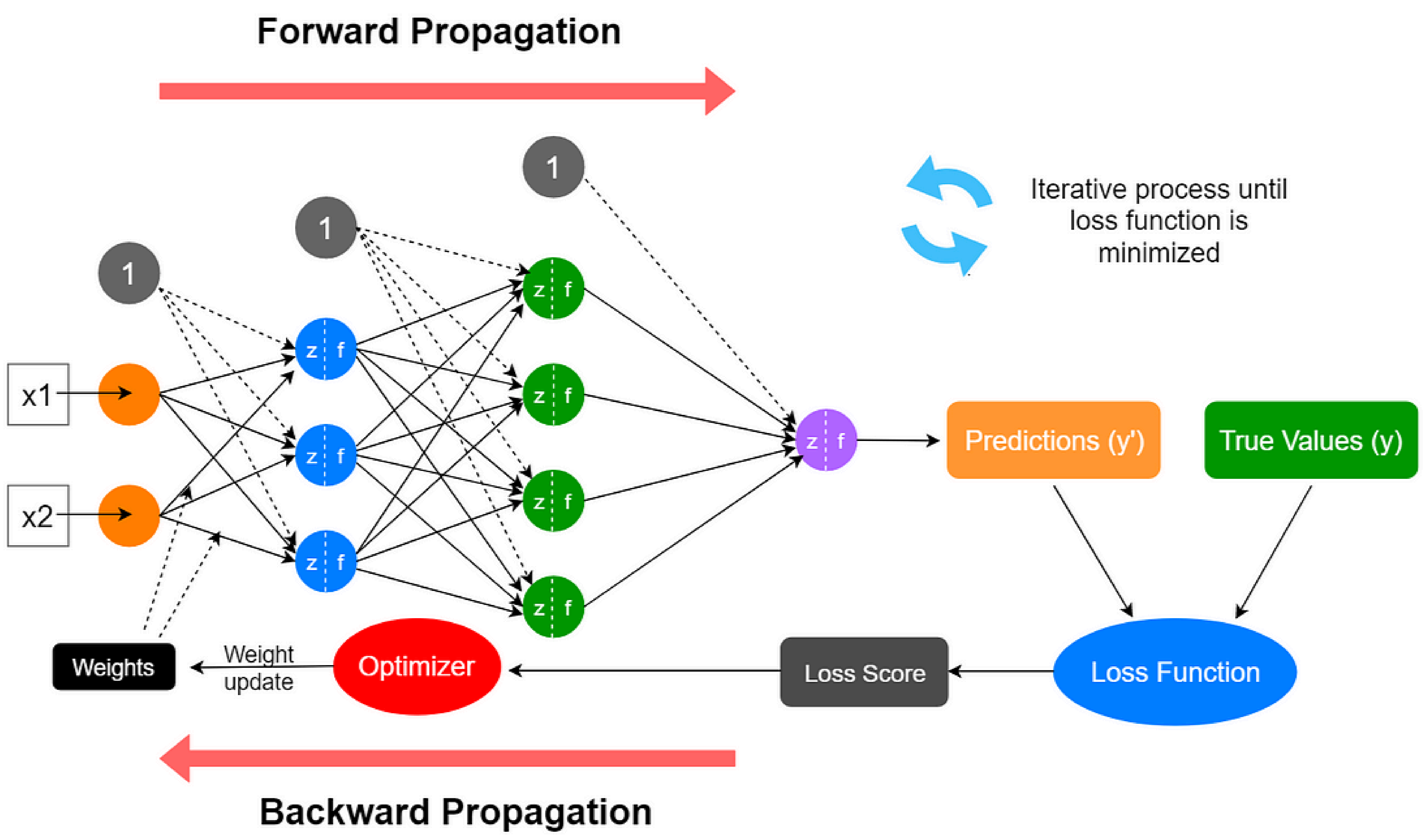
Activation Function	Output Range	Strengths	Weaknesses	Common Use Cases
Sigmoid	0 → 1	Interpretable as probability; good for binary classification	Saturates at extremes → vanishing gradients	Output layer for binary classification (e.g., spam detection)
Tanh	-1 → 1	Centered around 0; supports positive & negative outputs	Also suffers vanishing gradients at extremes	Older RNNs, NLP tasks needing ± outputs
ReLU	0 → ∞	Simple, efficient, avoids vanishing gradient, works well for deep nets	Can “die” if inputs are always negative (Dead ReLU problem)	Hidden layers in CNNs and modern deep networks (vision, NLP)

Neural Network

7. How does a neural network learn weights, and what are forward propagation and backward propagation?

- Forward Propagation: Passes input through the network to produce output.
- Backward Propagation: Computes the error between predicted and actual outputs and updates the weights to minimize the loss.
- In training, the network gradually adjusts its weights using gradient descent, so that its predictions become closer and closer to the true answers.

Neural Network



Neural Network

8. What do epoch, weights, bias, learning rate, and loss function mean in the context of neural networks?

- Epoch: One complete pass through the training dataset.
- Weights: Parameters controlling neuron connections, learned during training.
- Bias: Allows neurons to shift the activation function and provides flexibility. -In a neuron: with a bias, the activation function’s curve can shift left or right, instead of being locked at the origin.
- Learning Rate: Determines the size of weight updates during training.
- Loss Function: Measures prediction errors and guides the network to minimize them.

Neural Network

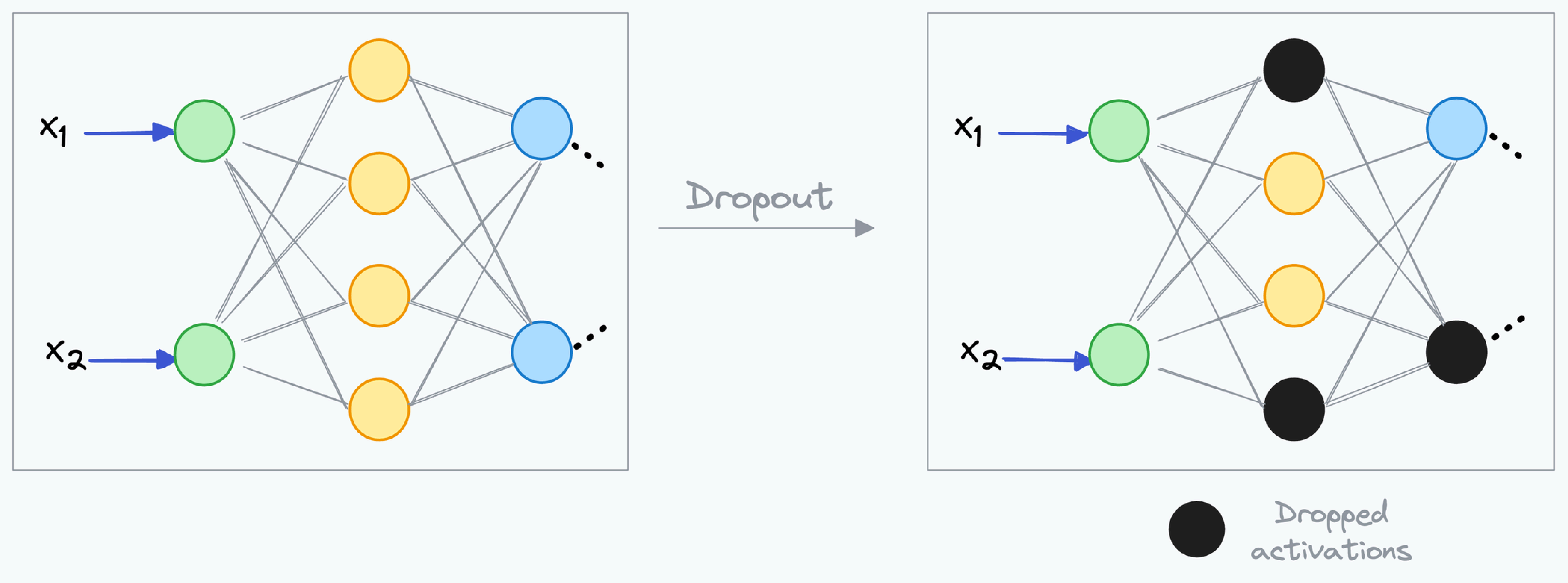
9. How can overfitting be avoided in a neural network, and what is dropout?

- **Overfitting: When a network performs well on training data but poorly on unseen data.**
- **Dropout: Randomly deactivates neurons during training to encourage generalization.**

Dropout is a clever trick: during training, some neurons are randomly “switched off” in each round. This forces the network to not rely too heavily on specific neurons, encouraging it to spread out learning and generalize better. At test time, all neurons are active again.

Neural Network

9. How can overfitting be avoided in a neural network, and what is dropout?



Neural Network

9. How can overfitting be avoided in a neural network, and what is dropout?

Other ways to fight overfitting include:

- **Data augmentation: make training data more diverse (e.g., flip or rotate images).**
- **Early stopping: stop training before the model starts memorizing.**
- **Weight regularization (L1/L2): add penalties to discourage overly complex weights.**
- **Reducing model complexity: fewer layers or parameters.**
- **Cross-validation: test on different data splits to check generalization.**

Neural Network

10. Play around with the TensorFlow Playground, create your neural networks and test it on the Circle and Gaussian datasets.

- Open TensorFlow Playground (https://playground.tensorflow.org/)
- Set the network to have two input features.
- Add two hidden layers, each with five neurons. • Select the Circle and Gaussian datasets.
- Adjust learning rate, activation functions, and optionally add noise to observe learning behavior.

Neural Network

11. What are some applications where neural networks have been successfully used?

• Computer Vision: Image classification, human pose estimation, gesture recognition.
• Natural Language Processing: Sentiment analysis, translation, text generation.
• Speech Recognition: Voice transcription and virtual assistants.
• Autonomous Vehicles: Object detection, lane tracking, path planning.
• Medical Applications: Tumor detection, disease prediction, drug discovery.
• Recommendation Systems: Movies, products, and content suggestions.
• Gaming: Chess and Go (e.g., AlphaZero, AlphaGo). • Finance: Fraud detection, stock prediction, credit scoring. • Other: Anomaly detection, deepfake creation, emotion recognition, environmental monitoring.