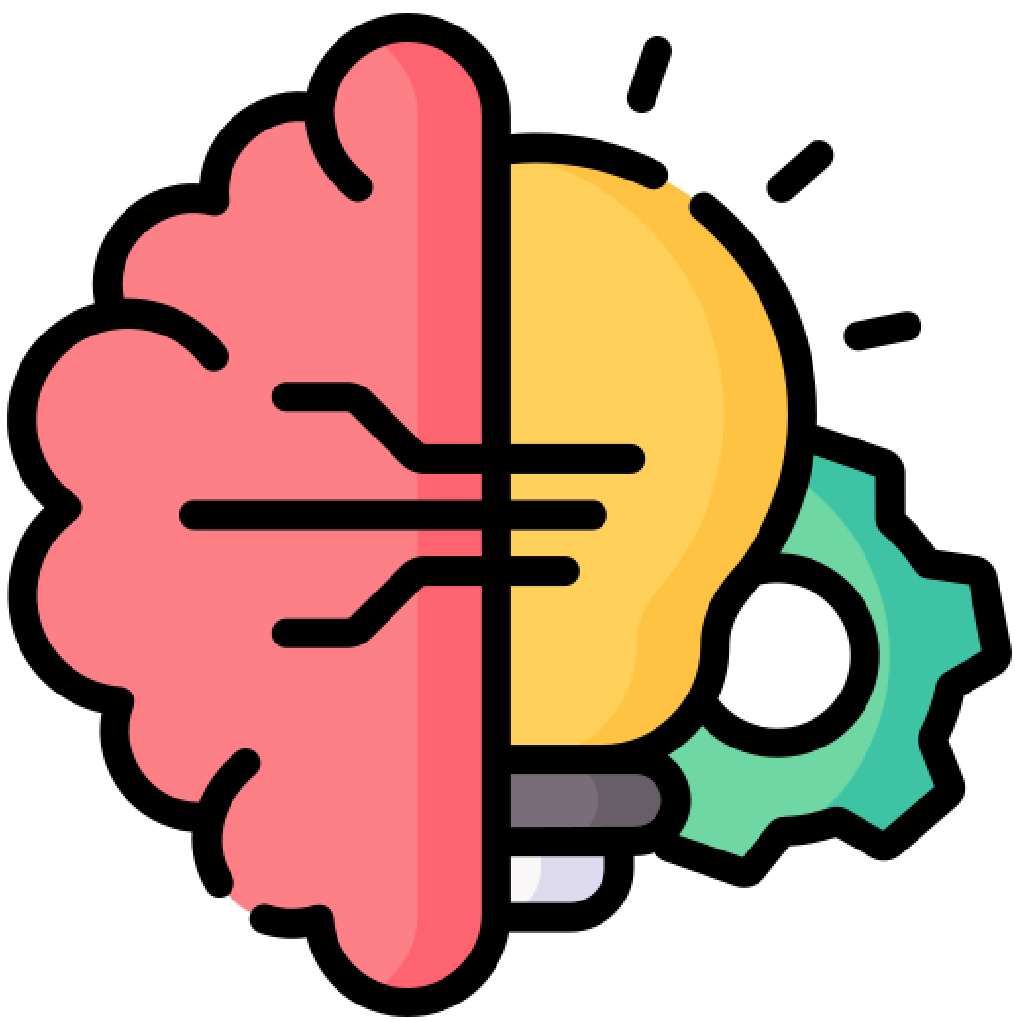
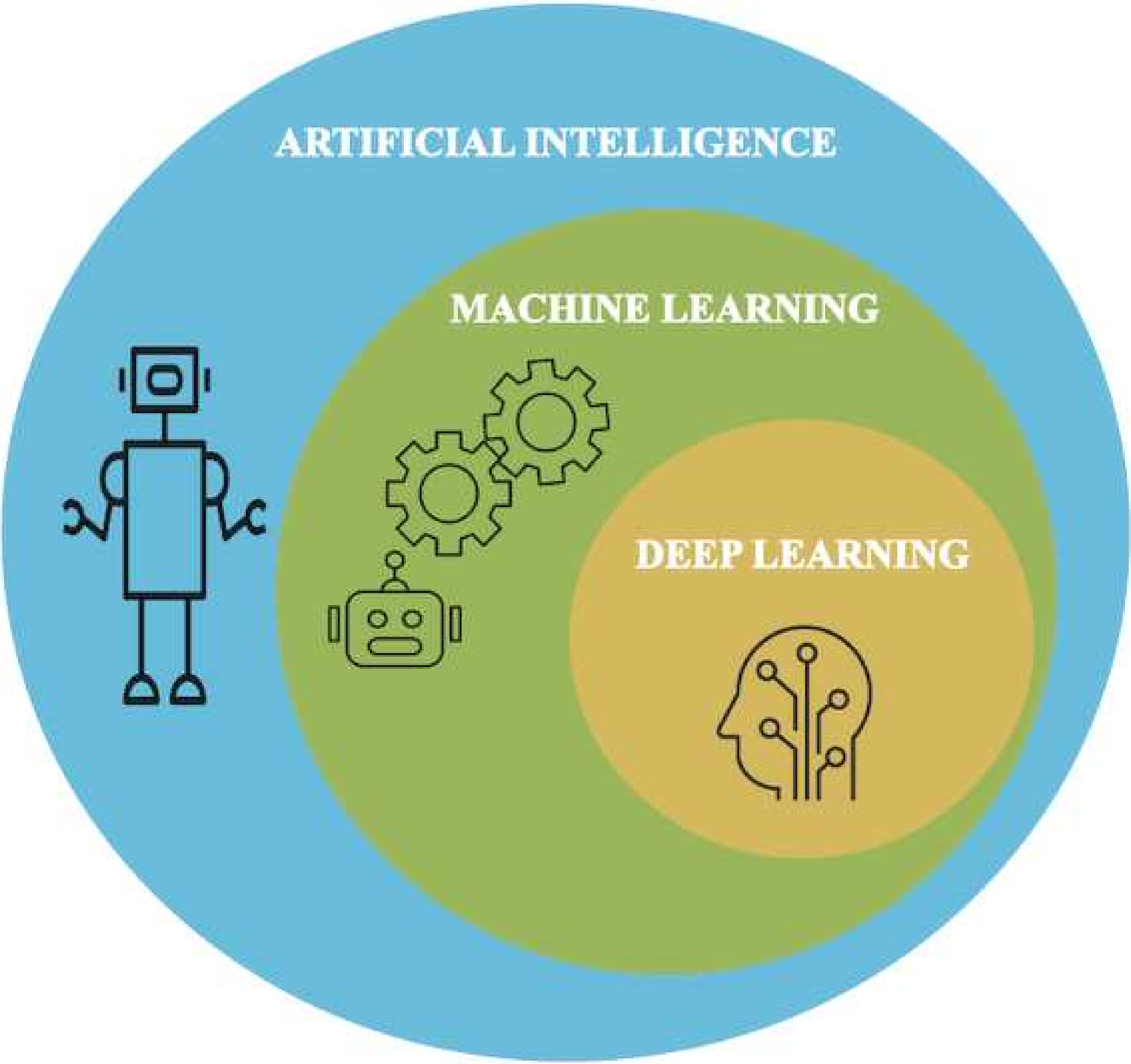
Deep Learning & Neural Networks



DATARANCH.org
VISUALIZE | ANALYZE | CAPITALIZE



DATARANCH.org
VISUALIZE | ANALYZE | CAPITALIZE

What is Deep Learning?

Deep learning is a subset of machine learning, which is essentially a neural network with three or more layers.

A neural network is a computational model inspired by the structure and functionality of the human brain. It consists of interconnected nodes, called neurons, which work together to process and analyze information.

While a neural network with a single layer can still make approximate predictions, additional hidden layers can help to increase the accuracy.

Key Terms

Neural Network: A computational model inspired by the structure and functionality of the human brain, consisting of interconnected nodes (neurons) that process and analyze information.

Deep Neural Network: A neural network with multiple hidden layers between the input and output layers, allowing for the learning of hierarchical representations.

Neuron: The basic building block of a neural network, also known as a perceptron. It receives input signals, performs computations on them, and produces an output signal.

Key Terms

Input Layer: The initial layer of a neural network that receives the raw input data. Each neuron in the input layer represents a feature or attribute of the input.

Hidden Layer: Layers in a neural network that sit between the input and output layers. They perform computations and transformations to learn hierarchical representations of the data.

Output Layer: The final layer of a neural network that produces the network's output or prediction. The number of neurons in the output layer depends on the nature of the task.

Key Terms

Activation Function: A function applied to the output of a neuron, introducing non-linearity and enabling the network to model complex relationships in the data.

Weight: A parameter associated with each connection between neurons in a neural network. It determines the strength or importance of the connection and is adjusted during training to optimize the network's performance.

Key Terms

Bias: An additional parameter in a neuron that allows the network to shift the activation function's output. It helps the network learn and represent patterns that may not necessarily pass through the origin.

Loss Function: A function that measures the discrepancy between the network's predictions and the true values in the training data. It quantifies the network's performance and is used to guide the training process.

Key Terms

Forward Propagation: The process of passing input data through the network from the input layer to the output layer. It involves applying the weights and activation functions of each neuron to generate predictions.

Backpropagation: A training algorithm for deep neural networks that calculates the gradient of the network's error with respect to the weights and updates them accordingly, propagating the error back through the layers.

Key Terms

Loss Function: A function that measures the discrepancy between the network's predictions and the true values in the training data. It quantifies the network's performance and is used to guide the training process.

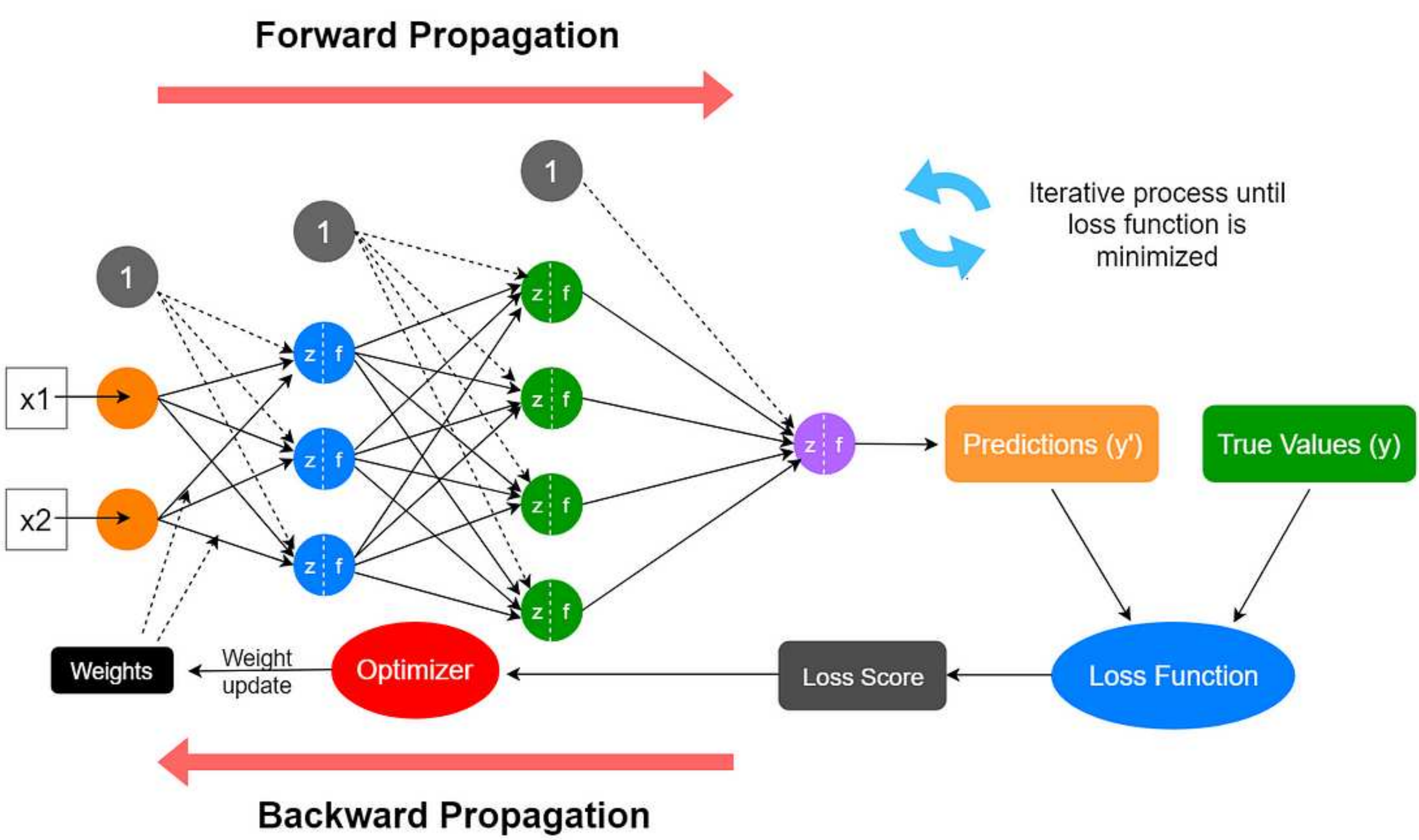
Learning Rate: A hyperparameter in the training process that determines the step size at each iteration of gradient descent. It controls the magnitude of weight and bias updates and affects the convergence and speed of training.

Key Terms

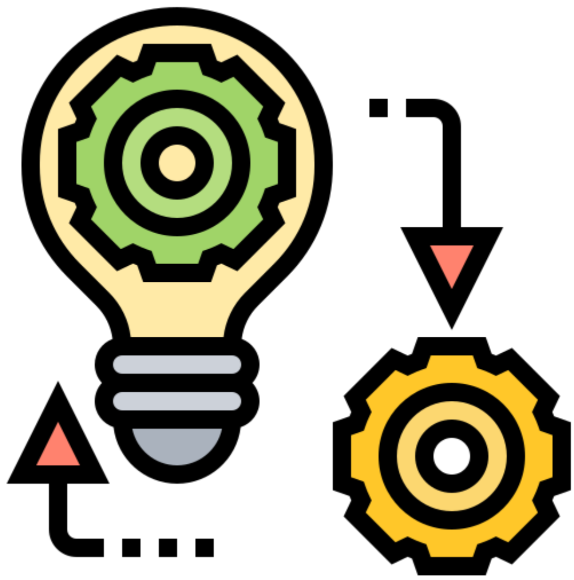
Recurrent Neural Network (RNN): A type of deep neural network designed to handle sequential data, where the output of a neuron is fed back as input to the same neuron in the next time step, allowing the network to maintain memory of past inputs.

Convolutional Neural Network (CNN): A specialized type of deep neural network commonly used for image and video processing, featuring convolutional layers that capture local patterns and spatial dependencies.

Example

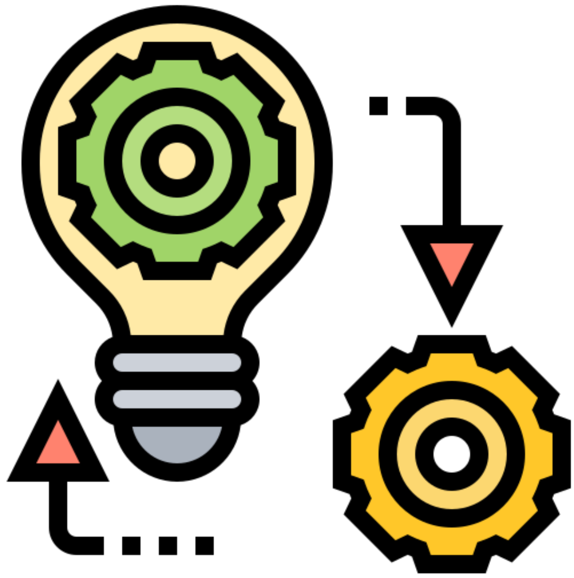

Forward Propagation
Iterative process until
loss function is
minimized
1
1
1
x1
x2
z f
Predictions (y')
True Values (y)
Weights
Weight
update
Optimizer
Loss Score
Loss Function
Backward Propagation

How Neural Network Works?



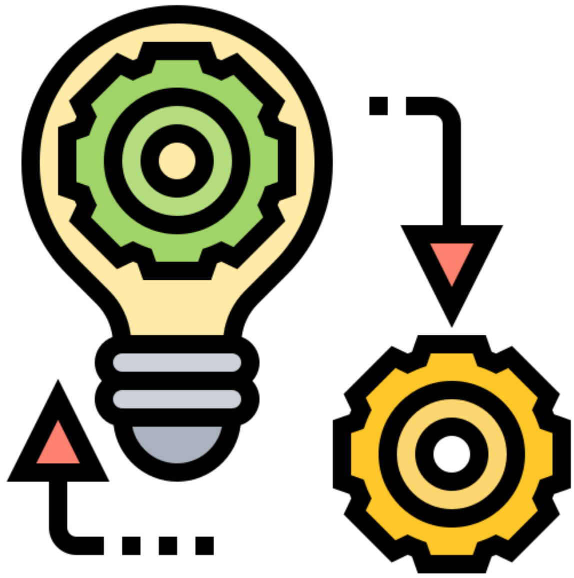
Neuron Activation: Each neuron in a neural network receives input from the previous layer or the input data itself. The input is multiplied by a weight associated with the connection between the neurons, and the products are summed. The neuron then applies an activation function to the summed value, which introduces non-linearity into the network. This transformed value becomes the output of the neuron and is passed on to the next layer.

How Neural Network Works?



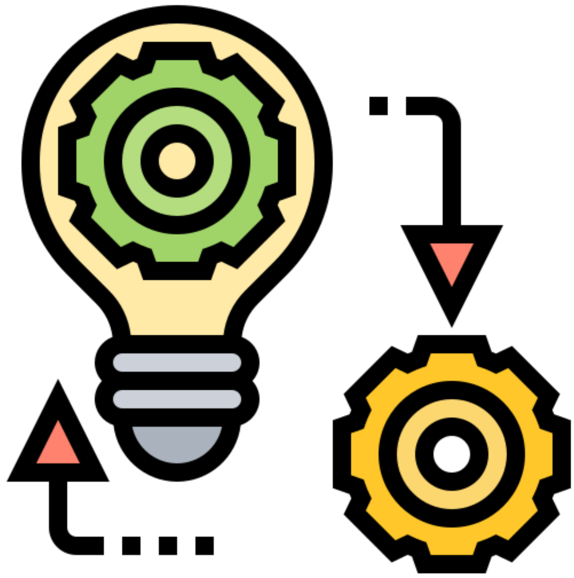
Step 1. Feedforward Process: The process of passing input data through the neural network is called feedforward. During feedforward, the input data is propagated through the network, layer by layer, starting from the input layer. Each neuron in a layer receives input from the previous layer and produces an output by applying the activation function. This process continues until the output layer is reached, and the final predictions or decisions are made.

How Neural Network Works?



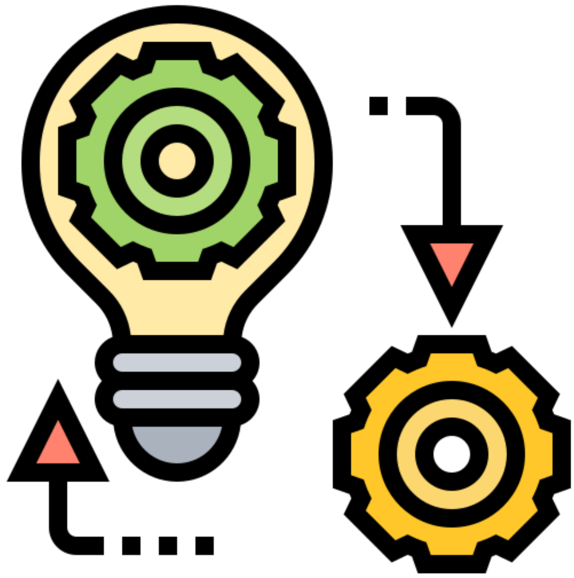
Step 2. Weights and Biases: The connections between neurons in a neural network are represented by weights. These weights determine the strength of the connection and control the impact of the input on the neuron's output. Additionally, each neuron typically has a bias term, which acts as an offset to the weighted sum of inputs. The weights and biases are learned during the training process, allowing the network to adjust its parameters to improve its predictions.

How Neural Network Works?



Step 3. Training with Backpropagation: Neural networks are trained using a process called backpropagation, which is based on the chain rule of calculus. Backpropagation calculates the gradients of the network's performance or loss function with respect to the weights and biases. The gradients indicate the direction and magnitude of the adjustments required to minimize the difference between the predicted outputs and the true labels in the training data.

How Neural Network Works?



Step 4. Model Evaluation: Once the neural network is trained, it can be evaluated on new, unseen data to assess its performance. Evaluation metrics depend on the task at hand, such as accuracy, precision, recall, F1 score, or mean squared error.

Advantages of Neural Networks

Ability to Learn Complex Patterns: Neural networks have the capability to learn and model highly complex patterns and relationships in data. They can automatically extract relevant features and representations from raw or high-dimensional input, enabling them to handle intricate and non-linear data patterns that may be challenging for traditional machine learning algorithms.

Adaptability and Generalization: Neural networks are capable of generalizing from training data to unseen examples. Once trained, they can make accurate predictions or decisions on new data that exhibits similar patterns to what they were trained on. This ability to generalize allows neural networks to handle a wide range of inputs and adapt to different scenarios, making them highly flexible models.

Advantages of Neural Networks

End-to-End Learning: Neural networks have the potential to perform end-to-end learning, where the entire learning process, from input to output, is captured within a single model. This eliminates the need for hand-engineering or manual feature extraction, as the neural network can automatically learn and discover relevant features directly from the raw input data.

Robustness to Noise: Neural networks can handle noisy or incomplete data to a certain extent. Through the learning process, they can learn to filter out irrelevant or noisy information and focus on the relevant features for making predictions or decisions. This robustness to noise allows neural networks to handle real-world data that may contain errors or uncertainties.

Advantages of Neural Networks

Parallel Processing and Scalability: Neural networks can be highly parallelized, meaning they can take advantage of modern computing architectures, such as GPUs or distributed systems, to process data in parallel. This enables efficient training and inference, allowing neural networks to scale and handle large datasets.

Feature Learning and Representation: Neural networks excel at learning hierarchical representations of data. Each layer of a deep neural network can capture increasingly abstract and complex features, allowing the network to automatically learn and encode multiple levels of abstraction in the data. This feature makes neural networks particularly effective for computer vision, natural language processing, and speech recognition.

Disadvantages of Neural Networks

Need for Large Amounts of Training Data: Neural networks typically require a large amount of labeled training data to perform well. Training a neural network from scratch with limited data can result in overfitting, where the model memorizes the training examples but fails to generalize to new, unseen data.

Computationally Intensive: Deep neural networks, especially those with numerous layers and parameters, can be computationally expensive to train and require substantial computational resources, including powerful GPUs or specialized hardware.

Disadvantages of Neural Networks

Difficulty in Interpretability: Neural networks are often considered black-box models, meaning they lack interpretability or transparency. Understanding the inner workings and the reasoning behind the network's predictions can be challenging. This lack of interpretability can be problematic in certain domains where explanations or justifications for decisions are required.

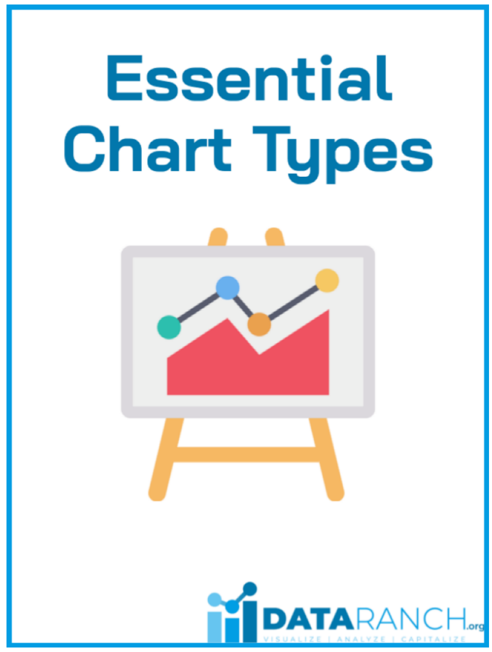
Vulnerability to Adversarial Attacks: Neural networks are susceptible to adversarial attacks, where intentionally crafted input samples can deceive the network into making incorrect predictions. These attacks exploit the network's sensitivity to small perturbations in the input data, leading to potential security risks in applications like image recognition or autonomous driving.

Disadvantages of Neural Networks

Lack of Robustness to Noisy or Out-of-Distribution Data: Neural networks may struggle to generalize well when encountering data that is significantly different from the training data distribution. They can be sensitive to noisy or out-of-distribution samples, leading to degraded performance or incorrect predictions.

Training Time and Convergence: Training deep neural networks can be a time-consuming process, particularly when dealing with large datasets and complex architectures. Additionally, convergence to an optimal solution is not guaranteed, and finding the right balance between training time and achieving good performance can be challenging.

Follow **#DataRanch** on LinkedIn for more...



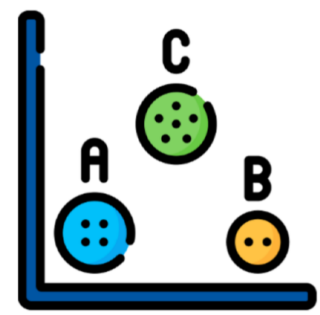
DATARANCH.org
VISUALIZE | ANALYZE | CAPITALIZE

Follow **#DataRanch** on LinkedIn for more...

What is Unsupervised Learning?

Principal Component Analysis

Clustering



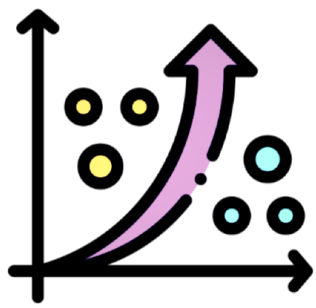
t-Distributed Stochastic Neighbour Embedding (t-SNE)

DATARANCH.org
VISUALIZE | ANALYZE | CAPITALIZE

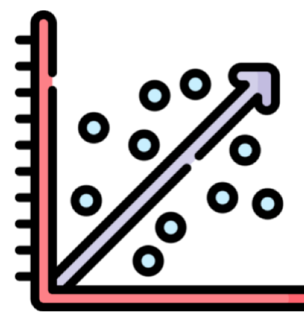
Follow **#DataRanch** on LinkedIn for more...

What is Supervised Learning?

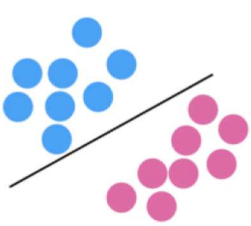
Logistic Regression



Regression Analysis

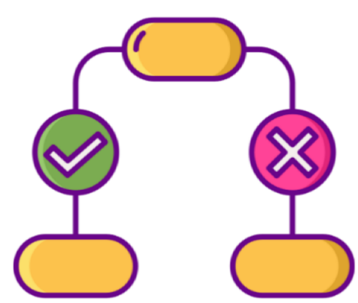


Support Vector Machine



Random Forest

Decision Trees



DATARANCH.org
VISUALIZE | ANALYZE | CAPITALIZE

DATARANCH.org

VISUALIZE | ANALYZE | CAPITALIZE

info@dataranch.org

linkedin.com/company/dataranch