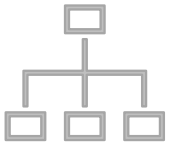
Neural Networks

Holberton

Why Neural Networks

Why Neural Networks



pattern recognition

automatic translation

stock market prediction

image recognition

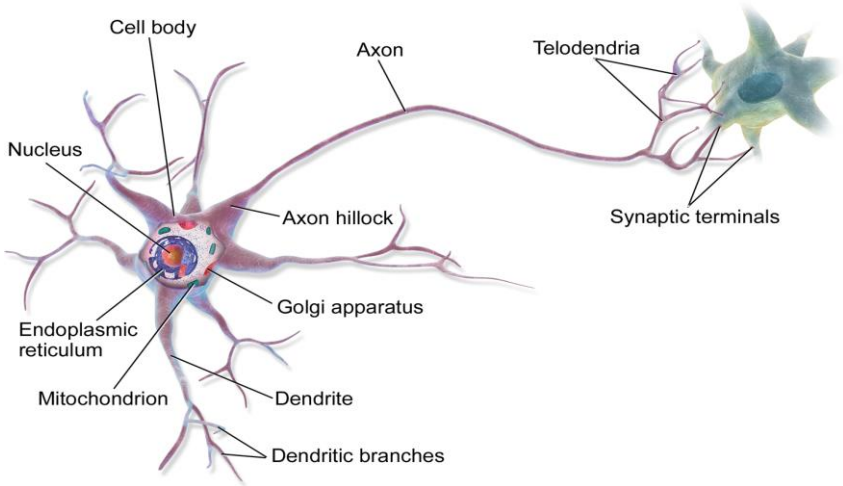
computer vision

recommendation

Introduction to (Artificial) Neural Networks

The biological brain network

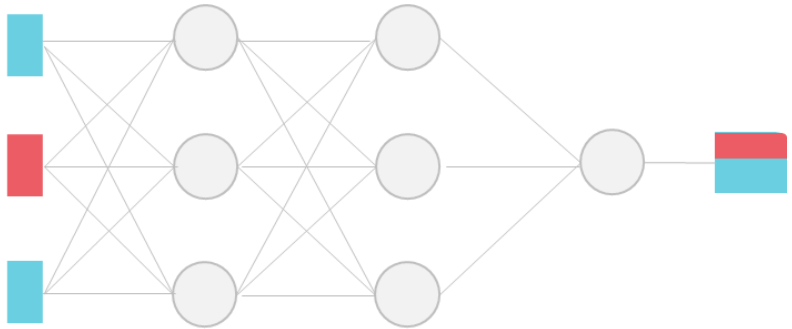
- process complex information
- the neuron is the most crucial unit
- connected neurons



Introduction to (Artificial) Neural Networks

The artificial network

- process complex information
- the neuron is the most crucial unit
- connected neurons

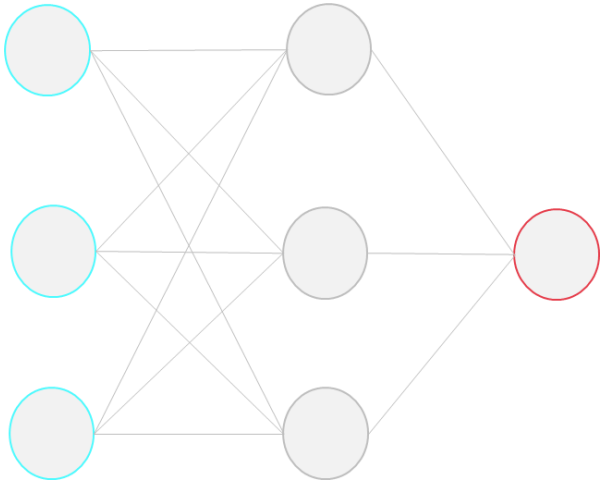


Feed-forward Neural Networks

Fundamentals of Artificial Neural Networks

Information is processed only in forward direction

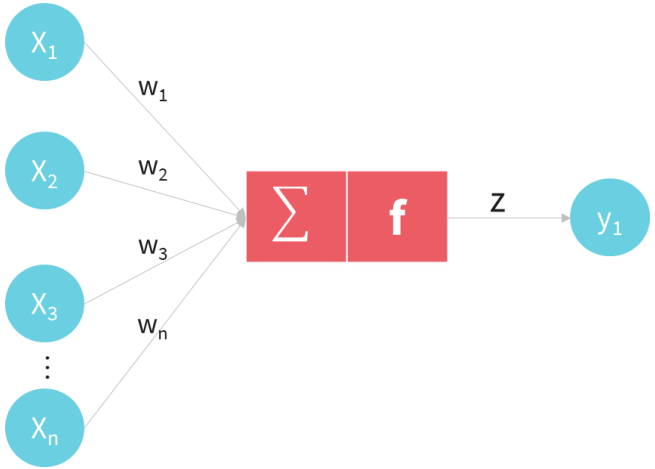
- network as a stack of layers
- weights → connection strength
- bias → adjustments within neurons



Fundamentals of Artificial Neural Networks

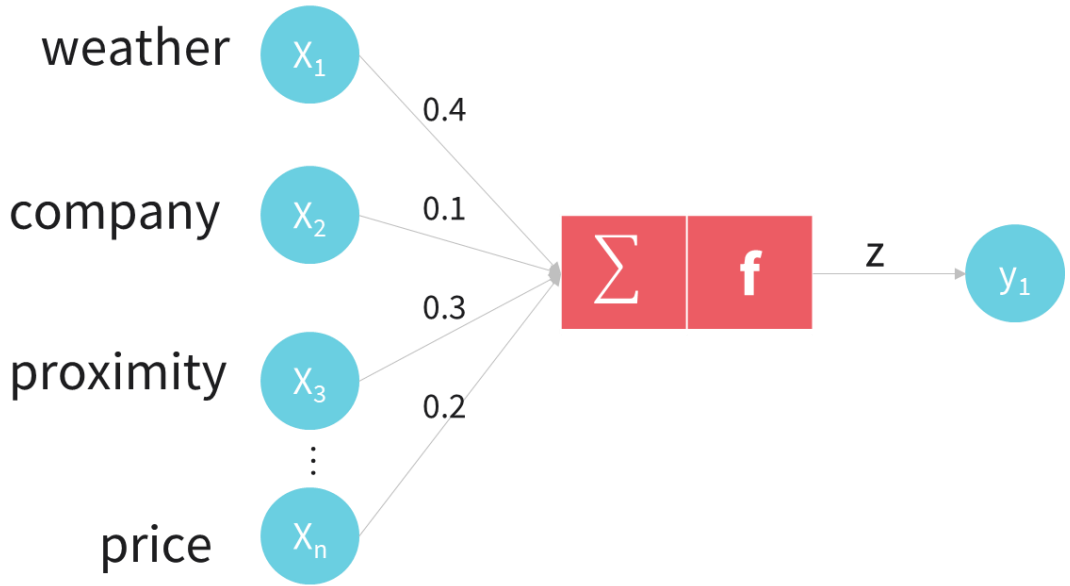
Perceptron: the most basic ANN architecture

- binary input values
- connection weights & bias
- activation function



Fundamentals of Artificial Neural Networks

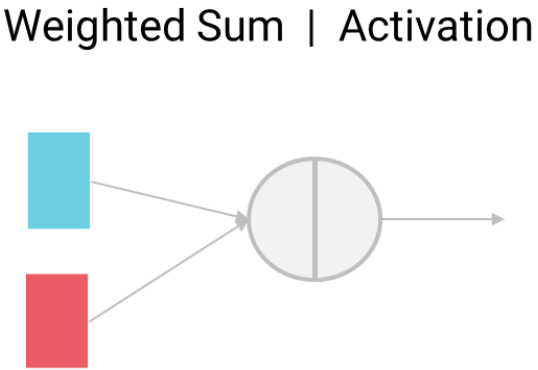
- Perceptron example: will you go to the movies?



Neural Network Architecture

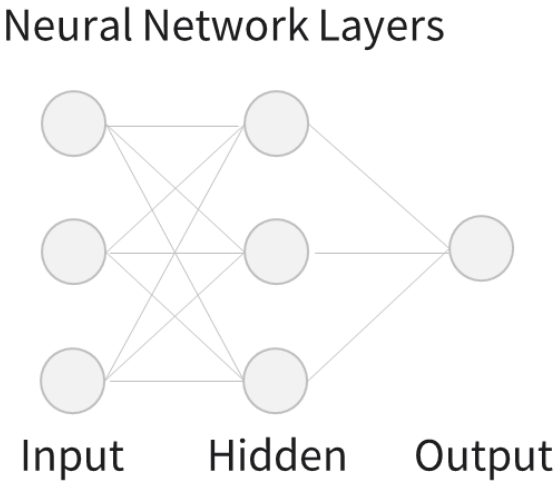
The Neuron

- fundamental unit of artificial network
- receives input and transforms it
- **activation function** decides importance of neuron in the network



The Layers

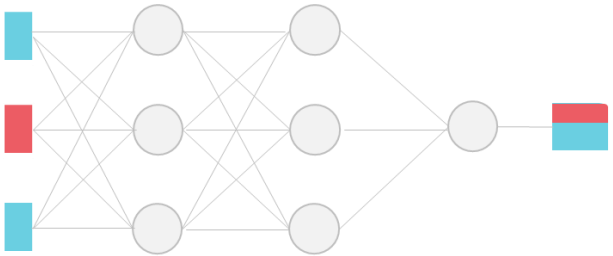
- core building blocks of neural network
- three types: input, hidden, output
- receive transform and output data



The Model

The model

combine layers into a network



Neural Network

Loss and optimization

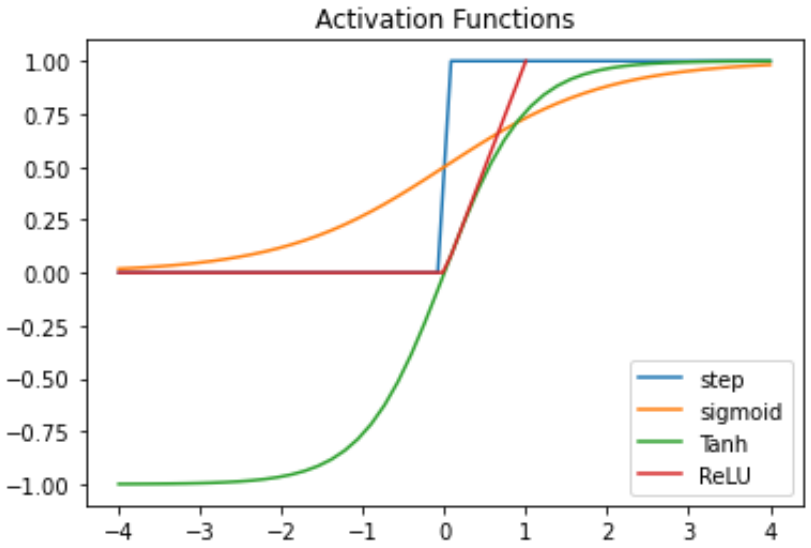
measure learning and adjust

Provide feedback

The Activation Functions

- define how inputs will be transformed to outputs

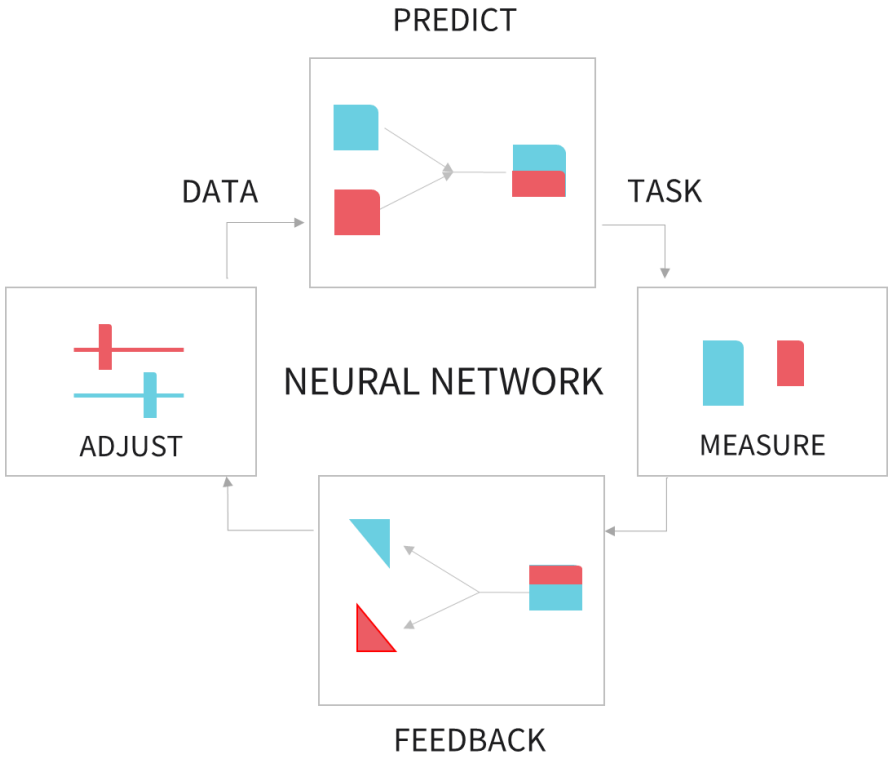
- several activation functions
  - step function
  - logistic function
  - hyperbolic tangent function
  - rectified linear unit function



Training Neural Networks

How to train a neural network

- Map features to target
- Train and test
- Adjust and repeat

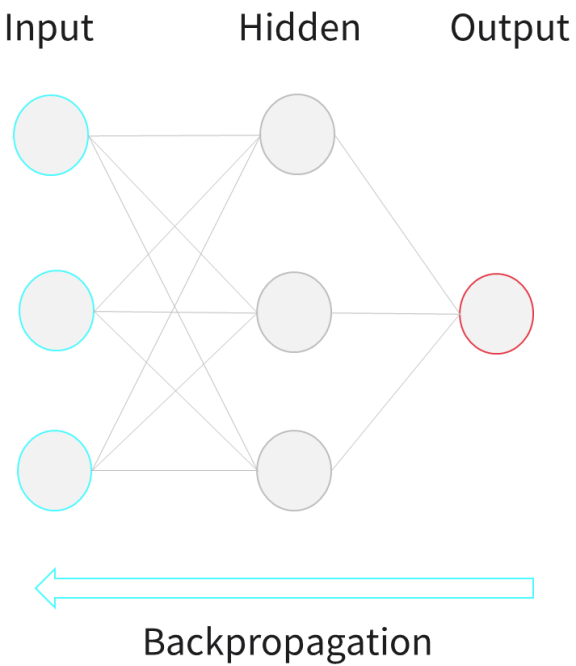


How to estimate training effectiveness

- How good is our feed-forward neural network → cost functions
  - Mean Absolute Error: $\mathrm{M}AE = \frac{1}{n} \cdot \sum_{i=1}^{n} |y_i - x_i|$
  - Mean Squared Error: $MSE = \frac{1}{n} \cdot \sum_{i=1}^{n} (y_i - \bar{y}_i)^2$
  - Cross Entropy $H(x) = \sum_{i=1}^{n} p(x) \cdot \log q(x)$

The Backpropagation Algorithm

- Backpropagation → the essence of neural network training
  - fine-tune network parameters based on error terms
  - propagates the errors from output to input nodes

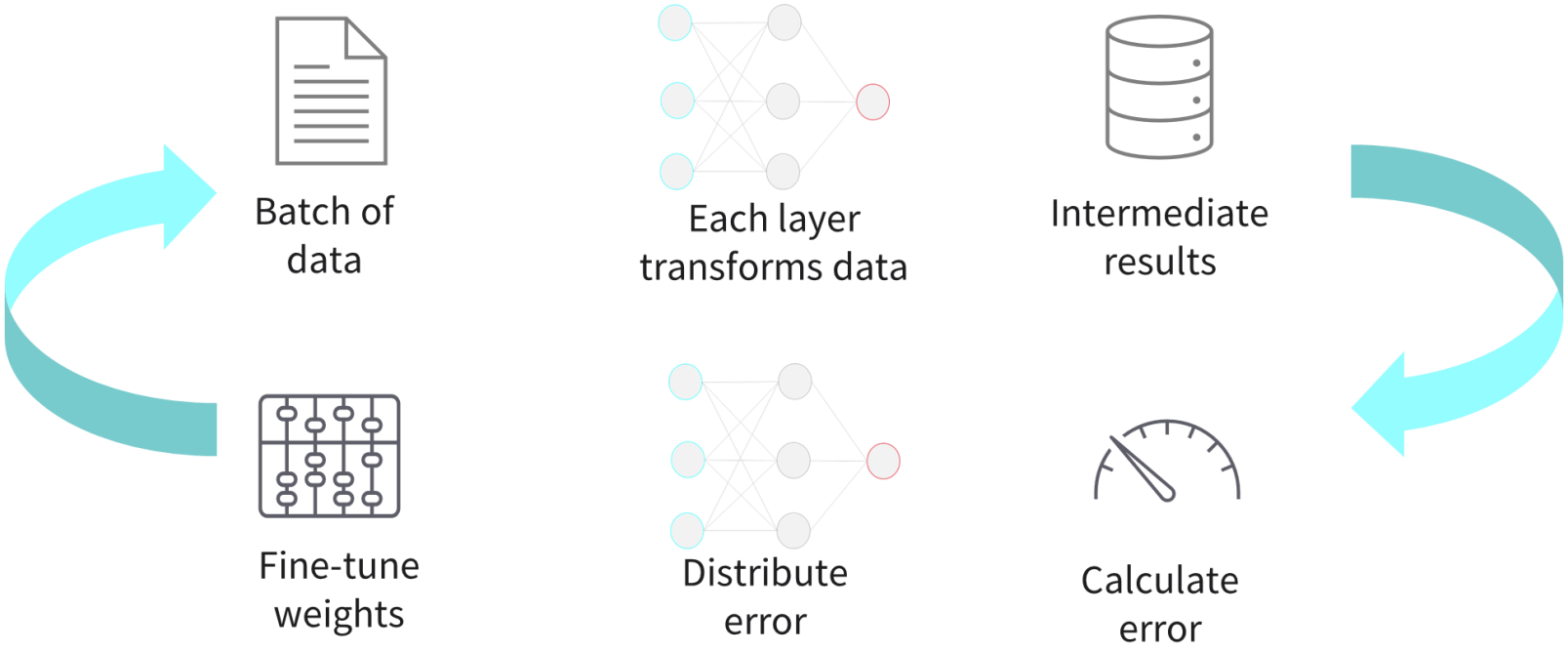


The Backpropagation Algorithm

- Forward pass: make prediction given data

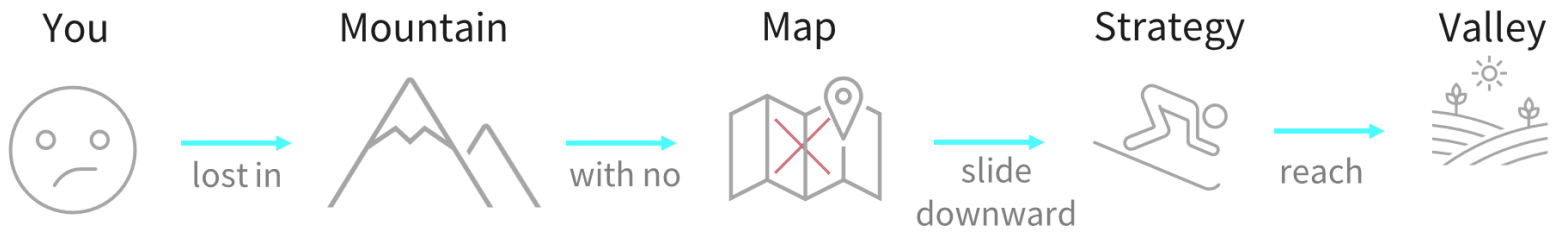
- Backward pass: adjust weights according to measured error

The Backpropagation Algorithm



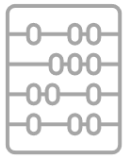
The Gradient Descent Algorithm

- Optimization technique for a wide range of problems
- Train a neural network by tweaking parameters until cost is minimized



The Gradient Descent Algorithm

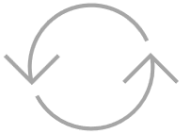
- Optimization strategy to train a neural network



1. Initialize random weights

2. Calculate gradient

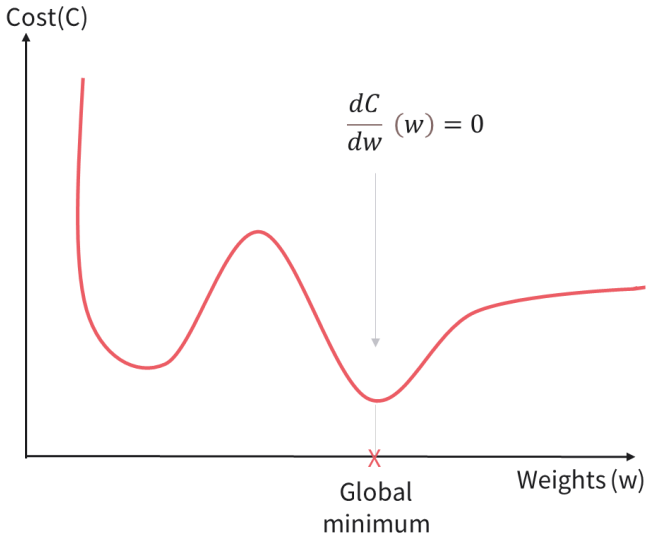
3. Update weights

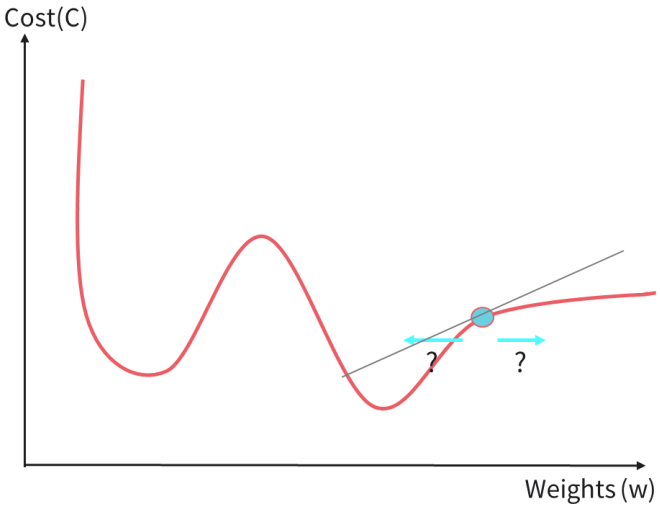


4. Repeat until cost is minimized

The Gradient Descent Algorithm

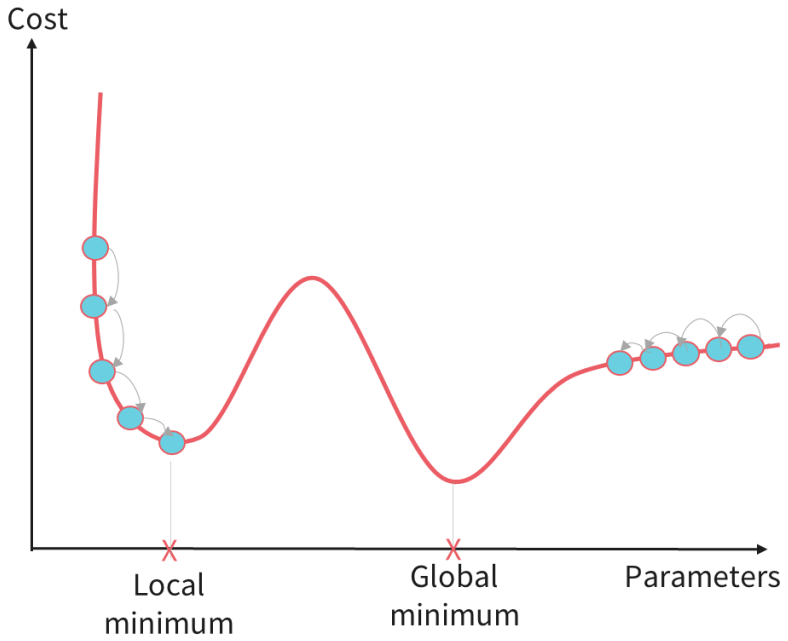
- Goal: minimize cost function





The Gradient Descent Algorithm: Pitfalls

- challenges when cost function is not regular
- gradient might become stuck or stagnant
- difficult to converge to global minimum



Any questions?