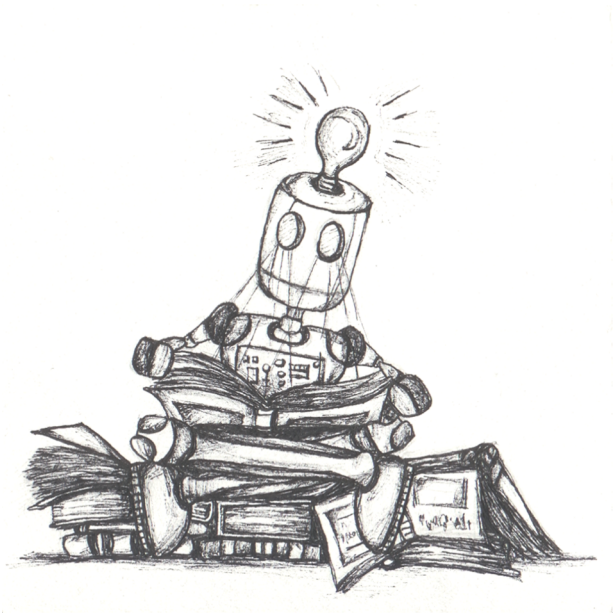
Introduction to Neural Networks

Lecture Notes: 2021, from Faraz Barzideh, Researcher UiS

What Is Machine Learning?



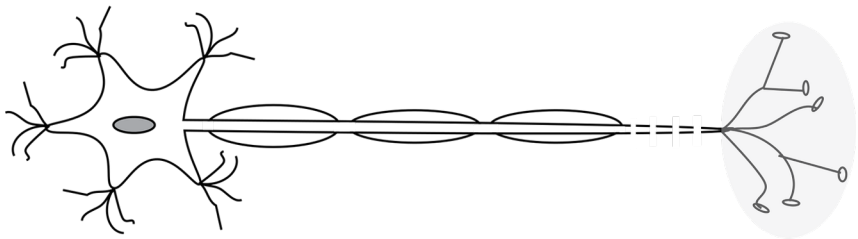
Link:https://pixabay.com/illustrations/art-artwork-robot-reading-books-3569271/

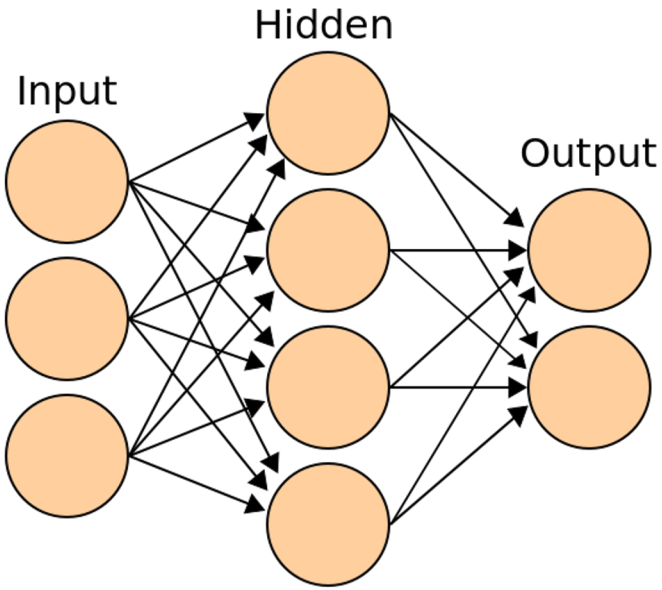
The field of **Machine learning** (**ML**) is the study of computer algorithms that can improve automatically through experience and by the use of data to imitate the way that humans learn.

What Is Learning?

Learning is the process of updating the parameters of the machine learning model so the model performs better for its task

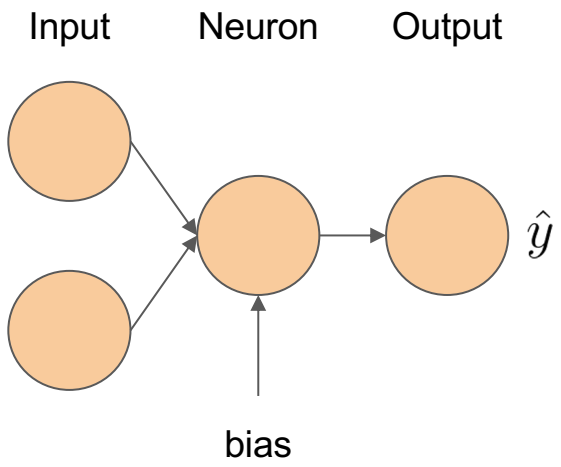
What Is An Artificial Neural Network?



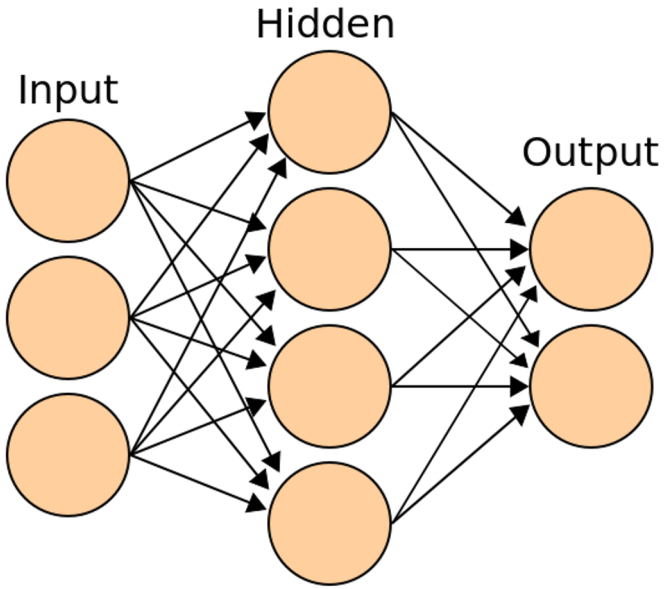


Neural Networks more accurately known as Artificial Neural Networks are inspired by the human brain, mimicking the way that biological neurons signal to one another.

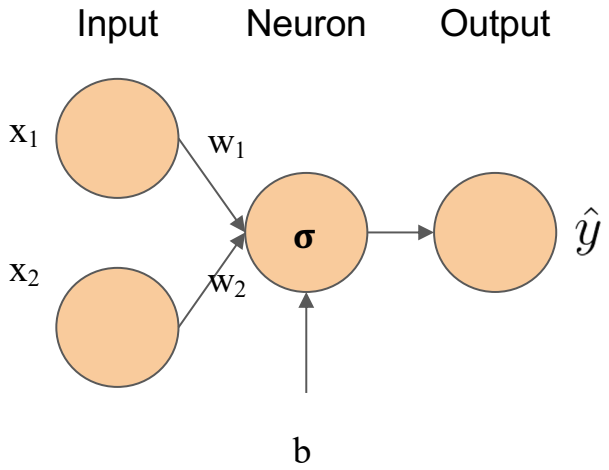
What Is A Neural Network?



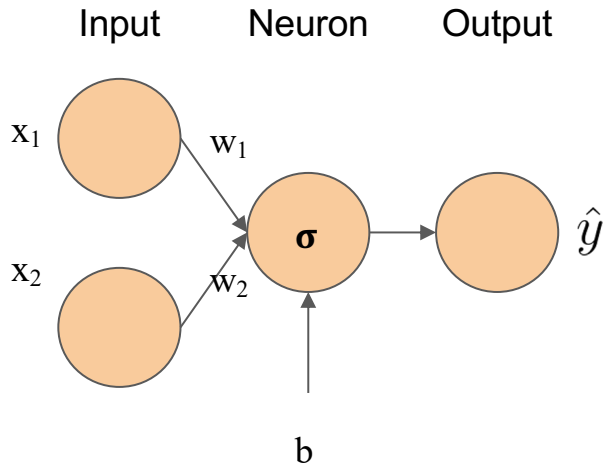
What Is A Neural Network?



What Is A Neural Network?



What Is A Neural Network?



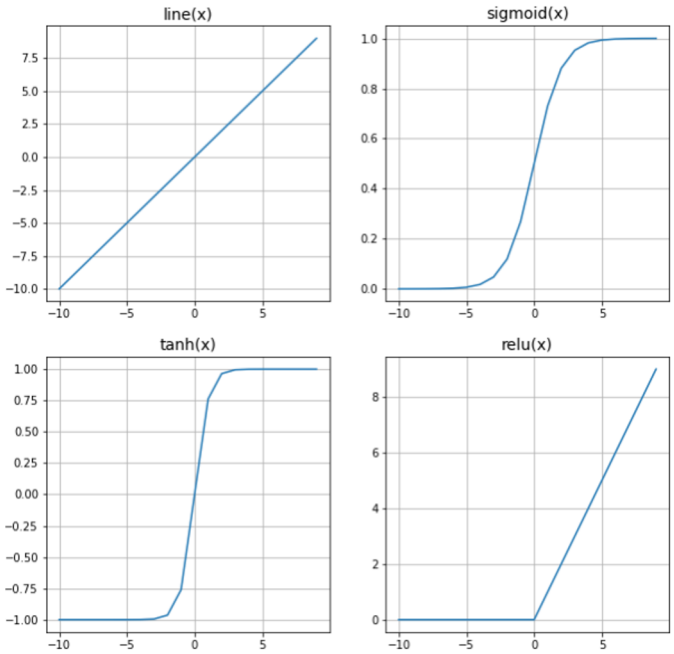
$$\hat{y} = \sigma([w_1 \quad w_2] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + b) = \sigma(\mathbf{w}^T\mathbf{x} + b) = \sigma(\sum w_i x_i + b)$$

Activation Function

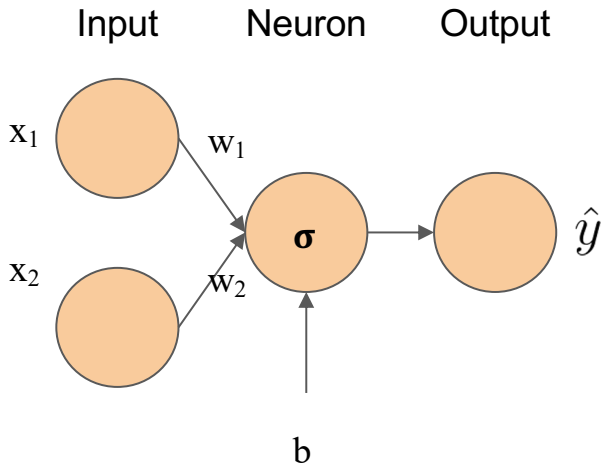
An activation function decides which neurons are active or inactive.

Types of activation function:

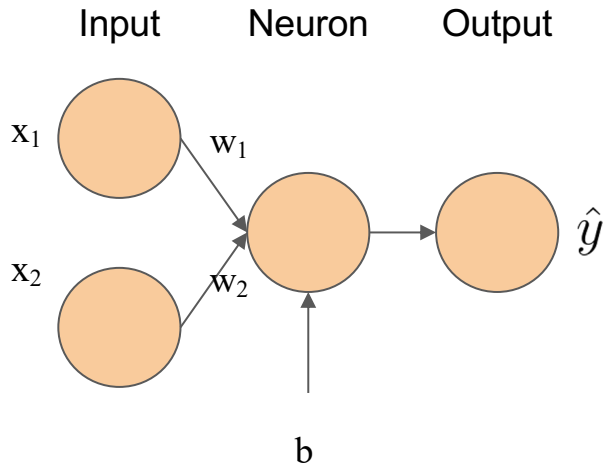
- Linear: f(x) = x
- Sigmoid: $\frac{1}{1+e^{-x}}$
- Tanh: $\frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$
- Relu: $max(0,x)$



Forward Propagation



Forward Propagation

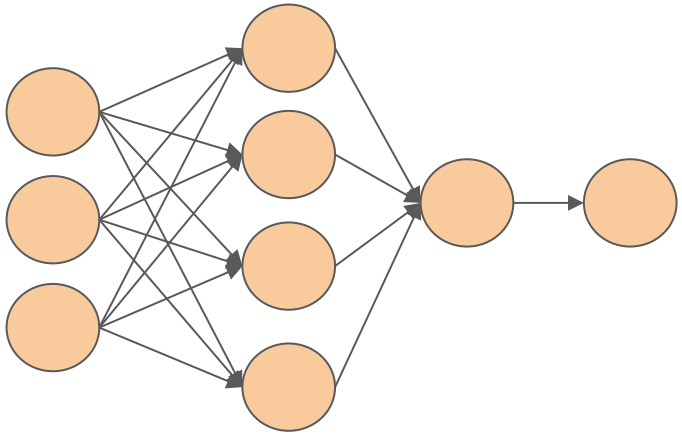


$$x_1 = 1$$
$$x_2 = 2$$
$$w_1 = 3$$
$$w_2 = 4$$
$$b = 1$$

$$\hat{y} = line(w_1 x_1 + w_2 x_2 + b) = (1 \times 3 + 2 \times 4 + 1) = 15$$

Forward Propagation

Input Hidden Layer Final layer Output



$$\mathbf{x} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \mathbf{W}_1 = \begin{bmatrix} 4 & 3 & 1 & 1 \\ 1 & 4 & 4 & 3 \\ 3 & 4 & 3 & 2 \end{bmatrix}, \mathbf{w}_2 = \begin{vmatrix} 4 \\ 2 \\ 1 \\ 3 \end{vmatrix}$$

How Does A Neural Network Learn?

A neural network learns by adjusting the weights and biases. This happens by using a cost function

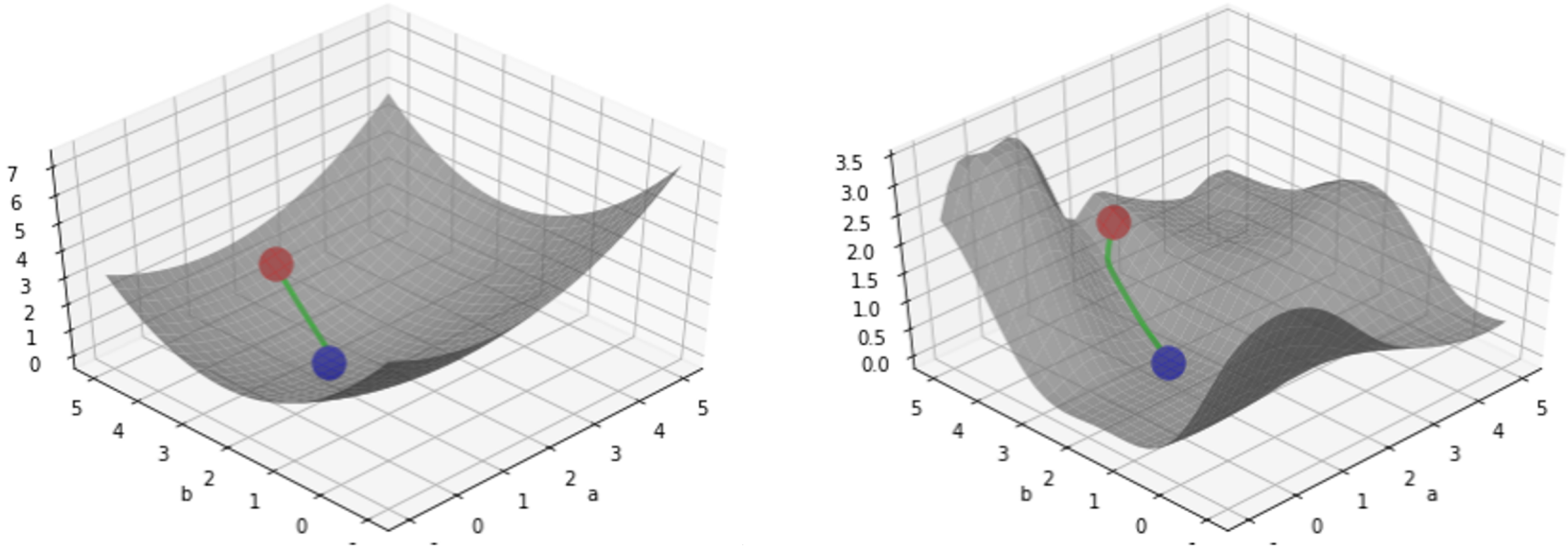
Cost Function

- The cost functions determines how good the network has learned its task.

- There are different cost functions for different applications
  - Quadratic: $\frac{1}{2}\sum_i(\hat{y}-y)^2$
  - Cross-entropy: $-\frac{1}{n}\sum_i(yln\hat{y}+(1-y)ln(1-\hat{y}))$
  - etc

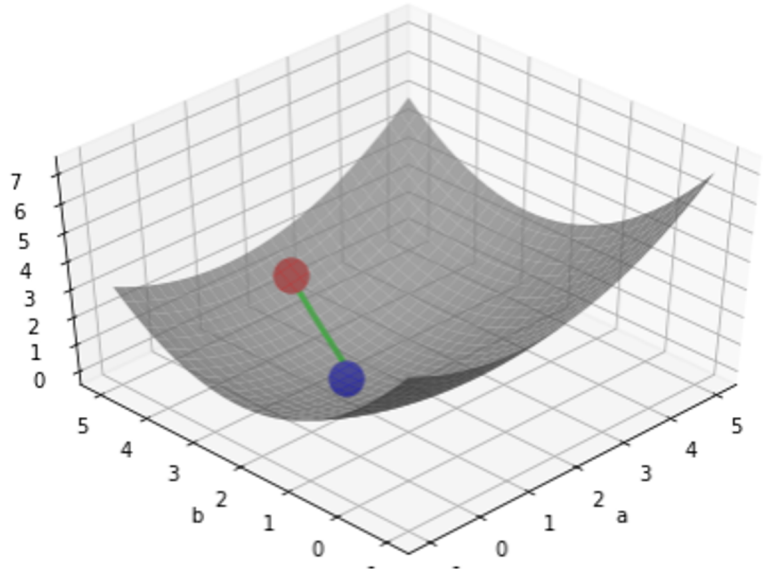
Gradient Descent

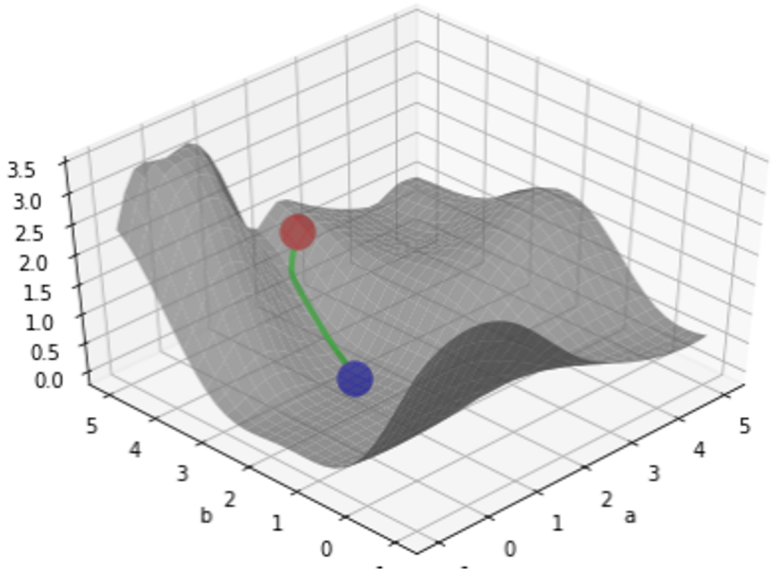
$$w_{(i)} = w_{(i-1)} - \gamma \nabla C(w_{(i-1)})$$



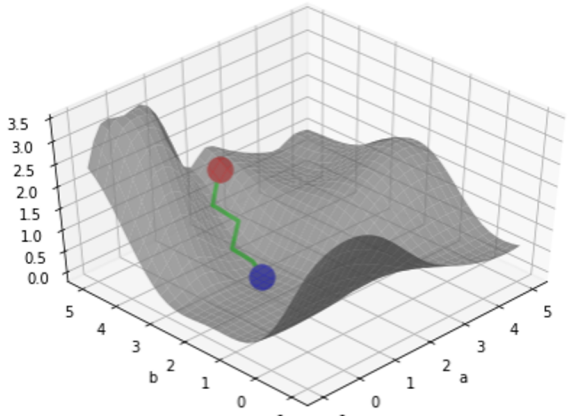
Gradient Descent

$$w_{(i)} = w_{(i-1)} - \gamma \nabla C(w_{(i-1)})$$

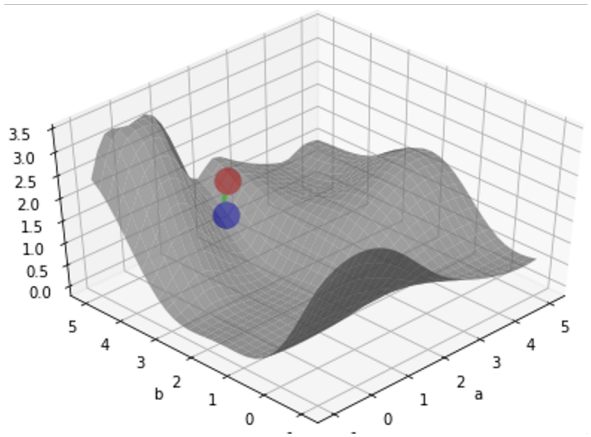




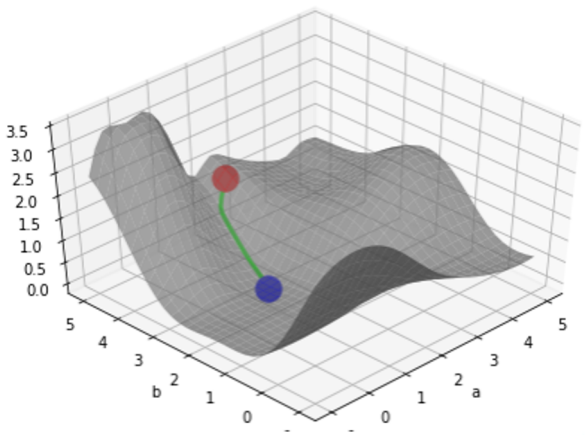
Learning Rate



LR = 0.5



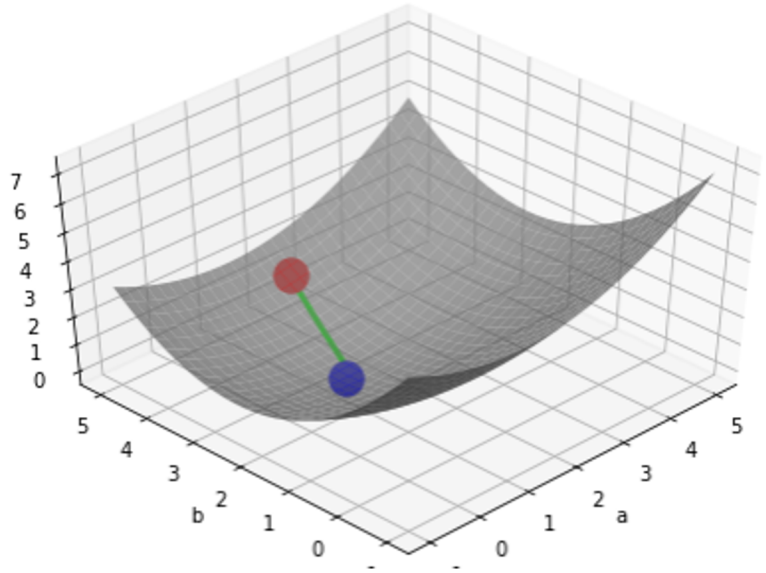
LR = 0.001

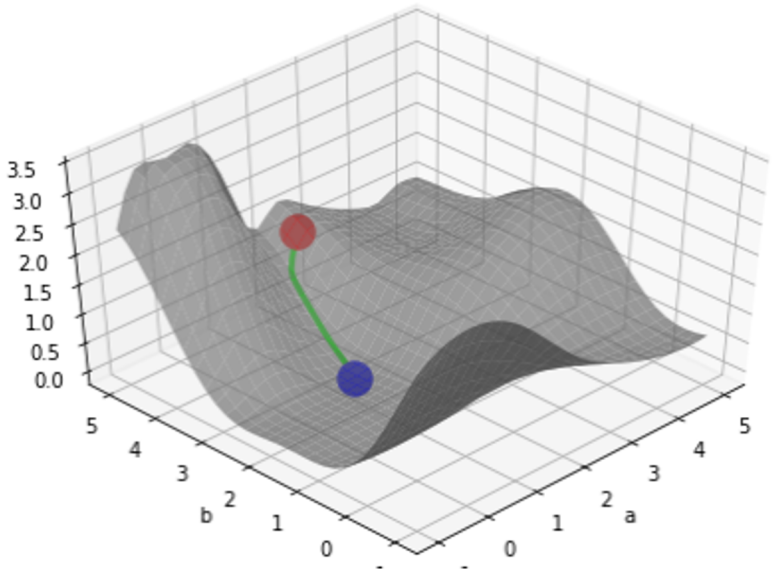


LR = 0.1

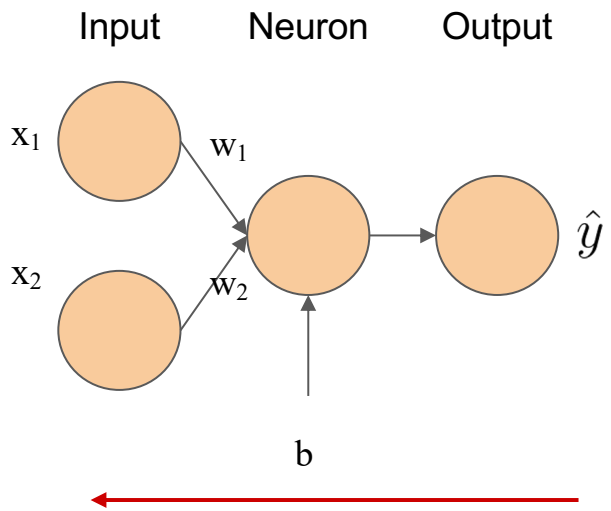
Gradient Descent

$$w_{(i)} = w_{(i-1)} - \gamma \nabla C(w_{(i-1)})$$

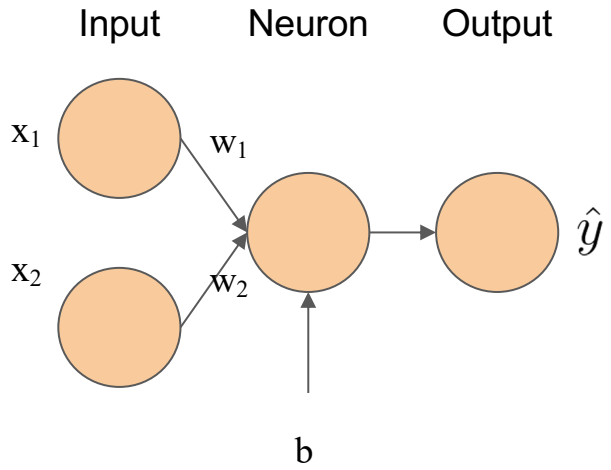




Backward Propagation



Backward Propagation



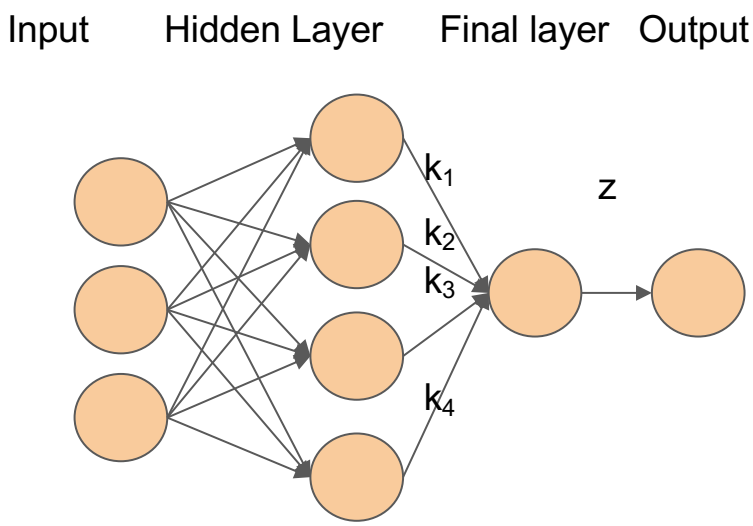
$$C(w_1, w_2, b) = \sum(\hat{y} - y)^2 = \sum(w_1x_1 + w_2x_2 + b - y)^2$$

$$\frac{\partial C(w_1, w2, b)}{\partial dw_1} = 2\sum x_1(w_1x_1 + w_2x_2 + b - y)$$

$$\frac{\partial C(w_1, w2, b)}{\partial dw_2} = 2\sum x_2(w_1x_1 + w_2x_2 + b - y)$$

$$\frac{\partial C(w_1, w2, b)}{\partial db} = 2\sum(w_1x_1 + w_2x_2 + b - y)$$

Chain Rule



$$\frac{\partial C(w,b)}{\partial w_{ij}} = \frac{\partial C(w,b)}{\partial z}\frac{\partial z}{\partial k_j}\frac{\partial k}{\partial w_{ij}}$$

References and resources:

- Deep Learning Playlists
  - DeepLearningAI
  - 3Blue1Brown
- List of cost functions
- Chain Rule
- Background image by ADDGRAPHIC, other link