Computational Intelligence

Samaneh Hosseini

Isfahan University of Technology

Outline

- Types of Learning Problems
- A simple mathematical model of a neuron
- Activation Function
  - Desirable Properties of The Activation Function
  - Common Activation Functions

Types of Learning

Types of learning problems

training data

- Supervised learning:
  Given input features, target features, and training examples, predict the value of the target features for new examples given their values on the input features.

  test data

- Unsupervised learning:
  Learning classifications when the examples do not have targets defined.

- Reinforcement Learning:
  Learning what to do based on rewards and punishments.

CQ: Supervised or Unsupervised Learning

CQ: We are given information on a user's credit card transactions. We would like to detect whether some of the transactions are fraudulent by finding some transactions that are different from the other transactions. We have no information on whether any particular transaction is fraudulent or not.

Is this a supervised or unsupervised learning problem?

(A) Supervised learning

(B) Unsupervised learning

CQ: Supervised or Unsupervised Learning

CQ: We are given information on a user's credit card transactions. We would like to detect whether some of the transactions are fraudulent by finding some transactions that are different from the other transactions. We have no information on whether any particular transaction is fraudulent or not.

Is this a supervised or unsupervised learning problem?

(A) Supervised learning

(B) Unsupervised learning

Learning complex relationships

- Image interpretation, speech recognition, and translation.
- The relationship between inputs and outputs can be extremely complex.
- How can we build a model to learn such complex relationships?

Humans can learn complex relationships well.
Can we build a model that mimics the human brain?

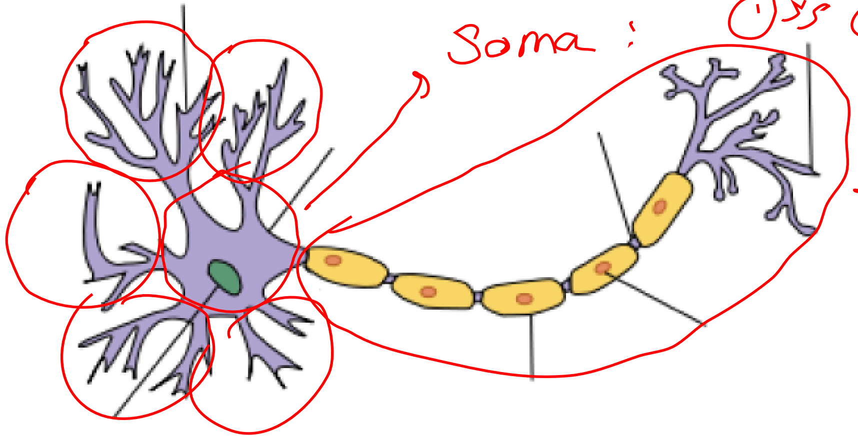
Human brains

- A brain is a set of densely connected neurons.
- Components of a neuron: dendrites, soma, axon, synapse
- Depending on the input signals, the neuron performs computations and decides to fire or not.

synapse : link between neurons

dentrites : receive input from neurons

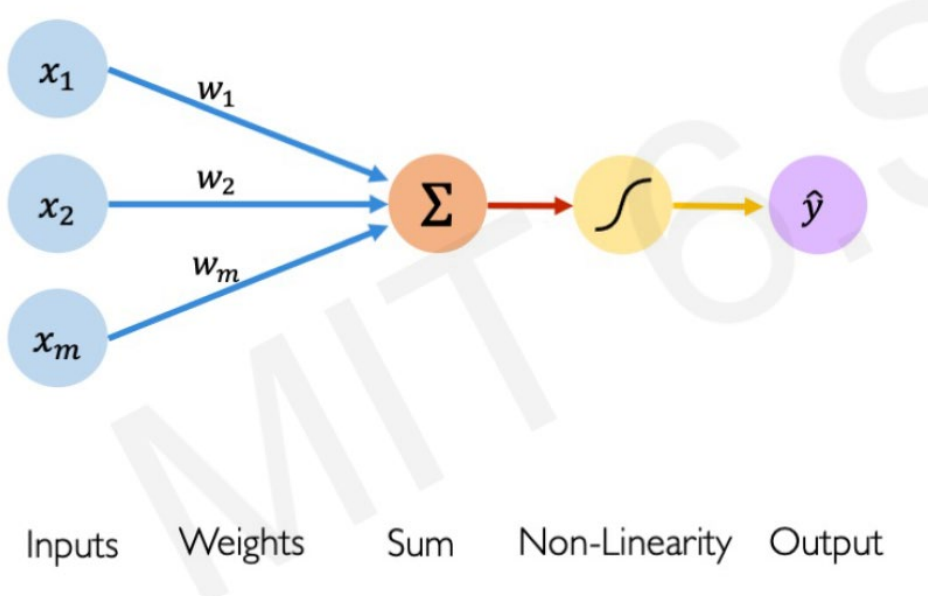
Soma : هسته پردازش نورون

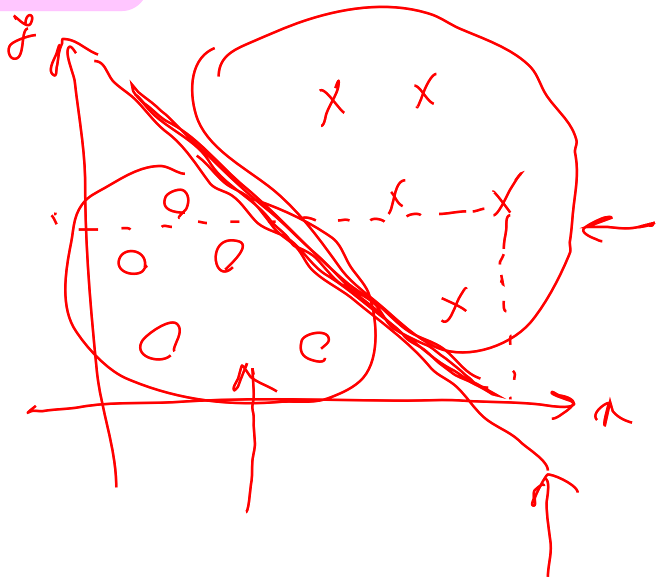


→ axon : send output to other neurons

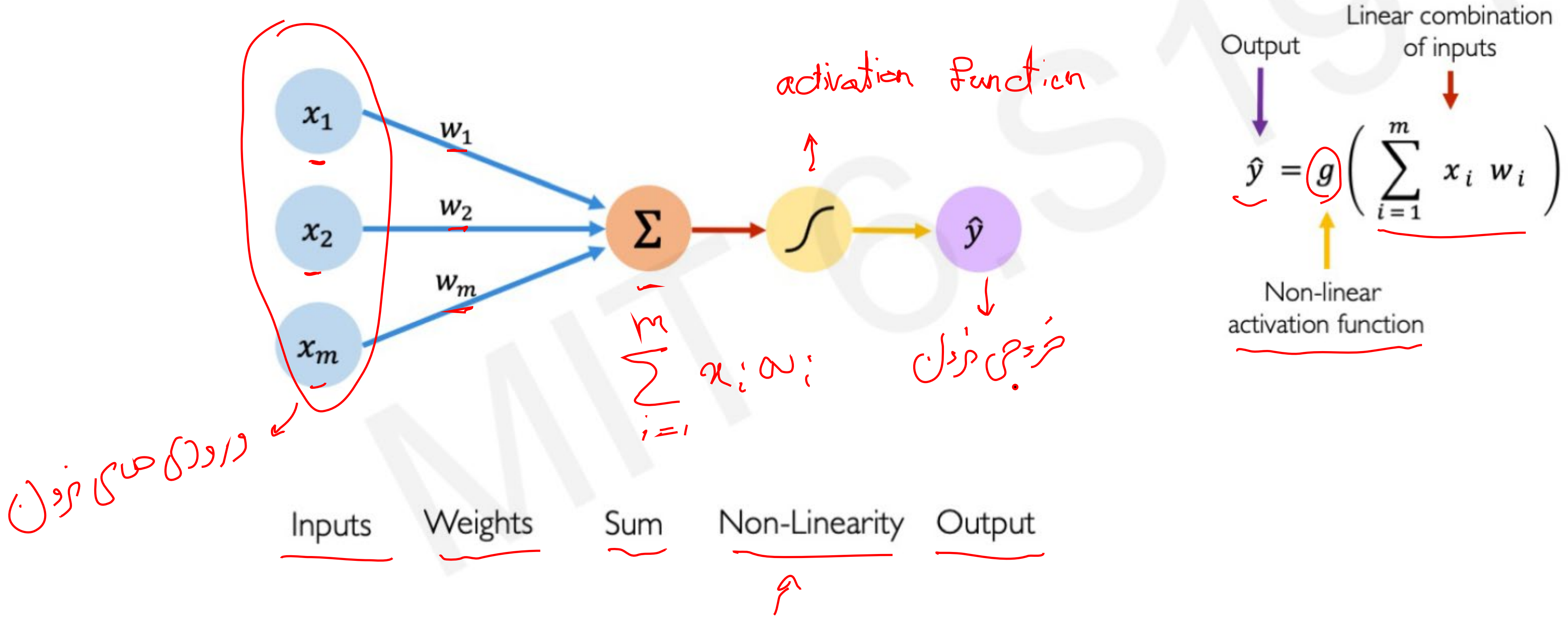
A simple mathematical model of a neuron

- McCulloch and Pitts 1943.
- A linear classfier — it "fires" when a linear combination of its inputs exceeds some threshold.



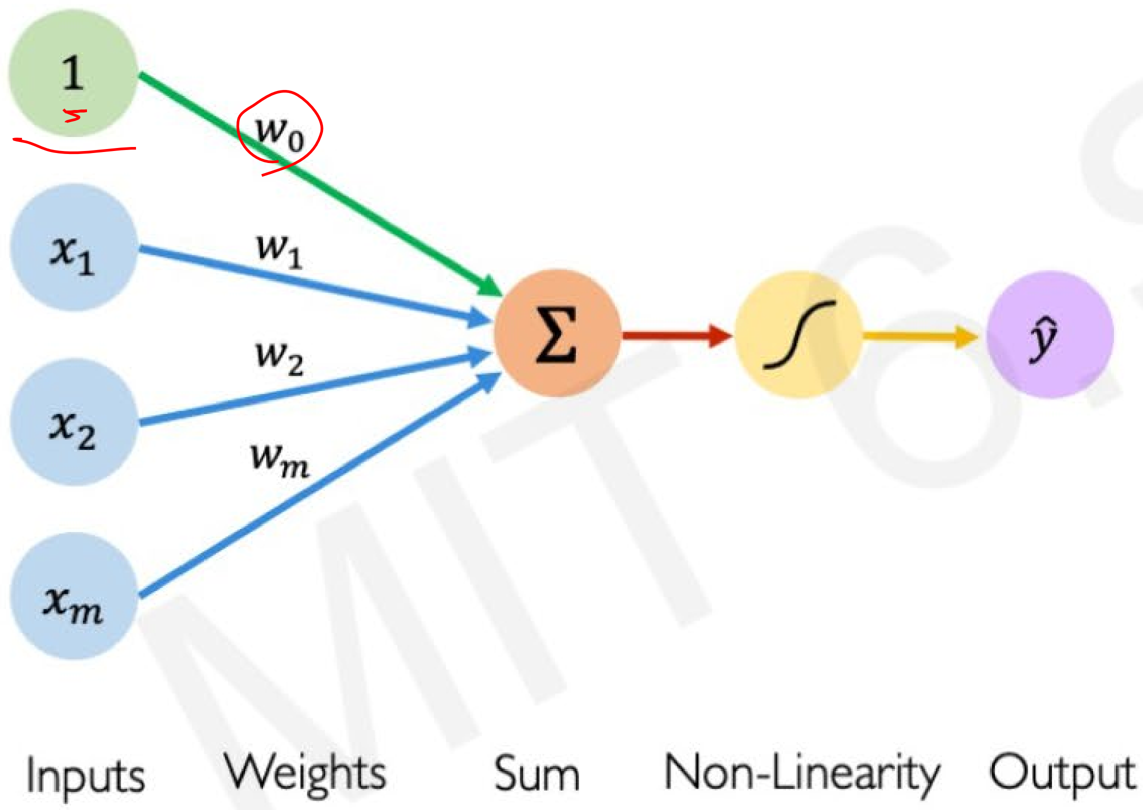


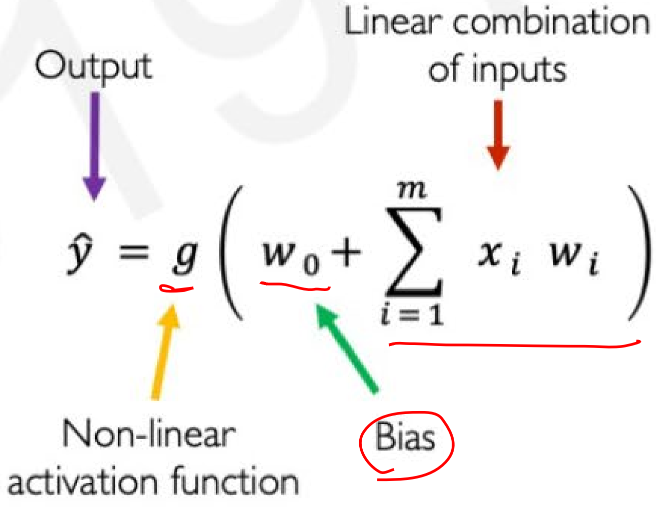
A simple mathematical model of a neuron



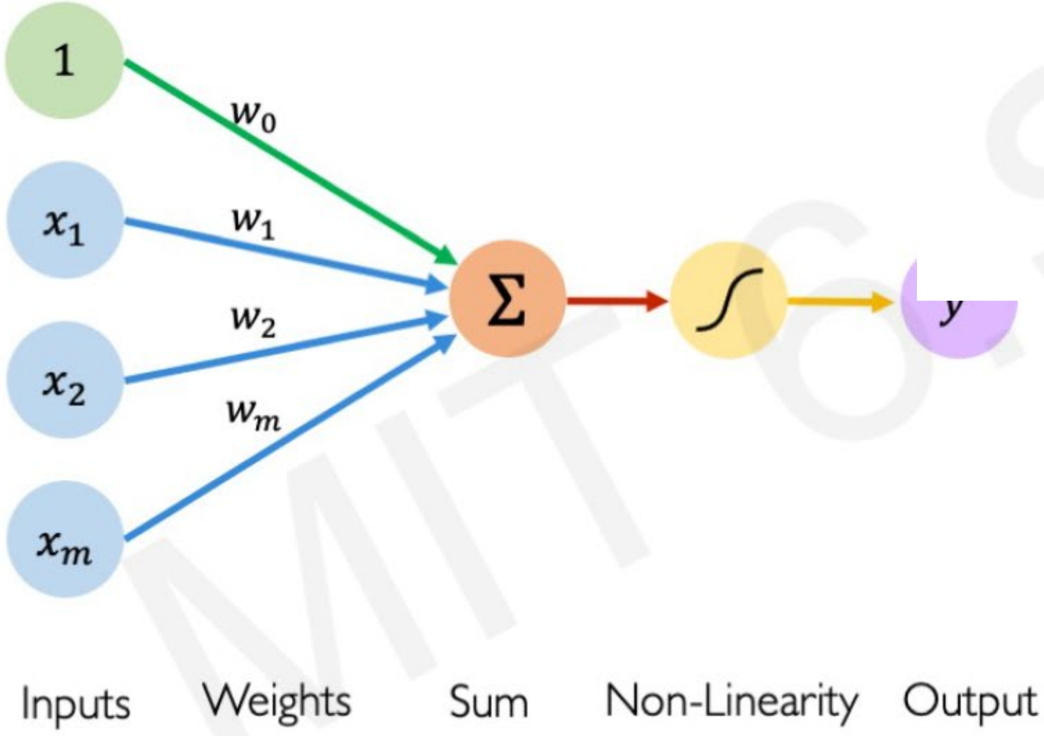
A simple mathematical model of a neuron

→ bias input (dummy input





A simple mathematical model of a neuron



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

$$\hat{y} = g\left(w_0 + \boldsymbol{W}^T \boldsymbol{X} \right)$$

where: $\boldsymbol{X} = \begin{bmatrix} x_1 \\ \vdots \\ x_m \end{bmatrix}_{m\times 1}$ and $\boldsymbol{W} = \begin{bmatrix} w_1 \\ \vdots \\ w_m \end{bmatrix}_{m\times 1}$

$$w^T = [w_1 \; \cdots \; w_m]_{1\times m}$$

$$w^T x = [w_1 \cdots w_m] \begin{bmatrix} x_1 \\ \vdots \\ x_m \end{bmatrix} = \boxed{w_1 x_1 + w_2 x_2 + \cdots + w_m x_m}$$

Desirable Properties of The Activation Function

- It should be non-linear.

- ترکیب توابع خطی نمی توانند یک رابطه غیرخطی را نشان دهد و روابط پیچیده اغلب غیرخطی هستند.

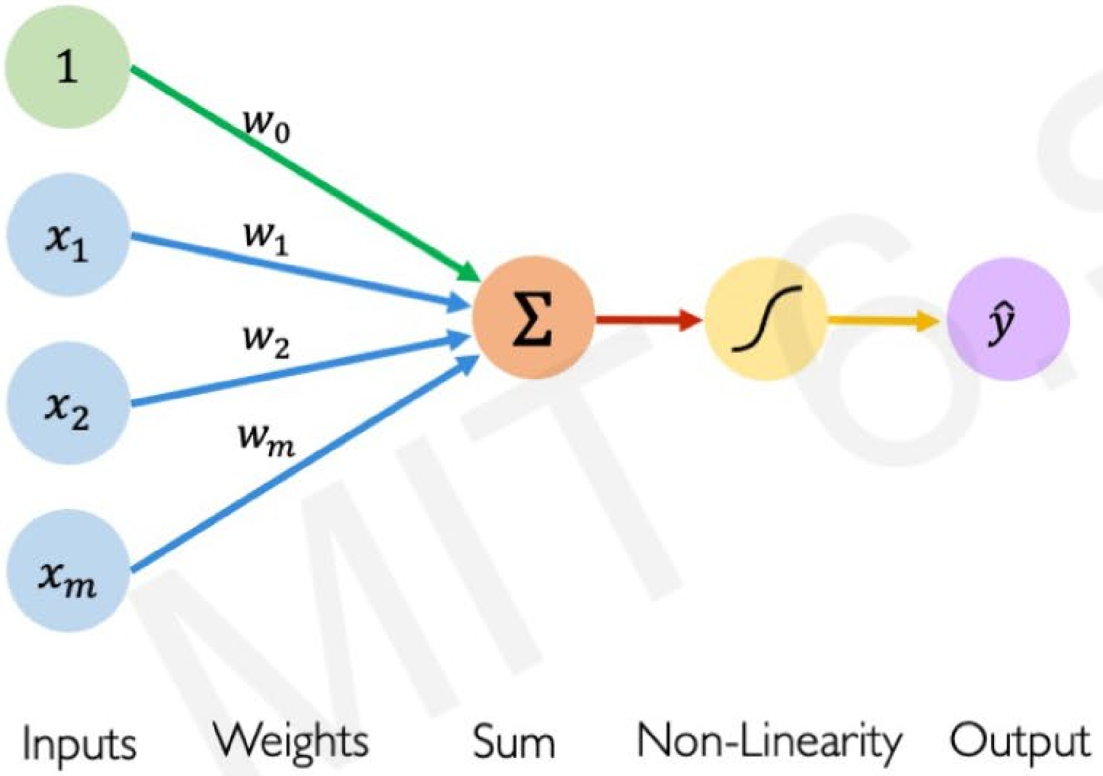
- It should mimic the behaviour of real neurons.

- اگر مجموع وزن دار ورودی ها به اندازه کافی بزرگ باشد نورون فعال می شود و در غیر این صورت fire نمی شود.

- It should be differentiable almost everywhere.

- یادگیری توسط الگوریتم gradient descent ← نیاز به مشتق پذیری دارد

Activation Function

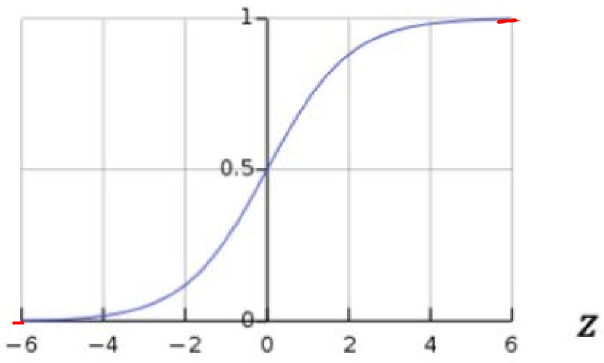


Activation Functions

$$\hat{y} = g(\, w_0 + \boldsymbol{w}^T \boldsymbol{X} \,)$$

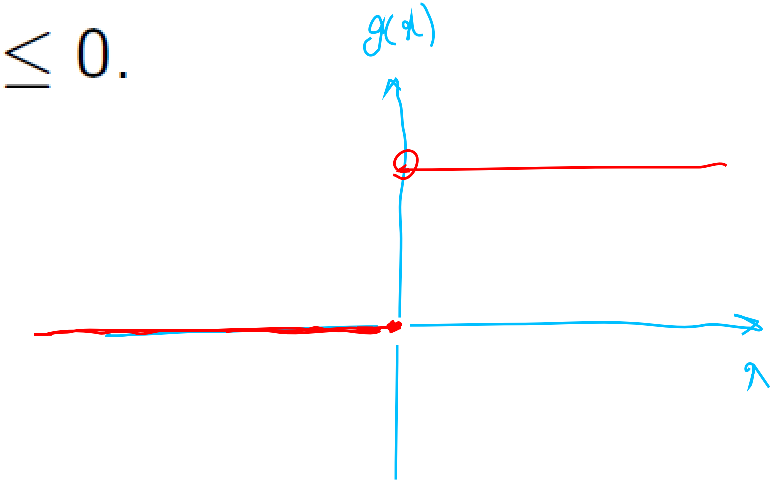
- Example: sigmoid function

$$g(z) = \sigma(z) = \frac{1}{1 + e^{-z}}$$



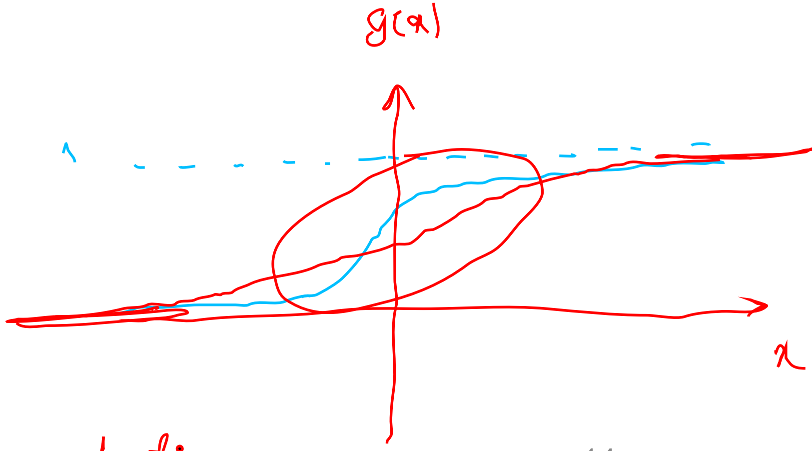
Common Activation Functions

- Step function: $g(x) = 1$ if $x > 0$. $g(x) = 0$ if $x \leq 0$.



- ساده
- مشتق پذیر نیست
- در عمل استفاده نمی‌شود
- مناسب برای تدریس داستان مطلب

- Sigmoid function: $g(x) = \dfrac{1}{1 + e^{-kx}}$.



- مشتق پذیر
- برد محدود به صفر و یک

$\downarrow K$ ← افقی تر (اوتر)

$\uparrow K$ ← نزدیک تر به step function

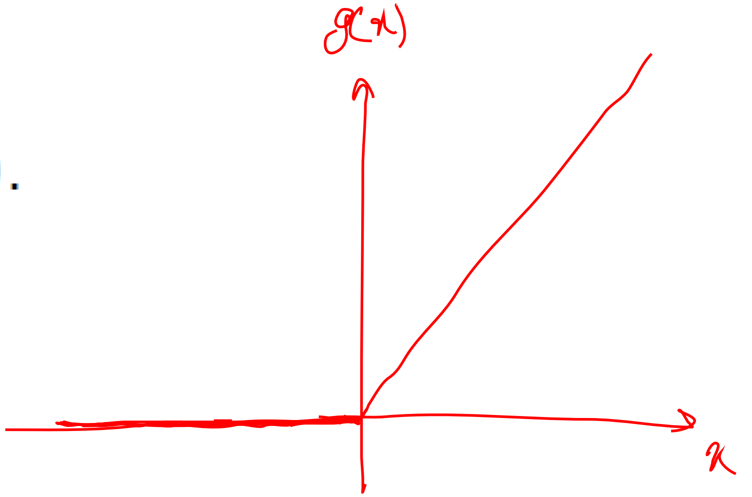
vanishing gradient → کندی learning → Complex Computation

Common Activation Functions

- Rectified linear unit (ReLu): $g(x) = max(0, x)$.

dying Relu

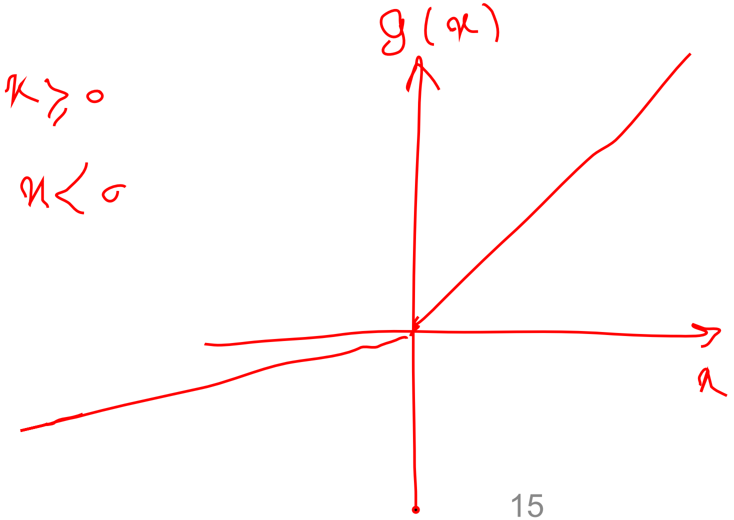
- غیرخطی
- همه‌جا مشتق‌پذیر بود
- vanishing gradient برای xهای بزرگ ندارد



- Leaky ReLU: $g(x) = (0.1 * x, x)$.

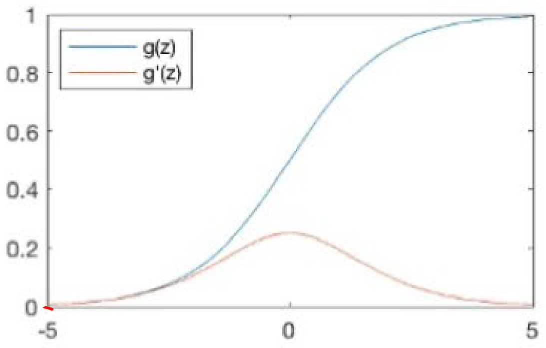
$$g(x)=\begin{cases} x & x \geq 0 \\ 0.1 * x & x < 0 \end{cases}$$

- برای مقادیر (ورودی‌ها) منفی هم قابل استفاده



Activation Function

Sigmoid Function

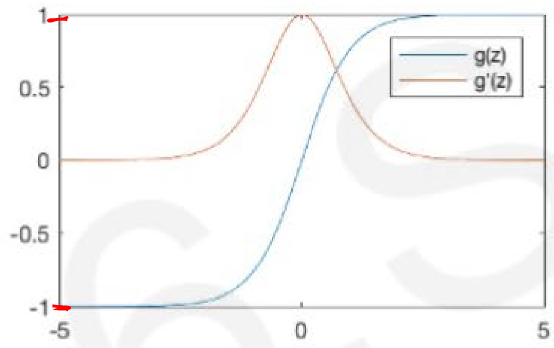


$$g(z) = \frac{1}{1 + e^{-z}}$$

$$g'(z) = g(z)(1 - g(z))$$

```
tf.math.sigmoid(z)
```

Hyperbolic Tangent

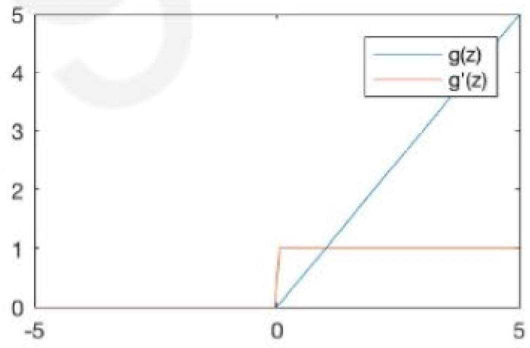


$$g(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}}$$

$$g'(z) = 1 - g(z)^2$$

```
tf.math.tanh(z)
```

Rectified Linear Unit (ReLU)



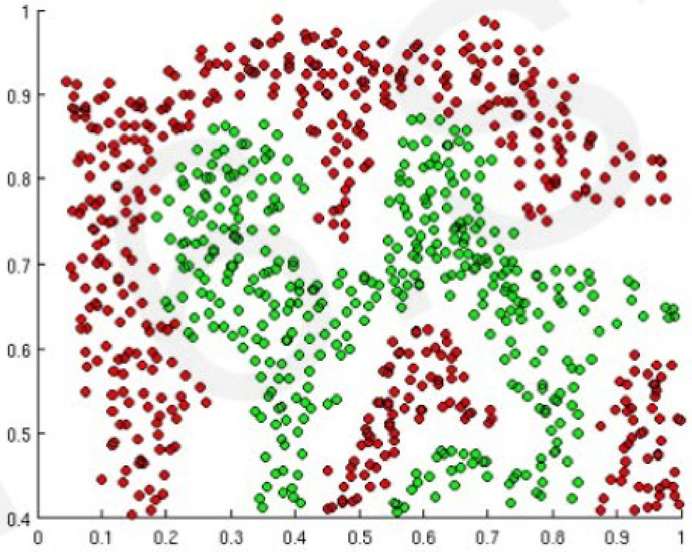
$$g(z) = \max(0, z)$$

$$g'(z) = \begin{cases} 1, & z > 0 \\ 0, & \text{otherwise} \end{cases}$$

```
tf.nn.relu(z)
```

Importance of Activation Function

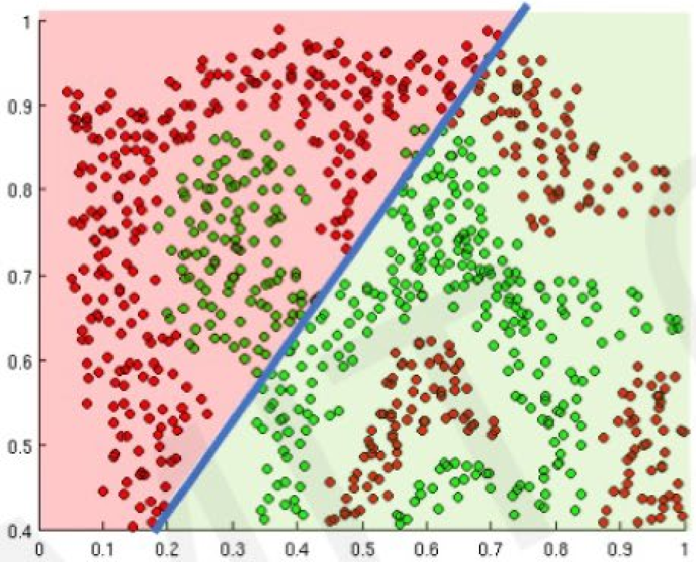
The purpose of activation functions is to ***introduce non-linearities*** *into the network*



What if we wanted to build a neural network to distinguish green vs red points?

Importance of Activation Function

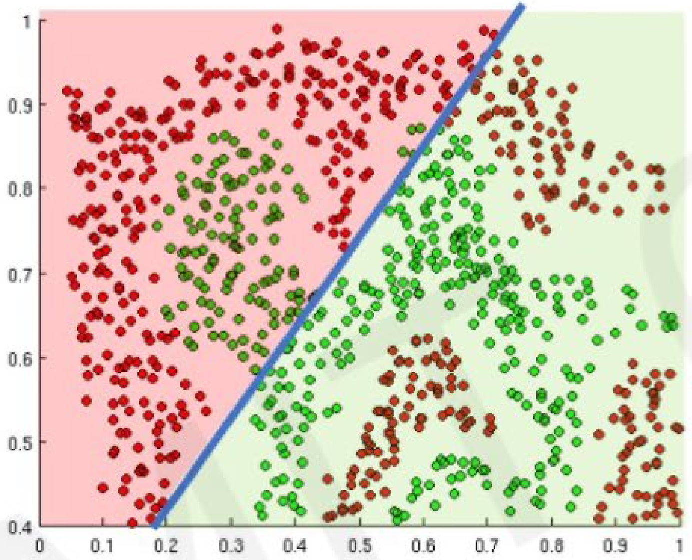
The purpose of activation functions is to ***introduce non-linearities*** *into the network*



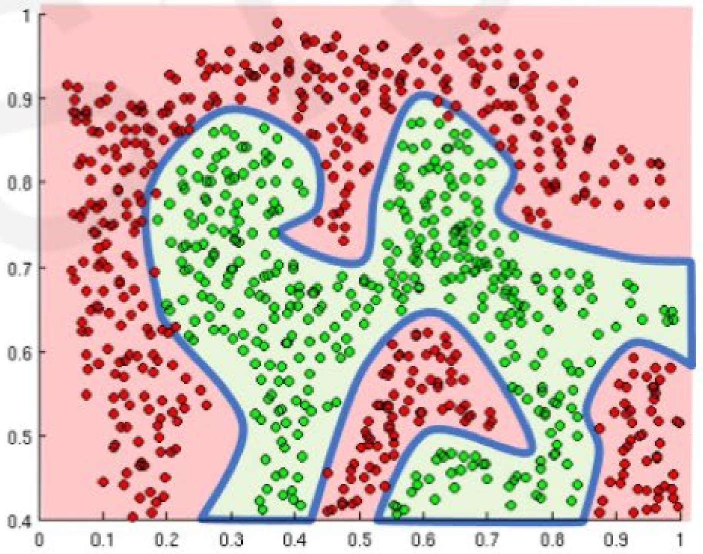
Linear activation functions produce linear decisions no matter the network size

Importance of Activation Function

The purpose of activation functions is to ***introduce non-linearities*** *into the network*



Linear activation functions produce linear decisions no matter the network size



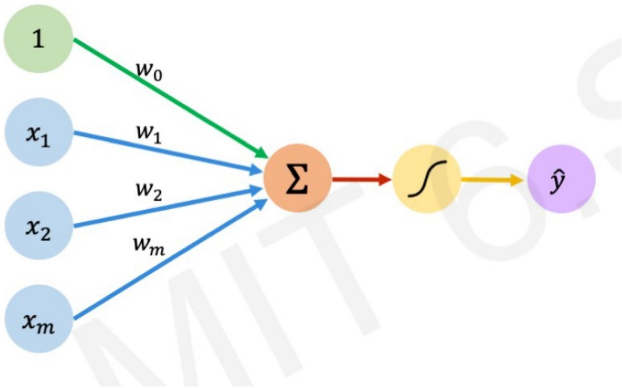
Non-linearities allow us to approximate arbitrarily complex functions

Core Foundation Review

Types of Learning Problems

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning

Simple model for neuron



Activation Function

- Desirable Properties of The Activation Function
- Common Activation Functions