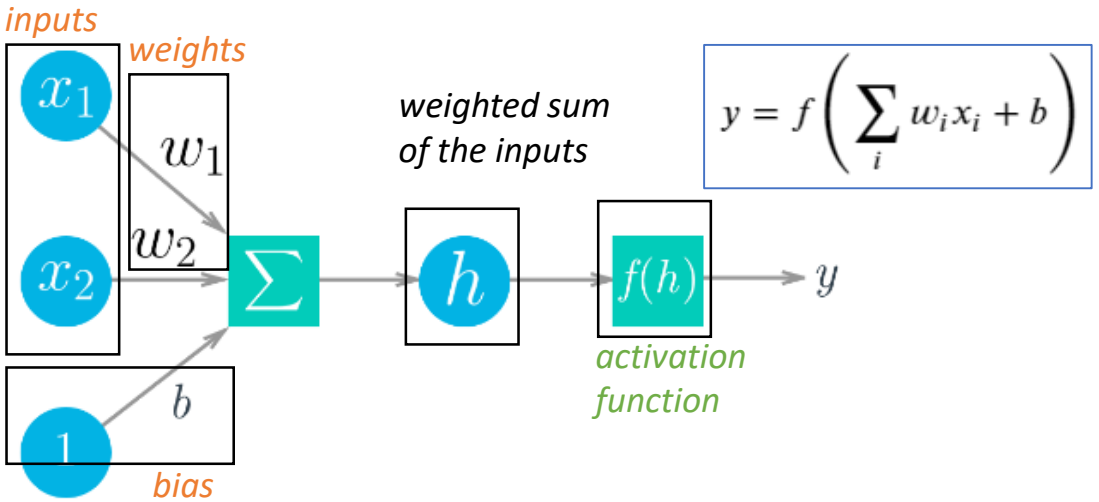
designed by Mariia Denysenko

Single layer neural network



$$y = f\left(\sum_i w_i x_i + b\right)$$

1

When a signal (**input**) arrives, it gets multiplied by a **weight** value. Weights show the strength of particular node.
Bias (offset) is an extra input to neurons and it is always 1, and has it's own connection weight. This makes sure that even when all the inputs are none (all 0's), the activation in the neuron will take place.

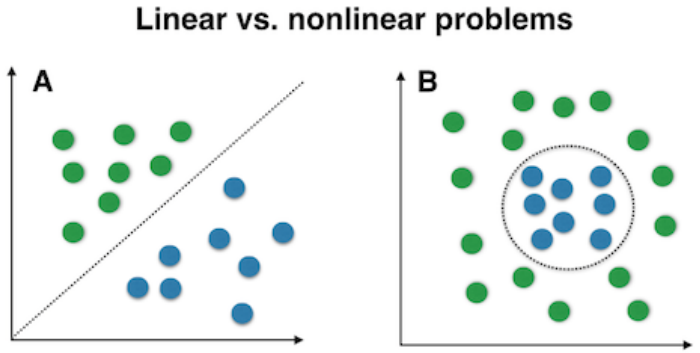
2

Activation function decides, whether a neuron should be activated or not by calculating weighted sum and further adding bias with it. The purpose of the activation function is to introduce ***non-linearity*** into the output of a neuron.

3

Why non-linearity?

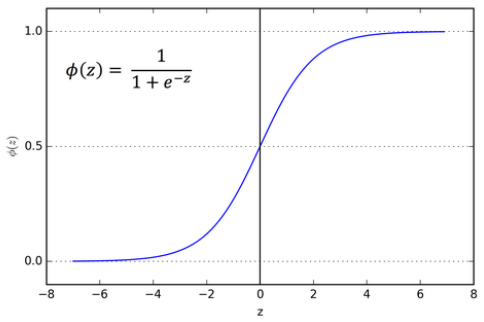
A linearly separable data (as to left) can be separated using a straight line in feature space.

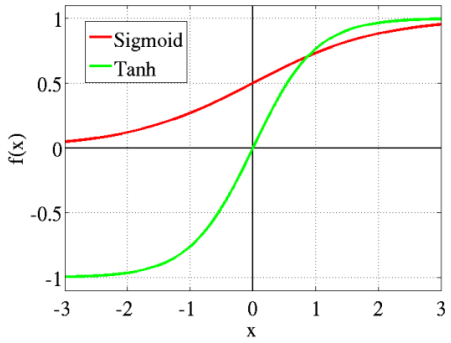


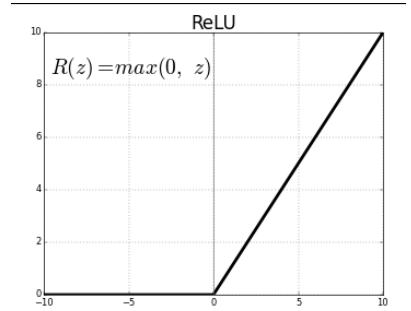
4

The most common types of activation functions

- Sigmoid or Logistic Activation Function
- Tanh or hyperbolic tangent Activation Function
- ReLU (Rectified Linear Unit) Activation Function

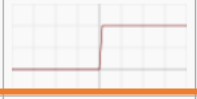
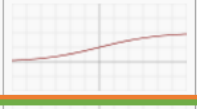
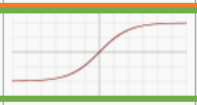
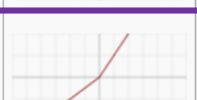






designed by Mariia Denysenko

Activation functions

Name	Plot	Equation	Derivative
Identity		$f(x) = x$	$f'(x) = 1$
Binary step		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x \neq 0 \\ ? & \text{for } x = 0 \end{cases}$
Logistic (a.k.a Soft step)		$f(x) = \frac{1}{1+e^{-x}}$	$f'(x) = f(x)(1-f(x))$
TanH		$f(x) = \tanh(x) = \frac{2}{1+e^{-2x}} - 1$	$f'(x) = 1 - f(x)^2$
ArcTan		$f(x) = \tan^{-1}(x)$	$f'(x) = \frac{1}{x^2+1}$
Rectified Linear Unit (ReLU)		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Parameteric Rectified Linear Unit (PReLU) [2]		$f(x) = \begin{cases} \alpha x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} \alpha & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Exponential Linear Unit (ELU) [3]		$f(x) = \begin{cases} \alpha(e^x - 1) & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} f(x) + \alpha & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
SoftPlus		$f(x) = \log_e(1+e^x)$	$f'(x) = \frac{1}{1+e^{-x}}$

1

Sigmoid or Logistic Activation Function

Sigmoid function exists between (0 to 1). Therefore, it is especially used for models where we have to predict the probability as an output.

The softmax function is a more generalized logistic activation function which is used for multiclass classification.

2

Tanh or hyperbolic tangent Activation Function

The range of the tanh function is from (-1 to 1). tanh is also sigmoidal (s - shaped).

The advantage is that the negative inputs will be mapped strongly negative and the zero inputs will be mapped near zero in the tanh graph.

The tanh function is mainly used for classification between two classes.

3

ReLU (Rectified Linear Unit) Activation Function

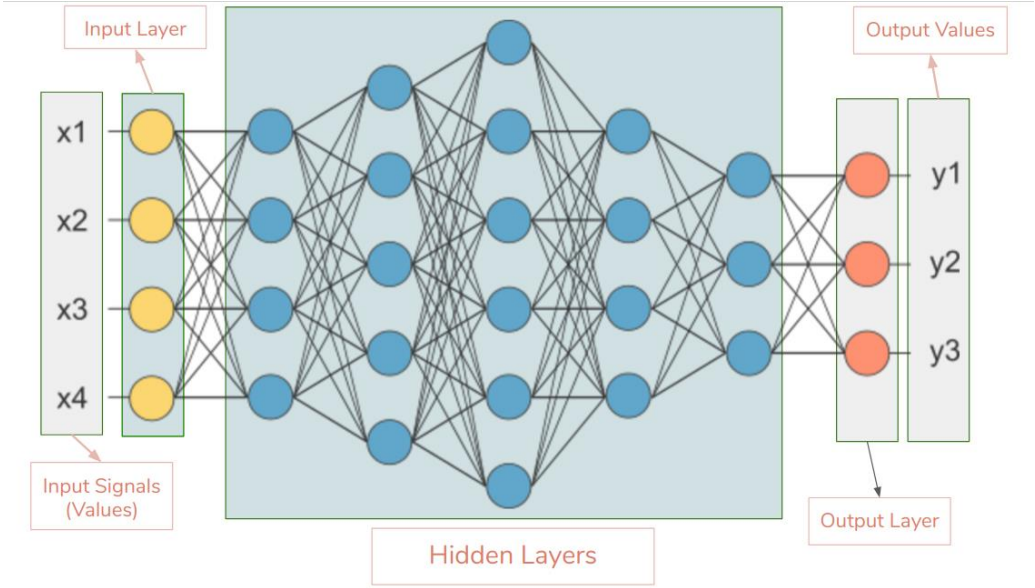
The ReLU is the most used activation function in the world right now.Since, it is used in almost all the convolutional neural networks or deep learning.

ReLU is half rectified (from bottom). The function ranges from 0 to infinity.

The fact that all the negative values become zero immediately decreases the ability of the model to fit or train from the data properly.

designed by Mariia Denysenko

Multilayer neural network and activation function



1

Input Layer is the first layer in the neural network. It takes input signals (values) and passes them on to the next layer.

Hidden Layers have neurons which apply different transformations to the input data.

Output Layer is the last layer in the network & receives input from the last hidden layer. With this layer we can get desired number of values and in a desired range.

2

How to select a correct activation function

Hidden layer:
A strictly positive function like a standard logistic (sigmoid) can be a bit awkward in multilayer models, since if the out layers' outputs are all positive, on the next layer's transition point being at zero, it can put a lot of pressure on bias to compensate it, and this can lead to slow learning. Therefore both sigmoid and tanh functions are not suitable for hidden layers.
Rectified linear unit (ReLU) is a preferred choice for all hidden layers. In some cases, Parametric ReLU can be used just to avoid exact zero derivatives.

Output layer:
For binary classification, the sigmoid function is a good choice for output layer because the actual output value is either 0 or 1.

Note: For non-classification problems such as prediction of housing prices, we shall use linear activation function at the output layer only.

Resources used:

- Udacity. Secure and Private AI Scholarship Challenge Nanodegree Program
- Muhammad Rizwan. How to Select Activation Function for Deep Neural Network. URL:
- https://engmrk.com/activation-function-for-dnn/
- Sagar Sharma. Activation Functions in Neural Networks. URL: https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6
- Kailash Ahirwar. Secret Sauce Behind Deep Learning. URL: https://hackernoon.com/everything-you-need-to-know-about-neural-networks-8988c3ee4491
- Alexander Ihler. Neural Networks: Basics. URL: https://www.youtube.com/watch?v=bH6VnezBZfI
- GeeksforGeeks. Activation Functions in Neural Networks. URL: https://www.geeksforgeeks.org/activation-functions-neural-networks/