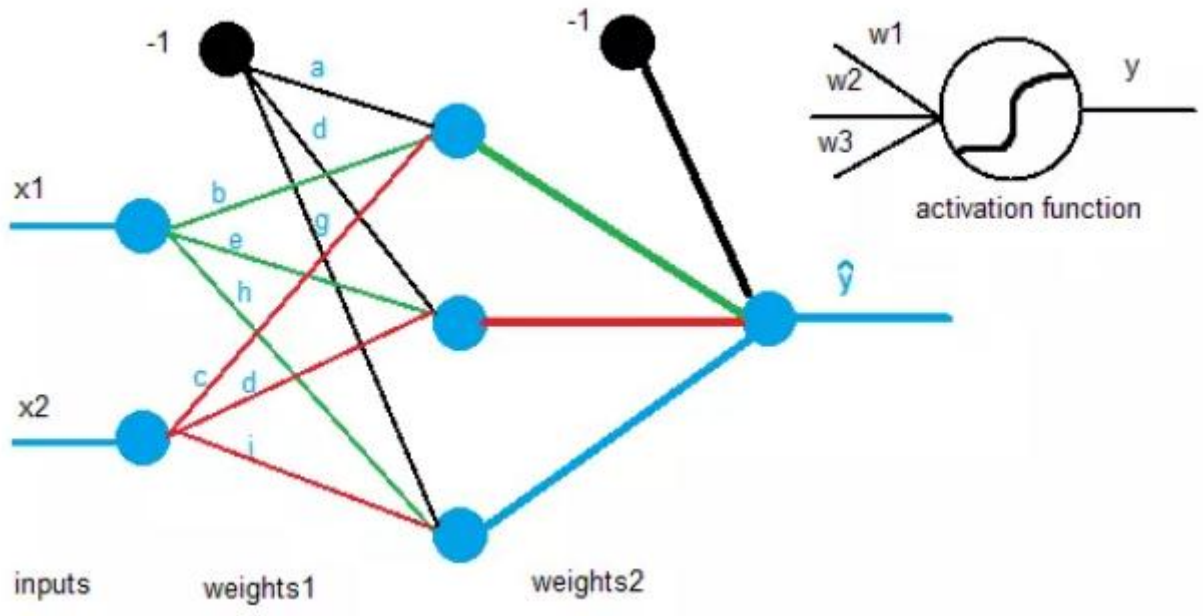
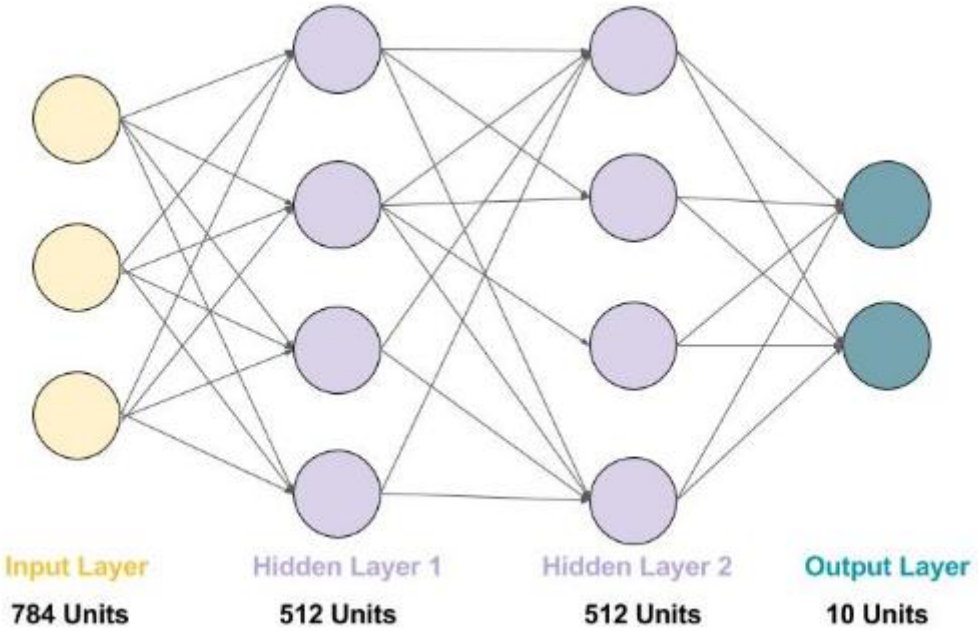
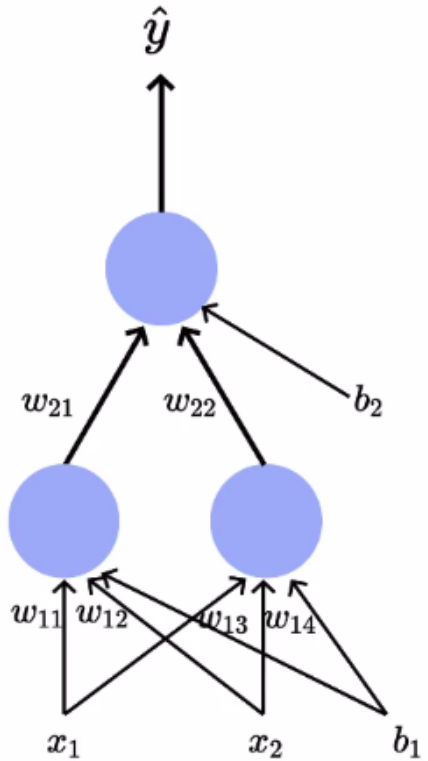

-1
a
d
x1
b
g
e
h
x2
c
d
i
-1
ŷ
w1
w2
w3
y
activation function
inputs
weights1
weights2

Hidden Layers



How to build complex functions using Deep Neural Networks?

Model



$$h_1 = f_1(x_1, x_2)$$

$$h_2 = f_2(x_1, x_2)$$

$$\hat{y} = g(h_1, h_2)$$

$$h_1 = \frac{1}{1+e^{-(w_{11}*x_1+w_{12}*x_2+b_1)}}$$

$$h_2 = \frac{1}{1+e^{-(w_{13}*x_1+w_{14}*x_2+b_1)}}$$

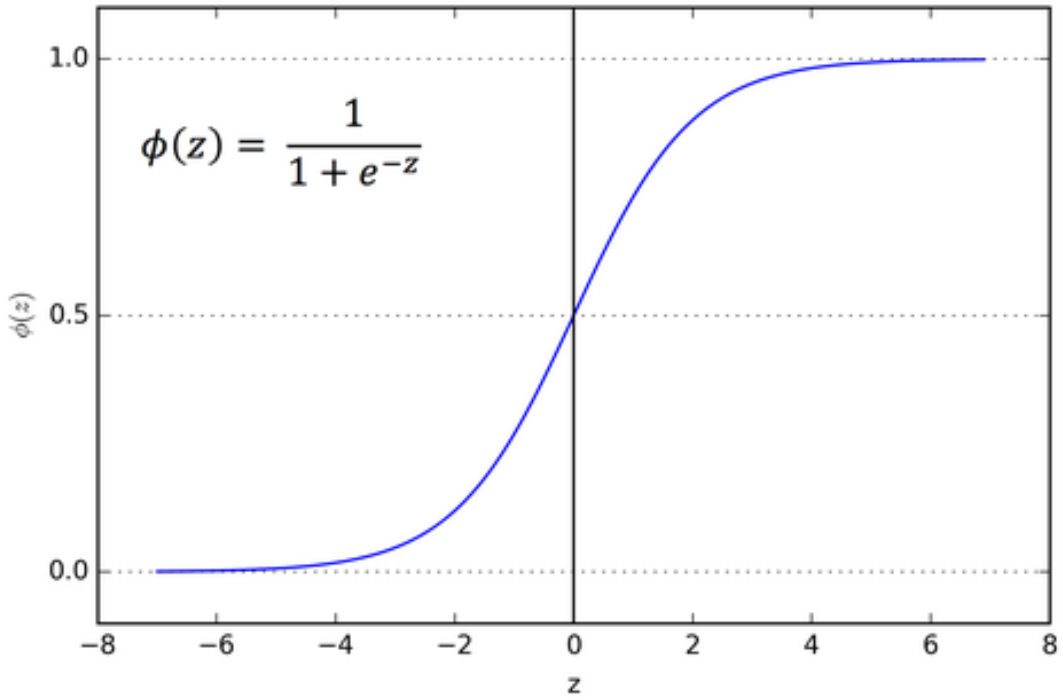
$$\hat{y} = \frac{1}{1+e^{-(w_{21}*h_1+w_{22}*h_2+b_2)}}$$

$$= \frac{1}{1+e^{-(w_{21}*(\frac{1}{1+e^{-(w_{11}*x_1+w_{12}*x_2+b_1)}})+w_{22}*(\frac{1}{1+e^{-(w_{13}*x_1+w_{14}*x_2+b_1)}})+b_2)}}$$

Activation Functions -Examples

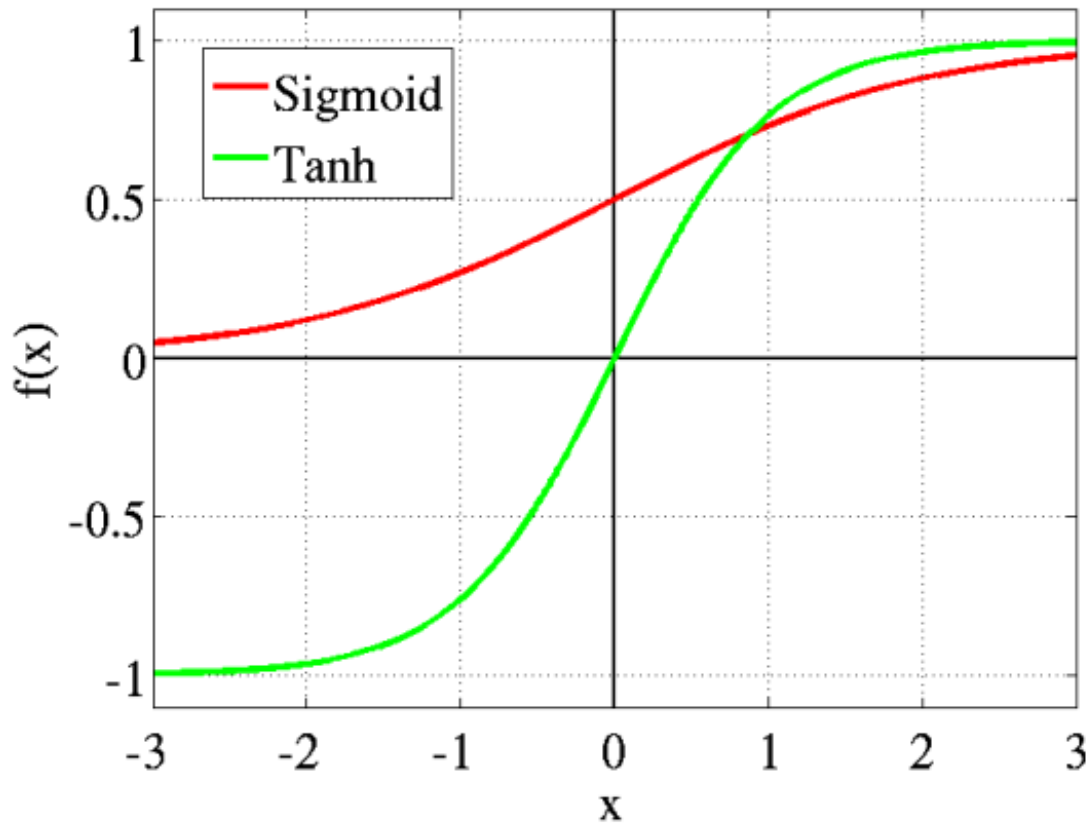
1. Sigmoid or Logistic Activation Function

The Sigmoid Function curve looks like a S-shape.



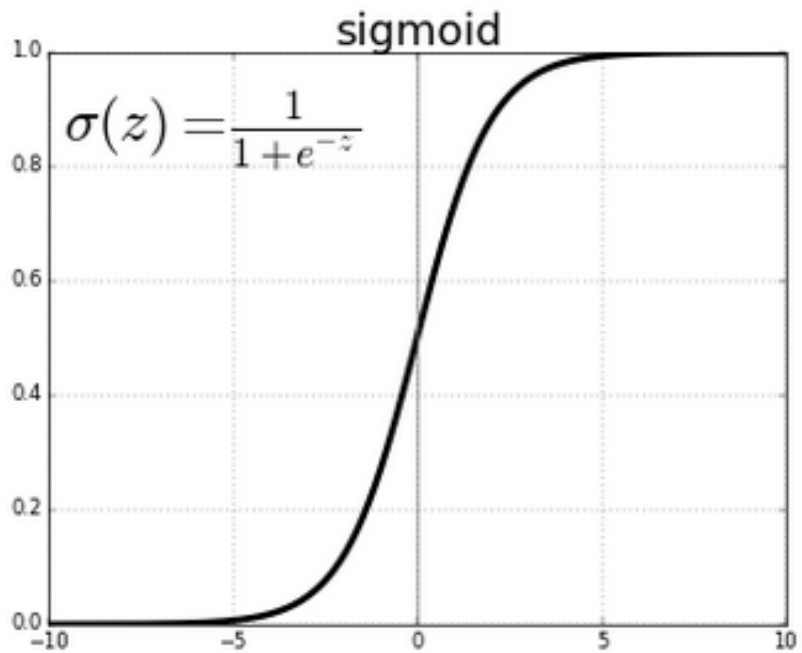
2. Tanh or hyperbolic tangent Activation Function

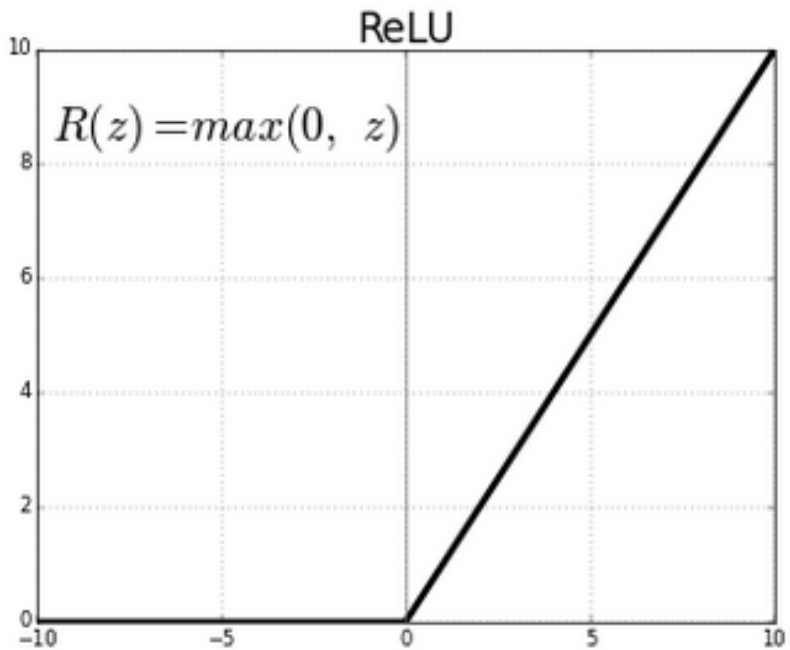
tanh is also like logistic sigmoid but better. The range of the tanh function is from (-1 to 1). tanh is also sigmoidal (s - shaped).



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.





4. Leaky ReLU

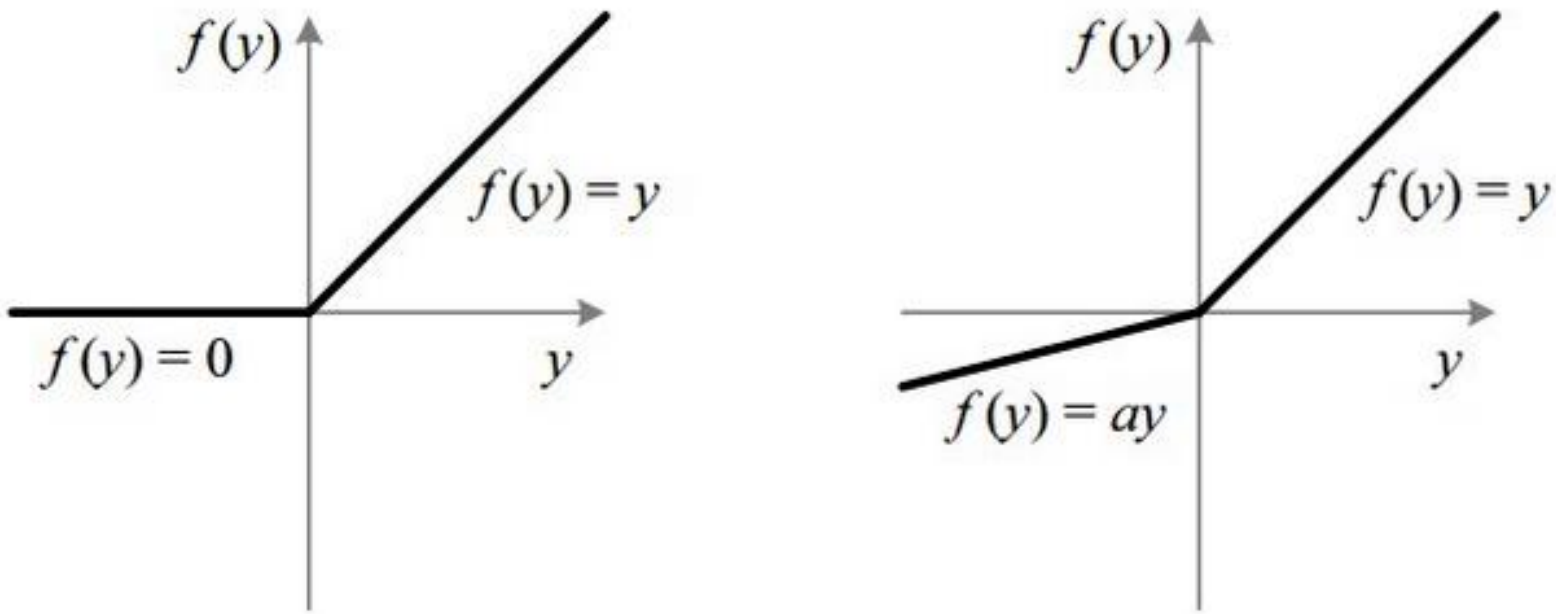
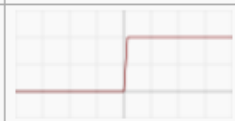
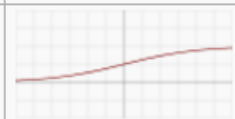
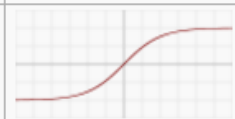
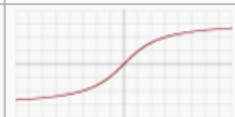
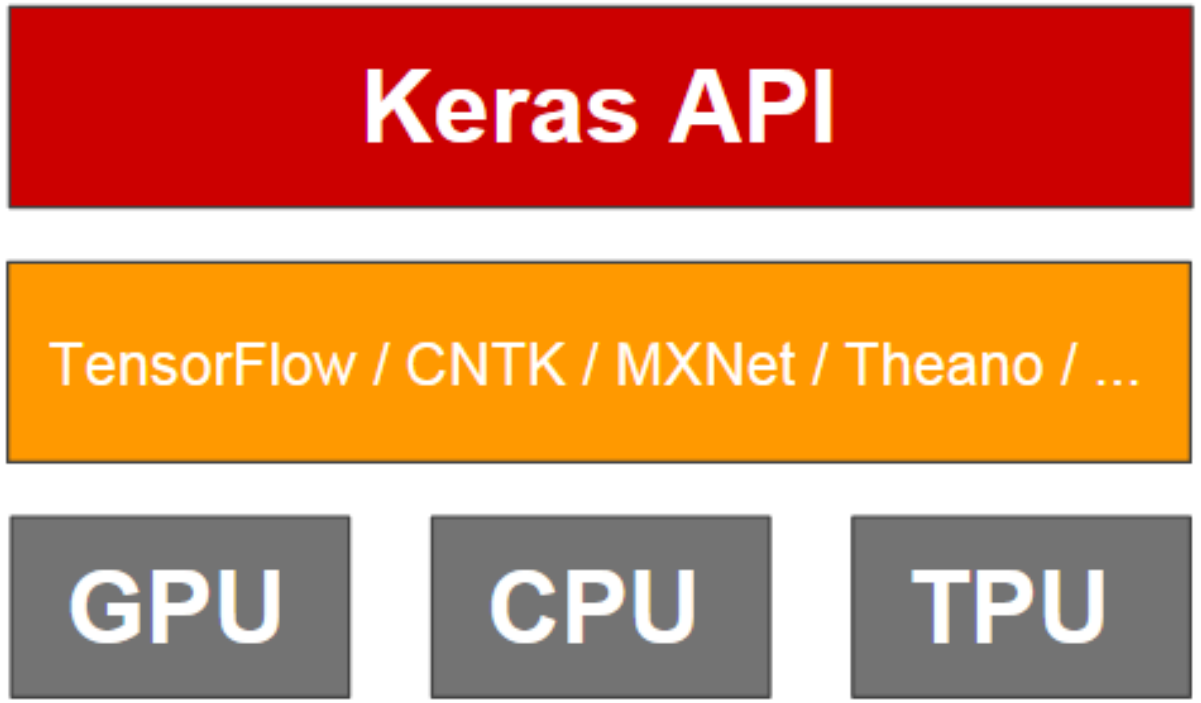


Fig : ReLU v/s Leaky ReLU

Activation Functions

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

Keras: an API for specifying & training Deep Learning Models



Who makes Keras? Contributors and backers

Reach

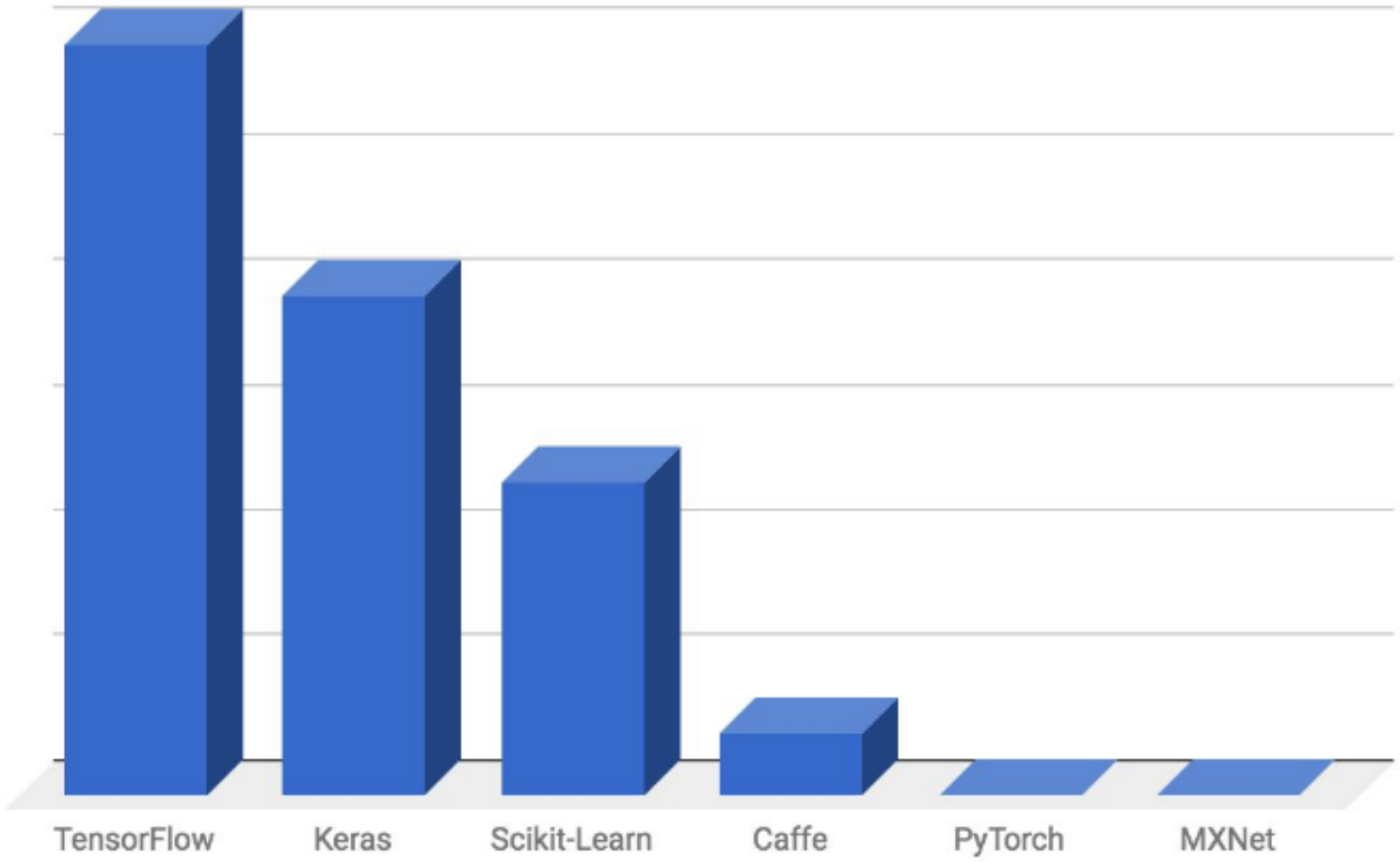
There are around 2,50,000 keras developers

> 2x Yearly growth

Industry traction

etc...

JOB Market



The Keras user experience

Keras is easy to use.

Keras follows best practices for reducing cognitive load: it offers consistent & simple APIs, it minimizes the number of user actions required for common use cases and it is easy to use. As a Keras user, you are more productive, allowing you to try more ideas faster .

Keras is Flexibile to use

Keras integrates with lower-level deep learning languages (in particular TensorFlow), it enables you to implement anything you could have built in the base language.

Loss Functions

- https://keras.io/losses/