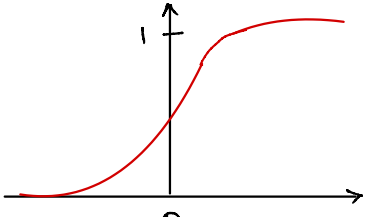
→ We can use different types of activation function instead of just sigmoid.

↓

classification won't always be binary

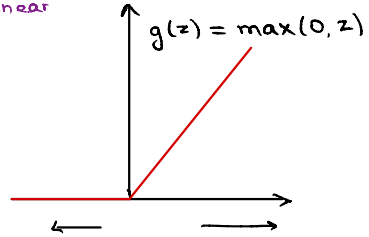
"sigmoid"



$0 \leq g(z) \leq 1$

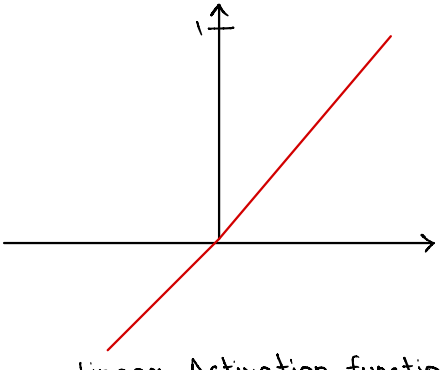
"ReLu"

Rectified Linear Unit



if $z \leq 0$, $g(z) = 0$

if $z \geq 0$, $g(z) = z$



Linear Activation function

"No activation function"

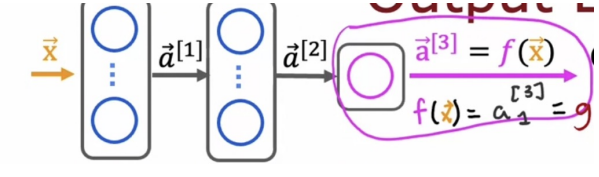
$a = g(z) = \underbrace{\vec{w} \cdot \vec{x} + b}_{z}$

In Summary :-

We can choose any activation function based on usage.

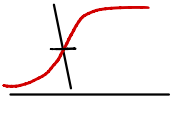
Choosing Activation Function

While choosing activation function for output layer



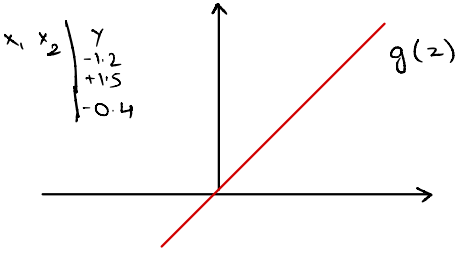
different activation function can be used for diff. layers.

- For binary classification :- sigmoid



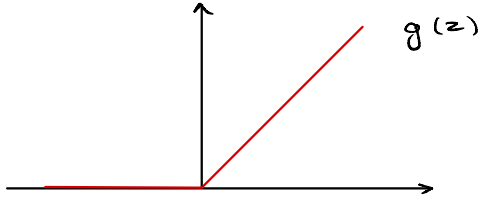
- For Regression i.e. output = +ve or -ve

  → use linear activation function:

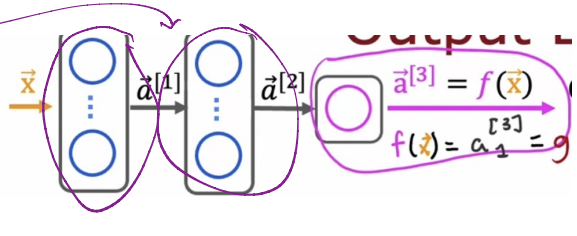


- For regression where output is 0 or +ve

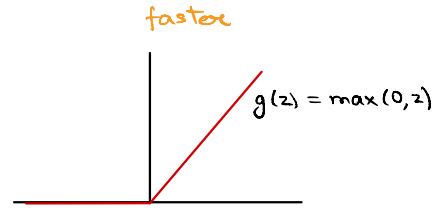
  → use ReLU



Choosing Activation Function for Hidden layer



most common choice :- ReLU.

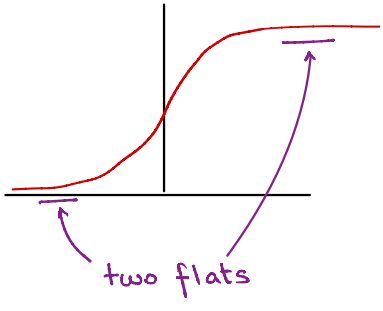


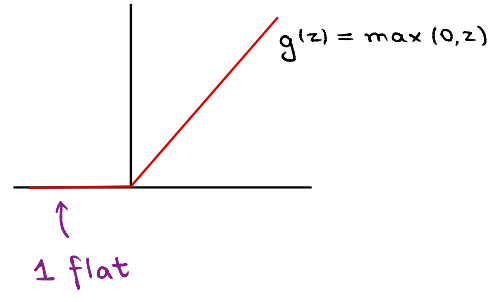
Q. Why is ReLU faster ?

A. ReLU is faster because :-

1. Computes very less. In comparison to sigmoid which computes inverse of $1 + \exp(-(wx+b))$, we just have to output all values > 0.

2. ReLU graph has less flats than sigmoid.





$\Rightarrow$ speed of gradient descent $\propto \frac{1}{\text{flat curves}}$

```
from tf.keras.layers import Dense
model = Sequential([
  Dense(units=25, activation='relu'),      layer1
  Dense(units=15, activation='relu'),      layer2
  Dense(units=1,  activation='sigmoid')    layer3
])
```

these are more activations
→ Leaky ReLU, tanh, etc.