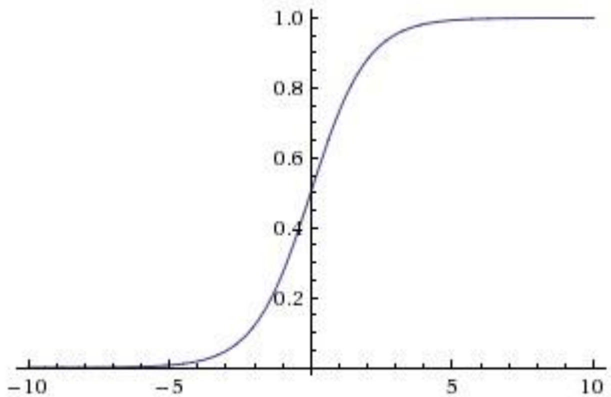
Activation Functions

Content from Andrej Karpathy
http://cs231n.github.io/neural-networks-1/#actfun

CSC411/2515: Machine Learning and Data Mining, Winter 2018

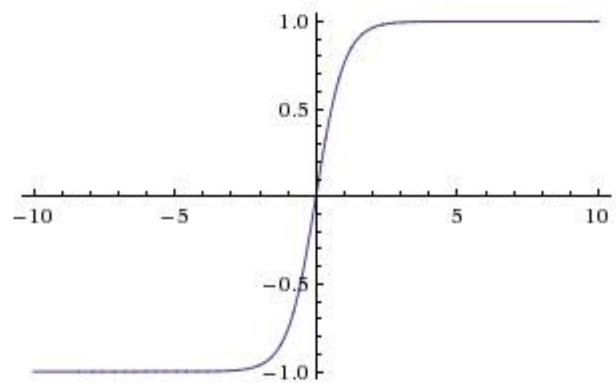
Michael Guerzhoy and Lisa Zhang

Sigmoid



- $\sigma(t) = 1/(1 + \exp(-t))$
- Disadvantages:
  - $\sigma'(t)$ is very small for t outside of t ∈ $[-5, 5]$
    - If that happens, the neuron "dies:" the weights below the n the value of the neuron remains fixed (since any change to the weights is multiplied by $\sigma'(t)$)
  - $\sigma(t)$ is always positive
    - *All* the weights will either move in the positive direction or the negative direction during a given step of gradient descent for a given neuron
      - Slower convergence

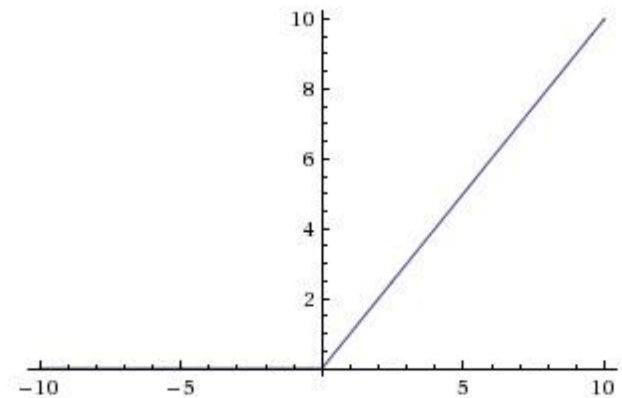
Tanh



- $\tanh(t) = \frac{1-\exp(-2t)}{1+\exp(-2t)}$
- (=$2\sigma(2t) - 1$)
- Not always positive
  - No problem with all the weights having to move in the same direction
  - Advantage over the sigmoid

ReLU

- Rectified Linear Unit
- $f(t) = \max(0, t)$
- Works well if you're careful – better than others (but needs care!)
- Cheap to compute
- Dies if t is too small
  - No gradient at all!



Summary

- Don’t use Sigmoid
- Try ReLU and then tanh