1 Weight initialization

1.1 Zero initialization

- $W = 0$
- never used
- all neurons will react the same way
    - learn the same function
    - output the same gradient
    - update in the same way

1.2 Random initialization

- $W = 0.01 * np.random.randn(n^{[l-1]}, n^{[l]})$
- small random numbers; gaussian with zero mean and $1e^{-2}$ standard deviation
- works okay for small networks
- problems with deeper networks
    - standard deviation drops to zero
    - activations become zero
    - no update

1.3 Xavier initialization

- $W = \frac{np.random.rand(n^{[l-1]}, n^{[l]})}{np.sqrt(n^{[l-1]})}$
- works with tanh activation
- does not work well with ReLU activation

1.4 MSRA initialization

- $W = \frac{np.random.rand(n^{[l-1]}, n^{[l]})}{np.sqrt(\frac{n^{[l-1]}}{2})}$
- works better with ReLU activation