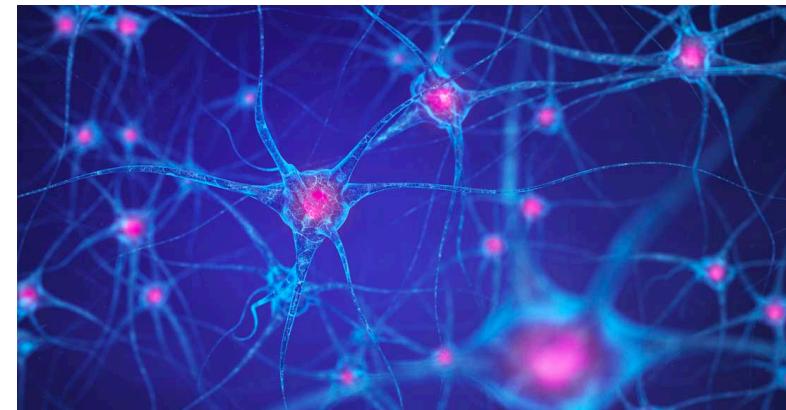
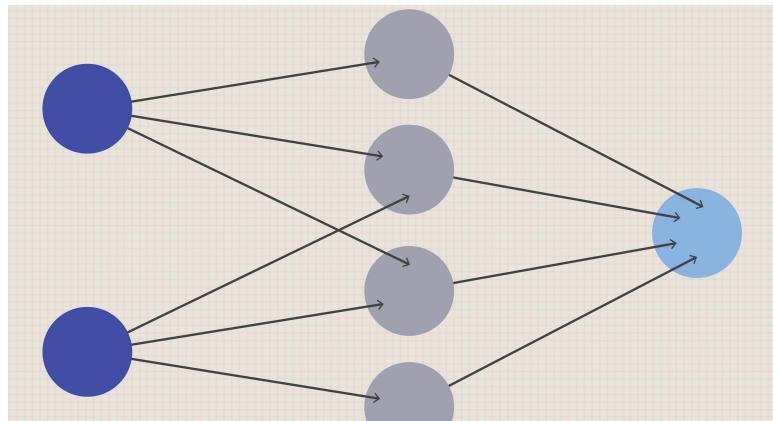


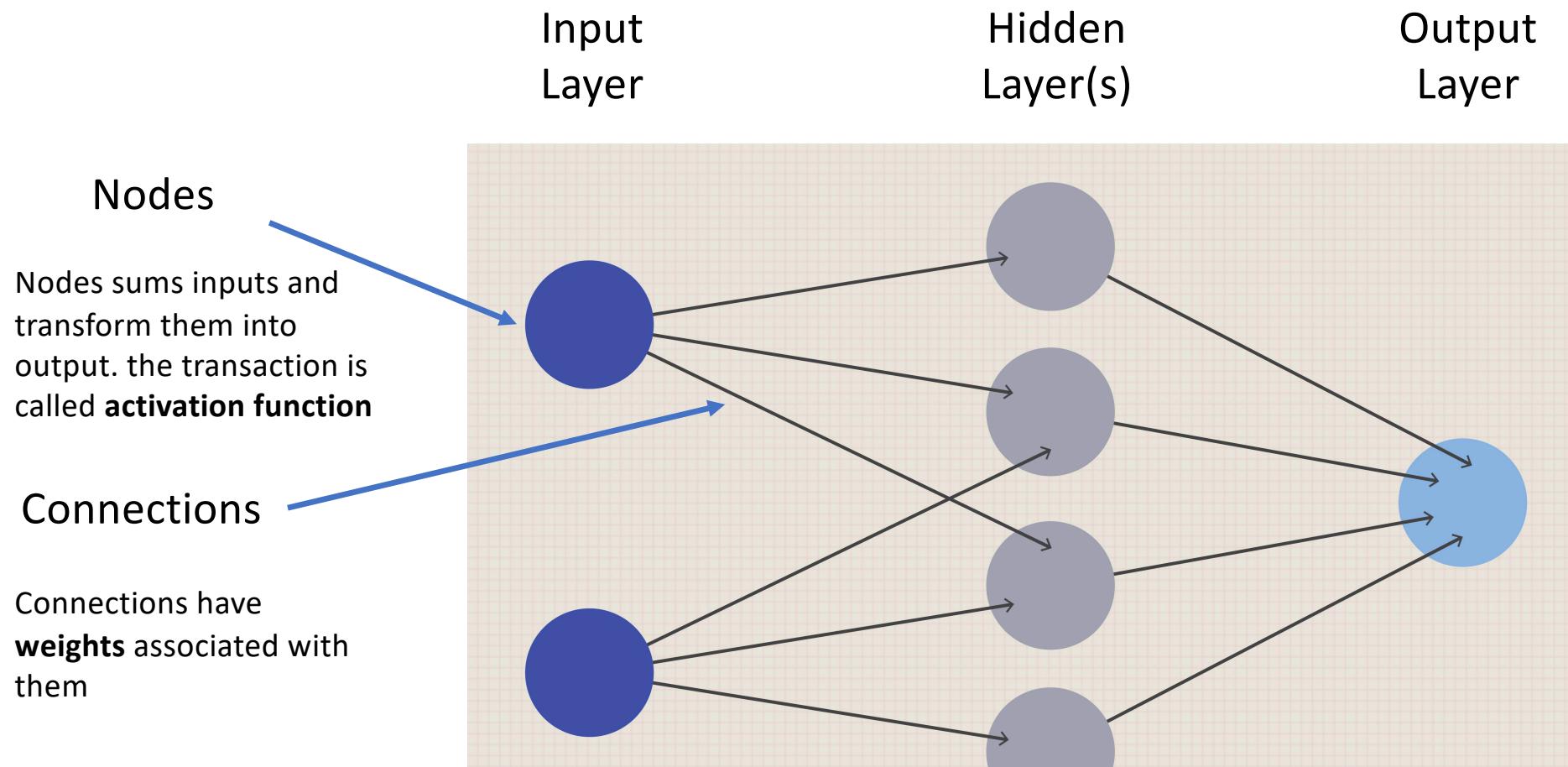
# Neural Network

# Neural Network

- Class of ML algorithms first devised over 60 years ago
- Inspired by the inner workings of how brain works



# Components of Neural Network

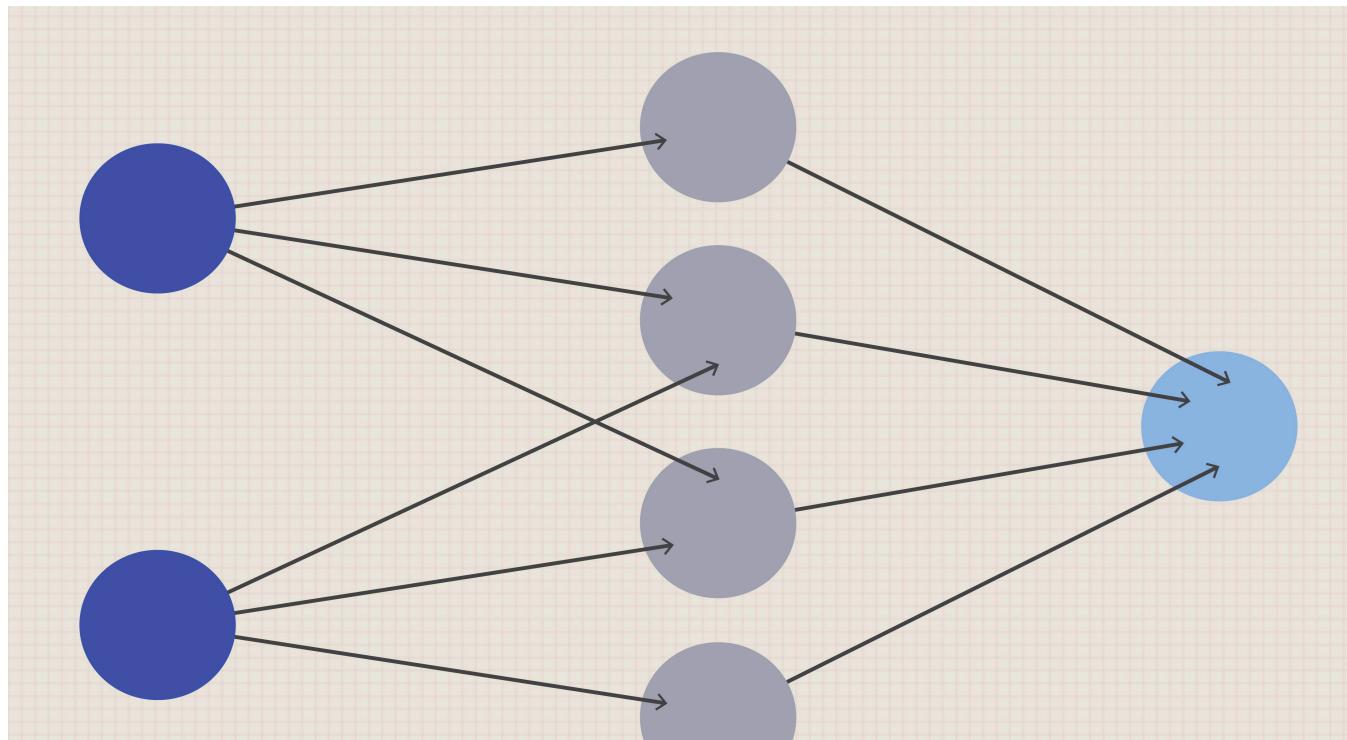


# Components of Neural Network

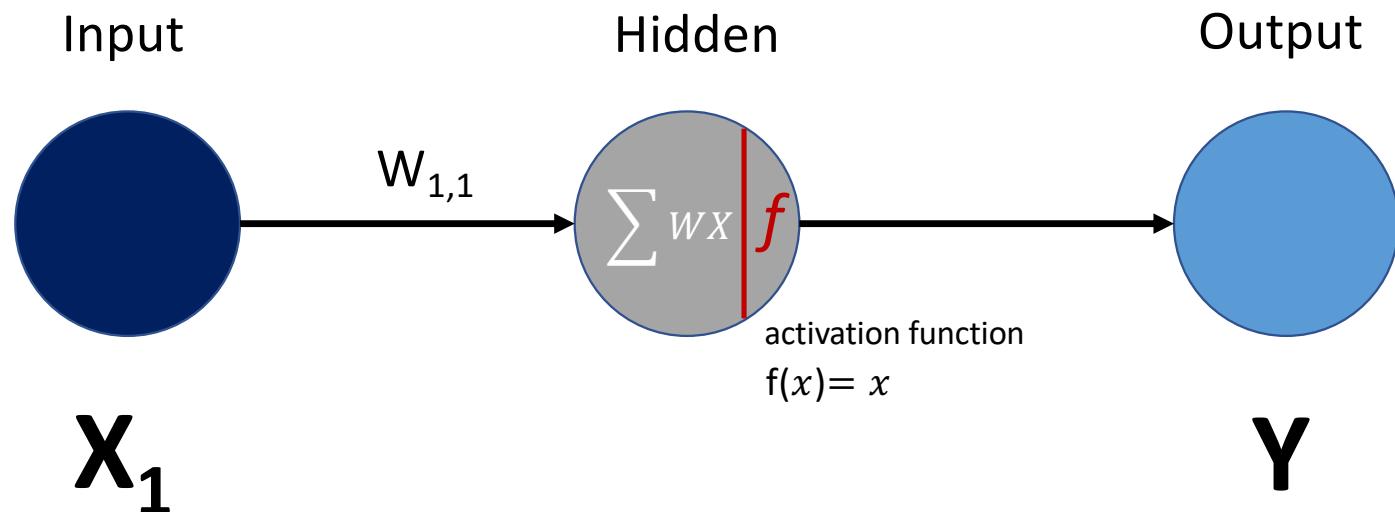
$x_1$

$x_2$

$y$



# Simple Neural Network

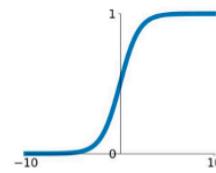


# Sidenote: Activation Function

Many different functions. Generally non-linear, for mathematical reasons

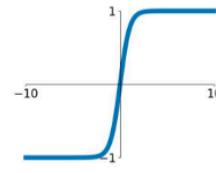
## Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



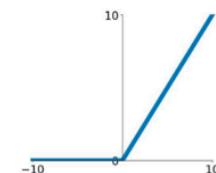
## tanh

$$\tanh(x)$$



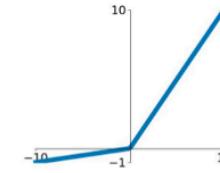
## ReLU

$$\max(0, x)$$



## Leaky ReLU

$$\max(0.1x, x)$$

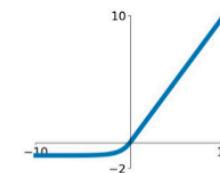


## Maxout

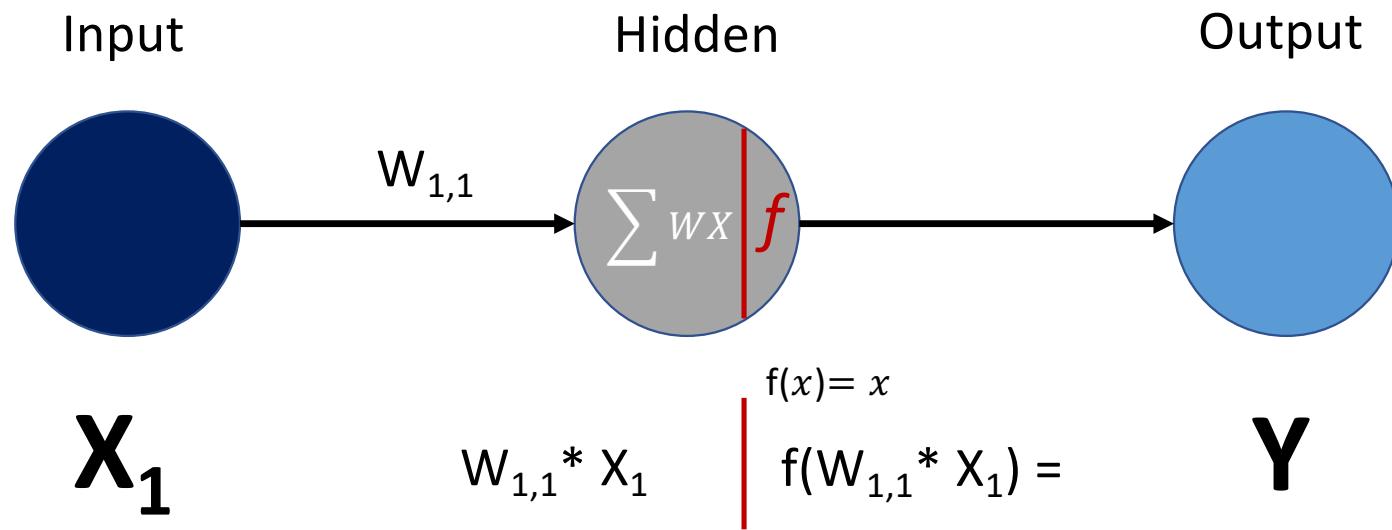
$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

## ELU

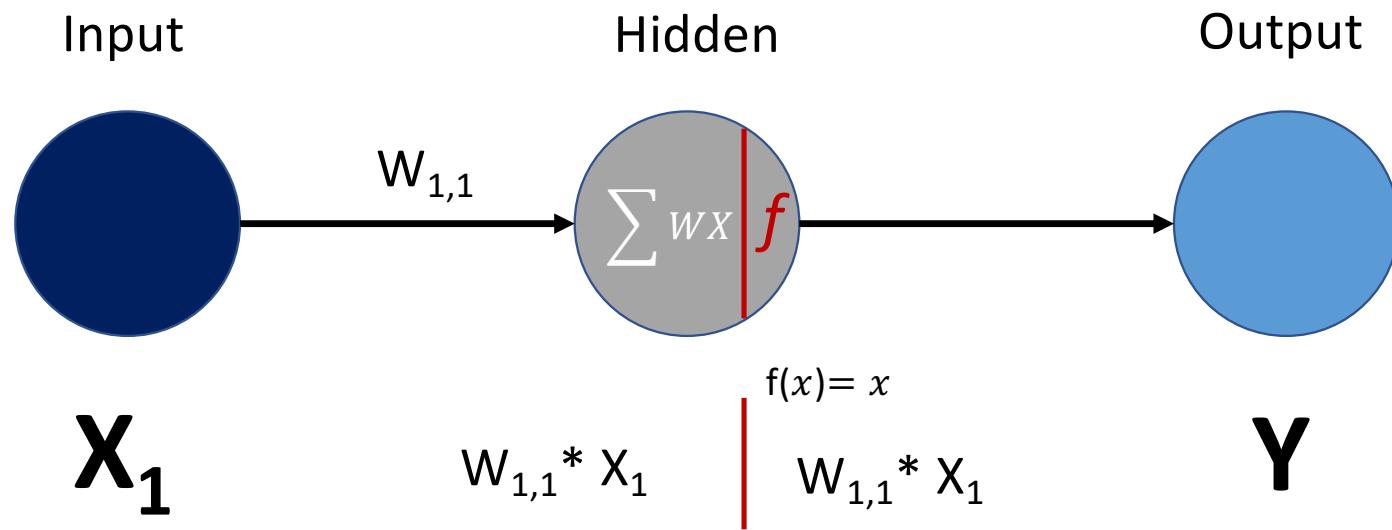
$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



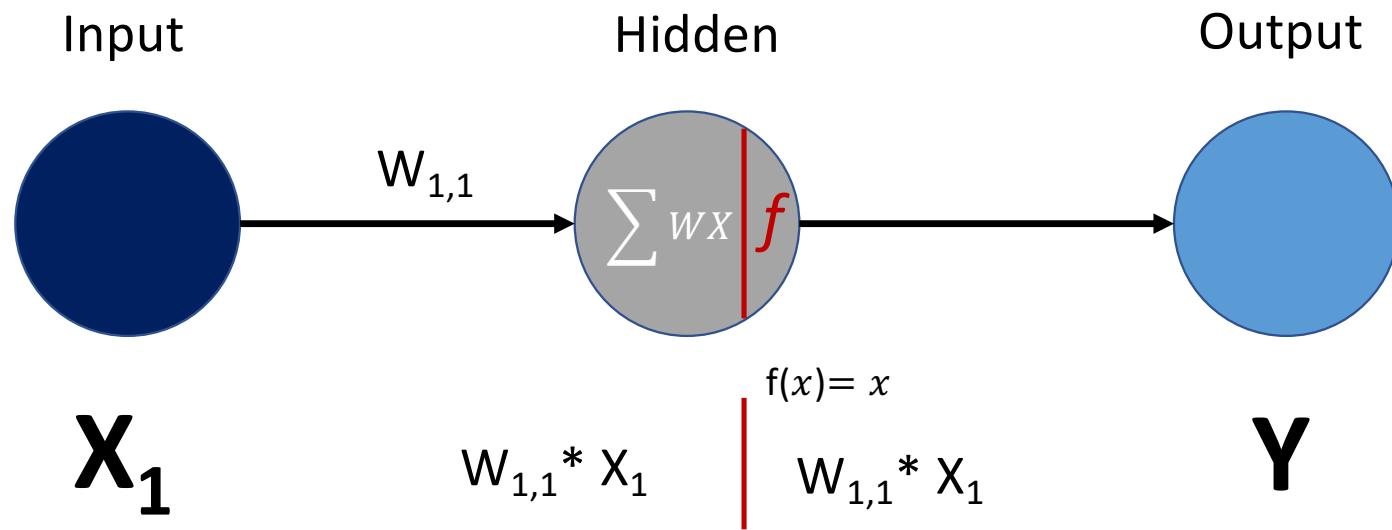
# Simple Neural Network



# Simple Neural Network



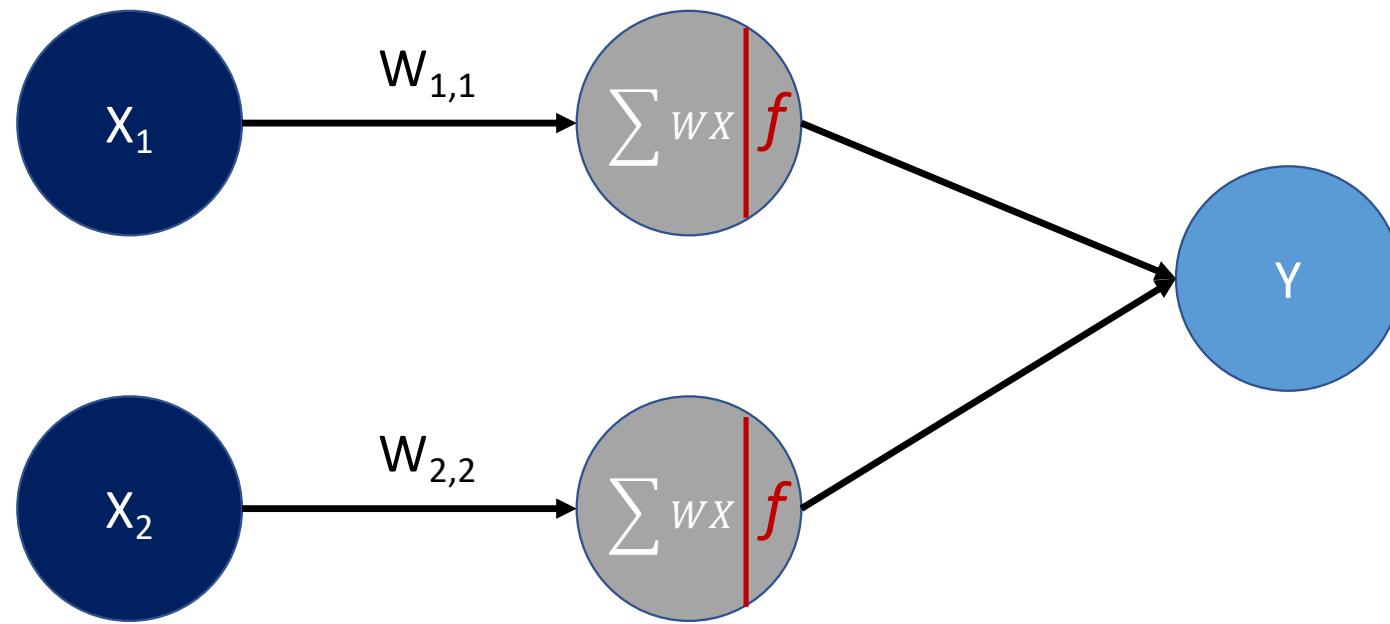
# Simple Neural Network



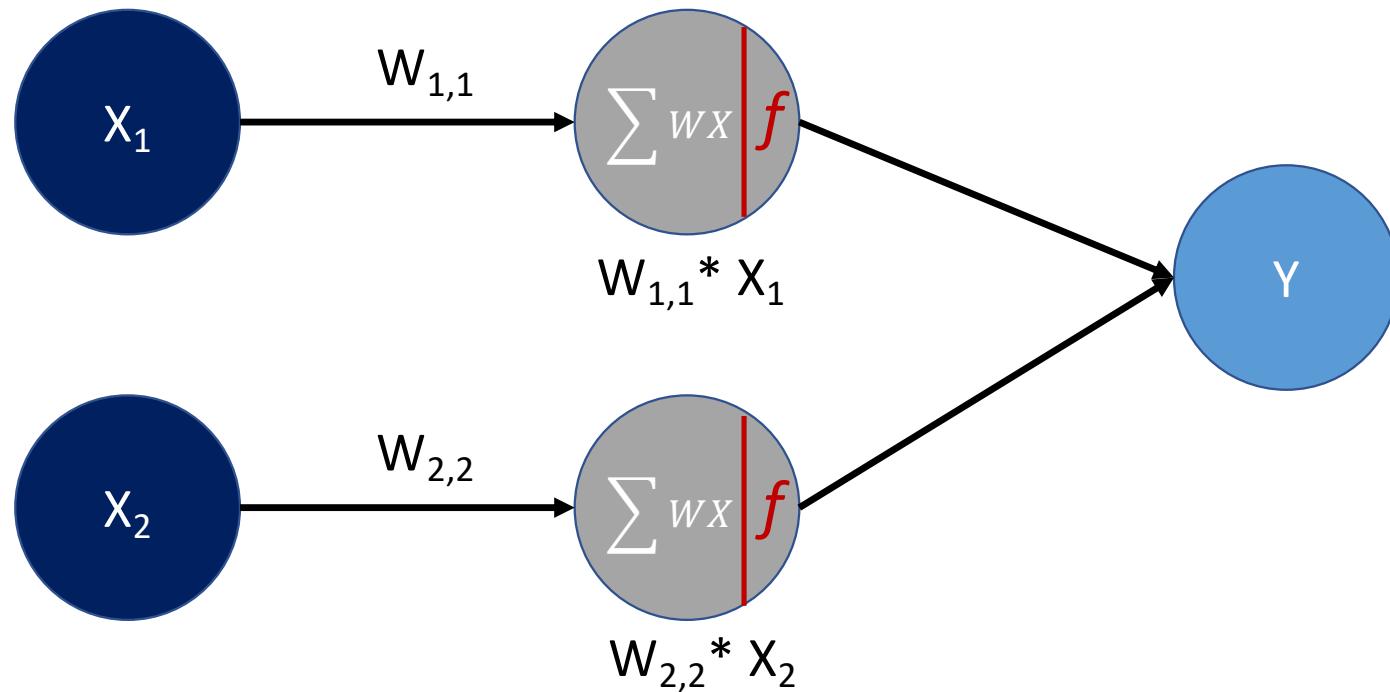
$$Y = W_{1,1} * X_1$$

Univariate Linear Regression

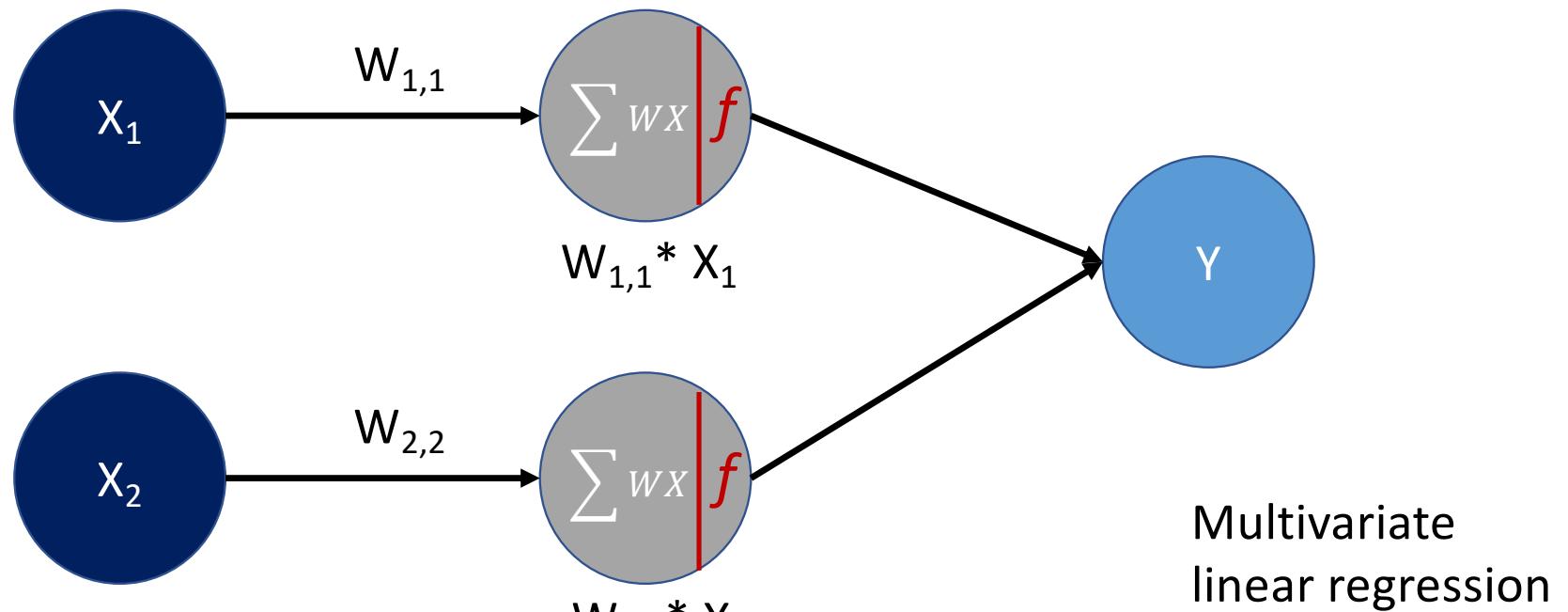
# A Slightly Complicated Neural Network



# A Slightly Complicated Neural Network

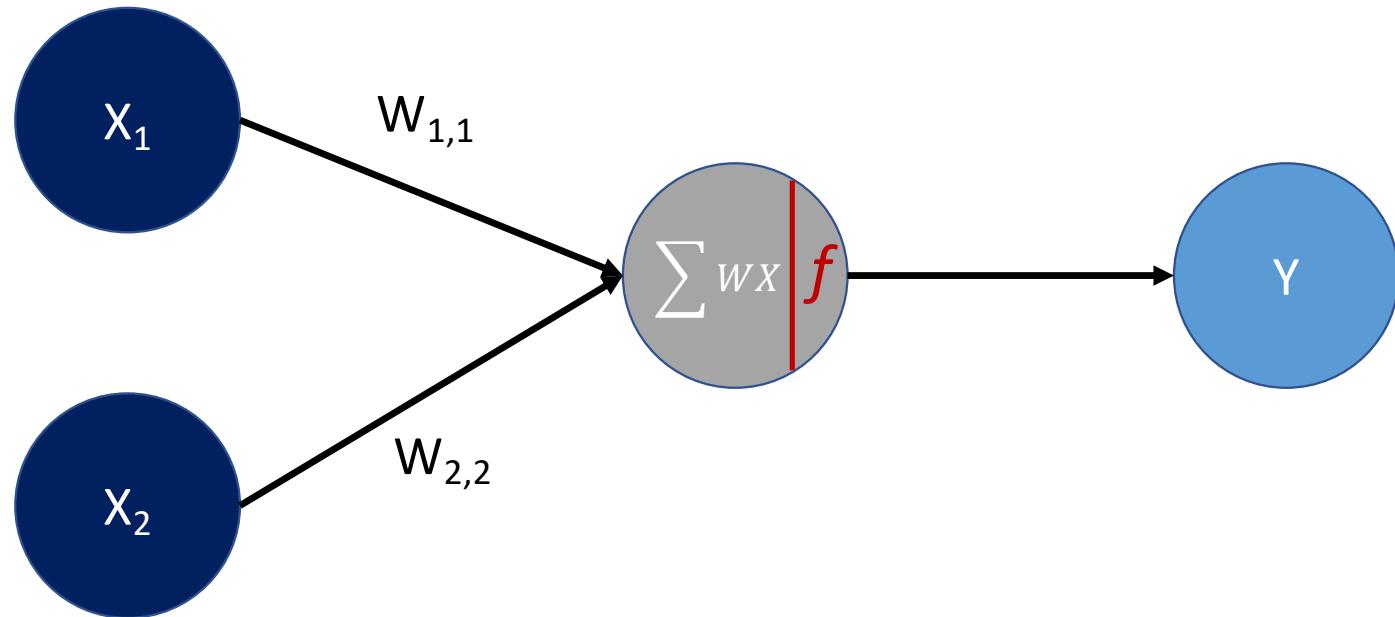


# A Slightly Complicated Neural Network

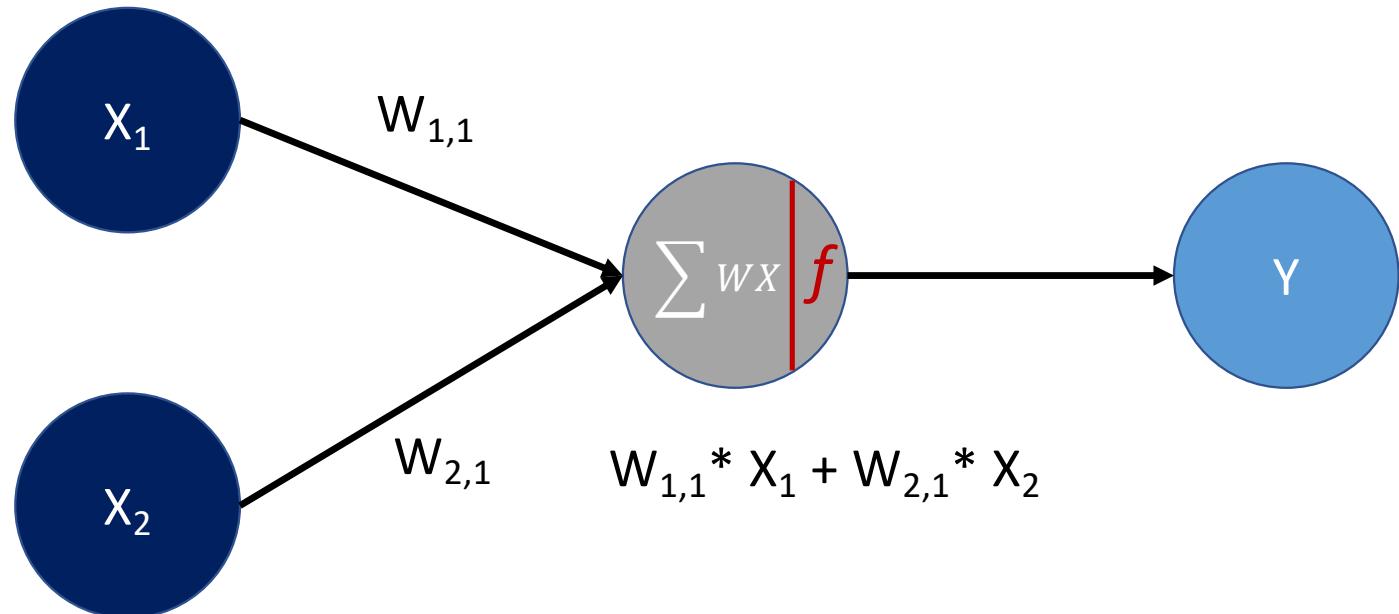


$$Y = W_{1,1} * X_1 + W_{2,2} * X_2$$

## A Slightly Complicated Neural Network(2)



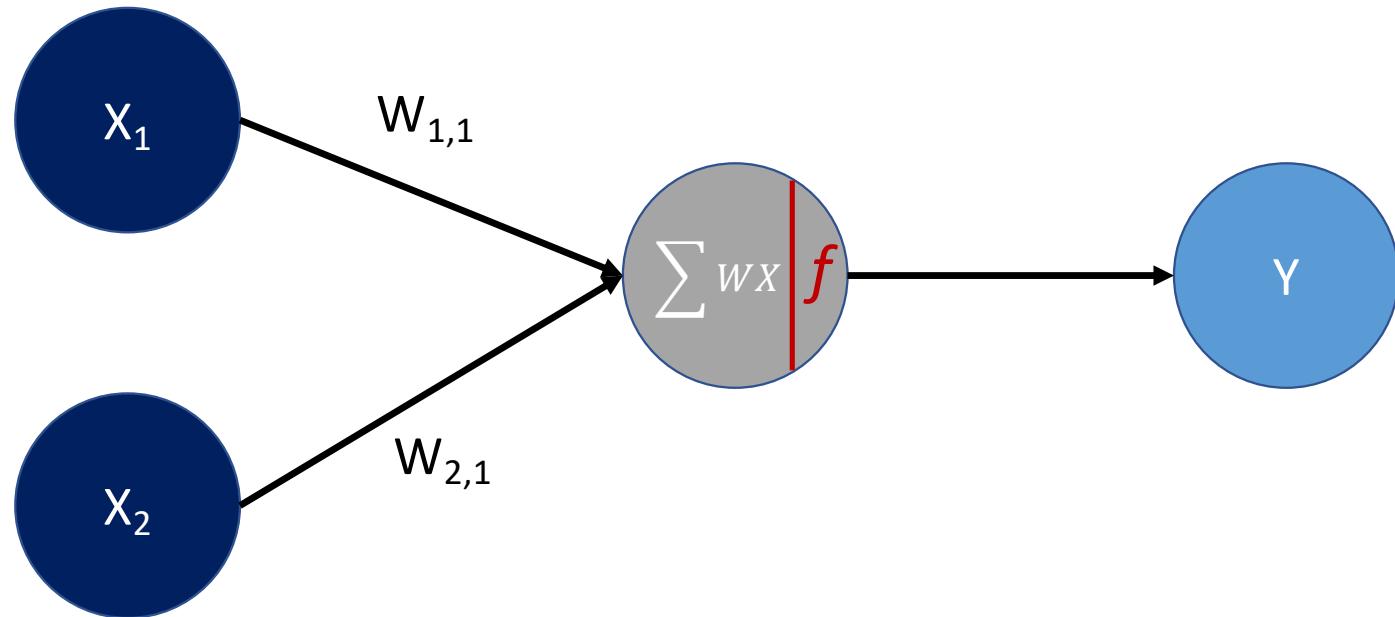
## A Slightly Complicated Neural Network(2)



Also Multivariate  
linear regression

$$Y = W_{1,1} * X_1 + W_{2,1} * X_2$$

# Objective of ‘training’ in Neural Network

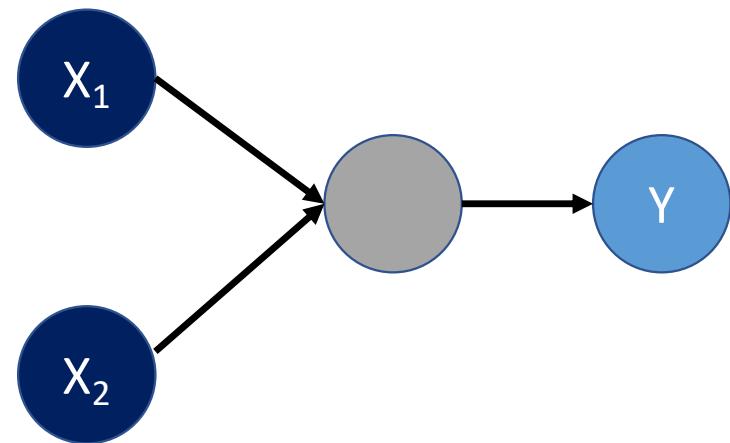


To learn what values of  $W_{1,1}$  and  $W_{2,1}$  would result in the best prediction of  $Y$ .

# How to train Neural Network?

1. (1) Assume random values for  $W_{i,j}$
2. Calculate  $Y_{predicted}$
3. Calculate error ( $Y_{predicted} - Y_{actual}$ )
4. Change values for  $W_{i,j}$  and repeat (2) and (3)

Choose  $W_{i,j}$  with the lowest error



# How to train Neural Network *faster*?

- Backpropagation and Gradient Descent

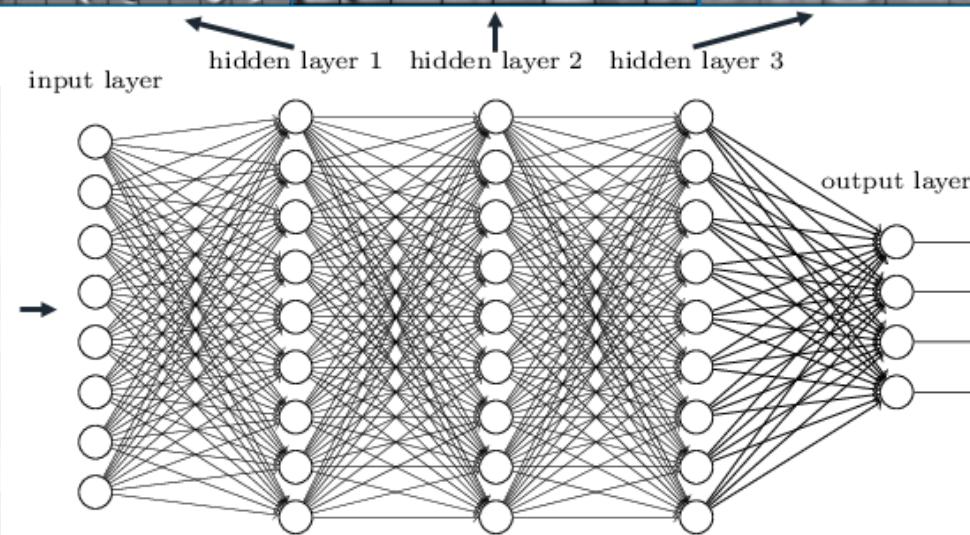
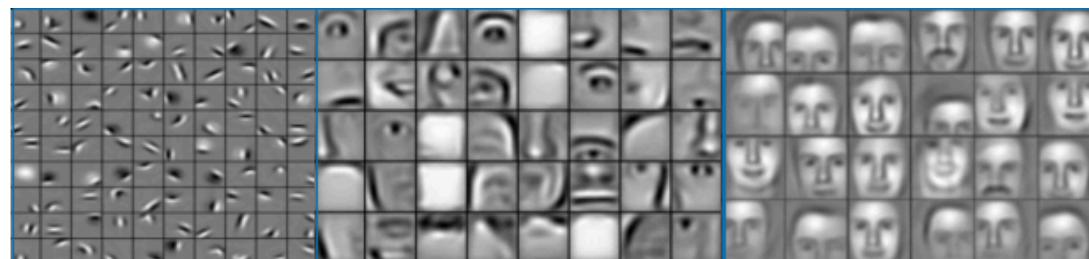
# Neural Network Topology

- You can change the number of nodes and the connections to achieve different network topology

# Neural Network Topology

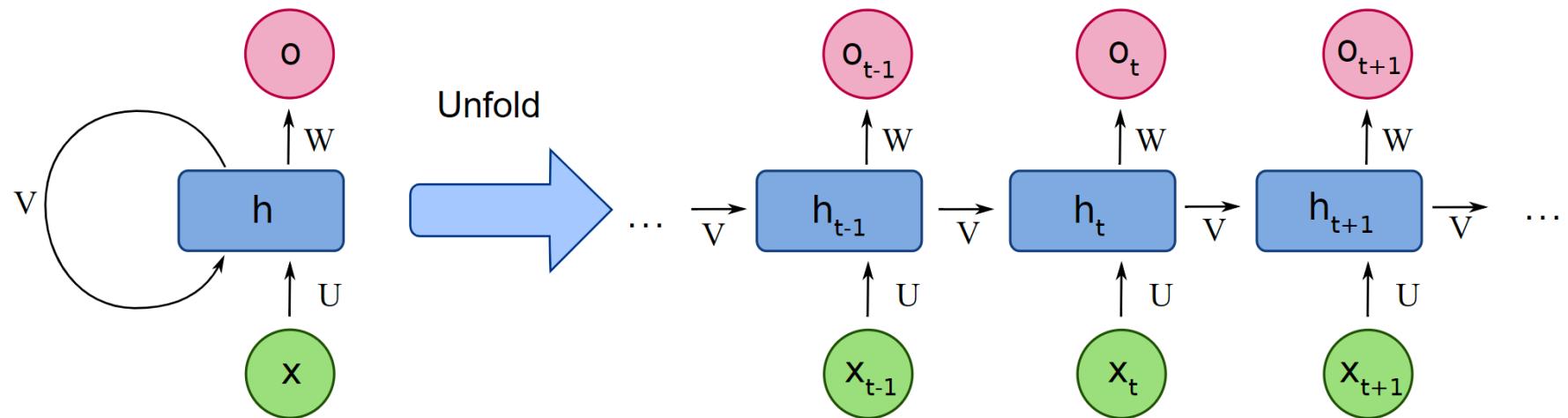
- Increase depth / number of hidden layers (aka Deep Learning)

Deep neural networks learn hierarchical feature representations



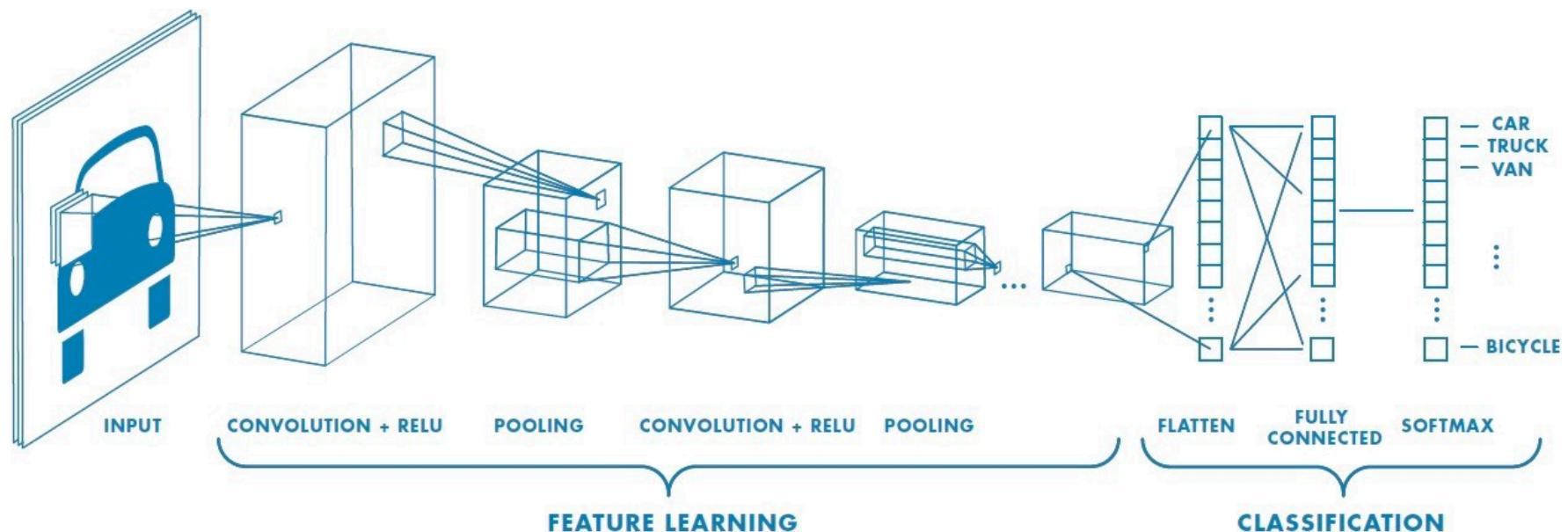
# Neural Network Topology

- Recurrent Neural Network



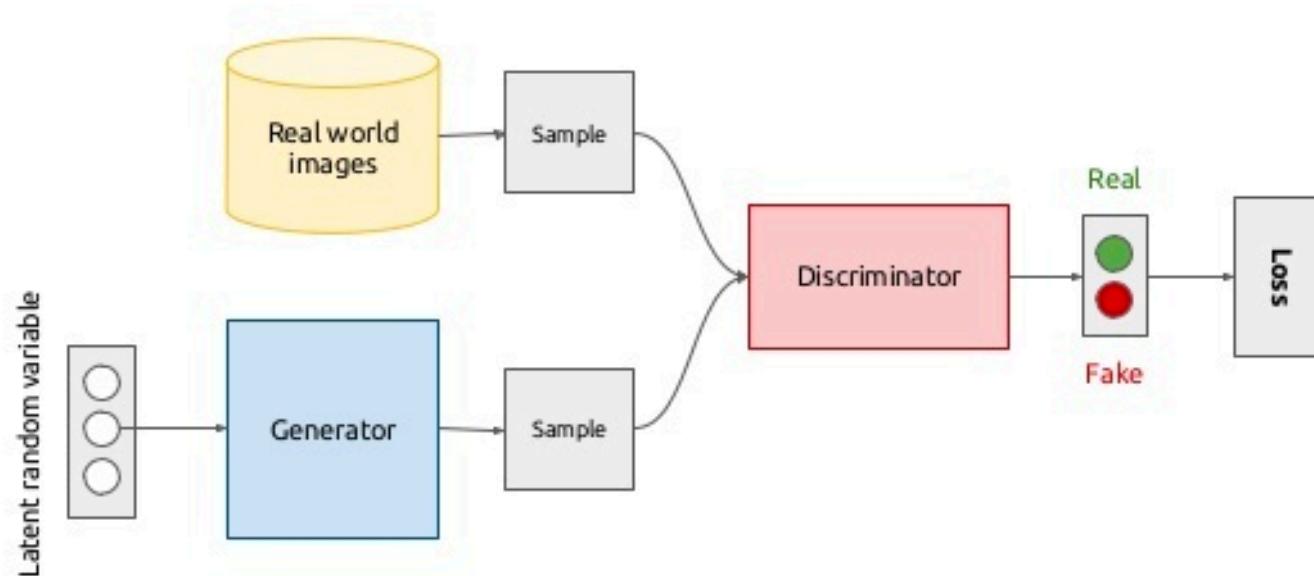
# Neural Network Topology

- Convolutional Neural Network



# Other Neural Network Advances

- Generative Adversarial Networks



# Other Neural Network Advances

- Reinforcement Learning

