



Neural Networks

# Input

Data vector  $x = (x_1, \dots, x_n)$

# Output

## Linear function of $x$

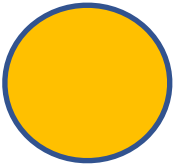
# Activation

Map output into range  
(e.g., probabilities  $[0,1]$ )

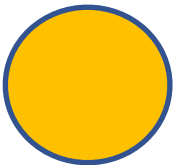
# Training

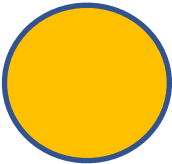
Learn model  $\theta = (W, b)$

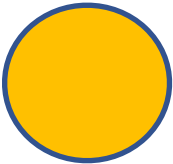
31

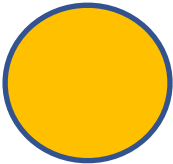


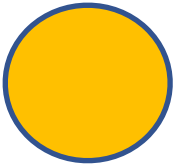












x

1

x

2

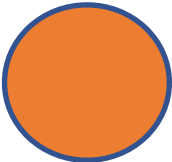


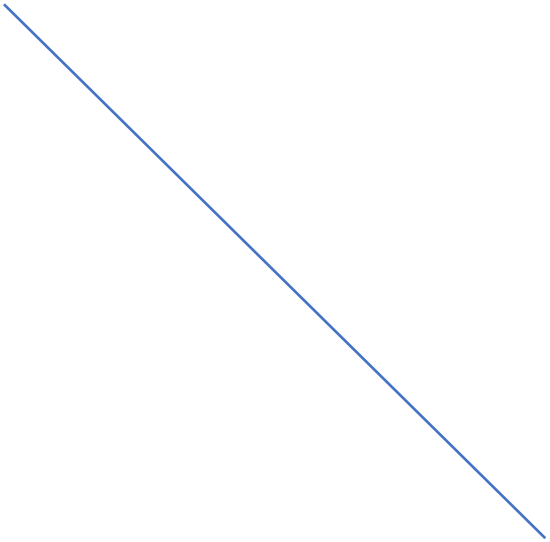


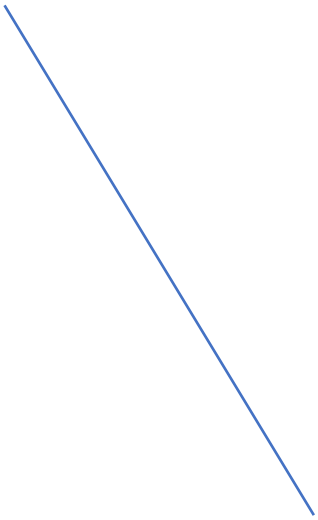
x

n



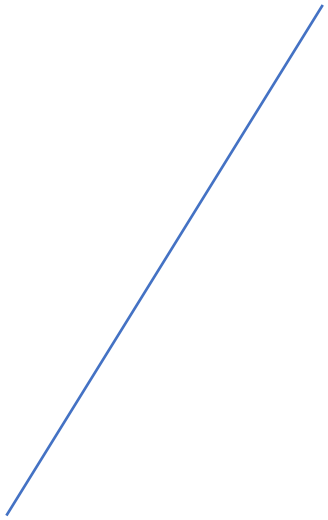




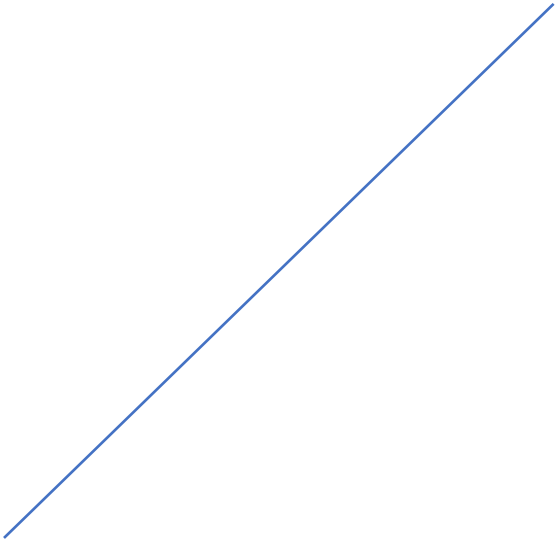










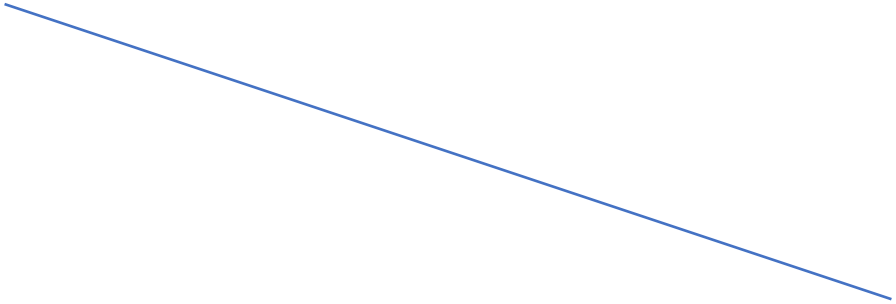


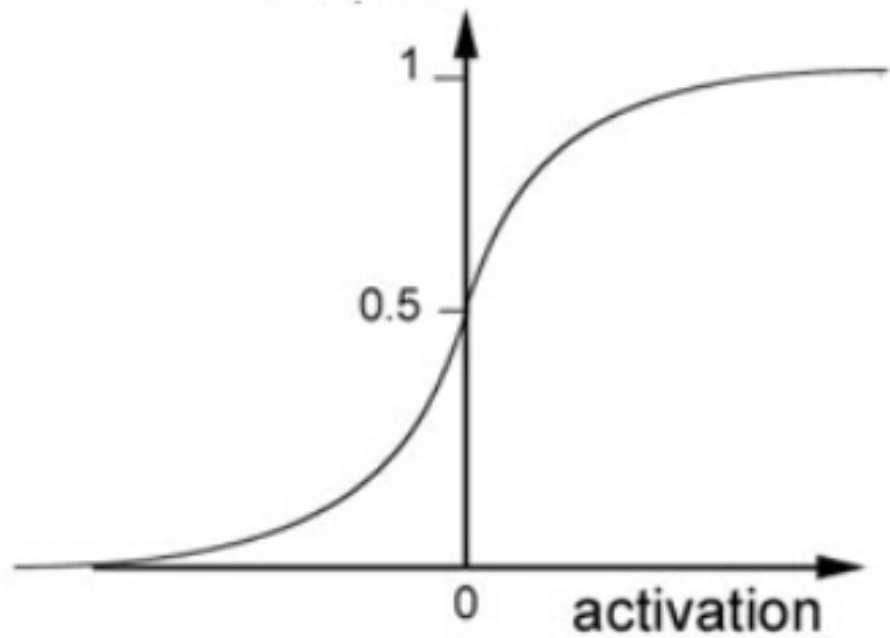
Weights  $w$

$$f(x) \equiv \sigma(N * x + b)$$

output output

Bias  $b$





# Neural Networks

## Input

Data vector  $x = (x_1, \dots, x_n)$

## Output

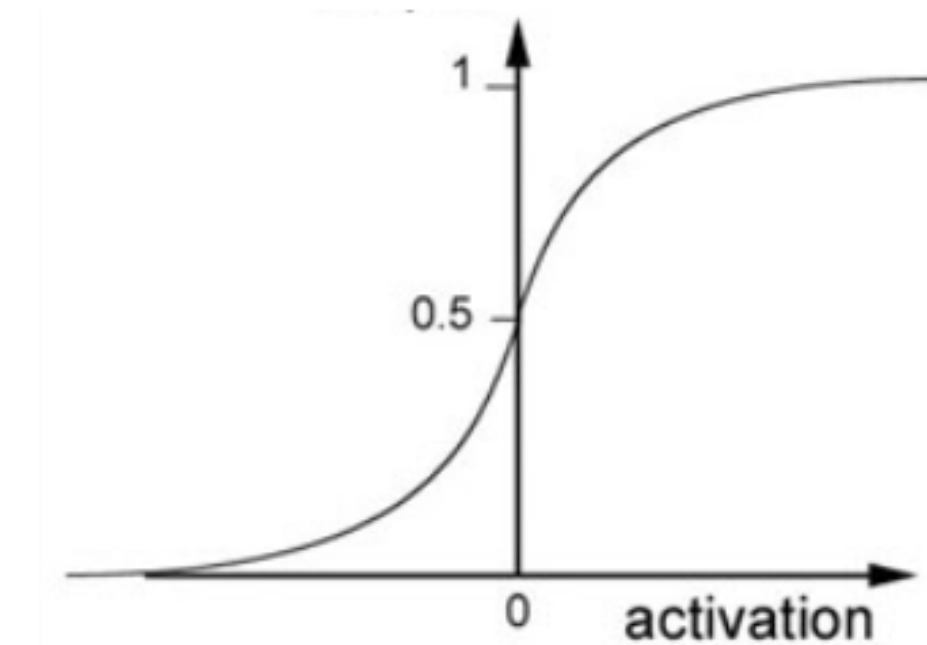
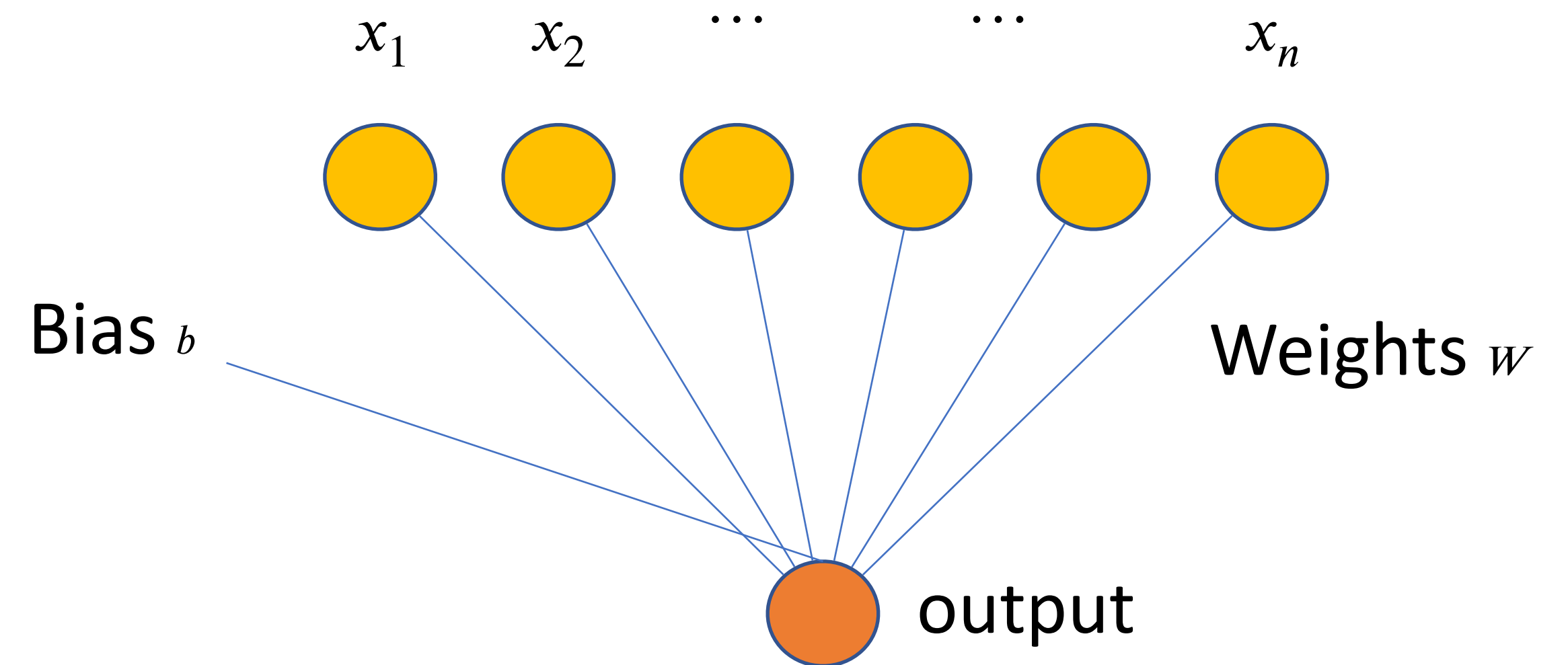
Linear function of  $x$

## Activation

Map output into range  
(e.g., probabilities [0,1])

## Training

Learn model  $\theta = (W, b)$
















$$f(x) = \sigma(W * x + b)$$

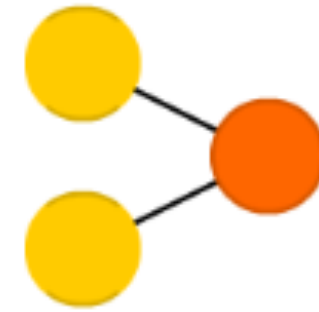


# A mostly complete chart of Neural Networks

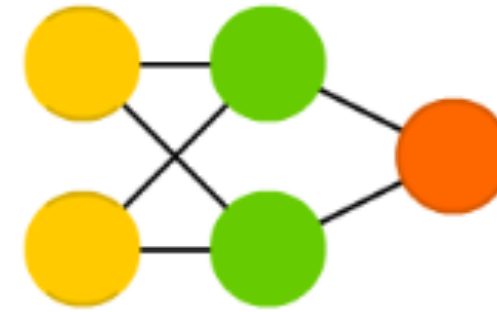
©2016 Fjodor van Veen - asimovinstitute.org

-  Backfed Input Cell
-  Input Cell
-  Noisy Input Cell
-  Hidden Cell
-  Probablistic Hidden Cell
-  Spiking Hidden Cell
-  Output Cell
-  Match Input Output Cell
-  Recurrent Cell
-  Memory Cell
-  Different Memory Cell
-  Kernel
-  Convolution or Pool

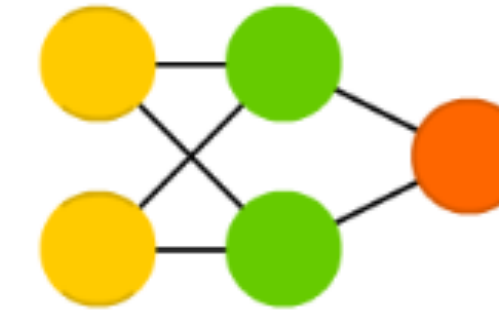
Perceptron (P)



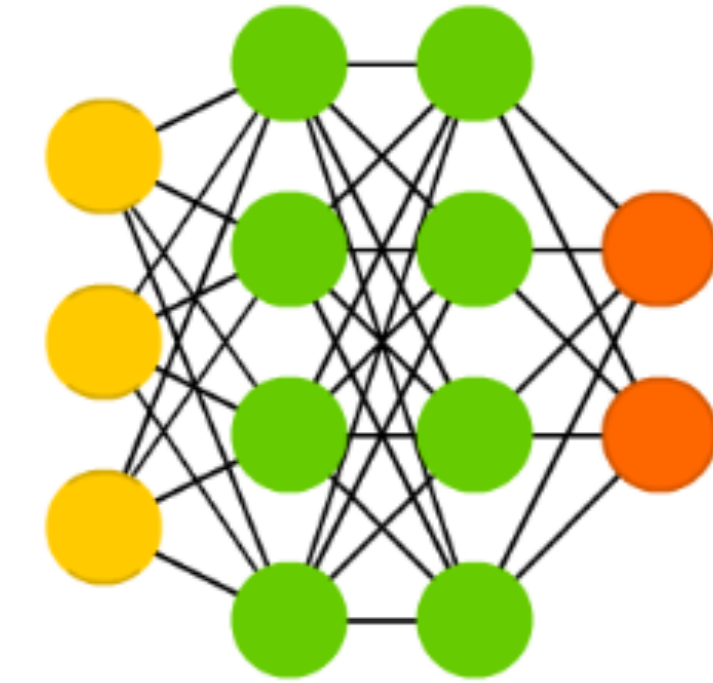
Feed Forward (FF)



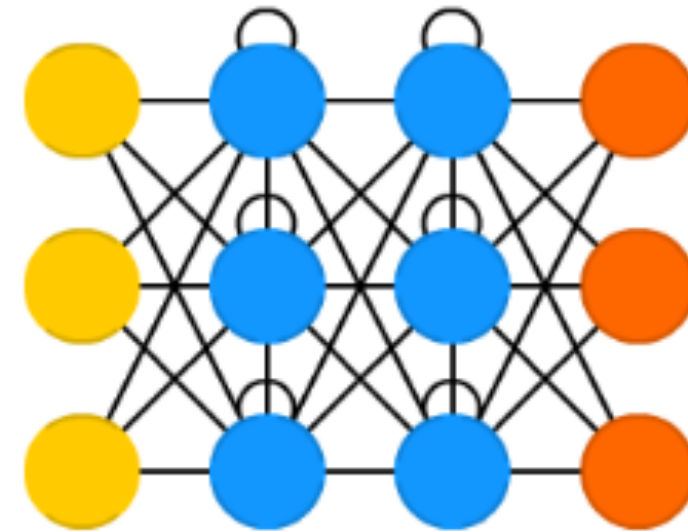
Radial Basis Network (RBF)



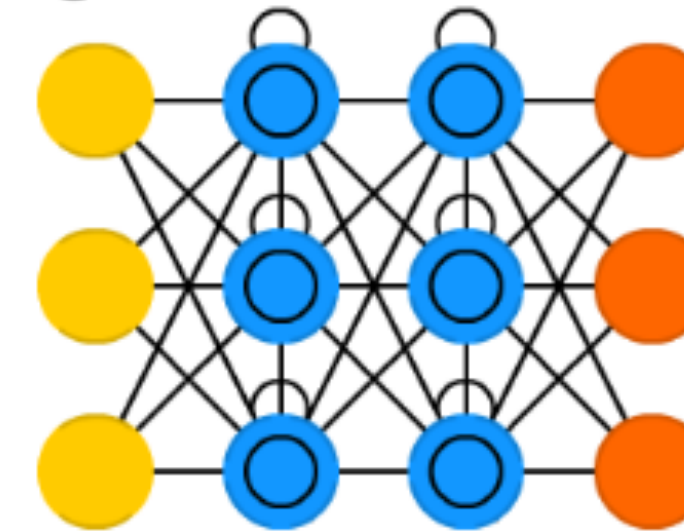
Deep Feed Forward (DFF)



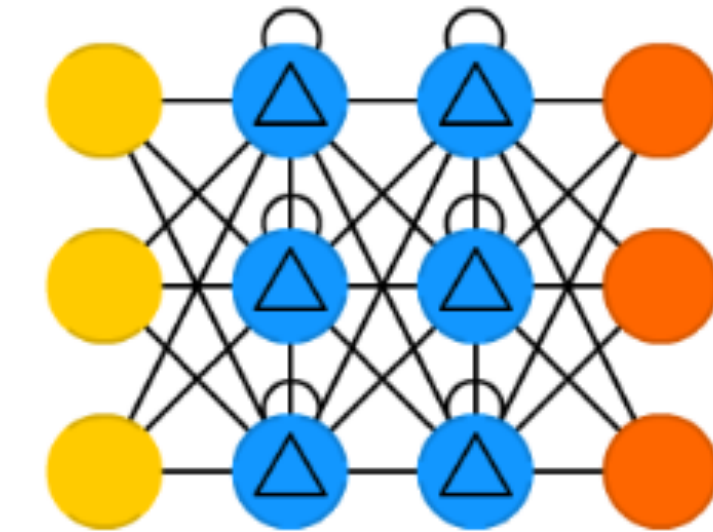
Recurrent Neural Network (RNN)



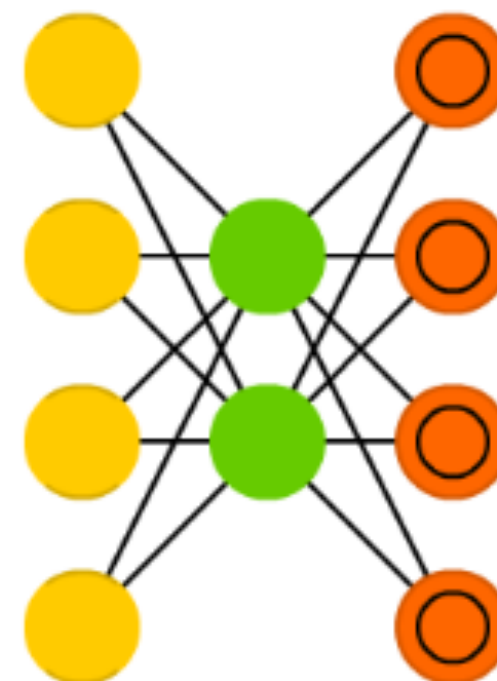
Long / Short Term Memory (LSTM)



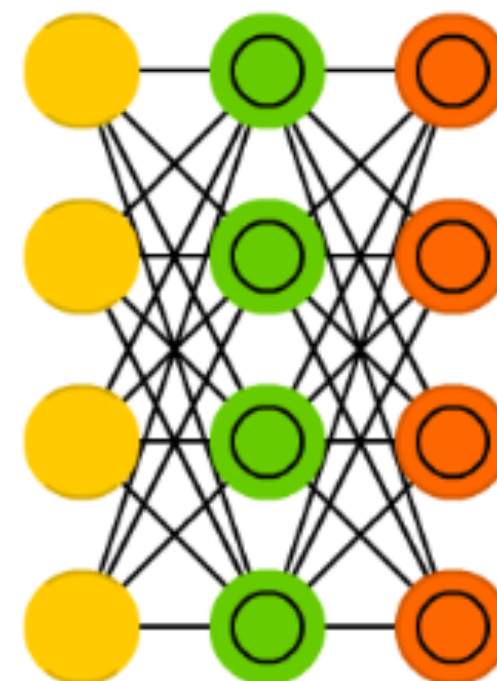
Gated Recurrent Unit (GRU)



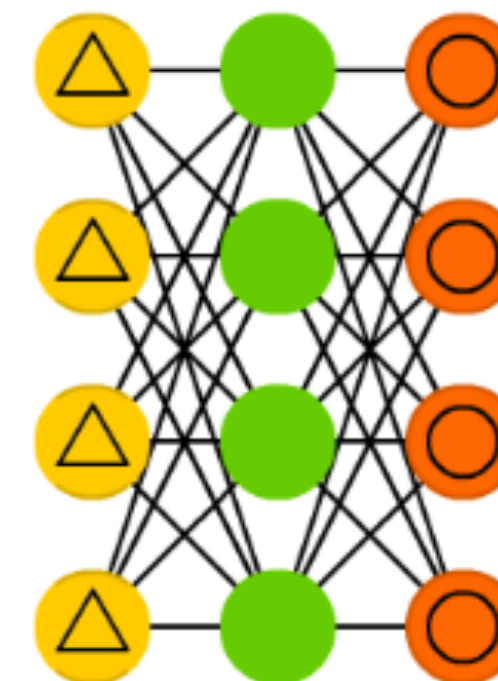
Auto Encoder (AE)



Variational AE (VAE)



Denoising AE (DAE)



Sparse AE (SAE)

