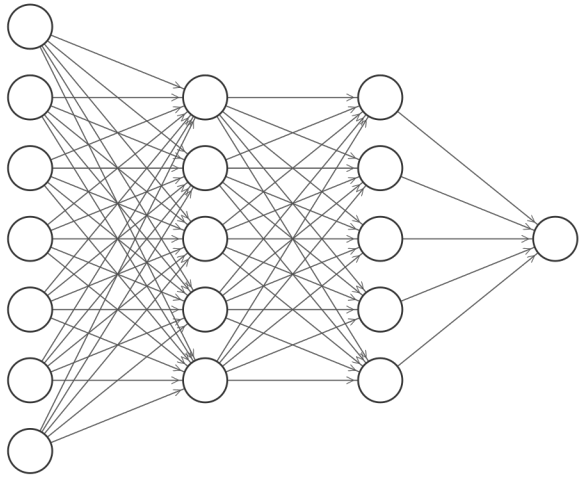
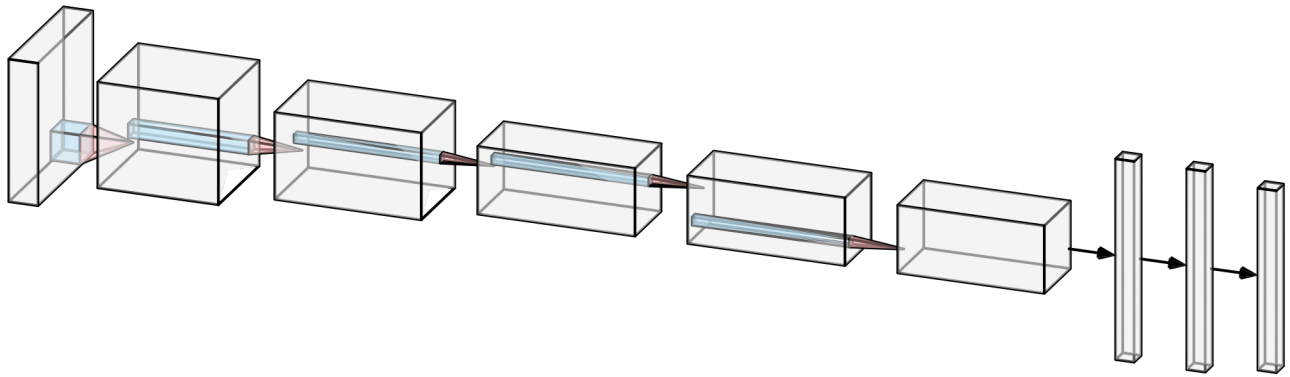
Overview, Schedule & Requirements

What are neural networks?

- (Artificial) Neural Networks are a Machine Learning model, “inspired” by the network of biological neurons in the brain.
- Adept at diverse tasks.
- Able to process large corpora of data, of different formats.

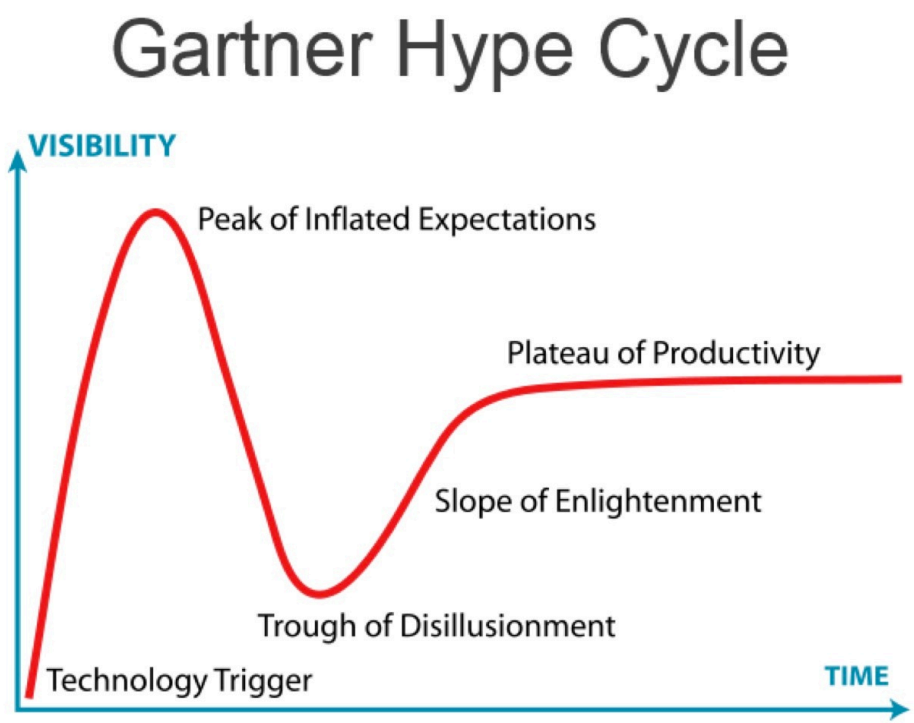


Input Layer $\in \mathbb{R}^7$ Hidden Layer $\in \mathbb{R}^5$ Hidden Layer $\in \mathbb{R}^5$ Output Layer $\in \mathbb{R}^1$



A (very) brief history of neural networks

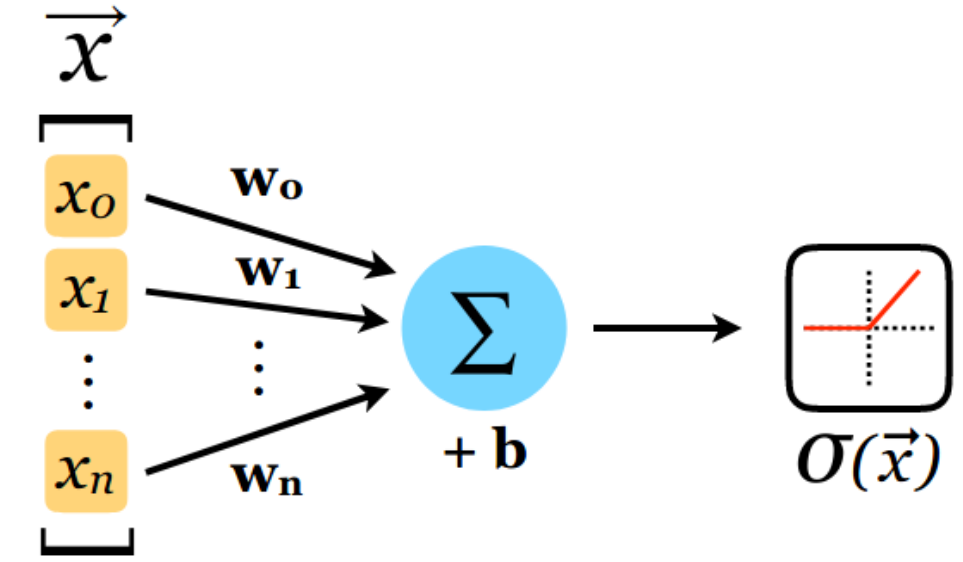
- 1943: McCulloch & Pitts model of biological neurons.
- 1957- Rosenblatt's Perceptron.
- 1969- First wave dies out. ☹
- 1984- Backpropagation.
- 1989- CNNs introduced.
- 1995(ish)- Alternate ML models outperform. Second wave dies out.☹
- 2010's- Successes in ImageNet Large Scale Visual Recognition Challenge.



Perceptrons (Neanderthal Neurons)

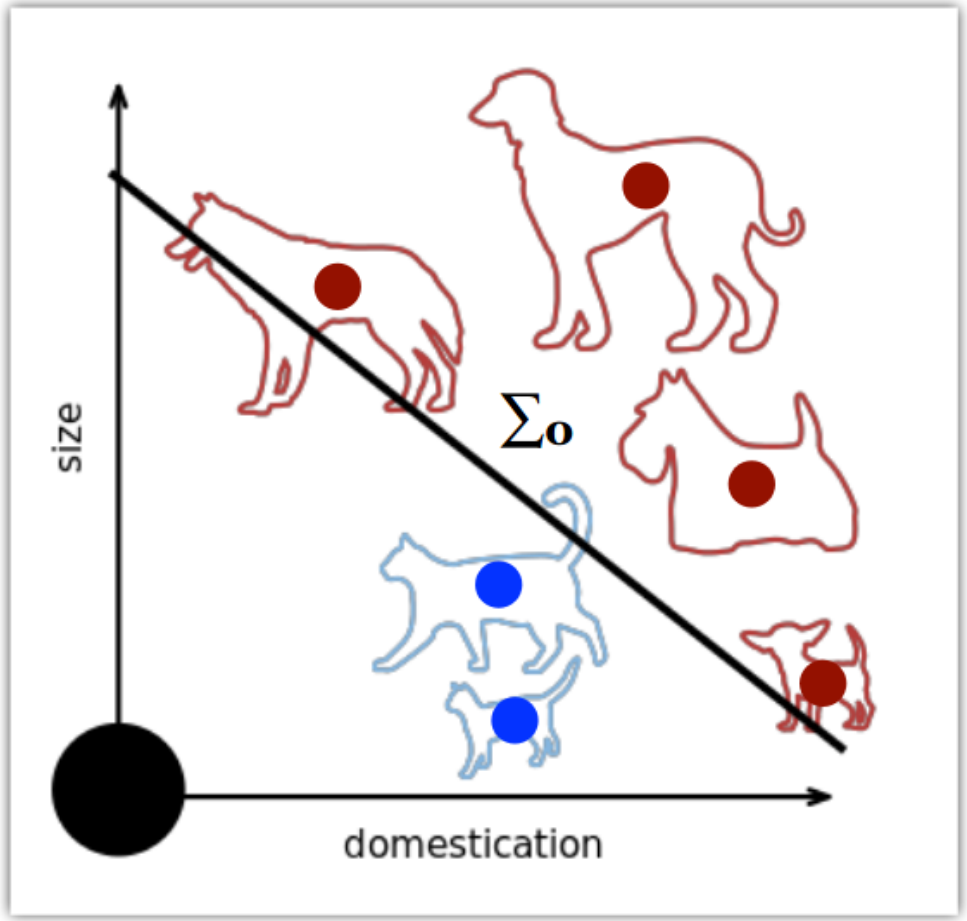
The basic unit of a neural net is the ***perceptron*** (loosely based on a real neuron)

Takes in a vector of inputs (x). Commonly inputs are summed with weights (w) and offset (b) then run through activation.



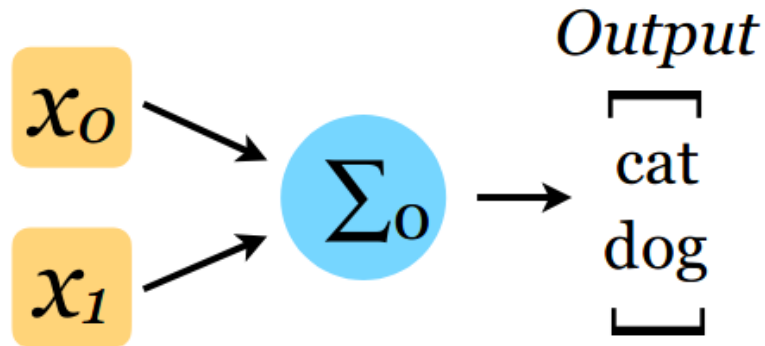
$$\sigma(\vec{x}) = \begin{cases} \vec{w}_i \cdot \vec{x} + b_i & \vec{w}_i \cdot \vec{x} + b_i \geq 0 \\ 0 & \vec{w}_i \cdot \vec{x} + b_i < 0. \end{cases}$$

Imagine using two features to separate cats and dogs



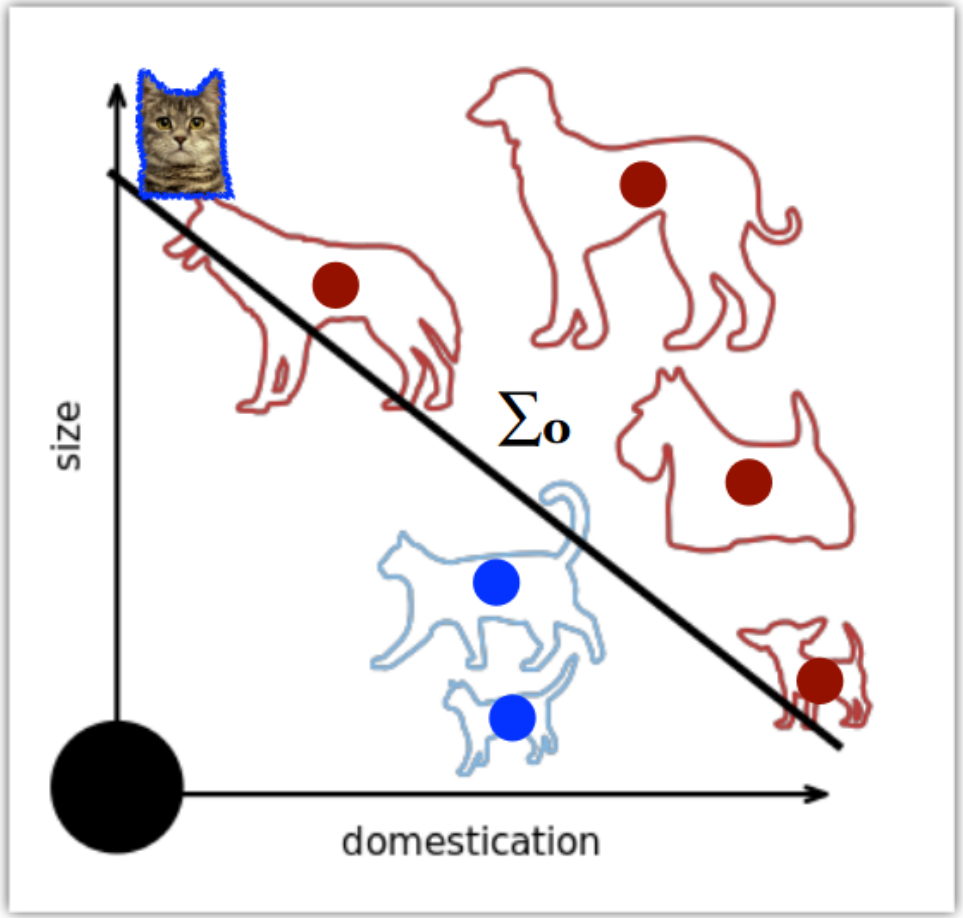
from [wikipedia]

$$\sigma(\vec{x}) = \begin{cases} \vec{w}_i \cdot \vec{x} + b_i & \vec{w}_i \cdot \vec{x} + b_i \geq 0 \\ 0 & \vec{w}_i \cdot \vec{x} + b_i < 0. \end{cases}$$

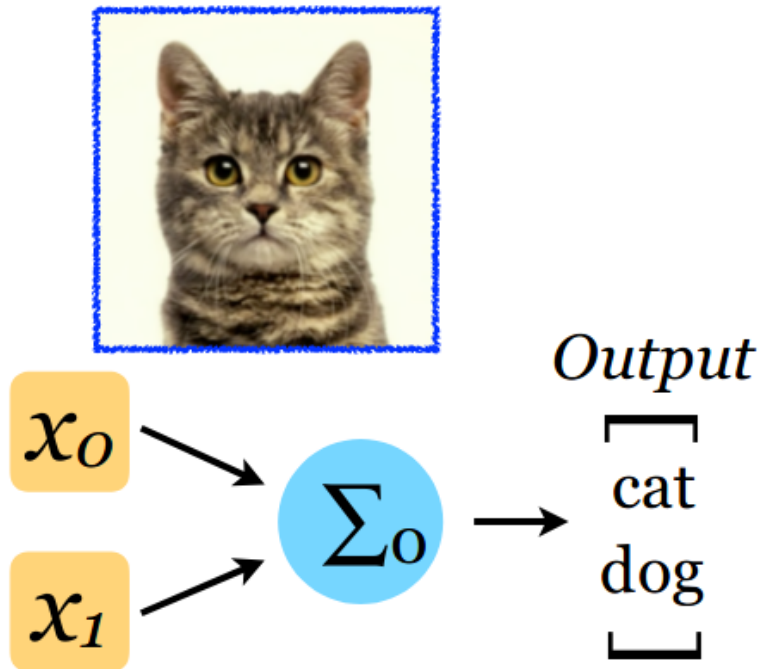


By picking a value for **w** and **b**, we define a boundary between the two sets of data

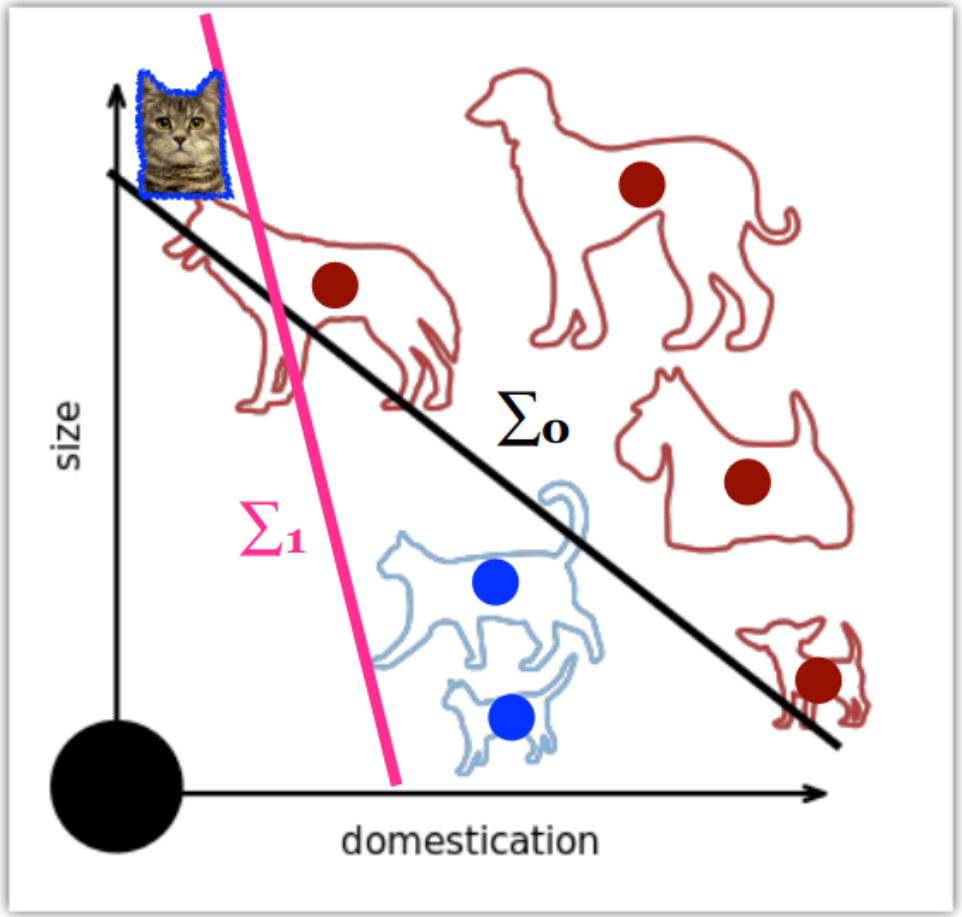
What if we have a new data point?



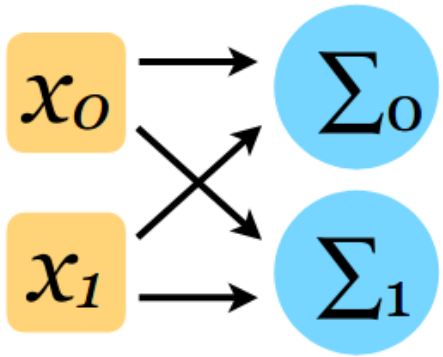
from [wikipedia]



What if we have a new data point?

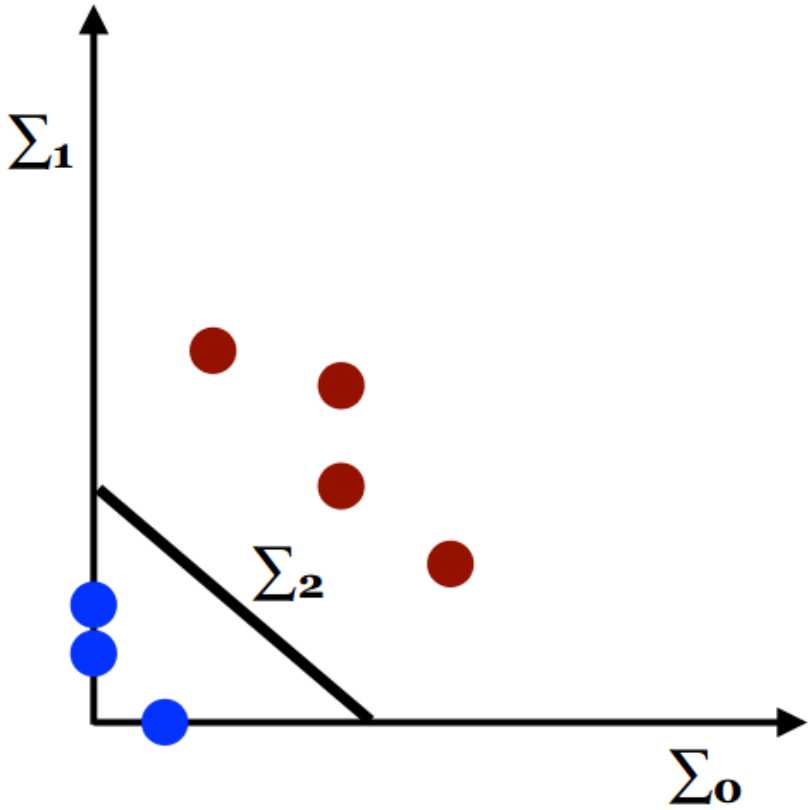


from wikipedia

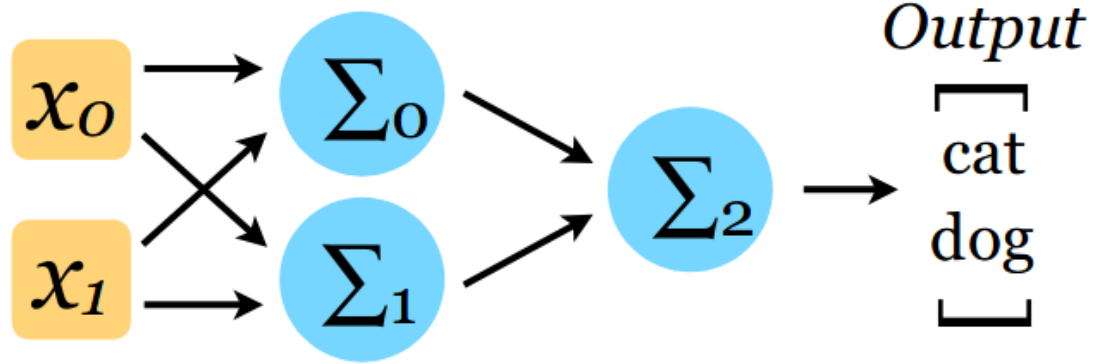


We can **add another perceptron** to help (but does not yet solve the problem)

What if we have a new data point?

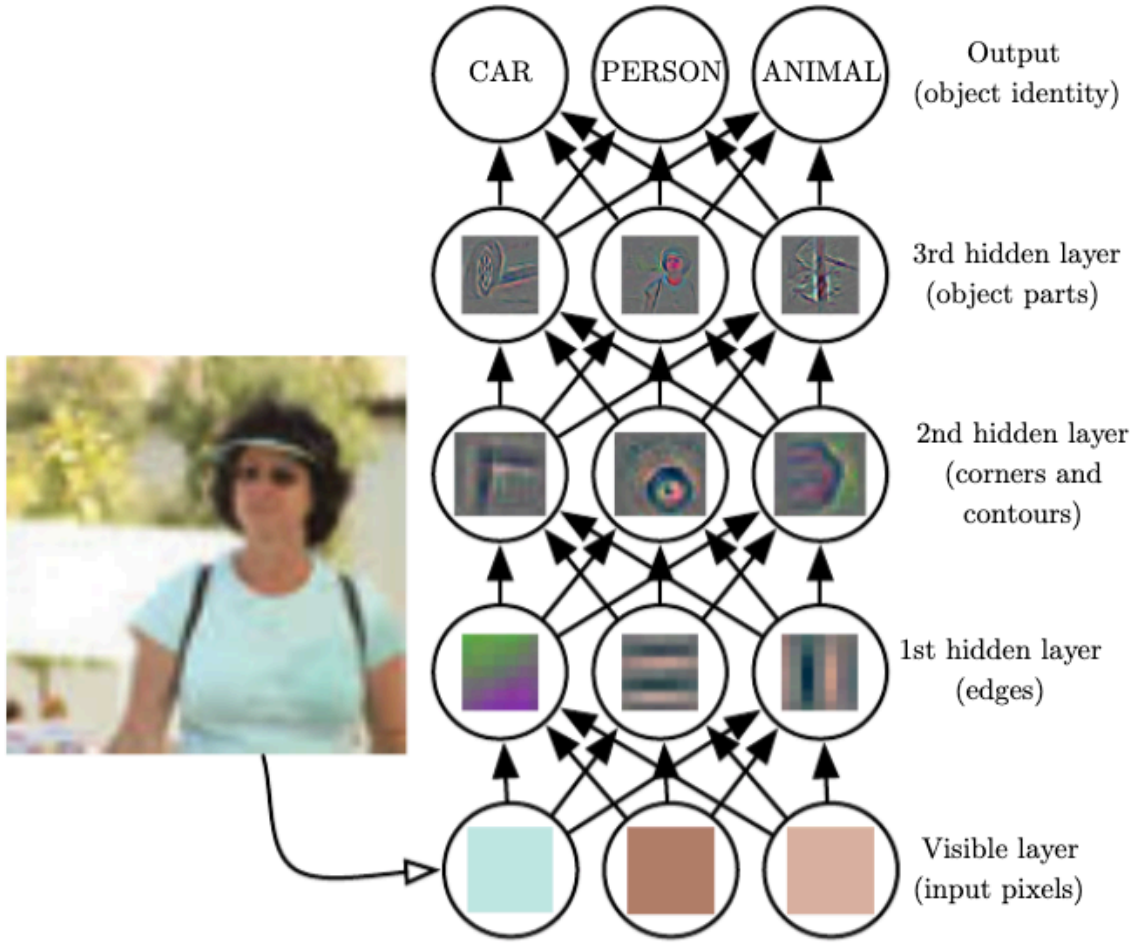


from wikipedia

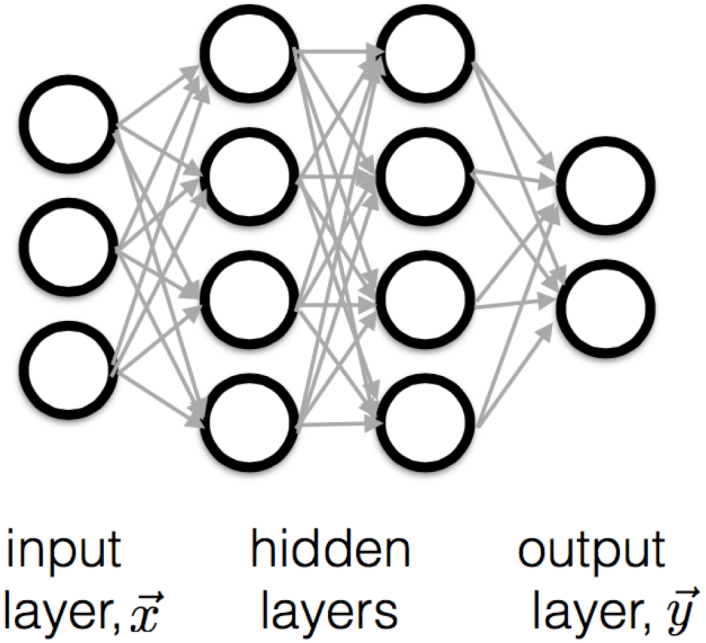


Another layer can classify based on preceding layer's output (of **non-linear activation**)

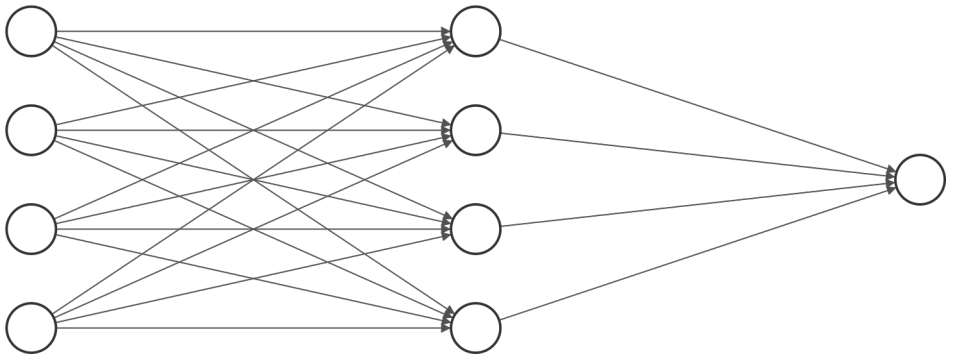
Hierarchical, Feature Representation Learning



Multi-Layer Perceptron
(aka Dense/Fully Connected Neural Network)



What is a neural network



Input Layer $\in \mathbb{R}^4$ Hidden Layer $\in \mathbb{R}^4$ Output Layer $\in \mathbb{R}^1$

$output$: $\hat{Y} = \sigma(W'\sigma(WX + b) + b')$

$input$: $\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$

$Layer\ 1$: $\begin{bmatrix} W_{11} & \cdots & W_{14} \\ \vdots & \ddots & \vdots \\ W_{41} & \cdots & W_{44} \end{bmatrix}$

Layer 2: $\begin{bmatrix} W'_1 & W'_2 & W'_3 & W'_4 \end{bmatrix}$

Training a neural network

- Forward Pass: Take a batch of data, process data through all the layers to get (current) predictions.
- Backward Pass: Compute error on predictions,
  Then compute the contribution of each parameter to the error by applying the chain rule,
  Apply gradient descent to tweak all the weights.

- Inference: Have trained neural network,
  Have samples that we need to predict,
  perform forward pass.