An introduction to neural networks

Yurui Ming (yrming@gmail.com)

CIBCI Lab, FEIT, UTS

How computer views an image

- What you mean a colourful world is just a number array for computer!
  - Your brain which is the outcome of millions of years evolution knowing how to deal with optical signal doesn't mean computer knows how to cope with this digital numbers effectively.

67	67	67	67	67	68	68	68	68
71	71	71	71	71	72	72	72	72
77	77	77	77	77	79	79	79	79
80	80	80	80	80	82	82	82	82
84	84	84	84	84	86	86	86	86
90	90	90	90	90	90	90	90	90
92	92	92	92	92	92	92	93	93
97	97	97	97	97	97	97	97	98
100	101	100	101	100	101	101	102	101
103	104	103	104	103	104	104	105	104
106	106	106	106	106	106	106	107	107
110	110	110	110	110	110	110	110	110
111	111	111	111	111	111	111	112	112
115	115	115	115	115	116	116	116	116
116	116	116	116	116	118	118	118	118

How we can help the computer

- Let computer deal with the essential information by applying filters:
  - Suppose to have computer recognise the harbor bridge and opera house;
  - Suppose the filter is of the shape: [filter_height, filter_width, in_channels, out_channels].

A filter of size 1x1x3x1:
[[[[0.2989], [0.5870], [0.1140]]]]

A filter of size 3x3x1x1:
[[[[0.1667]], [[0.6667]], [[0.1667]]],
[[[0.1667]], [[-3.3333]], [[0.1667]]],
[[[0.1667]], [[0.6667]], [[0.1667]]]]

What's wrong with manually crafted filter

- Laborious for handcrafted filters
  - Limited orientation covered and restricted feature extraction capability

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

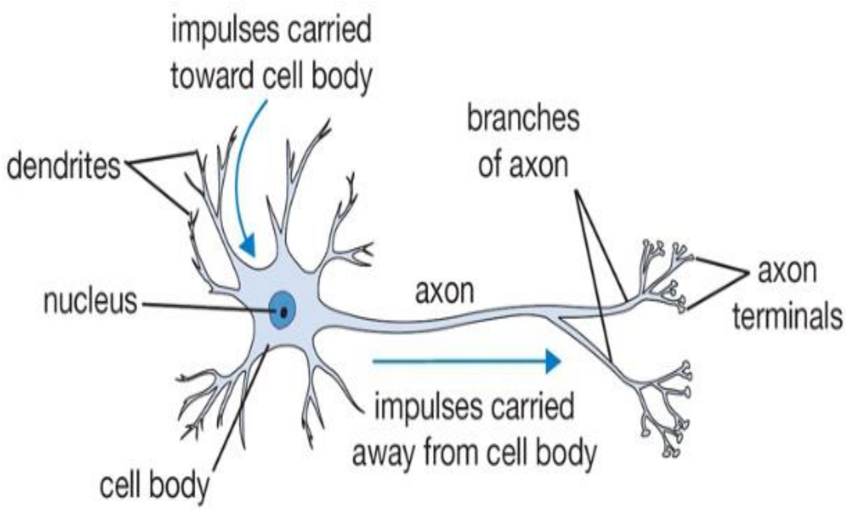
Sobel

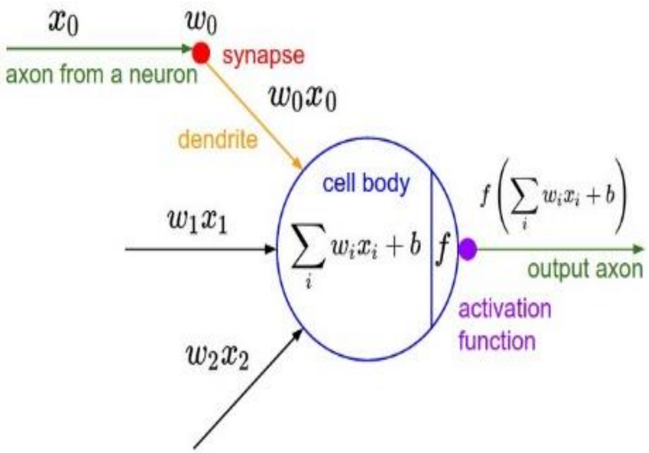
Retrospect of conventional image processing

- Any approach for automating the filter-design process?
- How mammals perceive images?
- What are the differences between computation performed by computer and human brain?
- Possibility of brain-inspired computation?

Artificial Neural Network

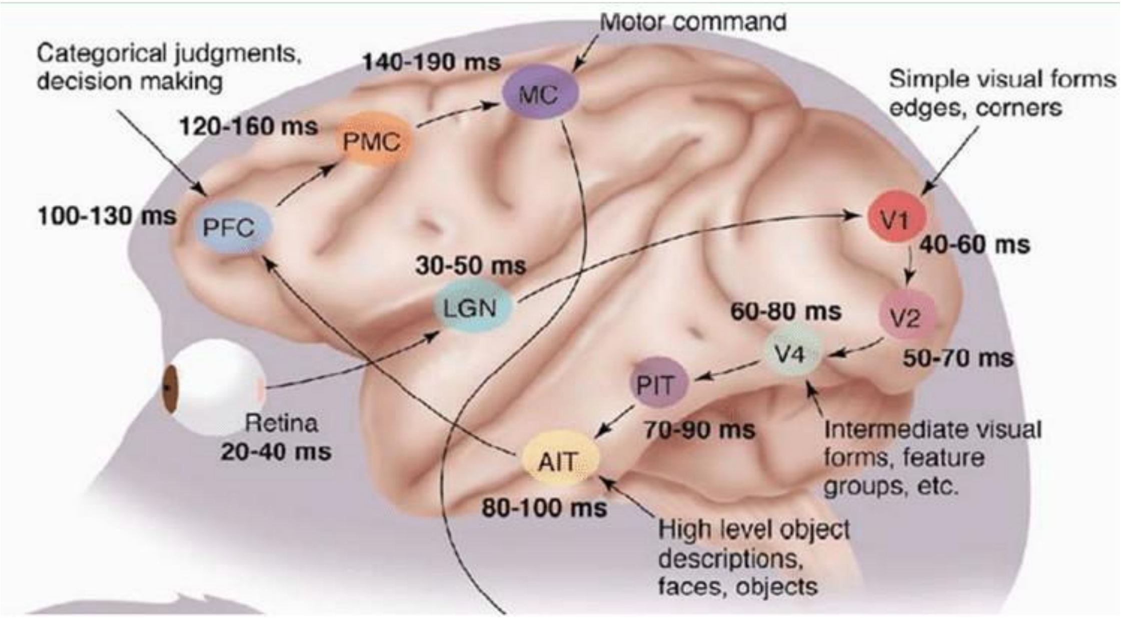
- Brain-inspired computation
  - Modelling how neuron processes information
  - (courtesy: Andrej Karpathy, Stanford’s CS231n lecture)

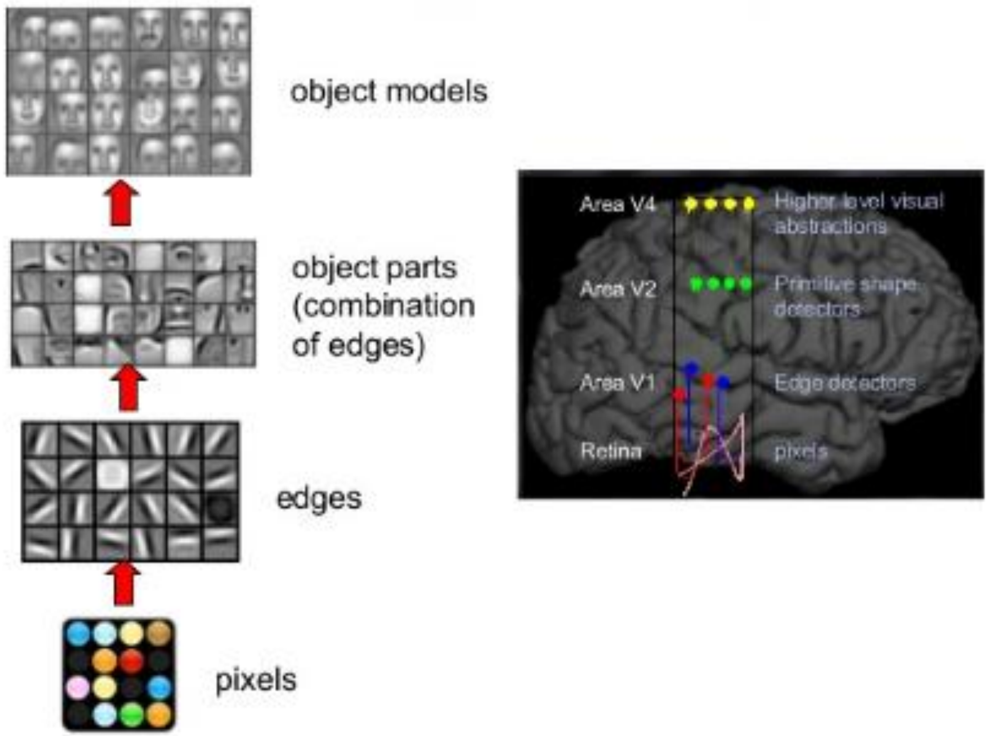




Deep Neural Network

- Inspiration from our visual system

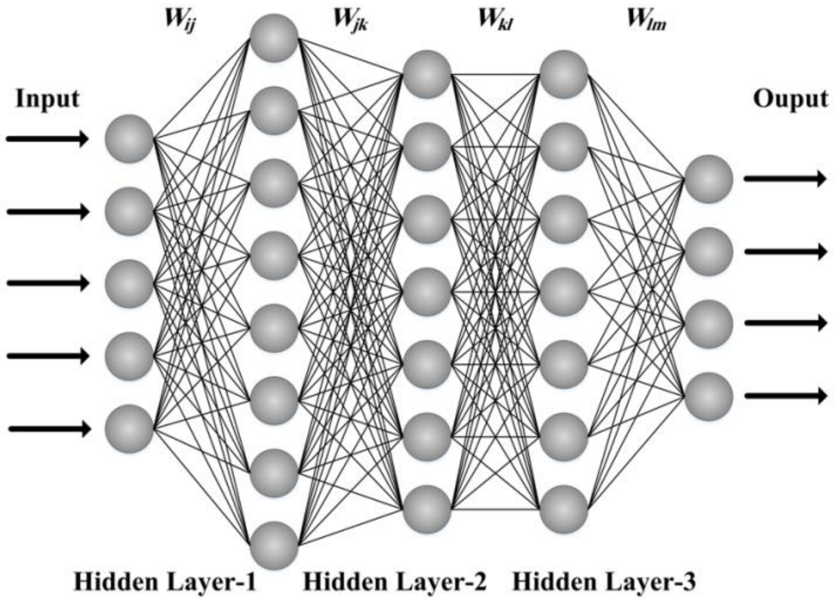


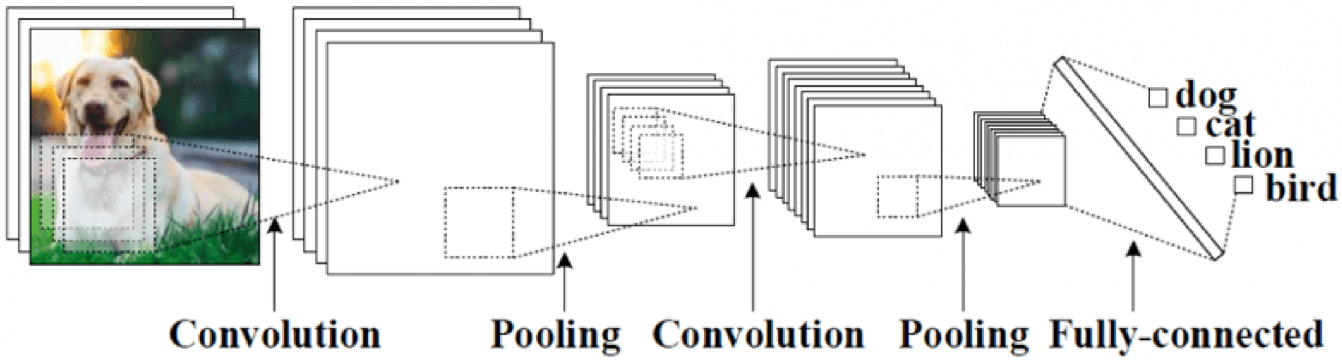


(courtesy: Simon J. Thorpe and Michele Fabre-Thorpe (left); Honglak Lee (right))

Neural network shallow vs. deep

- Evolvements with regard to the developments of neural science.





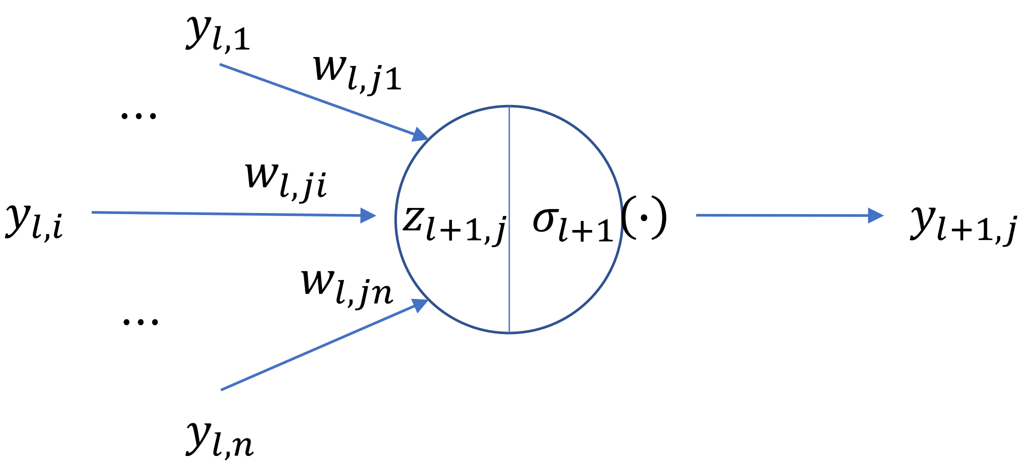
Key to Deep Learning’s Success

- Coincidence of tasks in image processing and operations of convolutional neural networks – designing of good filters suitable for specific problems.

- Two assumptions are made to simplify CNN computation: local reception filed and weights sharing
    - These simplifications harmonically coincide with convolutional operations;
    - However, no bio-physiological structure supports weights sharing, it just works.

The magic behind the scene: back propagation

- Exemplification by perceptron neural network
  - The output of i-th neuron at level l is denoted by $y_{l,i}$
  - The weight from the i-th neuron at level l to the j-th neuron at level $l+1$ is denoted by $w_{l,ji}$
  - The intermediate result for the j-th neuron at level $l+1$ is denoted by $z_{l+1,j}$



$$z_{l+1,j} = \sum_{i=1}^{n} y_{l,i} \cdot w_{l,ji}$$

$$y_{l+1,j} = \sigma_{l+1}(z_{l+1,j}) = \frac{1}{1+e^{-z_{l+1,j}}}$$

(sigmoid function)

Back propagation

- An alternative for back propagation is called error propagation or delta learning rule
  - If no ambiguity from the layer perspective, $w_{l,ji}$ is denoted by w_{ji}
  - For output from the i-th neurons at the final layer L, the ground-truth is denoted by $y_{L,i}^{d}$, or y_i^d

$$E = \frac{1}{2}\left(y_i - y_i^d\right)^2$$

$$w_{ij}(t+1) = w_{ij}(t) + \eta\left(-\frac{\partial E}{\partial w_{ij}}\right)$$

$$\frac{\partial E}{\partial w_{ij}} = \left(y_i - y_i^d\right)\frac{\partial y_i}{\partial w_{ij}} = \left(y_i - y_i^d\right)\frac{\partial y_i}{\partial z_i}\frac{\partial z_i}{\partial w_{ij}} = \left(y_i - y_i^d\right)y_i' y_j$$

$$w_{ij}(t+1) = w_{ij}(t) - \eta\left(y_i - y_i^d\right)y_i' y_j$$

Back propagation

- For simplicity, neurons are labelled sequentially

$$E = \frac{1}{2}\left(y_1 - y^d\right)^2 \qquad \Rightarrow \frac{\partial E}{\partial y_1} = \left(y_1 - y^d\right)$$

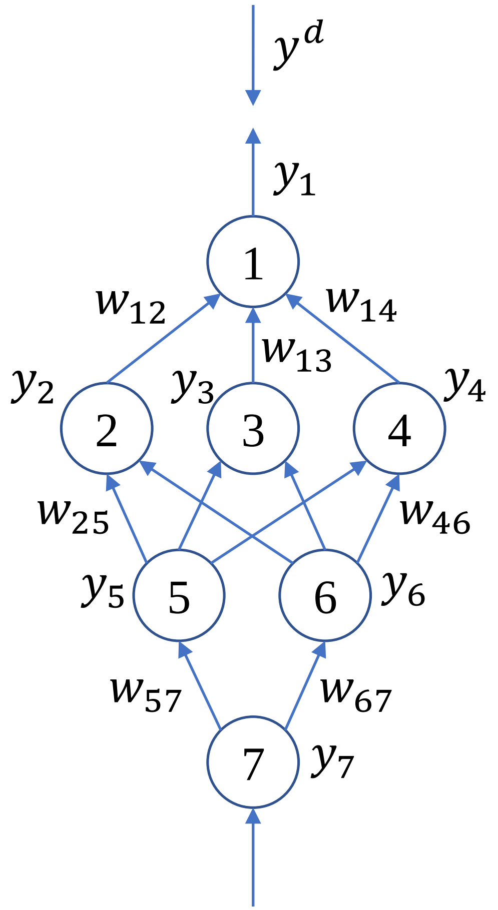
$$\frac{\partial E}{\partial w_{12}} = \frac{\partial E}{\partial y_1}\frac{\partial y_1}{\partial w_{12}} = \left(y_1 - y^d\right)\frac{\partial y_1}{\partial z_1}\frac{\partial z_1}{\partial w_{12}} = \left(y_i - y_i^d\right)y_1' y_2$$

$$\delta_0 \triangleq \left(y_1 - y^d\right) \qquad \Rightarrow \frac{\partial E}{\partial w_{12}} = \delta_0 y_1' y_2$$

$$\delta_1 \triangleq \delta_0 y_1' \qquad \Rightarrow w_{12}(t+1) = w_{12}(t) - \eta\delta_1 y_2$$

$$w_{13}(t+1) = w_{13}(t) - \eta\delta_1 y_3$$

$$w_{14}(t+1) = w_{14}(t) - \eta\delta_1 y_4$$



Back propagation

$$\frac{\partial E}{\partial w_{25}} = \frac{\partial E}{\partial y_1}\frac{\partial y_1}{\partial w_{25}} = \left(y_1 - y^d\right)\frac{\partial y_1}{\partial z_1}\frac{\partial z_1}{\partial w_{25}} = \left(y_i - y_i^d\right)y_1' \times$$

$$\frac{\partial}{\partial w_{25}}[w_{12}y_2 + w_{13}y_3 + w_{14}y_4] = \delta_1 w_{12}\frac{\partial y_2}{\partial w_{25}}$$

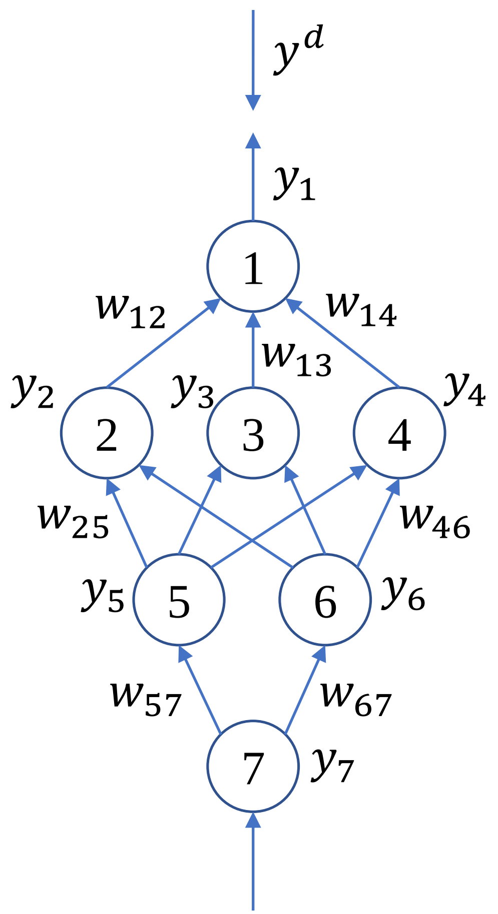
$$= \delta_1 w_{12}\frac{\partial y_2}{\partial z_2}\frac{\partial z_2}{\partial w_{25}} = \delta_1 w_{12} y_2' \frac{\partial z_2}{\partial w_{25}}$$

$$= \delta_1 w_{12}\frac{\partial y_2}{\partial z_2}\frac{\partial}{\partial w_{25}}[w_{25}y_5 + w_{26}y_6] = \delta_1 w_{12} y_2' y_5$$

$$\delta_2 \triangleq \delta_1 w_{12} y_2' \quad \Rightarrow \frac{\partial E}{\partial w_{25}} = \delta_2 y_5$$

$$w_{25}(t+1) = w_{25}(t) - \eta\delta_2 y_5$$

$$w_{26}(t+1) = w_{26}(t) - \eta\delta_2 y_6$$

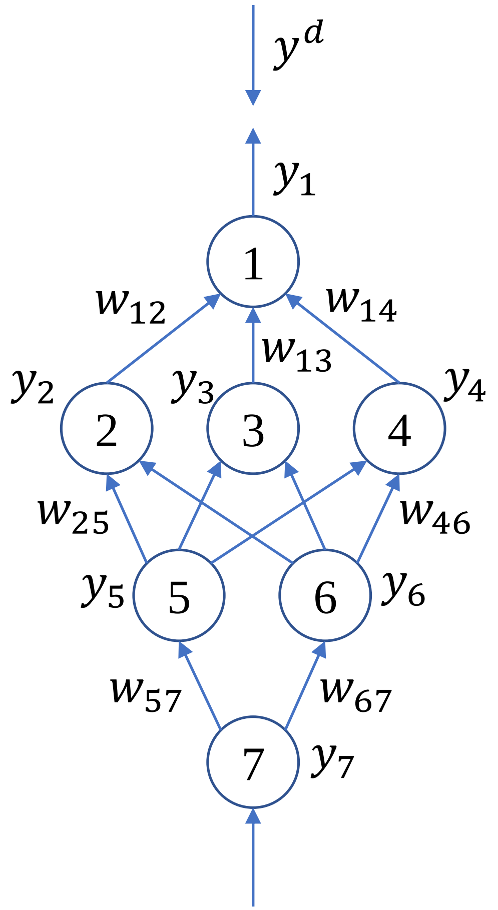


Back propagation

- Similarly, we have

$$\delta_3 \triangleq \delta_1 w_{13} y_3'$$
$$\delta_4 \triangleq \delta_1 w_{14} y_4'$$
$$\delta_5 \triangleq (\delta_2 w_{25} + \delta_3 w_{35} + \delta_4 w_{45}) y_5'$$
$$\delta_6 \triangleq (\delta_2 w_{26} + \delta_3 w_{36} + \delta_4 w_{46}) y_6'$$

$$w_{35}(t+1) = w_{35}(t) - \eta \delta_3 y_5$$
$$w_{36}(t+1) = w_{36}(t) - \eta \delta_3 y_6$$
$$w_{45}(t+1) = w_{45}(t) - \eta \delta_4 y_5$$
$$w_{46}(t+1) = w_{46}(t) - \eta \delta_4 y_6$$
$$w_{57}(t+1) = w_{57}(t) - \eta \delta_5 y_7$$
$$w_{67}(t+1) = w_{67}(t) - \eta \delta_6 y_7$$

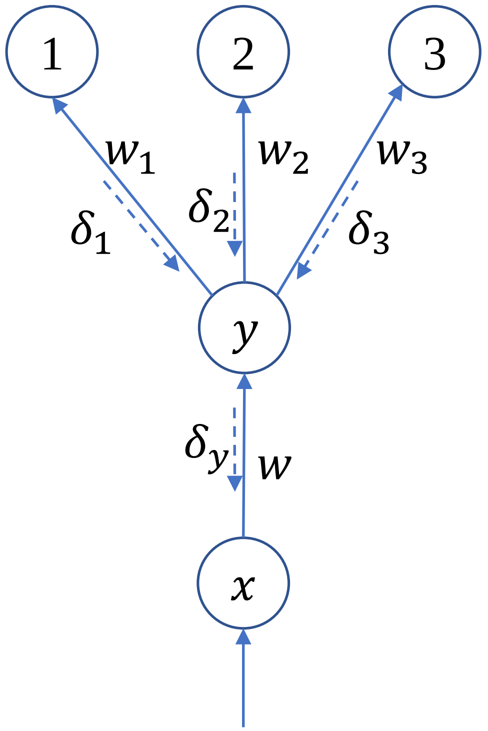


Back propagation

- General rule

$$\delta_y \triangleq (\delta_1 w_1 + \delta_2 w_2 + \delta_3 w_3) y'$$

$$w(t+1) = w(t) - \eta \delta_y x$$



Back propagation for CNN

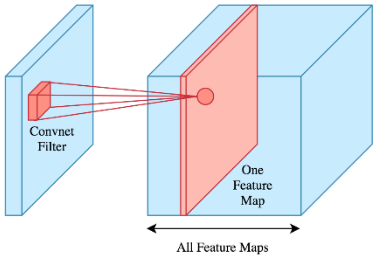
- Same principle but more complicated

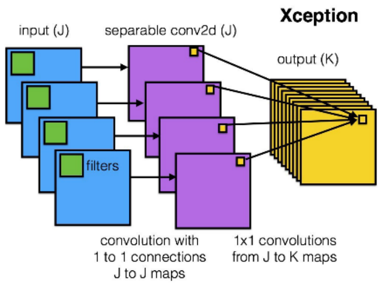
$$\frac{\partial E}{\partial \mathbf{k}_{ij}^{\ell}} = \text{rot180}(\text{conv2}(\mathbf{x}_i^{\ell-1},\ \text{rot180}(\boldsymbol{\delta}_j^{\ell}),\ '\text{valid}')).$$

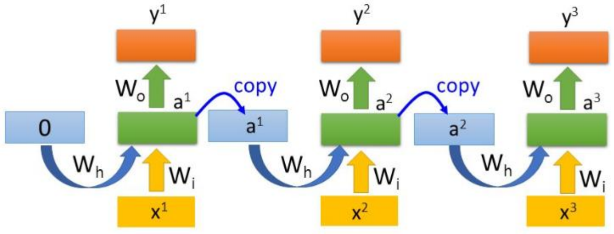
$$\boldsymbol{\delta}_j^{\ell} = f'(\mathbf{u}_j^{\ell}) \circ \text{conv2}(\boldsymbol{\delta}_j^{\ell+1},\ \text{rot180}(\mathbf{k}_j^{\ell+1}),\ '\text{full}').$$

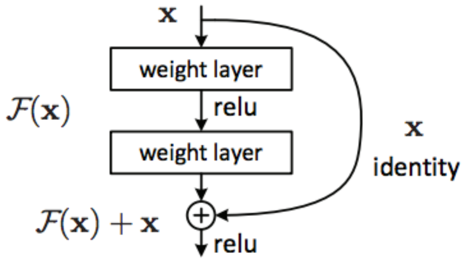
Common DNN structure

- (courtesy: Nameer Hirschkind, et al; Joyce Xu; Hung-yi Lee; Kaiming He)



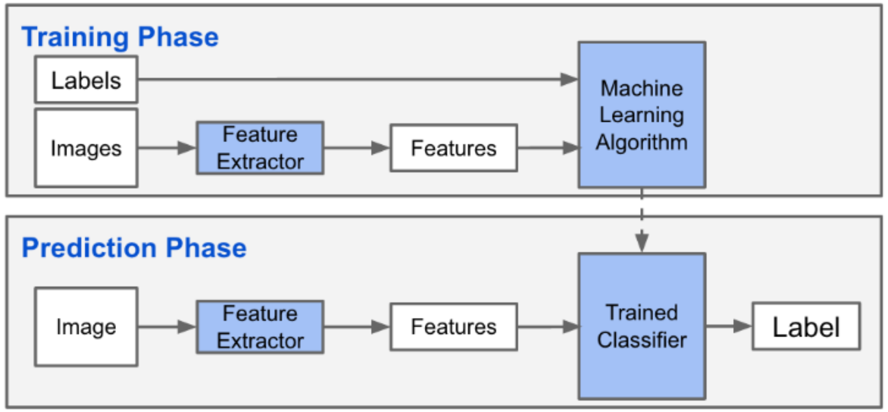


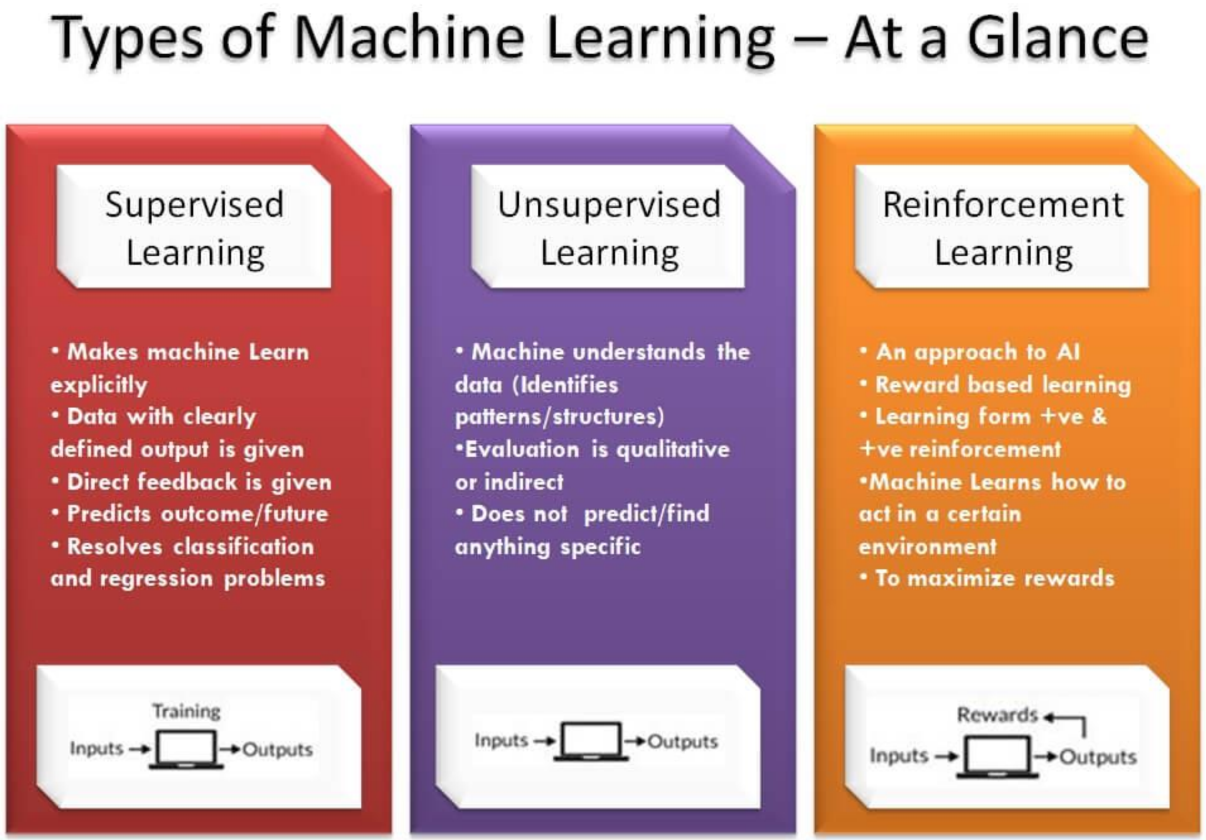




How does it work?

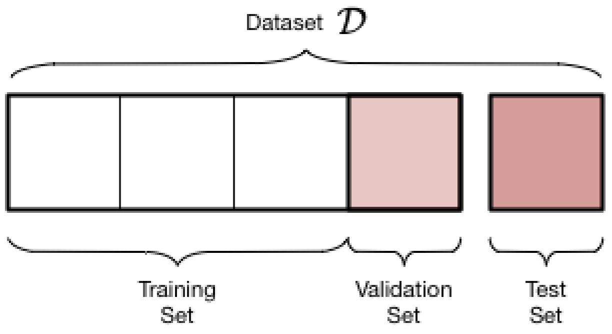
- Learning paradigms
  - Supervised learning
  - Unsupervised learning
  - Reinforcement learning

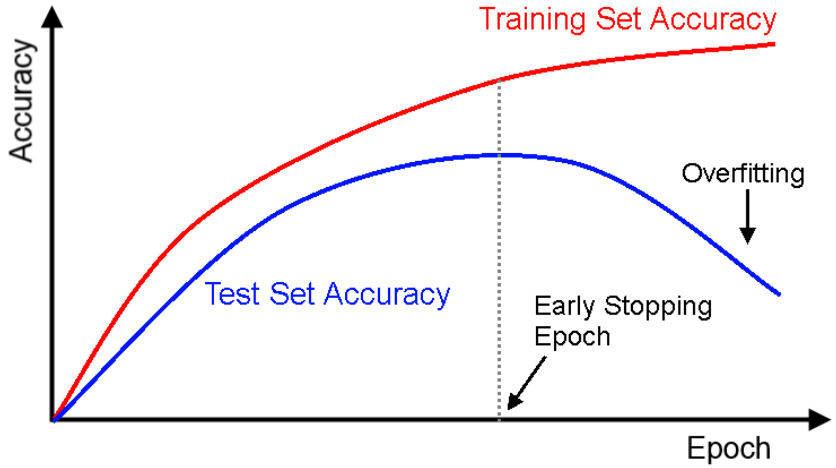




How does it work?

- Perform the following procedures in an iterative way:
  - Decide the neural network configuration and train the neural network
  - (Cross-)validate to evaluate the design and performance
- Real application inference





Q & A

Thanks