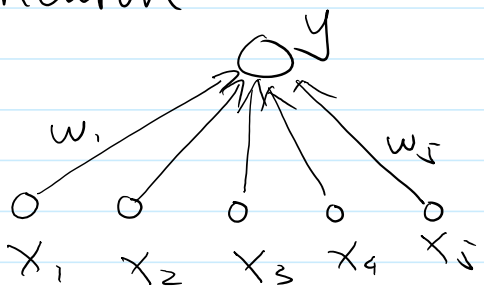
Lecture 19 Neural Networks

Sunday, November 6, 2016 3:24 PM

- single neuron

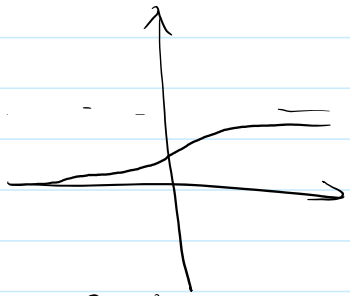


input $x \in \mathbb{R}^d$
weight $w \in \mathbb{R}^d$
bias $b \in \mathbb{R}$

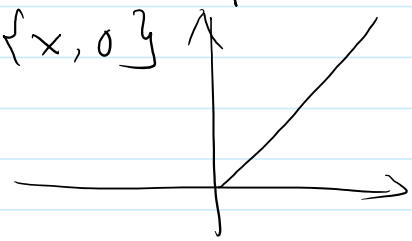
$$y = \sigma(w^T x + b)$$

σ : nonlinear function

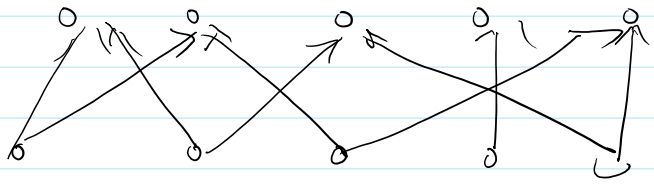
often used: sigmoid $\sigma(x) = \frac{e^x}{1+e^x}$



rectified linear $\sigma(x) = \max\{x, 0\}$
(used in deep learning)



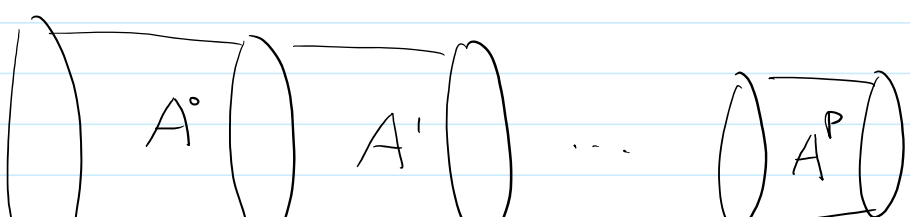
- single (fully connected) layer

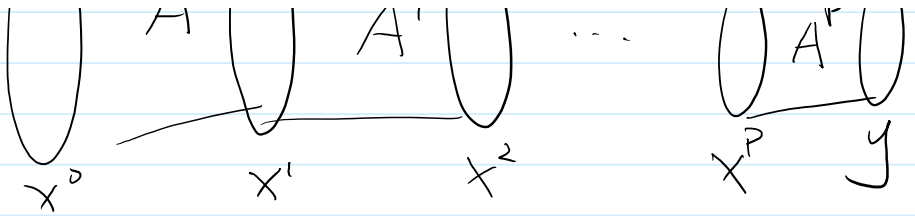


$x \in \mathbb{R}^{d_1}$
$y, b \in \mathbb{R}^{d_2}$
$A \in \mathbb{R}^{d_2 \times d_1}$

$y = \sigma(Ax+b)$ apply σ entry-wise to $Ax+b$.

- multi-layer



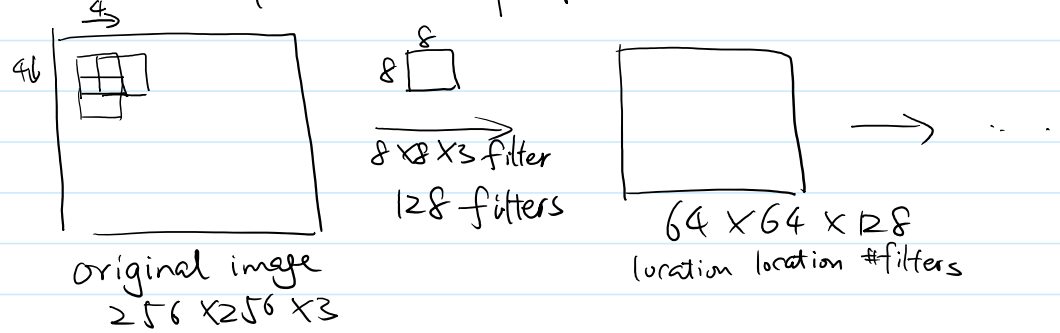


$$x^{i+1} = \sigma(A^i x^i + b^i)$$

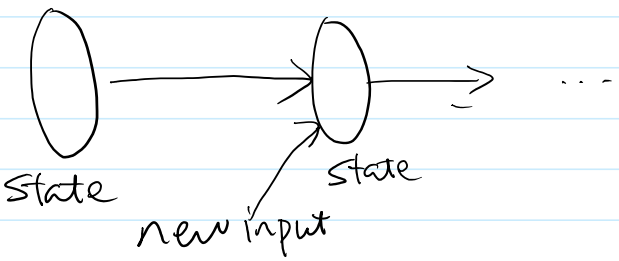
$$y = \sigma(A^P x^P + b^P)$$

- Deep learning: multilayer neural network + GPU + more.
  - structure of network
    - convolutional neural network

      for images, restrict weights A to act on a small patch, then perform the convolution



    - recurrent neural network



      many variants, used to process sequential data (e.g. text) and implement "memory"

  - training
    - regularization: $+\|A\|_F^2$ in objective

      (usual objective $\min \|y - f(x)\|^2$)

- backpropagation

  - stochastic gradient descent
  - how to compute gradient?

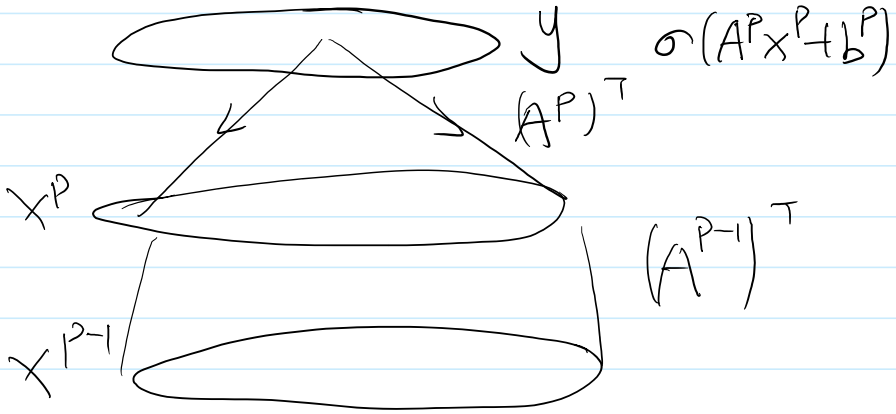
$$\frac{\partial y}{\partial x^p} = (A^p)^T f'(A^p x^p + b^p) \quad (\text{recall } f = \|y - (A^p x^p + b^p)\|^2)$$

$$\frac{\partial y}{\partial x^{p-1}} = \frac{\partial y}{\partial x^p} \cdot \frac{\partial x^p}{\partial x^{p-1}}$$

$$= (A^{p-1})^T \left(\sigma'(A^{p-1} x^{p-1} + b^{p-1}) \odot \left((A^p)^T f'(A^p x^p + b^p) \right) \right)$$

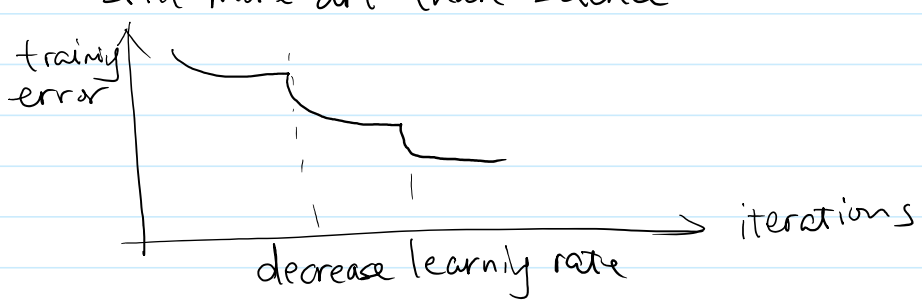
$\odot$ Hadamard product (entry-wise product)

- interpretation.



- dropout: disable a fraction (say $\frac{1}{2}$) of units.
  idea: avoid overfitting
  some analysis: might be interpreted as an instance specific l_2 regularization.

- turn learning rate
  still more art than science



- the neural net conjecture
  - Conjecture: any function we want to learn can be represented as a not too large neural net.
  - empirical results suggest this could be true, theoretically we don't know.
  - many recent advances in deep learning use this idea to replace steps in other learning procedure (e.g. reinforcement)
- what can we hope to do?
  ① understand the structure of deep net to offer insights in the neural net conjecture.
  Example: attempts to understand image/text
  [Zeiler Fergus] Visualizing and Understanding Convolutional Networks
  understanding word2vec (Arora et al.)
  ② even if there is a neural network, how do we find it?
  - non-convex optimization, saddle points
  ③ if we find a network, does it generalize?
  - may need new techniques
  ④ supervised vs. unsupervised.
  how much of deep learning is trying to learn a representation, how much is learning a mapping?