- Single neuron

\[ y = \sigma(\mathbf{w}^T \mathbf{x} + b) \]

\( \sigma \): 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

\[ y = \sigma(\mathbf{A} \mathbf{x} + b) \text{ apply } \sigma \text{ entry-wise to } \mathbf{A} \mathbf{x} + b. \]

- Multi-layer

\[ \ldots \]
\[ U \begin{bmatrix} U_0 & U_1 & \cdots & U_{p-1} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_p \\ y \end{bmatrix} \]

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

\[ y = \sigma (A^p x^p + b^p) \]

- 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

- Training
  - Regularization: \(+ \| A \|_F^2 \) in objective
    \( \text{(usual objective } \min \| y - f(x) \|^2) \)
- backpropagation

- stochastic gradient descent

- how to compute gradient:

\[
\frac{\partial y}{\partial x^p} = (A^p)^T f'(A^p x^p + b^p) \quad \text{(recall } f = \|y - (A^p x^p + b^p)\|_2^2)\]

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

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

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

- interpretation.

- dropout: disable a fraction (say \(\frac{1}{5}\)) of units.

  idea: avoid overfitting

  some analysis might be interpreted as an instance specific to regularization.

- tune learning rate

  still more art than science

- training error

  decrease learning rate
- 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?