Johns Hopkins Engineering

Applied Machine Learning for Mechanical Engineers

Machine Learning Fundamentals, Part 1, C

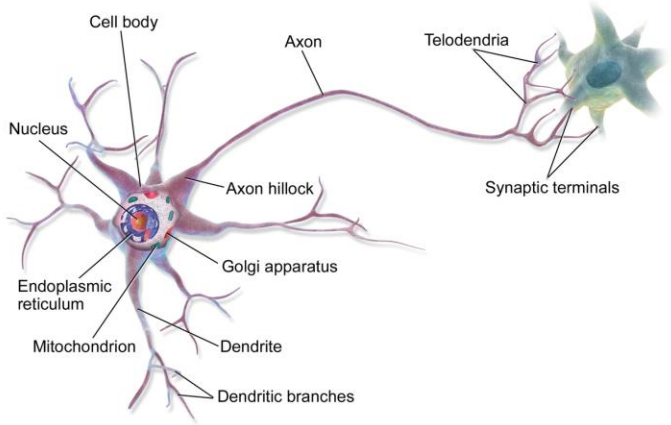
JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING

Perceptron and Neural Networks

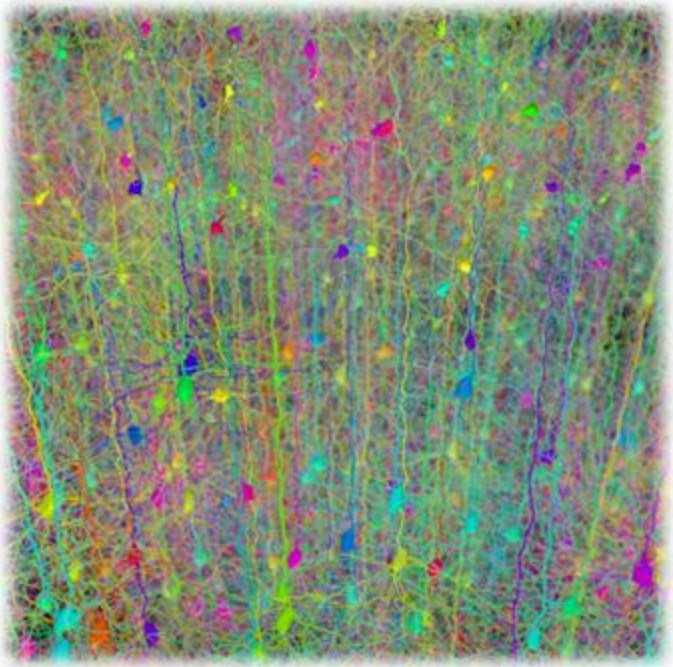
- By the end of this lecture you will be able to:
  - Describe perceptron
  - Describe activation/transfer functions
  - Describe multi-layer perceptron
  - Describe neural network optimization

Perceptron and Neural Networks

- Perceptron
  - Brain is a giant network of neurons connected to one another



Simplified biological model of a neuron in brain

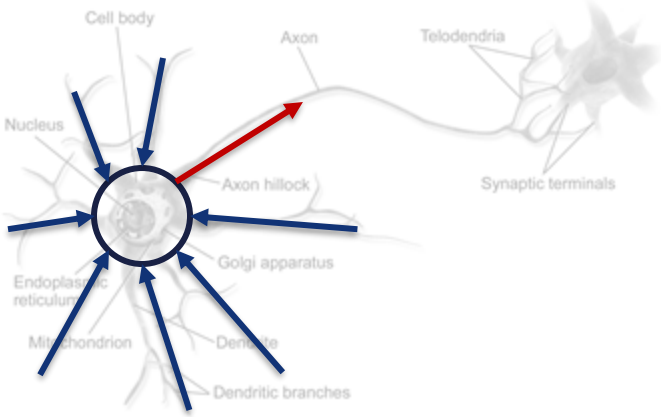


Computer simulation of dendrites of pyramidal network of neurons (biological neural networks)

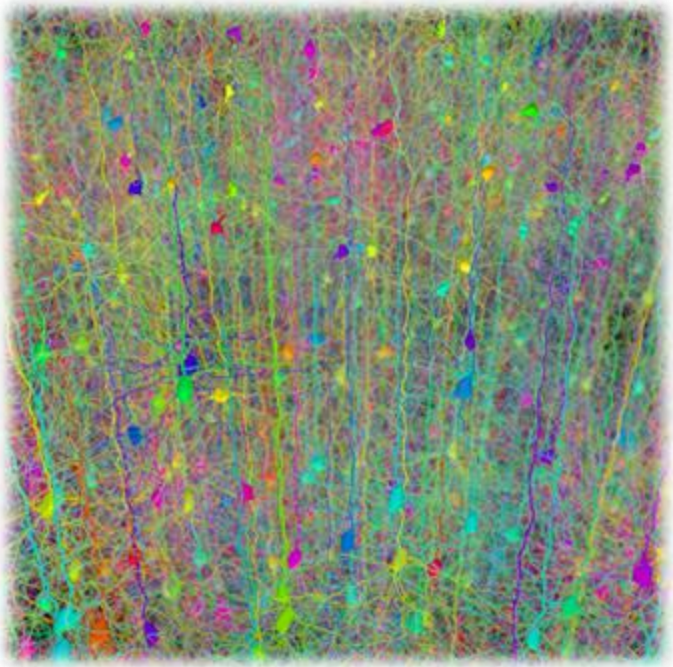
Cuntz, Hermann (2010). "PLoS Computational Biology Issue Image | Vol. 6(8) August 2010". PLOS Computational Biology. 6 (8): ev06.i08. doi:10.1371/image.pcbi.v06.i08
https://en.wikipedia.org/wiki/Neural_circuit

Perceptron and Neural Networks

- Perceptron
  - Brain is a giant network of neurons connected to one another



Simplified biological model of a neuron in brain

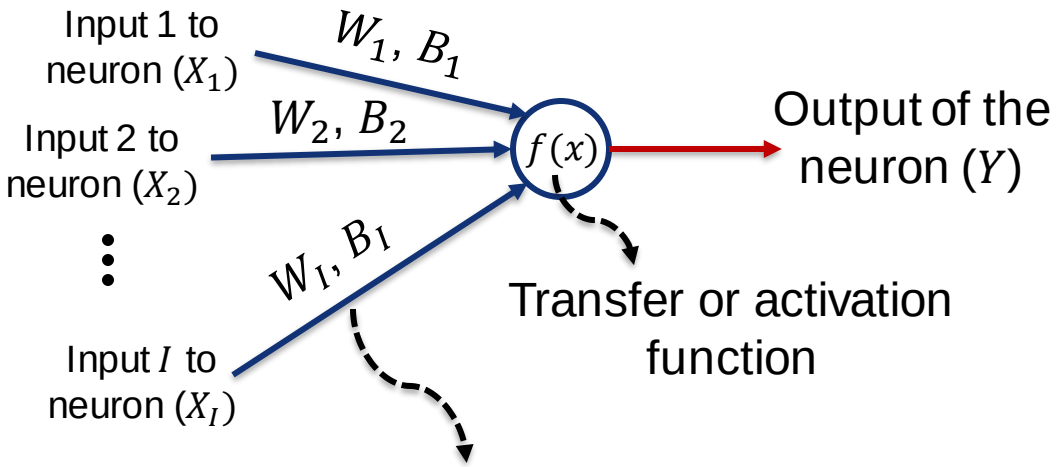


Computer simulation of dendrites of pyramidal network of neurons (biological neural networks)

Cuntz, Hermann (2010). "PLoS Computational Biology Issue Image | Vol. 6(8) August 2010". PLOS Computational Biology. 6 (8): ev06.i08. doi:10.1371/image.pcbi.v06.i08
https://en.wikipedia.org/wiki/Neural_circuit

Perceptron and Neural Networks

- Perceptron
  - Simplified perceptron as a simple neural network by Rosenblatt (1958) and Minsky and Seymour (1969), so-called artificial neural networks



Weight Bias

$$x = \sum_{i=1}^{I} X_i W_i + B_i$$

$$Y = f(x) = f\left(\sum_{i=1}^{I} X_i W_i + B_i\right)$$

Perceptron and Neural Networks

- Perceptron
  - Simplified perceptron as a simple neural network by Rosenblatt (1958) and Minsky and Seymour (1969), so-called artificial neural networks

Optimization problem: what are the optimum values of all W_n and B_n such minimize (maximize) error (accuracy) measurement of $Y_{estimated}$ versus Y_{real}?

Fitting one or very few datapoints

Weight Bias

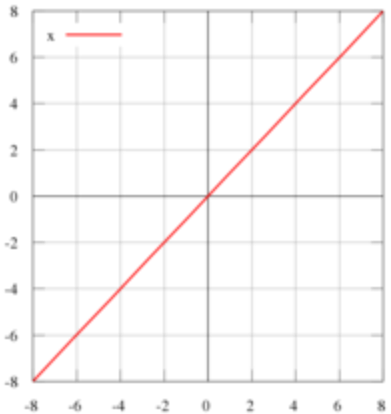
$$x = \sum_{i=1}^{I} X_i W_i + B_i$$

$$Y = f(x) = f\left(\sum_{i=1}^{I} X_i W_i + B_i\right)$$

Perceptron and Neural Networks

- Activation/transfer functions
  - Popular activation/transfer functions
    - Linear or identity function

$$f(x) = x$$

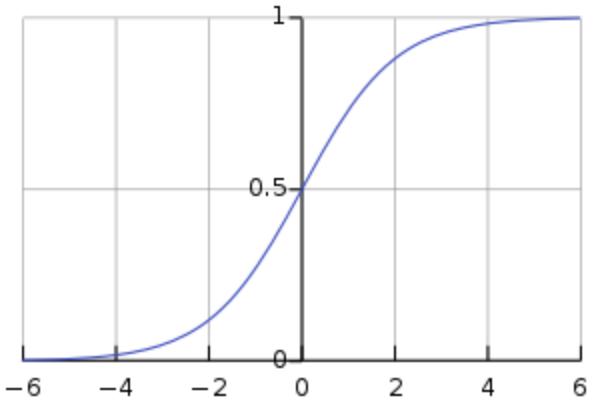


https://upload.wikimedia.org/wikipedia/commons/thumb/e/eb/Function-x.svg/330px-Function-x.svg.png

Perceptron and Neural Networks

- Activation/transfer functions
  - Popular activation/transfer functions
    - Sigmoid or logistic function

$$f(x) = \frac{1}{1 + e^{-x}}$$

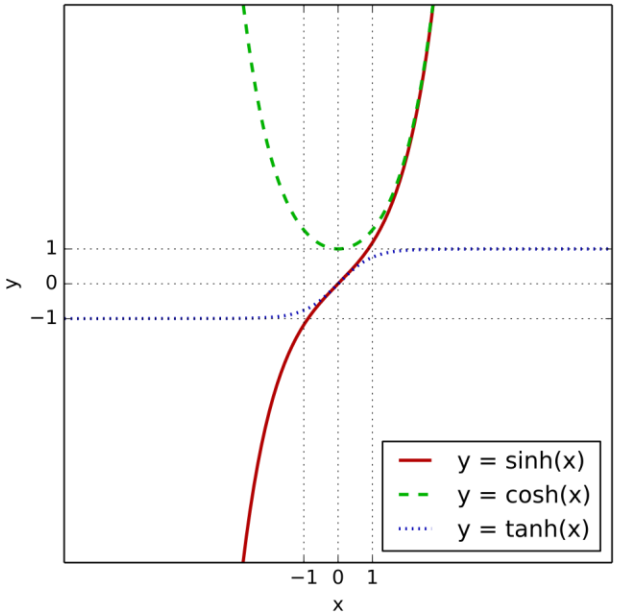


https://upload.wikimedia.org/wikipedia/commons/thumb/8/88/Logistic-curve.svg/480px-Logistic-curve.svg.png

Perceptron and Neural Networks

- Activation/transfer functions
  - Popular activation/transfer functions
    - Hyperbolic functions such as $\tanh(x)$

$$f(x) = \tanh(x)$$

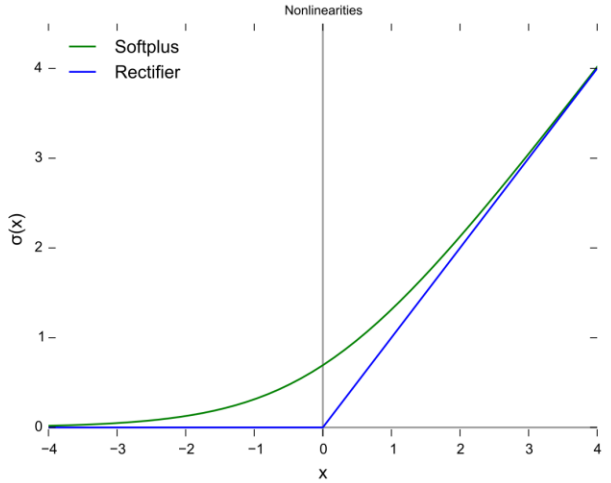


https://upload.wikimedia.org/wikipedia/commons/thumb/7/76/Sinh_cosh_tanh.svg/1024px-Sinh_cosh_tanh.svg.png

Perceptron and Neural Networks

- Activation/transfer functions
  - Popular activation/transfer functions
    - Rectified linear unit (ReLU)

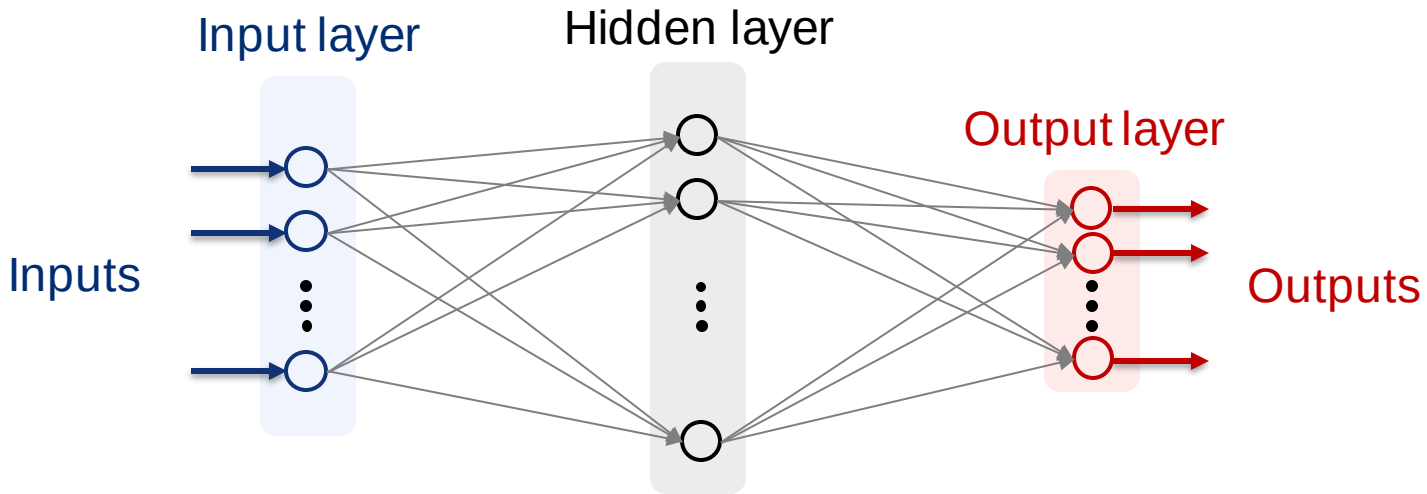
$$f(x) = \begin{cases} 0 & x \leq 0 \\ 1 & x > 0 \end{cases}$$



https://upload.wikimedia.org/wikipedia/commons/thumb/6/6c/Rectifier_and_softplus_functions.svg/1280px-Rectifier_and_softplus_functions.svg.png

Perceptron and Neural Networks

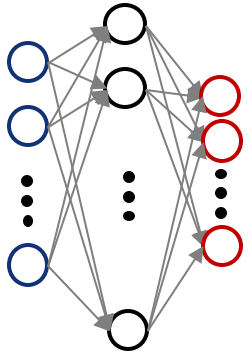
- Multi-Layer perceptron (feed-forward artificial neural network)
  - Increase memory by increasing complexity for complex problems



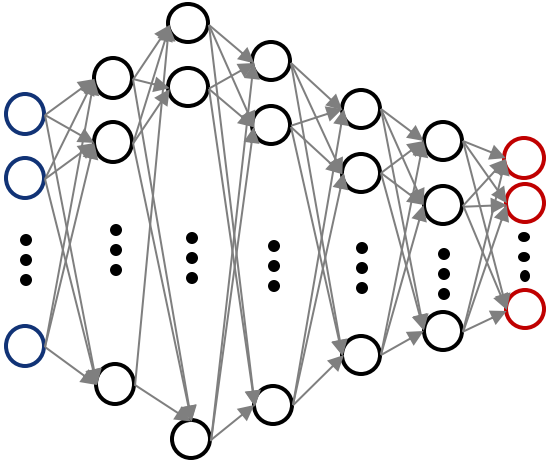
Perceptron and Neural Networks

- Multi-Layer perceptron (feed-forward artificial neural network)
  - Increase memory by increasing complexity for complex problems

Shallow
Neural Network
(1 hidden layer)



Deep
Neural Network
(2 or more hidden layers)



Perceptron and Neural Networks

- Neural network optimization
  - What are the optimum values of weights and biases of all layers of a neural network such when being fed (i.e., inputted and outputted) by every training datapoint in a repository, the estimated output of every training datapoint becomes very close to the actual output of that training datapoint?
  - What are the optimum values of weights and biases of all layers of a neural network such when being fed (i.e., inputted and outputted) by every training datapoint, the overall estimation error becomes minimized (or the accuracy becomes maximized)?
  - Fitness function?

Perceptron and Neural Networks

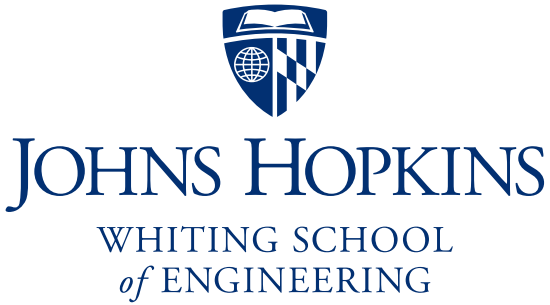
- Neural network optimization
  - The fitness function in neural network usually called **loss function** (sometimes is referred to as the cost function), and has a variety of forms depending on the problem (e.g., regression, classification, etc.) or the machine learning algorithm (e.g., supervised, unsupervised, classifier, regressor, etc.)
  - Example (with respect to notations used in slide 3 of lecture B of this module)

$$l(p) = \frac{1}{2N}\sum_{n=1}^{N}\left\|\boldsymbol{Y}_n - \widetilde{\boldsymbol{Y}}_n\right\|^2$$

where p is a set of optimization variables including weights and biases of different layers of neural networks, $\|.\|$ is the norm function, and $\boldsymbol{Y}_n$ and $\widetilde{\boldsymbol{Y}}_n$ are the actual and estimated outputs of n^{th} training datapoint. In this example, loss function computes the average of mean squared error of estimations.

Perceptron and Neural Networks

- In this lecture, you learned about:
  - Perceptron
  - Activation/transfer functions
  - Multi-layer perceptron
  - Neural network optimization
- In the next lecture, we will talk about learning algorithms



© The Johns Hopkins University 2020, All Rights Reserved.