Ch 10.1: Neural Nets

Lecture 29 - CMSE 381

Michigan State University
::
Dept of Computational Mathematics, Science & Engineering

Mon, Apr 6, 2026

Announcements

Last time:

- SVM

This lecture:

- Feed Forward Neural Nets

Announcements:

- Homework #6 due Sunday, 4/12 11:59pm
- Review class next Friday - start thinking about questions!

21	W	3/18	Polynomial & Step Functions	7.1-7.2		
22	F	3/20	Step Functions; Basis functions; Start Splines	7.2-7.4		
23	M	3/23	Regression Splines	7.4		Q7
24	W	3/25	Decision Trees	8.1		
25	F	3/27	Random Forests	8.2.1, 8.2.2	HW #5 Due Sun 3/29	
26	M	3/30	Maximal Margin Classifier	9.1		
27	W	4/1	SVC	9.2		Q8
28	F	4/3	SVM	9.3, 9.4		
29	M	4/6	Single Layer NN	10.1		
30	W	4/8	Multi Layer NN	10.2		Q9
31	F	4/10	CNN	10.3		
32	M	4/13	Unsupervised learning / clustering	12.1, 12.4	HW #6 Due Sun 4/12	
33	W	4/15	Virtual: Project Office Hours			Q10
	F	4/17	***Review***			
	M	4/20	**Midterm #3**			
	W	4/22				
	F	4/24			**Proiect Due**	

What will you learn today?

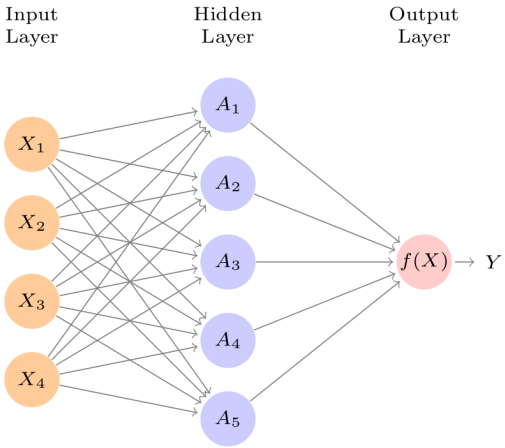
- What is the architecture of simple feed-forward neural network?
  - You should be able to describe what happens at each layer (e.g., input layer, hidden layer, output layer) conceptually and mathematically.
- What is activation? What types of activation functions are there?
  - You should be able to explain it mathematically.
  - Given example inputs and weights from the previous matrix, you should be able to calculate the activation A by hand (with the help of a calculator).
- How does a simple feed-forward neural network produce an output?
  - Given a β matrix and activation from the previous layer, you should be able to calculate the output by hand.
- What do the fitted parameters minimize in a simple feed-forward neural network?

Section 1

Neural Nets

The idea

Feed Forward Neural Network: The cartoon



What is activation? Neuroscience 101.

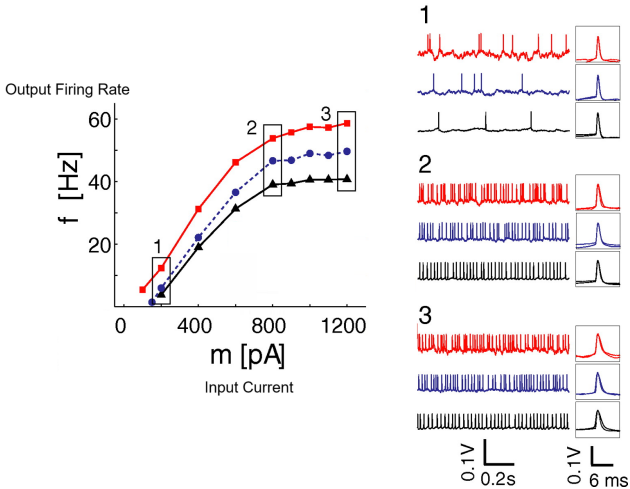
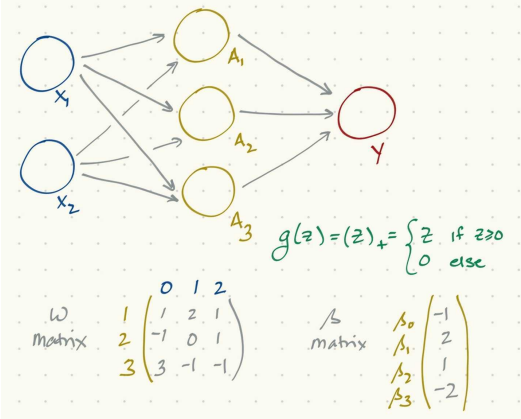


Figure adapted from Fig 1 of Arsiero et al. 2007 (J of Neurosci.)

A very simple example

Computing A_k for $(1, 0)$

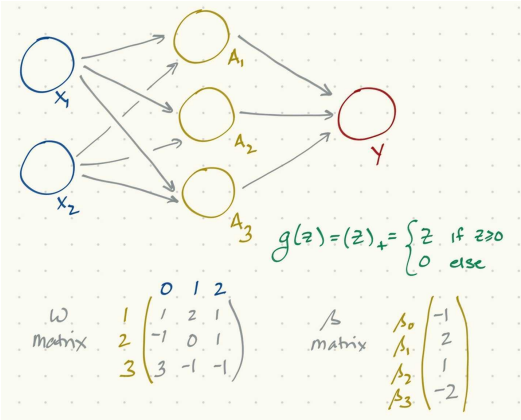
$$A_k = h_k(X) = g(w_{k0} + \sum_{j=1}^{p} w_{kj} X_j),$$



A very simple example

Computing Y for $(1, 0)$

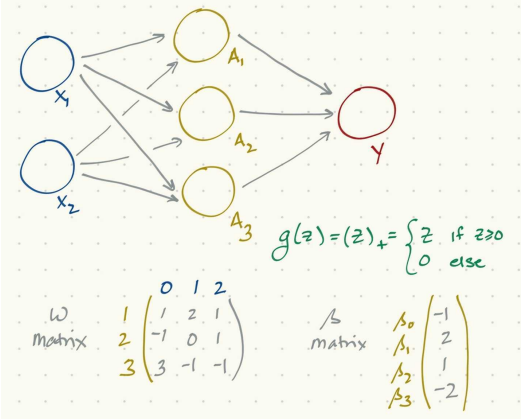
$$f(X) = \beta_0 + \sum_{k=1}^{K} \beta_k A_k$$



A very simple example

Computing Y for $(0, 1)$

$$A_k = h_k(X) = g(w_{k0} + \sum_{j=1}^{p} w_{kj} X_j), \qquad f(X) = \beta_0 + \sum_{k=1}^{K} \beta_k A_k$$



Test your understanding: PollEv!

A different example

- Draw the diagram for a neural net with input data points with $p = 3$ (i.e., (X_1, X_2, X_3)) and two units in the hidden layer.
- Using the ω and β matrices, what is the output predicted Y for the point $(2, 0, 1)$?

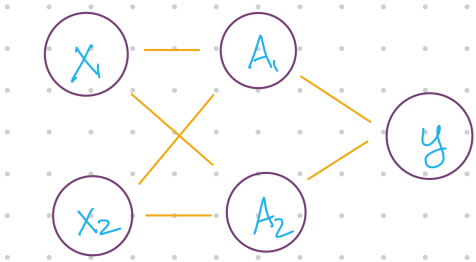
$$\omega = \begin{pmatrix} 1 & 0 & -2 & 2 \\ -3 & 1 & 0 & -1 \end{pmatrix} \qquad \beta = \begin{pmatrix} -1 \\ -2 \\ 1 \end{pmatrix}$$

- Use the activation function

$$g(z) = (z)_+ = \begin{cases} 0 & \text{if } z < 0 \\ z & \text{else.} \end{cases}$$

Extra space

What if there's no activation function?

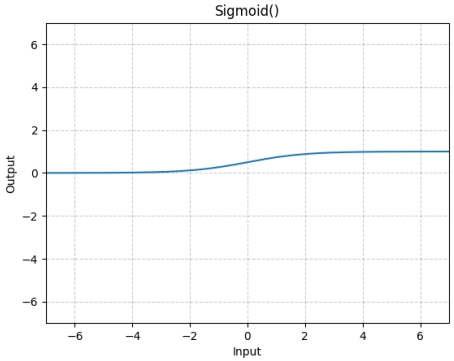


$$\omega = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix} \qquad \beta = \begin{pmatrix} -1 \\ -2 \\ 1 \end{pmatrix}$$

Choices for activation function

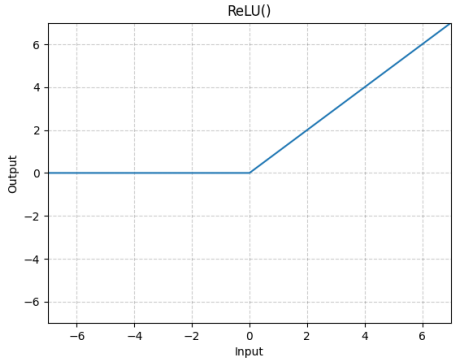
Sigmoid:

$$g(z) = \frac{e^z}{1+e^z} = \frac{1}{1+e^{-z}}$$

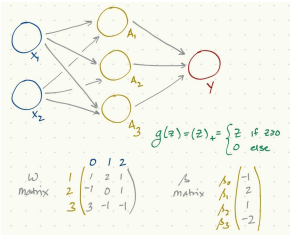


ReLU: Rectified linear unit

$$g(z) = (z)_+ = \begin{cases} 0 & \text{if } z < 0 \\ z & \text{else.} \end{cases}$$



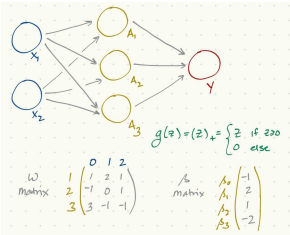
Matrix version: First layer



$$A_k = h_k(X) = g(w_{k0} + \sum_{j=1}^{p} w_{kj} X_j),$$

$$A = g(\mathbf{W} \cdot \mathbf{X}) \qquad \mathbf{X}^T = (1\ X_1\ X_2\ \cdots\ X_p)$$

Matrix version: Output



$$f(X) = \beta_0 + \sum_{k=1}^{K} \beta_k A_k$$

$$Y = \beta \cdot \mathbf{A} \qquad \mathbf{A}^T = (1\ A_1\ A_2\ \cdots\ A_K)$$

Now what?

Choose parameters by minimizing RSS, $\sum_{i=1}^{n}(y_i - f(x_i))^2$

Chosen in advance:

Tuned by the model:

Coding

Next time

21	W	3/18	Polynomial & Step Functions	7.1-7.2		
22	F	3/20	Step Functions; Basis functions; Start Splines	7.2-7.4		
23	M	3/23	Regression Splines	7.4		Q7
24	W	3/25	Decision Trees	8.1		
25	F	3/27	Random Forests	8.2.1, 8.2.2	HW #5 Due Sun 3/29	
26	M	3/30	Maximal Margin Classifier	9.1		Q8
27	W	4/1	SVC	9.2		
28	F	4/3	SVM	9.3, 9.4		
29	M	4/6	Single Layer NN	10.1		Q9
30	W	4/8	Multi Layer NN	10.2		
31	F	4/10	CNN	10.3	HW #6 Due Sun 4/12	
32	M	4/13	Unsupervised learning / clustering	12.1, 12.4		Q10
33	W	4/15	Virtual: Project Office Hours			
	F	4/17	***Review***			
	M	4/20	**Midterm #3**			
	W	4/22				
	F	4/24			**Proiect Due**	