CMSE 381, Fundamental Data Science Methods

November 23, 2025

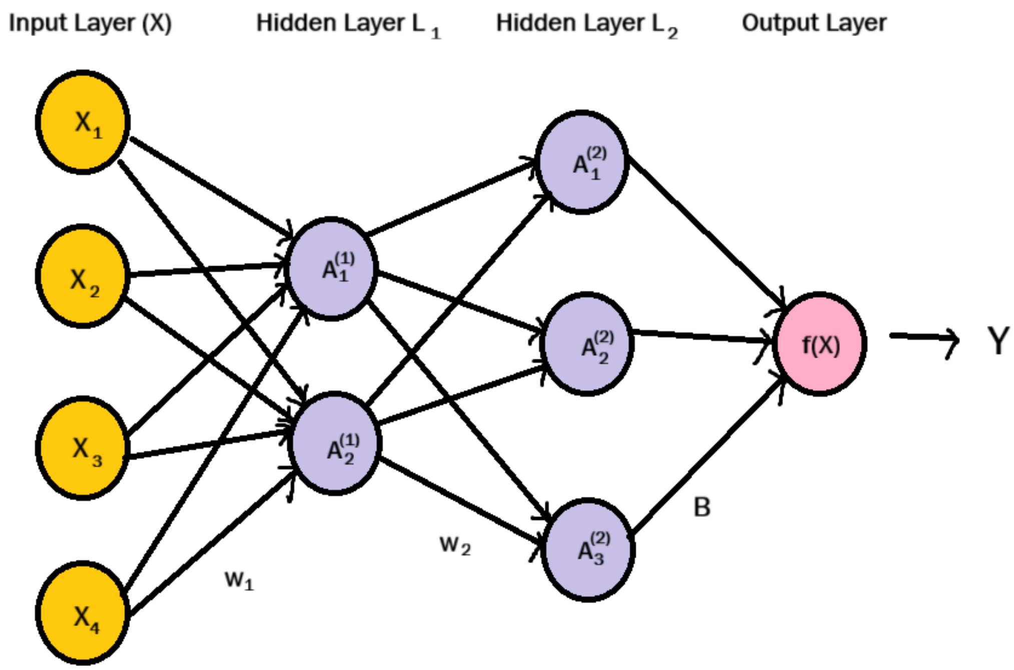
Homework 9

Lowell Monis

Instructor: Dr. Mengsen Zhang

Question 1: ISLP § 10.10.1

Consider a neural network with two hidden layers: $p = 4$ input units, 2 units in the first hidden layer, 3 units in the second hidden layer, and a single output.



(a) Draw a picture of the network.

(b) Write out an expression for $f(X)$, assuming ReLU activation functions. Be as explicit as you can! The ReLU activation function can be expressed as,

$$g(x) = (x)_+ = \max(0, x)$$

For hidden weight matrices $\mathbf{w}^{(1)}$ and $\mathbf{w}^{(2)}$, and output weight vector β, we can etch the first hidden layer's component vector, $A^{(1)}$ for $k \in [2]$ with the activation function applied:

$$A_k^{(1)} = g(\mathbf{w}_k^{(1)} \cdot \mathbf{X}) = \left(\mathbf{w}_k^{(1)} \cdot \mathbf{X}\right)_+ = \max\left(0, w_{k,0}^{(1)} + \sum_{j=1}^{4} w_{k,j}^{(1)} X_j\right)$$

Similarly, the weighted measures from the second hidden layer can be expressed as $A^{(2)}$ for $l \in [3]$:

$$A_l^{(2)} = g(\mathbf{w}_l^{(2)} \cdot \mathbf{A}^{(1)}) = \left(\mathbf{w}_l^{(2)} \cdot \mathbf{A}^{(1)}\right)_+ = \max\left(0, w_{l,0}^{(2)} + \sum_{j=1}^{2} w_{l,j}^{(2)} A_j^{(1)}\right)$$

Finally, we can express the result $f(\mathbf{X})$ as a function of the second hidden layer, weighted by the output weights β:

$$f(\mathbf{X}) = \boldsymbol{\beta} \cdot \mathbf{A}^{(2)} = \beta_0 + \sum_{l=1}^{3} \beta_l A_l^{(2)}$$

To elaborate in detail:

$$f(\mathbf{X}) = \beta_0 + \sum_{l=1}^{3} \beta_l \max\left(0, w_{l,0}^{(2)} + \sum_{k=1}^{2} w_{l,k}^{(2)} \max\left(0, w_{k,0}^{(1)} + \sum_{j=1}^{4} w_{k,j}^{(1)} X_j\right)\right)$$

(c) Now plug in some values for the coefficients and write out the value of $f(X)$. Pick any values you would like for the coefficients, but be sure to elaborate what they are. Then, plug in the vectors $(1, 0, 0, 0)$, $(0, 1, 0, 0)$, $(0, 0, 1, 0)$, and $(0, 0, 0, 1)$ and show the output.

I will first define the coefficients.

First hidden layer weights $\mathbf{w}^{(1)}$ (2×5 matrix, including column associated with the coefficient):

$\mathbf{w}_1^{(1)} = [0.5, 1, -1, 0.5, -0.5]$

$\mathbf{w}_2^{(1)} = [-0.3, 0.5, 1, -0.5, 1]$

Second hidden layer weights $\mathbf{w}^{(2)}$ (3×3 matrix, including column associated with the coefficient):

$\mathbf{w}_1^{(2)} = [0.2, 1, -0.5]$

$\mathbf{w}_2^{(2)} = [-0.1, 0.5, 1]$

$\mathbf{w}_3^{(2)} = [0.3, -1, 0.5]$

Output layer weights $\boldsymbol{\beta}$:

$\boldsymbol{\beta} = [1, 0.8, 1.2, -0.5]$

Now, I will compute $f(\mathbf{X})$ for each input vector:

1. For $\mathbf{X} = (1, 0, 0, 0)$:

- $A_1^{(1)} = \max(0, 0.5 + 1(1) + (-1)(0) + 0.5(0) + (-0.5)(0)) = \max(0, 1.5) = 1.5$
- $A_2^{(1)} = \max(0, -0.3 + 0.5(1) + 1(0) + (-0.5)(0) + 1(0)) = \max(0, 0.2) = 0.2$
- $A_1^{(2)} = \max(0, 0.2 + 1(1.5) + (-0.5)(0.2)) = \max(0, 1.6) = 1.6$
- $A_2^{(2)} = \max(0, -0.1 + 0.5(1.5) + 1(0.2)) = \max(0, 0.85) = 0.85$
- $A_3^{(2)} = \max(0, 0.3 + (-1)(1.5) + 0.5(0.2)) = \max(0, -1.1) = 0$

$$f(1, 0, 0, 0) = 1 + 0.8(1.6) + 1.2(0.85) + (-0.5)(0) = 1 + 1.28 + 1.02 + 0 = 3.30$$

2. For $\mathbf{X} = (0, 1, 0, 0)$:

- $A_1^{(1)} = \max(0, 0.5 + 0 - 1 + 0 + 0) = \max(0, -0.5) = 0$
- $A_2^{(1)} = \max(0, -0.3 + 0 + 1 + 0 + 0) = \max(0, 0.7) = 0.7$
- $A_1^{(2)} = \max(0, 0.2 + 0 - 0.35) = \max(0, -0.15) = 0$
- $A_2^{(2)} = \max(0, -0.1 + 0 + 0.7) = \max(0, 0.6) = 0.6$
- $A_3^{(2)} = \max(0, 0.3 + 0 + 0.35) = \max(0, 0.65) = 0.65$

$$f(0, 1, 0, 0) = 1 + 0 + 0.72 - 0.325 = 1.395$$

3. For $\mathbf{X} = (0, 0, 1, 0)$:

- $A_1^{(1)} = \max(0, 0.5 + 0.5) = 1.0$
- $A_2^{(1)} = \max(0, -0.3 - 0.5) = 0$
- $A_1^{(2)} = \max(0, 0.2 + 1.0) = 1.2$
- $A_2^{(2)} = \max(0, -0.1 + 0.5) = 0.4$
- $A_3^{(2)} = \max(0, 0.3 - 1.0) = 0$

$$f(0, 0, 1, 0) = 1 + 0.96 + 0.48 + 0 = 2.44$$

4. For $\mathbf{X} = (0, 0, 0, 1)$:

- $A_1^{(1)} = \max(0, 0.5 - 0.5) = 0$
- $A_2^{(1)} = \max(0, -0.3 + 1) = 0.7$
- $A_1^{(2)} = \max(0, 0.2 - 0.35) = 0$
- $A_2^{(2)} = \max(0, -0.1 + 0.7) = 0.6$
- $A_3^{(2)} = \max(0, 0.3 + 0.35) = 0.65$

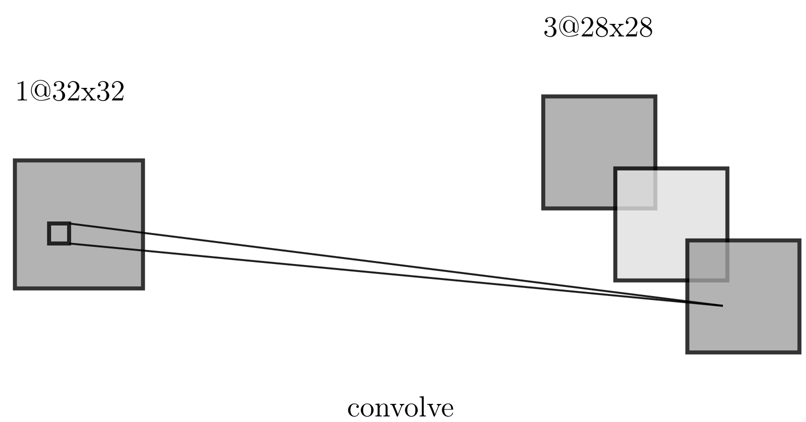
$$f(0, 0, 0, 1) = 1 + 0 + 0.72 - 0.325 = 1.395$$

(d) How many parameters are there? There are four predictors entering two units in the first hidden layer, and two parameters representing bias for each of the units in the hidden layer. This indicates there are $(4 \times 2) + 2 = 10$ parameters between the input and first hidden layers. There are two weighted units coming in from the first hidden layer towards 3 units in the second hidden layer, and three parameters representing bias. Thus, $(2 \times 3) + 3 = 9$ parameters exist between the two hidden layers. The final push towards the output layer includes three incoming weighted units towards a single final output plus one count of bias, giving us $(3 \times 1) + 1 = 4$ parameters. Thus, there are $10 + 9 + 4 = 23$ parameters in this neural network.

Question 2: ISLP § 10.10.4

Only a subset of the subproblems in this question are to be answered.

Consider a CNN that takes in 32×32 grayscale images and has a single convolution layer with three 5×5 convolution filters (without boundary padding).



(a) Draw a sketch of the input and first hidden layer.

(b) How many parameters are in this model? The parameters for a convolutional neural network consist of the weights in each filter and an extra bias term per filter.

For each of the three 5×5 convolution filter, there are $5 \times 5 \times 1 = 25$ weights, with the 1 representing the single input channel (grayscale), or the depth.

$$
\begin{aligned}
\text{Parameters} &= (\text{filter height} \times \text{filter width} \times \text{input channels} + 1) \times \text{number of filters} = (25 + 1) \times 3 \\
&= 26 \times 3 \\
&= 78
\end{aligned}
$$

Acknowledgments

I created the sketch of the convolutional neural network in Question 2 using a handy tool created by Alexander LeNail, a postdoc at Harvard Medical School, called NN-SVG. The tool can produce three types of neural network diagrams, allowing the user to customize the different layers and filters, and download an SVG.