Solution 1

Step 1: Identify the network architecture

We are given the following about a feedforward neural net:

- Input layer (L_i): 10 nodes
- Hidden layer 1 (L_{hl1}): 1000 nodes
- Hidden layer 2 (L_{hl2}): 1000 nodes
- Hidden layer 3 (L_{hl3}): 1000 nodes
- Hidden layer 4 (L_{hl4}): 1000 nodes
- Output layer (L_o): 1 node

Step 2: Calculate parameters between each layer pair

In a fully connected neural network, the number of parameters between two adjacent layers is the product of the number of nodes plus an offset term at each node. Since, we are only looking for the approximate number of parameters, we can ignore the offset terms and the number of parametersis the following:

$$\begin{aligned}
\text{number of parameters} &\approx (L_i \cdot L_{hl1}) + (L_{hl1} \cdot L_{hl2}) + (L_{hl2} \cdot L_{hl3}) + (L_{hl3} \cdot L_{hl4}) + (L_{hl4} \cdot L_o) \\
\text{number of parameters} &\approx (10 \cdot 1000) + (1000 \cdot 1000) + (1000 \cdot 1000) + (1000 \cdot 1) \\
\text{number of parameters} &\approx 10,000 + 1,000,000 + 1,000,000 + 1,000,000 + 1,000 \\
\text{number of parameters} &\approx 3,011,000
\end{aligned}$$

$\therefore$ The neural network has approximately 3,011,000 parameters.

Solution 2

Step 1: Define probability of label using softmax

The probability of a label j using softmax is defined as:

$$Pr(y_i) = \frac{e^{y_i}}{\sum_{j=1}^{n} e^{y_j}}$$

Step 2: Calculate probabilities

Now we can calculate the softmax probability for each label:

$$\begin{aligned}
Pr(y_1) &= \frac{e^{y_1}}{\sum_{j=1}^{4} e^{y_j}} = \frac{e^{y_1}}{e^{y_1}+e^{y_2}+e^{y_3}+e^{y_4}} = \frac{e^{1.0}}{5.0862} \approx \frac{2.7183}{5.0862} \approx 0.5344 \\
Pr(y_2) &= \frac{e^{y_2}}{\sum_{j=1}^{4} e^{y_j}} = \frac{e^{y_1}}{e^{y_1}+e^{y_2}+e^{y_3}+e^{y_4}} = \frac{e^{0.0}}{5.0862} = \frac{1.0000}{5.0862} \approx 0.1966 \\
Pr(y_3) &= \frac{e^{y_3}}{\sum_{j=1}^{4} e^{y_j}} = \frac{e^{y_1}}{e^{y_1}+e^{y_2}+e^{y_3}+e^{y_4}} = \frac{e^{-1.0}}{5.0862} \approx \frac{0.3679}{5.0862} \approx 0.0723 \\
Pr(y_4) &= \frac{e^{y_4}}{\sum_{j=1}^{4} e^{y_j}} = \frac{e^{y_1}}{e^{y_1}+e^{y_2}+e^{y_3}+e^{y_4}} = \frac{e^{0.0}}{5.0862} = \frac{1.0000}{5.0862} \approx 0.1966
\end{aligned}$$

Step 3: Identify the most likely label

Comparing the probabilities:

$$\begin{aligned}
Pr(y_1) &\approx 0.5344 \\
Pr(y_2) &\approx 0.1966 \\
Pr(y_3) &\approx 0.0723 \\
Pr(y_4) &\approx 0.1966
\end{aligned}$$

Hence, $P(y_1)$ has the highest probability.

$\therefore$ The most likely label is label 1 (y_1), with a probability of approximately 0.5344 (or about 53.44%).

Solution 3

Step 1: Neural Network architecture

The neural network has the following properties:

- 2 input units: x_1 and x_2
- 2 hidden units with ReLU activation
- 1 output unit

Where the ReLU (Rectified Linear Unit) activation function is defined as:

$$\text{ReLU}(z) = \max(0, z)$$

Step 2: Define structure

Let:

- h_1 and h_2 as the outputs of the two hidden units
- $W^{(1)}$ as the weight matrix from input to hidden layer
- $b^{(1)}$ as the bias vector for the hidden layer
- $W^{(2)}$ as the weight vector from hidden to output layer
- $b^{(2)}$ as the bias for the output unit

The forward pass through the network can be written as:

$$\begin{aligned}
h_1 &= \text{ReLU}(w_{11}^{(1)} x_1 + w_{12}^{(1)} x_2 + b_1^{(1)}) \\
h_2 &= \text{ReLU}(w_{21}^{(1)} x_1 + w_{22}^{(1)} x_2 + b_2^{(1)}) \\
y &= w_1^{(2)} h_1 + w_2^{(2)} h_2 + b^{(2)}
\end{aligned}$$

We need to find values for these weights and biases such that the output y matches the XOR function.

Step 3: Determine the weights and biases

The hidden units should have the following logic:

- h_1 should activate (output positive value) when at least one input is 1
- h_2 should activate (output positive value) when both inputs are 1

For h_1, we want:

$$\begin{aligned}
h_1(0,0) &= \text{ReLU}(w_{11}^{(1)} \cdot 0 + w_{12}^{(1)} \cdot 0 + b_1^{(1)}) = 0 \\
h_1(0,1) &= \text{ReLU}(w_{11}^{(1)} \cdot 0 + w_{12}^{(1)} \cdot 1 + b_1^{(1)}) > 0 \\
h_1(1,0) &= \text{ReLU}(w_{11}^{(1)} \cdot 1 + w_{12}^{(1)} \cdot 0 + b_1^{(1)}) > 0 \\
h_1(1,1) &= \text{ReLU}(w_{11}^{(1)} \cdot 1 + w_{12}^{(1)} \cdot 1 + b_1^{(1)}) > 0
\end{aligned}$$

Let $w_{11}^{(1)} = 1$, $w_{12}^{(1)} = 1$, and $b_1^{(1)} = -0.5$, then subsitiute into equations above to see if conditions are met:

$$\begin{aligned}
h_1(0,0) &= \text{ReLU}(0 + 0 - 0.5) = \text{ReLU}(-0.5) = 0 \\
h_1(0,1) &= \text{ReLU}(0 + 1 - 0.5) = \text{ReLU}(0.5) = 0.5 \\
h_1(1,0) &= \text{ReLU}(1 + 0 - 0.5) = \text{ReLU}(0.5) = 0.5 \\
h_1(1,1) &= \text{ReLU}(1 + 1 - 0.5) = \text{ReLU}(1.5) = 1.5
\end{aligned}$$

We have: $h_1(0,0) = 0$, $h_1(1,0) = 0.5 > 0$, $h_1(0,1) = 0.5 > 0$, and $h_1(1,1) = 1.5 > 0$. Hence, the conditions for h_1 are met.

For h_2, we want:

$$\begin{aligned}
h_2(0,0) &= \text{ReLU}(w_{21}^{(1)} \cdot 0 + w_{22}^{(1)} \cdot 0 + b_2^{(1)}) = 0 \\
h_2(0,1) &= \text{ReLU}(w_{21}^{(1)} \cdot 0 + w_{22}^{(1)} \cdot 1 + b_2^{(1)}) = 0 \\
h_2(1,0) &= \text{ReLU}(w_{21}^{(1)} \cdot 1 + w_{22}^{(1)} \cdot 0 + b_2^{(1)}) = 0 \\
h_2(1,1) &= \text{ReLU}(w_{21}^{(1)} \cdot 1 + w_{22}^{(1)} \cdot 1 + b_2^{(1)}) > 0
\end{aligned}$$

Let, $w_{21}^{(1)} = 1$, $w_{22}^{(1)} = 1$, and $b_2^{(1)} = -1.5$, then subsitiute into equations above to see if conditions are met:

$$\begin{aligned}
h_2(0,0) &= \text{ReLU}(0 + 0 - 1.5) = \text{ReLU}(-1.5) = 0 \\
h_2(0,1) &= \text{ReLU}(0 + 1 - 1.5) = \text{ReLU}(-0.5) = 0 \\
h_2(1,0) &= \text{ReLU}(1 + 0 - 1.5) = \text{ReLU}(-0.5) = 0 \\
h_2(1,1) &= \text{ReLU}(1 + 1 - 1.5) = \text{ReLU}(0.5) = 0.5
\end{aligned}$$

We have: $h_2(0,0) = 0$, $h_2(1,0) = 0$, $h_2(0,1) = 0$, and $h_2(1,1) = 0.5 > 0$. Hence, the conditions for h_2 are met.

Step 5: Determine the output layer weights

The output y is defined as:

$$y = w_1^{(2)} h_1 + w_2^{(2)} h_2 + b^{(2)}$$

We want:

$$\begin{aligned}
y(0,0) &= w_1^{(2)} \cdot 0 + w_2^{(2)} \cdot 0 + b^{(2)} = 0 \\
y(0,1) &= w_1^{(2)} \cdot 0.5 + w_2^{(2)} \cdot 0 + b^{(2)} = 1 \\
y(1,0) &= w_1^{(2)} \cdot 0.5 + w_2^{(2)} \cdot 0 + b^{(2)} = 1 \\
y(1,1) &= w_1^{(2)} \cdot 1.5 + w_2^{(2)} \cdot 0.5 + b^{(2)} = 0
\end{aligned}$$

From the $y(0,0)$ equation, we get $b^{(2)} = 0$. From the $y(0,1)$ and $y(1,0)$ equations, we get $w_1^{(2)} \cdot 0.5 = 1$, so $w_1^{(2)} = 2$. From the $y(1,1)$ equation, we get $2 \cdot 1.5 + w_2^{(2)} \cdot 0.5 = 0$, which gives $3 + 0.5w_2^{(2)} = 0$, so $w_2^{(2)} = -6$.

Verify the weights and biases:

$$\begin{aligned}
y(0,0) &= 2 \cdot 0 + (-6) \cdot 0 + 0 = 0 \\
y(0,1) &= 2 \cdot 0.5 + (-6) \cdot 0 + 0 = 1 \\
y(1,0) &= 2 \cdot 0.5 + (-6) \cdot 0 + 0 = 1 \\
y(1,1) &= 2 \cdot 1.5 + (-6) \cdot 0.5 + 0 = 3 - 3 = 0
\end{aligned}$$

$\therefore$ the neural network implements the XOR function with the following parameters:

Input to hidden layer weights:

$$W^{(1)} = \begin{bmatrix} w_{11}^{(1)} & w_{12}^{(1)} \\ w_{21}^{(1)} & w_{22}^{(1)} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

Hidden layer biases:

$$b^{(1)} = \begin{bmatrix} b_1^{(1)} \\ b_2^{(1)} \end{bmatrix} = \begin{bmatrix} -0.5 \\ -1.5 \end{bmatrix}$$

Hidden to output layer weights:

$$W^{(2)} = \begin{bmatrix} w_1^{(2)} & w_2^{(2)} \end{bmatrix} = \begin{bmatrix} 2 & -6 \end{bmatrix}$$

Output layer bias:

$$b^{(2)} = 0$$

Solution 4 (a)

Step 1

Let:

- $h_1 = \text{ReLU}(x_1 + x_2)$
- $h_2 = \text{ReLU}(x_1 + x_2 - 1)$
- $y = h_1 - h_2$

Now, calculate h_1 and h_2 for each input combination:
For $(x_1, x_2) = (0, 0)$:

$$h_1 = \text{ReLU}(0+0) = \text{ReLU}(0) = 0 \quad \text{and} \quad h_2 = \text{ReLU}(0+0-1) = \text{ReLU}(-1) = 0$$

For $(x_1, x_2) = (0, 1)$:

$$h_1 = \text{ReLU}(0+1) = \text{ReLU}(1) = 1 \quad \text{and} \quad h_2 = \text{ReLU}(0+1-1) = \text{ReLU}(0) = 0$$

For $(x_1, x_2) = (1, 0)$:

$$h_1 = \text{ReLU}(1+0) = \text{ReLU}(1) = 1 \quad \text{and} \quad h_2 = \text{ReLU}(1+0-1) = \text{ReLU}(0) = 0$$

For $(x_1, x_2) = (1, 1)$:

$$h_1 = \text{ReLU}(1+1) = \text{ReLU}(2) = 2 \quad \text{and} \quad h_2 = \text{ReLU}(1+1-1) = \text{ReLU}(1) = 1$$

Step 2

Now we compute $y = h_1 - h_2$ for each case:
For $(x_1, x_2) = (0, 0)$: $y = h_1 - h_2 = 0 - 0 = 0$

For $(x_1, x_2) = (0, 1)$: $y = h_1 - h_2 = 1 - 0 = 1$

For $(x_1, x_2) = (1, 0)$: $y = h_1 - h_2 = 1 - 0 = 1$

For $(x_1, x_2) = (1, 1)$: $y = h_1 - h_2 = 2 - 1 = 1$

Step 3

Summarizing our results in a truth table:

x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	1

This truth table corresponds to the logical OR function. The output is 1 when at least one of the inputs is 1, and 0 only when both inputs are 0.

$\therefore$ The neural network realizes the OR function: $\text{OR}(x_1, x_2) = x_1 \vee x_2$.

Solution 4 (b)

Step 1

Let:

- $h_1 = \text{ReLU}(x)$
- $h_2 = \text{ReLU}(-x)$
- $y = h_1 + h_2$

Determine behavior of each hidden unit for different values of $x \in \mathbb{R}$:
For $h_1 = \text{ReLU}(x)$:

$$h_1 = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases}$$

For $h_2 = \text{ReLU}(-x)$:

$$h_2 = \begin{cases} -x & \text{if } x \leq 0 \\ 0 & \text{if } x > 0 \end{cases}$$

Step 2

Now we compute $y = h_1 + h_2$ by considering different cases:
Case 1: $x > 0$:

$$\begin{aligned} h_1 &= x \\ h_2 &= 0 \\ y &= h_1 + h_2 = x + 0 = x \end{aligned}$$

Case 2: $x < 0$:

$$\begin{aligned} h_1 &= 0 \\ h_2 &= -x \\ y &= h_1 + h_2 = 0 + (-x) = -x \end{aligned}$$

Case 3: $x = 0$:

$$\begin{aligned} h_1 &= \text{ReLU}(0) = 0 \\ h_2 &= \text{ReLU}(-0) = \text{ReLU}(0) = 0 \\ y &= h_1 + h_2 = 0 + 0 = 0 \end{aligned}$$

Hence, we can write the output function as:

$$y = \begin{cases} x & \text{if } x \geq 0 \\ -x & \text{if } x < 0 \end{cases}$$

$\therefore$ The neural network realizes the absolute value function: $f(x) = |x|$.

Solution 5

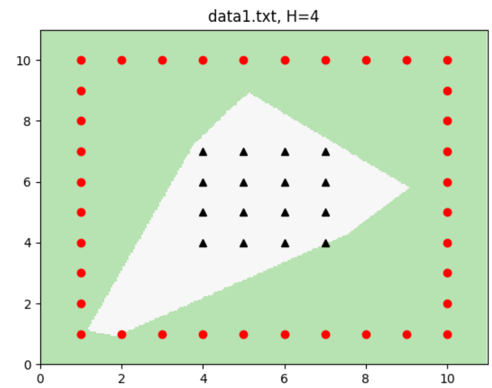
Results Table

Below is a summary of the best models found for each dataset.

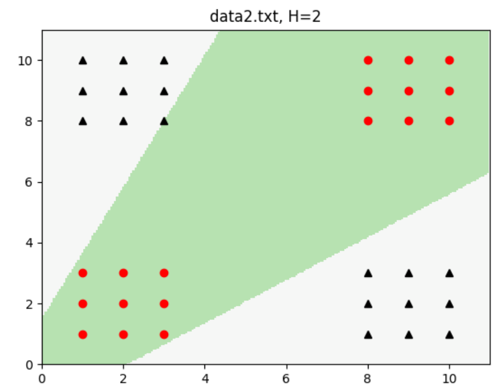
data	H value	iterations	error rate
data1.txt	4	15955	0
data2.txt	2	386	0
data3.txt	2	15257	5
data4.txt	4	46	0
data5.txt	2	2044	0
noisy	4	1533	125

Note: The noisy dataset was run with two hidden layers.

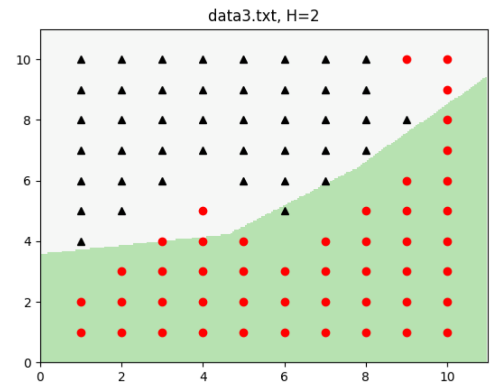
Plots



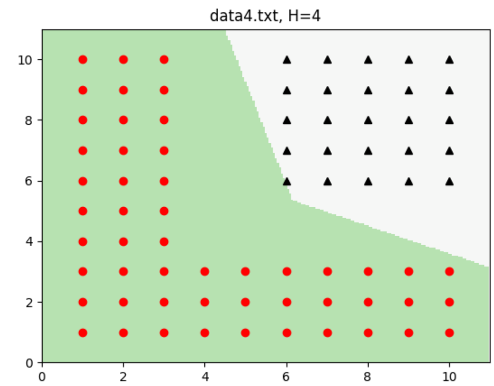
(a) Plot for data1.txt



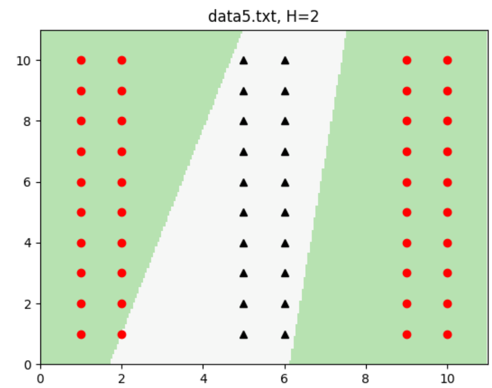
(b) Plot for data2.txt



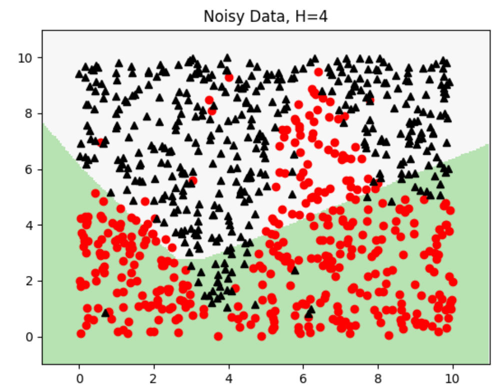
(c) Plot for data3.txt



(d) Plot for data4.txt



(e) Plot for data5.txt



(f) Plot for noisy data and fit with two hidden layers

Figure 1: Summary of model plots for different datasets.

Solution 6

Image Comparison: Orginal vs Normalized

Below is the image comparision.

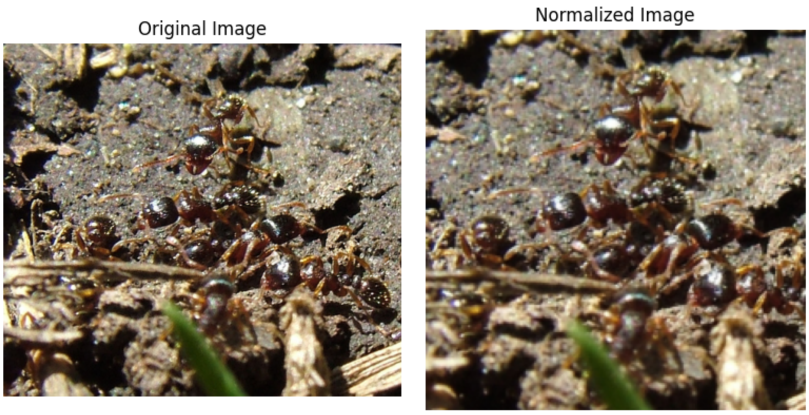


Figure 2: Left: Original image. Right: Normalized image that was resized and normalized for ResNet50 input.

Model Test Accuracies

classifier	**test accuracy**
Logistic Regression	0.9542
k-NN ($k = 1$)	0.9673
k-NN ($k = 3$)	0.9542
k-NN ($k = 5$)	0.9608

Bonus: Evaluation

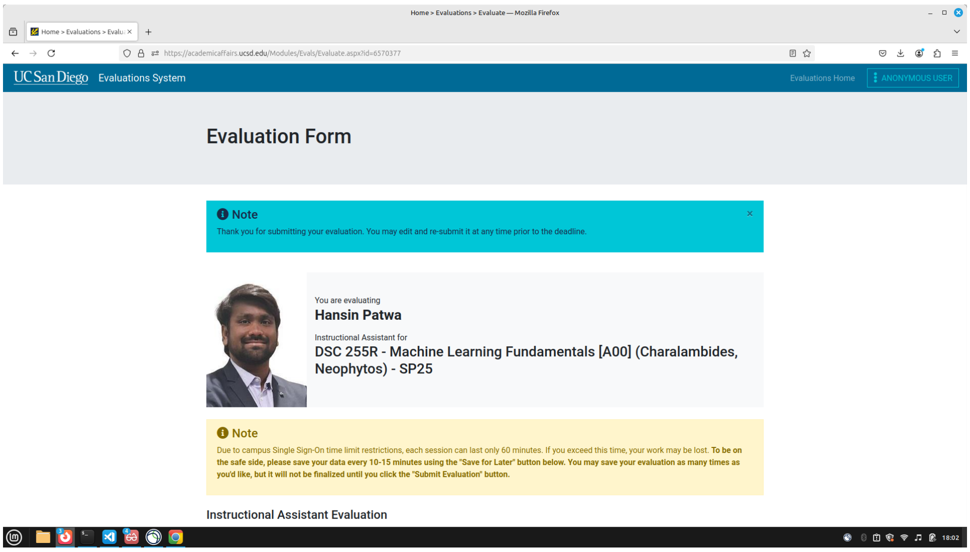


Figure 3: Hansin Evaluation

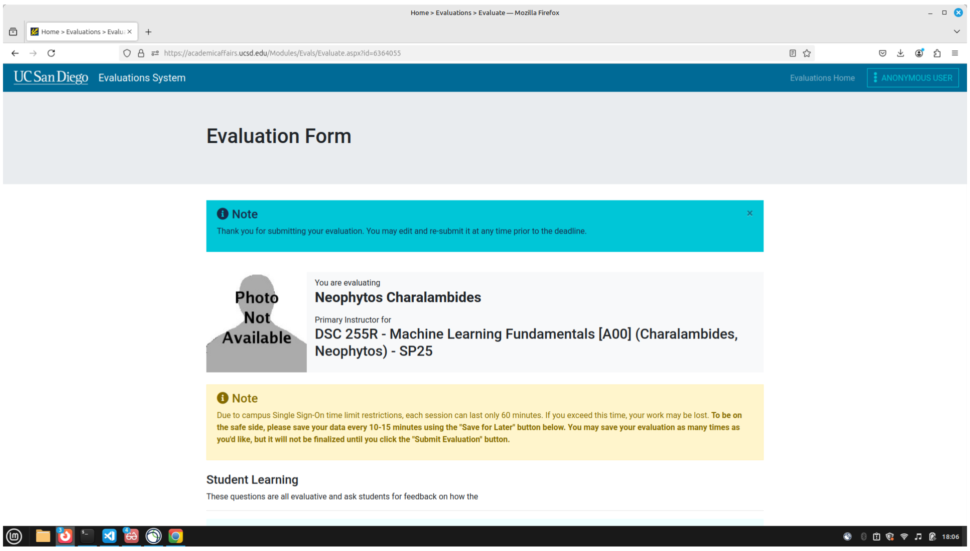


Figure 4: Neo Evaluation