ENGR 3321: Introduction to Deep Learning for Robotics

Neural Network N11:

Multi-Input, One-Hidden Layer, One-Output Model

09/29/2025

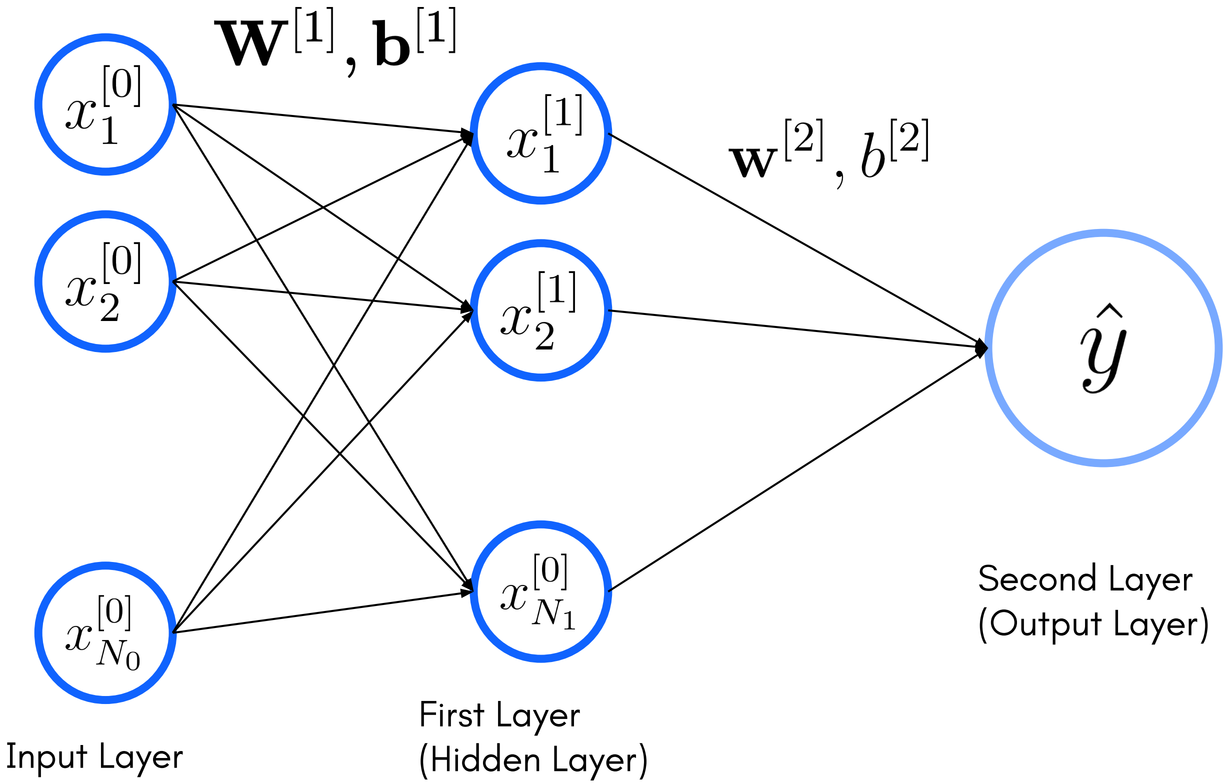
Outline

- Representations
- Training Process Review
  - Validation Dataset
  - Back-Propagation

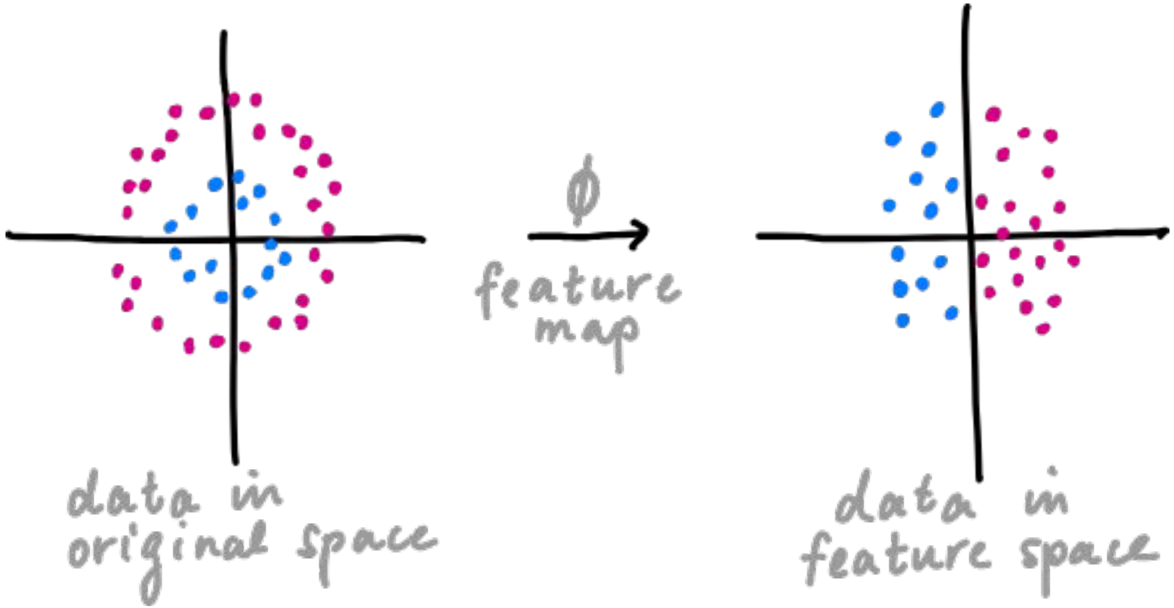
Review: Model Training

1. Prepare datasets: train, validation
2. (Randomly) Initialize model parameters: w, b.
3. Evaluate the model with a metric (e.g. BCE).
4. Calculate gradient of loss.
5. Update parameters a small step on the directions descending the gradient of loss.
6. Repeat 3 to 5 until converge.

1 Hidden Layer Neural Network



Feature Transformation



Individual Representation

$$\hat{y} = \sigma\left(w_1^{[2]}x_1^{[1]} + w_2^{[2]}x_2^{[1]} + \ldots + w_{N_1}^{[2]}x_{N_1}^{[1]} + b^{[2]}\right)$$

Where

$$x_1^{[1]} = \sigma\left(w_{11}^{[1]}x_1^{[0]} + w_{21}^{[1]}x_2^{[0]} + \ldots + w_{N_01}^{[1]}x_{N_0}^{[0]} + b_1^{[1]}\right)$$

$$x_2^{[1]} = \sigma\left(w_{12}^{[1]}x_1^{[0]} + w_{22}^{[1]}x_2^{[0]} + \ldots + w_{N_02}^{[1]}x_{N_0}^{[0]} + b_2^{[1]}\right)$$

$$\vdots$$

$$x_{N_1}^{[1]} = \sigma\left(w_{1N_1}^{[1]}x_1^{[0]} + w_{2N_1}^{[1]}x_2^{[0]} + \ldots + w_{N_0N_1}^{[1]}x_{N_0}^{[0]} + b_{N_1}^{[1]}\right)$$

Input Feature Matrix

$$\mathbf{X}^{[0]} = \begin{bmatrix} {}^{(1)}x_1^{[0]} & {}^{(1)}x_2^{[0]} & \cdots & {}^{(1)}x_{N_0}^{[0]} \\ {}^{(2)}x_1^{[0]} & {}^{(2)}x_2^{[0]} & \cdots & {}^{(2)}x_{N_0}^{[0]} \\ & & \cdots & \\ {}^{(M)}x_1^{[0]} & {}^{(1)}x_2^{[0]} & \cdots & {}^{(M)}x_{N_0}^{[0]} \end{bmatrix}_{(M,N_0)}$$

First-Layer Parameters

$$\mathbf{W}^{[1]} = \begin{bmatrix} w_{11}^{[1]} & w_{21}^{[1]} & \dots & w_{N_0 1}^{[1]} \\ w_{12}^{[1]} & w_{22}^{[1]} & \dots & w_{N_0 2}^{[1]} \\ & & \dots & \\ w_{1N_1}^{[1]} & w_{2N_1}^{[1]} & \dots & w_{N_0 N_1}^{[1]} \end{bmatrix}_{(N_1, N_0)}$$

$$\mathbf{b}^{[1]} = \begin{bmatrix} b_1^{[1]} & b_2^{[1]} & \dots & b_{N_1}^{[1]} \end{bmatrix}_{(1, N_1)}$$

Second-Layer Parameters

$$\mathbf{w}^{[2]} = \begin{bmatrix} w_1^{[2]} & w_2^{[2]} & \dots & w_{N_1}^{[2]} \end{bmatrix}_{(1,N_1)}$$

$b^{[2]}$, scalar

Forward Propagation

$$\boldsymbol{X}^{[1]}=\sigma(\boldsymbol{X}^{[0]}\cdot\boldsymbol{W}^{[1]T}+\boldsymbol{b}^{[1]})=\sigma(\boldsymbol{Z}^{[1]}))$$

$$\mathbf{X}^{[1]}=\sigma\left(\begin{bmatrix} {}^{(1)}x_1^{[0]} & {}^{(1)}x_2^{[0]} & \dots & {}^{(1)}x_{N_0}^{[0]} \\ {}^{(2)}x_1^{[0]} & {}^{(2)}x_2^{[0]} & \dots & {}^{(2)}x_{N_0}^{[0]} \\ & & \dots & \\ {}^{(M)}x_1^{[0]} & {}^{(M)}x_2^{[0]} & \dots & {}^{(M)}x_{N_0}^{[0]} \end{bmatrix} \cdot \begin{bmatrix} w_{11}^{[1]} & w_{12}^{[1]} & \dots & w_{1N_1}^{[1]} \\ w_{21}^{[1]} & w_{22}^{[1]} & \dots & w_{2N_1}^{[1]} \\ & & \dots & \\ w_{N_01}^{[1]} & w_{N_02}^{[1]} & \dots & w_{N_0N_1}^{[1]} \end{bmatrix} + \begin{bmatrix} b_1^{[1]} & b_2^{[1]} & \dots & b_{N_1}^{[1]} \\ b_1^{[1]} & b_2^{[1]} & \dots & b_{N_1}^{[1]} \\ & & \dots & \\ b_1^{[1]} & b_2^{[1]} & \dots & b_{N_1}^{[1]} \end{bmatrix}\right)$$

$$\hat{\mathbf{y}}=\sigma\left(\mathbf{X}^{[1]}\mathbf{w}^{[\mathbf{2}]\mathrm{T}}+b^{[2]}\right)=\sigma\left(\mathbf{Z}^{[2]}\right)$$

$$\hat{\mathbf{y}}=\sigma\left(\begin{bmatrix} {}^{(1)}x_1^{[0]} & {}^{(1)}x_2^{[1]} & \dots & {}^{(1)}x_{N_1}^{[1]} \\ {}^{(2)}x_1^{[1]} & {}^{(2)}x_2^{[1]} & \dots & {}^{(2)}x_{N_1}^{[1]} \\ & & \dots & \\ {}^{(M)}x_1^{[1]} & {}^{(M)}x_2^{[1]} & \dots & {}^{(M)}x_{N_1}^{[1]} \end{bmatrix} \cdot \begin{bmatrix} w_1^{[2]} \\ w_2^{[2]} \\ \dots \\ w_{N_1}^{[2]} \end{bmatrix} + \begin{bmatrix} b^{[2]} \\ b^{[2]} \\ \dots \\ b^{[2]} \end{bmatrix}\right)$$

Target and Prediction

$$\mathbf{y} = \begin{bmatrix} {}^{(1)}y \\ {}^{(2)}y \\ . \\ . \\ . \\ {}^{(M)}y \end{bmatrix}_{(M,1)}$$

$$\hat{\mathbf{y}} = \begin{bmatrix} {}^{(1)}\hat{y} \\ {}^{(2)}\hat{y} \\ . \\ . \\ . \\ {}^{(M)}\hat{y} \end{bmatrix}_{(M,1)}$$

Matrix Form

$$\hat{\mathbf{y}} = \sigma(\mathbf{X}^{[1]} \cdot \mathbf{w}^{[2]T} + b^{[2]})$$

$(M, 1)$

$$= \sigma(\sigma(\mathbf{X}^{[0]} \cdot \mathbf{W}^{[1]T} + \mathbf{b}^{[1]}) \cdot \mathbf{w}^{[2]T} + b^{[2]})$$

(M, N_0) (N_0, N_1) (M, N_1) $(N_1, 1)$ $(M, 1)$

Prepare Datasets: Training

A dataset with M_{tr} samples:

- Each sample has N features: $x_1, x_2, \ldots, x_N$
- Each sample is labeled: y ($y \in \{0, 1\}$ for binary classification)

$$\mathcal{D} = \{({}^{(1)}x_1^{[0]}, {}^{(1)}x_2^{[0]}, \ldots, {}^{(1)}x_N^{[0]}, {}^{(1)}y), ({}^{(2)}x_1^{[0]}, {}^{(2)}x_2^{[0]}, \ldots, {}^{(2)}x_N^{[0]}, {}^{(2)}y), \ldots, ({}^{(M_{tr})}x_1^{[0]}, {}^{(M_{tr})}x_2^{[0]}, \ldots, {}^{(M_{tr})}x_N^{[0]}, {}^{(M_{tr})}y)\}$$

$$= \{({}^{(1)}\mathbf{x}^{[0]}, {}^{(1)}y), ({}^{(2)}\mathbf{x}^{[0]}, {}^{(2)}y), \ldots, ({}^{(M_{tr})}\mathbf{x}^{[0]}, {}^{(M_{tr})}y)\}$$

Prepare Datasets: Validation

A dataset with M_v ($M_v < M_{tr}$) samples:

- Each sample has N features: $\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_N$
- Each sample is labeled: y
- Validation dataset can be used to evaluate model.
- Validation dataset does not participate into model updating

$$\mathcal{D} = \{({}^{(1)}\tilde{x}_1, {}^{(1)}\tilde{x}_2, \ldots, {}^{(1)}\tilde{x}_N, {}^{(1)}y), ({}^{(2)}\tilde{x}_1, {}^{(2)}\tilde{x}_2, \ldots, {}^{(2)}\tilde{x}_N, {}^{(2)}y), \ldots, ({}^{(M_v)}\tilde{x}_1, {}^{(M_v)}\tilde{x}_2, \ldots, {}^{(M_v)}\tilde{x}_N, {}^{(M_v)}y)\}$$

$$= \{({}^{(1)}\tilde{\mathbf{x}}, {}^{(1)}y), ({}^{(2)}\tilde{\mathbf{x}}, {}^{(2)}y), \ldots, ({}^{(M_v)}\tilde{\mathbf{x}}, {}^{(M_v)}y)\}$$

Binary Cross Entropy Loss

$$\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) = \frac{1}{M} \sum_{i=1}^{M} -{}^{(i)}y \ln {}^{(i)}\hat{y} - (1 - {}^{(i)}y) \ln(1 - {}^{(i)}\hat{y}) = \overline{-\mathbf{y} \ln \hat{\mathbf{y}} - (1 - \mathbf{y}) \ln(1 - \hat{\mathbf{y}})}$$

Back-Propagation (2nd layer)

$$\nabla \mathcal{L} = \left[\frac{\partial \mathcal{L}}{\partial w_{11}{}^{[1]}} \quad \cdots \quad \frac{\partial \mathcal{L}}{\partial w_{\mathrm{N_1 N_0}}{}^{[1]}} \quad \frac{\partial \mathcal{L}}{\partial b_1{}^{[1]}} \quad \cdots \quad \frac{\partial \mathcal{L}}{\partial b_{\mathrm{N_1}}{}^{[1]}} \quad \frac{\partial \mathcal{L}}{\partial w_1{}^{[2]}} \quad \cdots \quad \frac{\partial \mathcal{L}}{\partial w_{\mathrm{N_1}}{}^{[2]}} \quad \frac{\partial \mathcal{L}}{\partial b^{[2]}} \right]$$

$$= \left[\frac{\partial \mathcal{L}}{\partial \mathbf{W}^{[1]}} \quad \frac{\partial \mathcal{L}}{\partial \mathbf{b}^{[1]}} \quad \frac{\partial \mathcal{L}}{\partial \mathbf{w}^{[2]}} \quad \frac{\partial \mathcal{L}}{\partial \mathrm{b}^{[2]}} \right]$$

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}^{[2]}} = \frac{\partial \mathcal{L}}{\partial \hat{\mathbf{y}}} \frac{\partial \hat{\mathbf{y}}}{\partial \mathbf{Z}^{[2]}} \frac{\partial \mathbf{Z}^{[2]}}{\partial \mathbf{w}^{[2]}} = \frac{1}{M} (\hat{\mathbf{y}} - \mathbf{y})^T \cdot \mathbf{X}^{[1]}$$

$$\frac{\partial \mathcal{L}}{\partial b^{[2]}} = \frac{\partial \mathcal{L}}{\partial \hat{\mathbf{y}}} \frac{\partial \hat{\mathbf{y}}}{\partial \mathbf{Z}^{[2]}} \frac{\partial \mathbf{Z}^{[2]}}{\partial b^{[2]}} = \overline{\hat{\mathbf{y}} - \mathbf{y}}$$

$$\frac{\partial \mathcal{L}}{\partial \mathbf{X}^{[1]}} = \frac{\partial \mathcal{L}}{\partial \hat{\mathbf{y}}} \frac{\partial \hat{\mathbf{y}}}{\partial \mathbf{Z}^{[2]}} \frac{\partial \mathbf{Z}^{[2]}}{\partial \mathbf{X}^{[1]}} = (\hat{\mathbf{y}} - \mathbf{y}) \cdot \mathbf{w}^{[2]}$$

Back-Propagation (1st layer)

$$\frac{\partial \mathcal{L}}{\partial \mathbf{W}^{[1]}} = \frac{\partial \mathcal{L}}{\partial \mathbf{X}^{[1]}} \frac{\partial \mathbf{X}^{[1]}}{\partial \mathbf{Z}^{[1]}} \frac{\partial \mathbf{Z}^{[1]}}{\partial \mathbf{W}^{[1]}} = \frac{1}{M}[(\hat{\mathbf{y}} - \mathbf{y}) \cdot \mathbf{w}^{[2]} * \mathbf{X}^{[1]} * (1 - \mathbf{X}^{[1]})]^T \cdot \mathbf{X}^{[0]}$$

$$\frac{\partial \mathcal{L}}{\partial \mathbf{b}^{[1]}} = \frac{\partial \mathcal{L}}{\partial \mathbf{X}^{[1]}} \frac{\partial \mathbf{X}^{[1]}}{\partial \mathbf{Z}^{[1]}} \frac{\partial \mathbf{Z}^{[1]}}{\partial \mathbf{b}^{[1]}} = \overline{(\hat{\mathbf{y}} - \mathbf{y}) \cdot \mathbf{w}^{[2]} * \mathbf{X}^{[1]} * (1 - \mathbf{X}^{[1]})}, \ \mathrm{axis} = 0$$

Gradient Descent Optimization

Given dataset: $\left\{ \left({}^{(1)}\mathbf{x}, {}^{(1)}y \right), \left({}^{(2)}\mathbf{x}, {}^{(2)}y \right), \ldots, \left({}^{(M)}\mathbf{x}, {}^{(M)}y \right) \right\}$

Initialize $\mathbf{W}^{[1]}$, $\mathbf{w}^{[\mathbf{2}]}$, $\mathbf{b}^{[1]}\, and\, b^{[2]}$

Repeat until converge {

$$\mathbf{W}^{[1]} := \mathbf{W}^{[1]} - \alpha \frac{\partial \mathcal{L}}{\partial \mathbf{W}^{[1]}}$$

$$\mathbf{w}^{[2]} := \mathbf{w}^{[2]} - \alpha \frac{\partial \mathcal{L}}{\partial \mathbf{w}^{[2]}}$$

$$\mathbf{b}^{[1]} := \mathbf{b}^{[1]} - \alpha \frac{\partial \mathcal{L}}{\partial \mathbf{b}^{[1]}}$$

$$b^{[2]} := b^{[2]} - \alpha \frac{\partial \mathcal{L}}{\partial b^{[2]}}$$

}

where α is learning rate