ENGR 3321: Introduction to Deep Learning for Robotics

Neural Network 201:
Two-Input One-Output Linear Function

09/10/2025

Outline

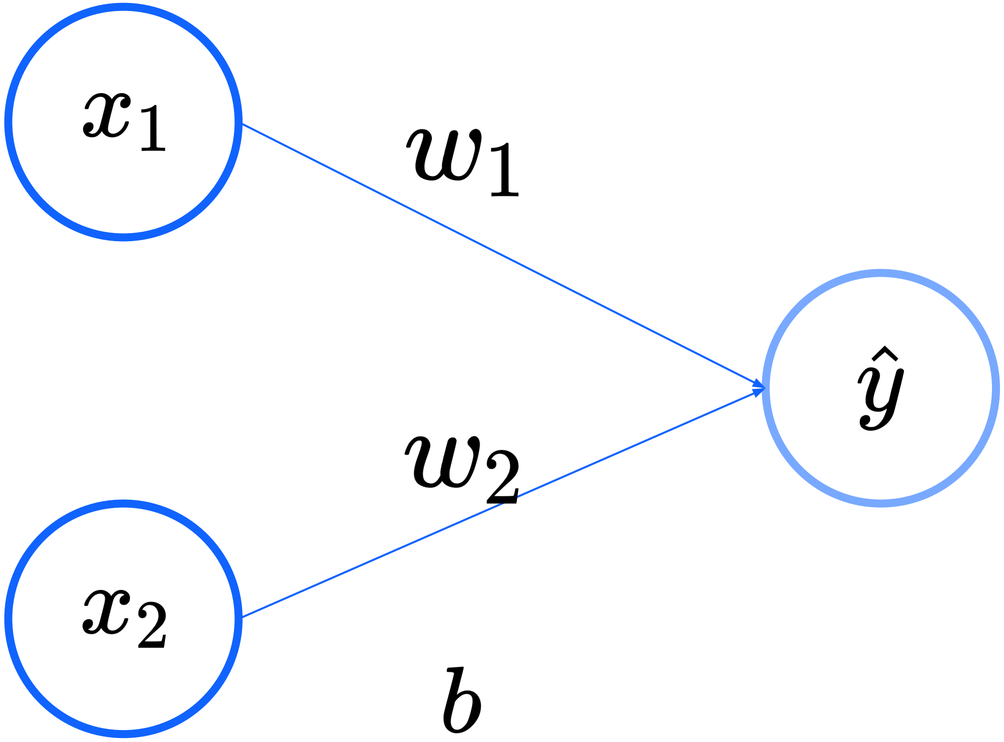
- Matrix Format
- Review: Model Training

Two-Input One-Output Linear Function

A Linear model predicts a sample's property using two of its features.

$$f(x_1, x_2) = \hat{y} = w_1 x_1 + w_2 x_2 + b$$

Neural Network Form



Input (Feature) Matrix

$$\mathbf{x_1} = \begin{bmatrix} {}^{(1)}x_1 \\ {}^{(2)}x_1 \\ \cdot \\ \cdot \\ \cdot \\ {}^{(M)}x_1 \end{bmatrix}_{(M,1)} \qquad \mathbf{x_2} = \begin{bmatrix} {}^{(1)}x_2 \\ {}^{(2)}x_2 \\ \cdot \\ \cdot \\ \cdot \\ {}^{(M)}x_2 \end{bmatrix}_{(M,1)} \qquad \mathbf{X} = \begin{bmatrix} \mathbf{x_1} & \mathbf{x_2} \end{bmatrix}_{(M,2)}$$

Model Parameters

$$\mathbf{w} = \begin{bmatrix} w_1 & w_2 \end{bmatrix}_{(1,2)}$$

$$\mathbf{b} = \begin{bmatrix} b \\ b \\ \cdot \\ \cdot \\ \cdot \\ b \end{bmatrix}_{(M,1)}$$

Review: Model Training

1. Load dataset: X (features), y (labels)
2. (Randomly) Initialize model parameters: w, b.
3. Evaluate the model with a metric (e.g. MSE).
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.

Load Dataset

A dataset with M samples:

- Each sample has 2 features: x_1 and x_2
- Each sample is labeled: y

$$\mathcal{D} = \{({}^{(1)}x_1, {}^{(1)}x_2, {}^{(1)}y), ({}^{(2)}x_1, {}^{(2)}x_2, {}^{(2)}y), \ldots, ({}^{(M)}x_1, {}^{(M)}x_2, {}^{(M)}y)\}$$

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

Initialize Model

$$\hat{\mathbf{y}} = \mathbf{X} \cdot \mathbf{w}^{\mathbf{T}} + \mathbf{b}$$

$(M, 1)$ $(M, 2)$ $(2, 1)$ $(M, 1)$

Prediction

Matrix Multiplication

$${}^{(1)}\hat{y} = {}^{(1)}x_1 w_1 + {}^{(1)}x_2 w_2 + b$$

$${}^{(2)}\hat{y} = {}^{(2)}x_1 w_1 + {}^{(2)}x_2 w_2 + b$$

$$\vdots$$

$${}^{(M)}\hat{y} = {}^{(M)}x_1 w_1 + {}^{(M)}x_2 w_2 + b$$

Evaluate Model (MSE)

$$\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) = \frac{1}{M}\sum_{i=1}^{M}\left({}^{(i)}\hat{y} - {}^{(i)}y\right)^2 = \overline{(\hat{\mathbf{y}} - \mathbf{y})^2}$$

Loss

Gradient of Loss

$$\nabla \mathcal{L} = \begin{bmatrix} \frac{\partial \mathcal{L}}{\partial w_1} & \frac{\partial \mathcal{L}}{\partial w_2} & \frac{\partial \mathcal{L}}{\partial b} \end{bmatrix}$$

$$\frac{\partial \mathcal{L}}{\partial w_1} = \frac{\partial \mathcal{L}}{\partial \hat{\mathbf{y}}} \frac{\partial \hat{\mathbf{y}}}{\partial w_1} = \frac{1}{M} \sum_{i=1}^{M} ({}^{(i)}\hat{y} - {}^{(i)}y)^{(i)} x_1$$

$$\frac{\partial \mathcal{L}}{\partial w_2} = \frac{\partial \mathcal{L}}{\partial \hat{\mathbf{y}}} \frac{\partial \hat{\mathbf{y}}}{\partial w_2} = \frac{1}{M} \sum_{i=1}^{M} ({}^{(i)}\hat{y} - {}^{(i)}y)^{(i)} x_2$$

$$\frac{\partial \mathcal{L}}{\partial b} = \frac{\partial \mathcal{L}}{\partial \hat{\mathbf{y}}} \frac{\partial \hat{\mathbf{y}}}{\partial b} = \frac{1}{M} \sum_{i=1}^{M} ({}^{(i)}\hat{y} - {}^{(i)}y)$$

Vectorized Gradient

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = [\frac{\partial \mathcal{L}}{\partial w_1} \ \frac{\partial \mathcal{L}}{\partial w_2}] = \frac{1}{M}(\hat{\mathbf{y}} - \mathbf{y})^T \cdot \mathbf{X}$$

$(1, M)$ $(M, 2)$

Matrix Multiplication

$$\frac{\partial \mathcal{L}}{\partial b} = \overline{\hat{\mathbf{y}} - \mathbf{y}}$$

Vectorized Gradient Descent

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 $\mathbf{w}\, and\, b$

Repeat until converge {

$$\mathbf{w} := \mathbf{w} - \alpha \frac{\partial \mathcal{L}}{\partial \mathbf{w}}$$

$$b := b - \alpha \frac{\partial \mathcal{L}}{\partial b}$$

}

where α is learning rate