CSE 403: Machine Learning

Chapter 02: Supervised Machine Learning

Md Atikuzzaman
Lecturer
Department of Computer Science & Engineering
atik@cse.green.edu.bd

GREEN UNIVERSITY
OF BANGLADESH

Outline

What is Supervised Learning?

Definition

Supervised learning aims to learn a parametric function

$$f_\theta : \mathcal{X} \to \mathcal{Y}$$

from labeled training data.

- Training examples are pairs:

$$(x^{(i)}, y^{(i)}), \quad i = 1, \dots, m$$

- The model is trained to predict $\hat{y} = f_\theta(x)$.
- The central objective is **generalization** to unseen data.

A good model performs well not only on training data, but also on new samples from the same distribution.

Core Elements of Supervised Learning

1. **Input space** $\mathcal{X}$
2. **Output space** $\mathcal{Y}$
3. **Training dataset** $\mathcal{D}$
4. **Model class (hypothesis space)** $\mathcal{H}$
5. **Loss function** $\ell(\cdot,\cdot)$ and objective $J(\theta)$
6. **Optimization algorithm** (e.g., gradient descent)

$$\textbf{Data} + \textbf{Model} + \textbf{Loss} + \textbf{Optimizer} \quad\longrightarrow\quad \textbf{Predictive model}$$

Mathematical Structure of the Dataset

Training Dataset

$$\mathcal{D} = \{(x^{(i)}, y^{(i)})\}_{i=1}^{m} \subseteq \mathcal{X} \times \mathcal{Y}$$

- $x^{(i)} \in \mathbb{R}^d$: feature vector
- $y^{(i)} \in \mathcal{Y}$: target variable (label)
- m: number of training samples
- d: number of features

Example Dataset: House Price Prediction

Size (m^2)	Rooms	Price ($)
60	2	180000
80	2	220000
100	3	260000
120	3	300000
140	4	340000

$$x = \begin{bmatrix} \text{Size} \\ \text{Rooms} \end{bmatrix} \in \mathbb{R}^2, \qquad y = \text{Price} \in \mathbb{R}$$

Feature Representation and Dataset Splitting

Feature Representation

An input example $x^{(i)} \in \mathcal{X}$ is represented as a d-dimensional vector:

$$x^{(i)} = \begin{bmatrix} x_1^{(i)} \\ x_2^{(i)} \\ \vdots \\ x_d^{(i)} \end{bmatrix} \in \mathbb{R}^d.$$

- The set $\mathcal{X}$ is called the **feature space** (often $\mathcal{X} = \mathbb{R}^d$).

Dataset Splitting

Training	Validation	Test

- Training: learn parameters θ
- Validation: tune hyperparameters, select model
- Test: final evaluation (used once at the end)

All splits are assumed to be independent and identically distributed (i.i.d.) samples from the same underlying distribution $P_{X,Y}$.

Learning Objective

Empirical Risk Minimization (ERM)

Define the empirical risk (training objective):

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} \ell\big(f_\theta(x^{(i)}),\, y^{(i)}\big).$$

Optimization Problem

$$\hat{\theta} = \arg\min_{\theta \in \Theta} J(\theta).$$

- $\ell(\cdot,\cdot)$: loss for a single example
- $J(\theta)$: average loss over the dataset

Model Class (Hypothesis Space)

Hypothesis Space

$$\mathcal{H} = \left\{ f_\theta : \mathcal{X} \to \mathcal{Y} \;\middle|\; \theta \in \Theta \right\}.$$

Linear Regression Example

$$f_\theta(x) = \theta_0 + \theta^\top x, \qquad \theta \in \mathbb{R}^d.$$

- $\mathcal{H}$ determines the set of functions the learner is allowed to choose from.

Loss Functions

Regression (Squared Loss)

$$\ell\big(f_\theta(x), y\big) = \big(f_\theta(x) - y\big)^2$$

Binary Classification (Logistic Loss)

$$\ell\big(\hat{y}, y\big) = -\Big[y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})\Big], \qquad \hat{y} = \sigma(f_\theta(x))$$

Multiclass Classification (Cross-Entropy)

$$\ell(\hat{\mathbf{y}}, \mathbf{y}) = -\sum_{k=1}^{K} y_k \log(\hat{y}_k)$$

Optimization Algorithm

Gradient Descent (Vector Form)

$$\theta^{(t+1)} = \theta^{(t)} - \alpha \nabla_\theta J\big(\theta^{(t)}\big).$$

- $\alpha > 0$: learning rate
- Iterate until convergence (or early stopping using validation set)

Learning Workflow

Dataset $\mathcal{D} = \{(x^{(i)}, y^{(i)})\}$ → **Learning Setup** $\mathcal{H}$ + ℓ + optimizer → **Trained Model** parameters $\hat{\theta}$

Predict $\hat{y} = f_\theta(x)$ — Compute Loss $\ell(\hat{y}, y)$ — Update $\theta \leftarrow \theta - \alpha \nabla J(\theta)$

Training repeats the loop until validation performance stops improving.

Supervised Learning: Formal Framework

$$\underbrace{\mathcal{D} = \{(x^{(i)}, y^{(i)})\}_{i=1}^{m}}_{\text{Training data}} + \underbrace{(\mathcal{H}, \ell, \text{optimizer})}_{\text{Learning setup}} \longrightarrow \underbrace{f_{\hat{\theta}}}_{\text{Predictive model}}$$

Key equation

$$\hat{\theta} = \arg\min_{\theta \in \Theta} \frac{1}{m} \sum_{i=1}^{m} \ell\big(f_\theta(x^{(i)}), y^{(i)}\big).$$

Key Takeaways

- Supervised learning uses labeled data to learn $f_\theta : \mathcal{X} \to \mathcal{Y}$.
- The dataset $\mathcal{D}$ contains pairs (x, y) drawn i.i.d. from $P_{X,Y}$.
- A model class $\mathcal{H}$ restricts which functions can be learned.
- A loss function ℓ quantifies prediction error.
- Optimization minimizes $J(\theta)$ to obtain $\hat{\theta}$.

Supervised learning = Data + Model class + Loss + Optimization.

References

1. C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.
2. R. O. Duda, P. E. Hart, D. G. Stork, *Pattern Classification*, Wiley, 2000.
3. I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning*, MIT Press, 2016.
4. A. Zhang, Z. C. Lipton, M. Li, A. J. Smola, *Dive into Deep Learning*, Cambridge University Press, 2023.