# Chapter 1

# Supervised Learning Formulations

In this chapter, we will set up the standard theoretical formulation of supervised learning and introduce the *empirical risk minimization* (ERM) paradigm. The setup will apply to almost the entire monograph and the ERM paradigm will be the main focus of Chapter 2, 3, and 4.

## 1.1 Supervised learning

label 추론

In supervised learning, we have a dataset where each data point is associated with a label, and we aim to learn from the data a function that maps data points to their labels. The learned function can be used to infer the labels of test data points. More formally, suppose the data points, also called inputs, belong to some input space $\mathcal{X}$ (e.g. images of birds), and labels belong to the output space $\mathcal{Y}$ (e.g. bird species). Suppose we are interested in a specific joint probability distribution $P$ over $\overset{\text{input}}{\mathcal{X}} \times \overset{\text{output}}{\mathcal{Y}}$ (e.g. images of birds in North America), from which we draw a *training set*, (i.e we draw a a set of $n$ independent and identically distributed (i.i.d.) data points $\{(x^{(i)}, y^{(i)})\}_{i=1}^{n}$ from $P$). The goal of supervised learning is to learn a mapping (i.e. a function) from $\mathcal{X}$ to $\mathcal{Y}$ using the training data. Any such function $h : \mathcal{X} \to \mathcal{Y}$ is called a *predictor* (also *hypothesis* or *model*).

$h$ (mapping from $\mathcal{X}$ to $\mathcal{Y}$: predictor)

$h$(predictor)간 비교:
- "loss func. $\ell$"
- from $\mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$
  (예측값)(실제값) → 얼마나 다른지
- non-neg.
- loss made by the predictor $h$: $\ell(h(x), y)$
  ↓
  difference b/w $h(x)$ and the true label $y$
- best possible loss: 0
  $L(h)$가 0에 가깝게!

Given two predictors, how do we decide which is better? For that, we define a *loss function* over the predictions. There are several ways to define loss functions: for now, define a loss function $\ell$ as a function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$. Intuitively, the loss function takes two labels, the prediction made by a model $\hat{y}$ and the true label $y$, and gives a number that captures how different the two labels are. We assume $\ell$ is non-negative, i.e $\ell(\hat{y}, y) \geq 0$. Then, the loss of a model $h$ on an example $(x, y)$ is $\ell(h(x), y)$, i.e. the difference (as measured by $\ell$) between the prediction made by $h$ and the true label.

With these definitions, we are able to formalize the problem of supervised learning. Precisely, we seek to find a model $h$ that minimizes what we call the expected loss (or population loss or expected risk or population risk):

find $h$
that 최소화 $L(h)$ ← population loss

$$L(h) \triangleq \underbrace{\mathop{\mathbb{E}}_{(x,y)\sim p}\left[\underbrace{\ell(h(x), y)}_{\text{"loss"}}\right]}_{\text{의 expectation}}. \quad \geq 0 \tag{1.1}$$

Note that $L$ is nonnegative because $\ell$ is nonnegative. Typically, the loss function is designed so that the best possible loss is zero when $\hat{y}$ matches $y$ exactly. Therefore, the goal is to find $h$ such that $L(h)$ is as close to zero as possible.

**Examples: regression and classification problems.** Here are two standard types of supervised learning problems based on the properties of the output space:

- In the problem of *regression*, predictions are real numbers ($\mathcal{Y} = \mathbb{R}$). We would like predictions to be as close as possible to the real labels. A classical loss function that captures this is the squared error, $\ell(\hat{y}, y) = (\hat{y} - y)^2$.

- In the problem of *classification*, predictions are in a discrete set of $k$ unordered classes $\mathcal{Y} = [k] = \{1, \cdots, k\}$. One possible classification loss is the $0-1$ loss: $\ell(\hat{y}, y) = \mathbb{1}(\hat{y} \neq y)$, i.e. 0 if the prediction is equal to the true label, and 1 otherwise.

**Hypothesis class.** So far, we said we would like to find *any function* that minimizes population risk. However, in practice, we do not have a way of optimizing over arbitrary functions. Instead, we work within a more constrained set of functions $\mathcal{H}$, which we call the *hypothesis family* (or *hypothesis class*). Each element of $\mathcal{H}$ is a function $h : \mathcal{X} \to \mathcal{Y}$. Usually, we choose a set $\mathcal{H}$ that we know how to optimize over (e.g. linear functions, or neural networks).

이걸 bound 하려고 한다!

Given one particular function $h \in \mathcal{H}$, we define the *excess risk of h* with respect to $\mathcal{H}$ as the difference between the population risk of $h$ and the best possible population risk inside $\mathcal{H}$:

$$\underset{\substack{\text{excess risk of } h \\ \geq 0}}{E(h)} \triangleq \overbrace{\underset{\mathbb{E}(\ell(h), y)}{L(h)}}^{\text{pop. risk}} - \overbrace{\inf_{g \in \mathcal{H}} L(g)}^{\mathcal{H}\text{ 내 func. 중 best}}.$$

(가장 예에서 멀어지려고 하는 상태에서의 최... inf.)

ERM minimizer $\theta$ (혹은 $\theta$를 사용한 $h_\theta$)의 excess risk가 bounded 됨을 보이는게! 목표

Generally we need more assumptions about a specific problem and hypothesis class to bound absolute population risk, hence we focus on bounding the excess risk.

Usually, the family we choose to work with can be parameterized by a vector of parameters $\theta \in \Theta$. In that case, we can refer to an $\overset{\text{function}}{\text{element}}$ of $\mathcal{H}$ by $h_\theta$, making that explicit. An example of such a parametrization of the hypothesis class is $\mathcal{H} = \{h : h_\theta(x) = \theta^\top x, \theta \in \mathbb{R}^d\}$.

## 1.2 Empirical risk minimization

Optimizing the "training set loss"를 하는 것이 "low population loss"를 가진 model 로 이끈다.

Our ultimate goal is to minimize population risk. However, in practice we do not have access to the entire population: we only have a *training set* of $n$ data points, (drawn from the same distribution as the entire population.) While we cannot compute population risk, we can compute *empirical risk*, the loss over the training set, and try to minimize that. This is, in short, the paradigm known as *empirical risk minimization* (ERM): we optimize the training set loss, with the hope that this leads us to a model that has low population loss. From now on, with some abuse of notation, we often write $\ell(h_\theta(x), y)$ as $\ell((x, y), \theta)$ and use the two notations interchangeably. Formally, we define the empirical risk of a model $h$ as:

hypothesis family 내의 function $h_\theta$

$$\underset{\text{empirical risk}}{\widehat{L}(h_\theta)} \triangleq \underbrace{\frac{1}{n}\sum_{i=1}^{n}}_{\text{평균}} \ell(h_\theta(x^{(i)}), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n} \ell((x^{(i)}, y^{(i)}), \theta). \tag{1.2}$$

*Empirical risk minimization* is the method of finding the minimizer of $\widehat{L}$, which we call $\hat{\theta}$:

minimizer:

$$\hat{\theta} \triangleq \operatorname*{argmin}_{\theta \in \Theta} \widehat{L}(h_\theta). \tag{1.3}$$

얘가.. $\mathcal{H}$ 내의 어떤 $h$(function)을 고를지 결정(?) 하니까?

Since we are assuming that our training examples are drawn from the same distribution as the whole population, we know that empirical risk and population risk are equal *in expectation* (over the randomness of the training dataset):

$$\begin{aligned}
\overbrace{\mathop{\mathbb{E}}_{(x^{(i)}, y^{(i)}) \overset{\text{iid}}{\sim} P} \overset{\text{empirical risk}}{\widehat{L}(h_\theta)}}^{\text{의 expectation}} &= \mathop{\mathbb{E}}_{(x^{(i)}, y^{(i)}) \overset{\text{iid}}{\sim} P} \frac{1}{n}\sum_{i=1}^{n} \ell(h_\theta(x^{(i)}), y^{(i)}) \quad \text{*정의 of } \widehat{L}(h_\theta) &(1.4)\\
&= \frac{1}{n}\sum_{i=1}^{n} \mathop{\mathbb{E}}_{(x^{(i)}, y^{(i)}) \overset{\text{iid}}{\sim} P} \ell(h_\theta(x^{(i)}), y^{(i)}) \quad \text{*iid} &(1.5)\\
&= \frac{1}{n} \cdot n \cdot \mathop{\mathbb{E}}_{(x^{(i)}, y^{(i)}) \overset{\text{iid}}{\sim} P} \underbrace{\ell(h_\theta(x^{(i)}), y^{(i)})}_{h_\theta\text{를 사용했을 때의 loss}} &(1.6)\\
&= \underset{\substack{\text{*정의.} \\ (\text{population loss of } h_\theta)}}{L(h_\theta)}. &(1.7)
\end{aligned}$$

1이어도 상관없음..?

의 평균 : $i = 1 \cdots n$까지 다 같음.

즉, empirical risk is an unbiased estim. of population risk!

This is one reason why it makes sense to use empirical risk: it is an unbiased estimator of the population risk.

The key question that we seek to answer in the first part of this course is: **what guarantees do we have on the excess risk for the parameters learned by ERM?** The hope with ERM is that minimizing the training error will lead to small testing error. One way to make this rigorous is by showing that the ERM minimizer's excess risk is bounded.