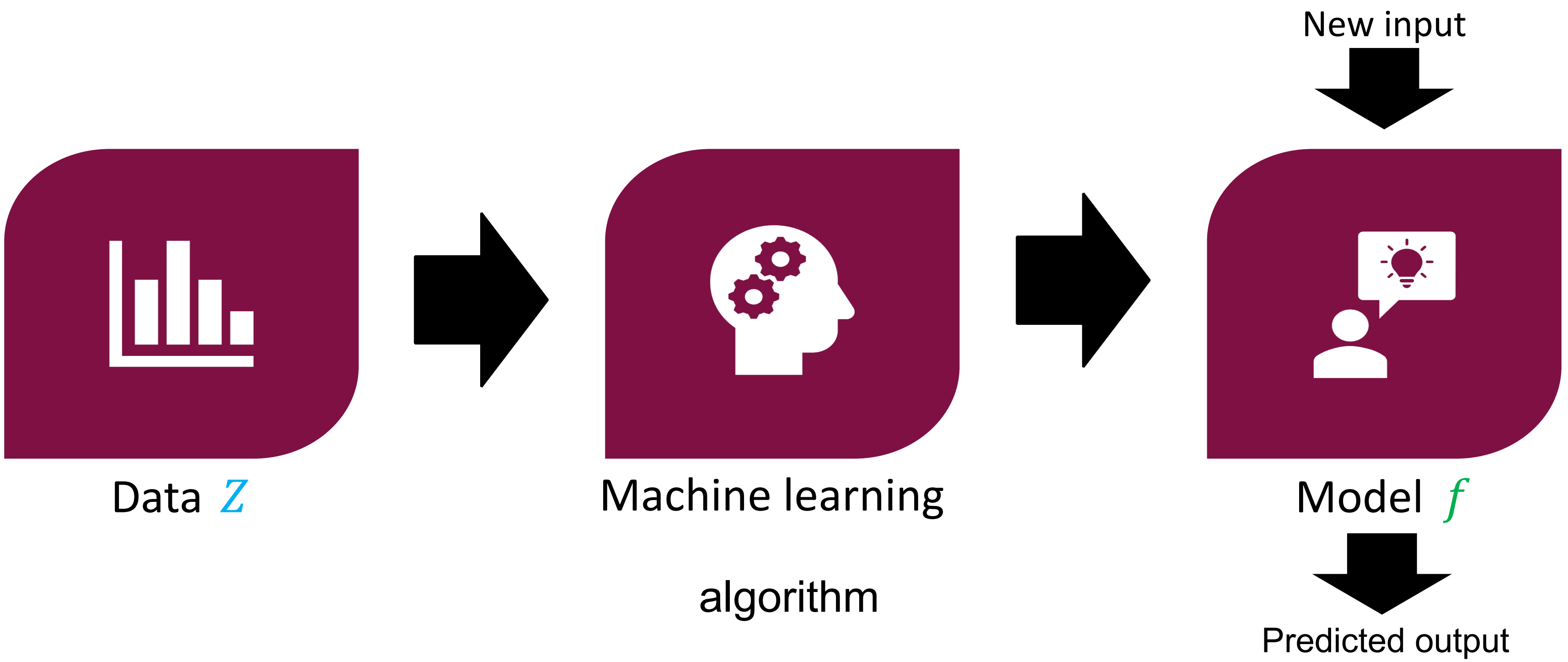
Introduction to Regression Problem

Loss, Risk, Optimal Rule

송 준

고려대학교
통계학과 / 융합데이터과학 대학원

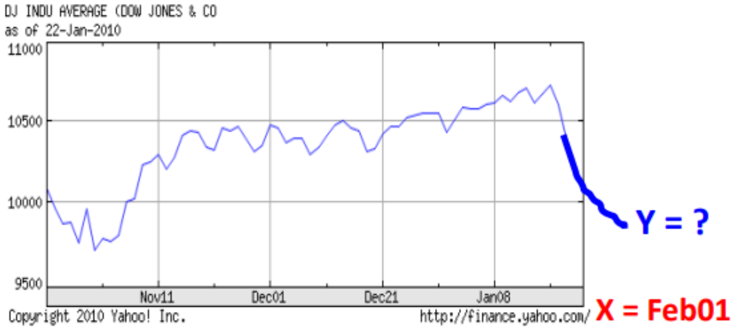
Machine Learning for Prediction



Supervised Learning

Goal: *Construct a predictor* $f: X \mapsto Y$ *that minimizes a risk* $R(f)$, **performance measure**

Sports
Science
News



✓ **Classification** output: a class

$$R(Y,f) = P\big(Y \neq f(X)\big)$$

✓ **Regression** output: a number

$$R(Y,f) = E\left[\big(Y - f(X)\big)^2\right]$$

Performance Measure & Decision Making

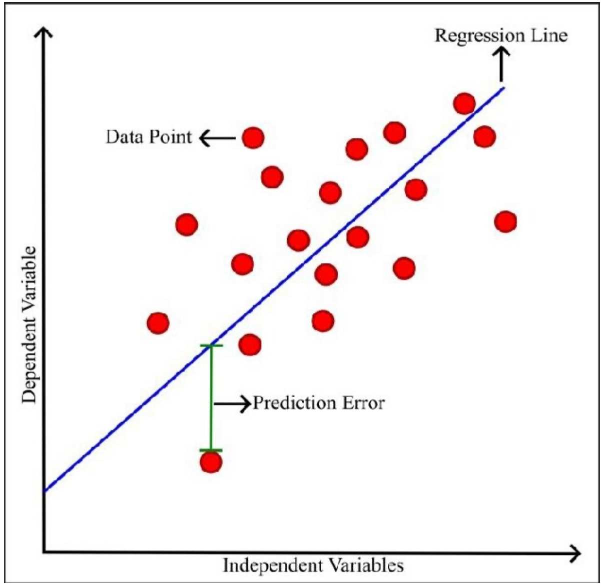
Performance Measures : Loss

Prediction error of the i-th observation(y_i) and the i-th prediction($\hat{y}_i$)

New input

Model f

Predicted output



Performance Measures : Risk

Performance:

- $\boldsymbol{loss(Y, f(X))}$ *: true label Y* 와 *prediction f(X)* 의 **가까운 정도** *(*이를 줄이고자 함*)*
- *We want to perform well on any test data :* $(\boldsymbol{X}, \boldsymbol{Y}) \sim \boldsymbol{P_{XY}}$
- *Given an X drawn randomly from a distribution, how well does the predictor perform on average?*

$$\boxed{\textbf{Risk}\ \ \mathbf{R(f)} \equiv \mathbb{E}_{\mathbf{XY}}[\mathbf{loss}(\mathbf{Y}, \mathbf{f(X)})]}$$

Performance Measures

Performance of supervised learning:

$$\textbf{Risk}\ \ \boldsymbol{R(f) \equiv \mathbb{E}_{XY}[loss(Y, f(X))]}$$

	Classification	**Regression**
$loss(Y, f(X))$	$\mathbb{1}_{\{f(X) \neq Y\}}$	$P(f(X) \neq Y)$
Risk $R(f)$	$(f(X) - Y)^2$	$\mathbb{E}[(f(X) - Y)^2]$

Bayes Optimal Rule

Ideal goal:

Construct **prediction rule** $f^* : \mathcal{X} \rightarrow \mathcal{Y}$

$$f^* = argmin_f \mathbb{E}_{XY}[loss\big(Y, f(X)\big)]$$

Bayes optimal rule

Best possible performance:

$$R(f^*) \leq R(f) \text{ for all } f$$

Bayes Risk

But Optimal rule is not computable in practice - depends on unknown P_{XY}

Bayes Optimal Rule

We can't just simply minimize the risk since P_{XY} *unknown!*

Training data (experience) provides a glimpse of P_{XY}

(observed) $\{(\boldsymbol{X_i}, \boldsymbol{Y_i})\}_{\boldsymbol{i=1}}^{\boldsymbol{n}} \sim \boldsymbol{i.i.d}\ \boldsymbol{P_{XY}}$ **(unknown)**

Supervised Learning

Goal of ML :

Improve the performance on some task with experience

Task: *Given* $\mathrm{X} \in \mathcal{X}$, *predict* $\mathrm{Y} \in \mathcal{Y}$

$\equiv$ *Construct prediction rule* $f : \mathcal{X} \to \mathcal{Y}$

Performance: $R(f) \equiv \mathbb{E}_{XY}[loss(Y, f(X))]$, $(X, Y) \sim P_{XY}$

Experience: *Training data* $\{(X_i, Y_i)\}_{i=1}^{n} \sim i.i.d\ P_{XY}$ (unknown)

Performance: Are we done?

Performance of a learning algorithm

Given a data set $D_n = \{(X_i, Y_i)\}_{i=1}^{n}$, *the performance of the algorithm at a random test point (X,Y) is :*

$$\textbf{Risk: } \mathbf{R}(\widehat{\boldsymbol{f}_n}) \doteq \mathbb{E}_{XY}[\boldsymbol{loss}(\boldsymbol{Y}, \widehat{\boldsymbol{f}_n}(\boldsymbol{X}))]$$

This quantity, however, depends on the data set D_n, *and therefore it is random in* D_n.

Often we want to discuss the average performance of the algorithm, and remove the randomness (D_n *) from the performance:*

Expected Risk(Generalization Error)

$$\mathbb{E}_{D_n}[\mathbf{R}(\widehat{\boldsymbol{f}_n})] \doteq \mathbb{E}_{XY}[\boldsymbol{loss}(\boldsymbol{Y}, \widehat{\boldsymbol{f}_n}(\boldsymbol{X}))]$$

Performance: Are we done?

Ideal goal: *Construct prediction rule* $f^* : \mathcal{X} \rightarrow \mathcal{Y}$

$$f^* = argmin_f \mathbb{E}_{XY}[loss(Y, f(X))]$$

Bayes optimal rule

Practical goal: *Given* $\{(X_i, Y_i)\}_{i=1}^n$*, learn prediction rule* $\widehat{f_n} : \mathcal{X} \rightarrow \mathcal{Y}$

$$\text{Often: } \widehat{f_n} = argmin_{f \in F} \frac{1}{n}\sum_{i=1}^n [loss(Y_i, f(X_i))]$$

Empirical Risk minimizer

$$\boxed{\frac{1}{n}\sum_{i=1}^n [loss(Y_i, f(X_i))] \xrightarrow{L.L.N} \mathbb{E}_{XY}[loss(Y, f(X))]}$$

Performance : 현실적 한계

Ideal goal: Construct prediction rule $f^*: \mathcal{X} \rightarrow \mathcal{Y}$

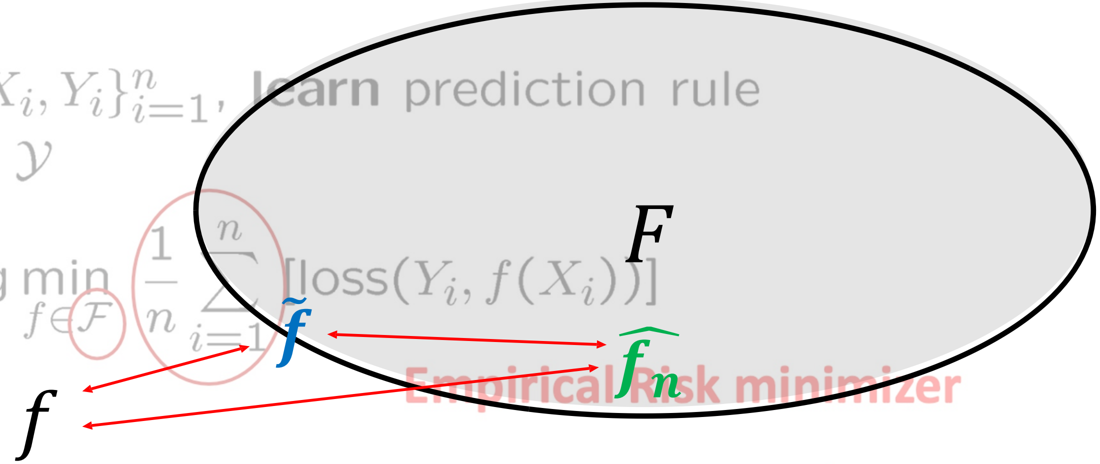
$$f^* = \arg\min_{f} \mathbb{E}_{XY}\left[\text{loss}(Y, f(X))\right]$$

Bayes optimal rule

- ✓ 우리가 현실적으로 고려할 수 있는(계산할 수 있는) 함수공간 F
- ✓ space for 선형함수, B-spline basis 함수, Neural Network (딥러닝 모형) 등

Practical goal: Given $\{X_i, Y_i\}_{i=1}^n$, learn prediction rule $\widehat{f}_n: \mathcal{X} \rightarrow \mathcal{Y}$

Often: $\widehat{f}_n = \arg\min_{f \in \mathcal{F}} \frac{1}{n}\sum_{i=1}^{n}\left[\text{loss}(Y_i, f(X_i))\right]$



Empirical Risk minimizer

$$\frac{1}{n}\sum_{i=1}^{n}\left[\text{loss}(Y_i, f(X_i))\right] \xrightarrow{\text{Law of Large Numbers}} \mathbb{E}_{XY}\left[\text{loss}(Y, f(X))\right]$$

Introduction to Linear Regression

✓ **Data Assumption:**

- $(x_1, y_1), \dots, (x_n, y_n), x_i \in R^p,\ \ y_i \in R$
- (x_i, y_i): *a realization of* $(X_i, Y_i) \sim i.i.d.(X, Y)$

✓ **Model Assumption:** *X*와 *Y*는 **선형관계**를 가짐.

$$\boldsymbol{Y = f(X) + \epsilon}$$

✓ ***f*** **의 형태제약:**

$$\mathrm{F} = \{f: f(x) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p, for\ some\ \beta_0, \beta_1, \dots, \beta_p\}$$

*1*차 목표: $\boldsymbol{\beta_0, \beta} = \left(\boldsymbol{\beta_1}, \dots, \boldsymbol{\beta_p}\right)^{\boldsymbol{T}}$ 찾기,