Statistical Learning

Master Spécialisé Intelligence Artificielle de Confiance (IAC)
@ Centrale Supélec en partenariat avec l'IRT SystemX
2024/2025.

Faïcel Chamroukhi

chamroukhi.com

Objectives

The objective of this lecture is to understand :

- The foundational principles of decision-making in machine learning, including from a probabilistic perspective.
- The different errors and risk measures associated with a machine learning problem.
- Their optimal formulations and key decompositions, including the bias-variance decomposition.
- The intuitions behind standard decision rules.
- Practical applications showcased through selected machine learning algorithms.

Outline

- Supervised Learning
- Prediction function
- Loss function
- Risk function
- Bayes Risk

Supervised Learning

- The data are represented by a random pair $(\boldsymbol{X}, Y) \in \mathcal{X} \times \mathcal{Y}$ where $\boldsymbol{X}$ is a vector of descriptors for some variable of interest Y
- The objective is **Prediction**, i.e. to seek for a prediction function $h : \mathcal{X} \to \mathcal{Y}$ for which $\widehat{y} = h(\boldsymbol{x})$ is a good approximation of the true output y
- Problems : typically $\boldsymbol{X}_i \in \mathbb{R}^p$, $Y \in \mathcal{Y} = \mathbb{R}^d$ for **regression** and $Y \in \mathcal{Y} = \{0, 1\}, \{-1, +1\}$ or $\{1, \cdots, K\}$ for **classification**

$\hookrightarrow$ We will mainly focus on parametric probabilistic models of the form

$$Y = h(X) + \epsilon, \epsilon \sim p_\theta$$

with the conditional distr. $P(Y|X, h)$ can be computed in terms of $P_\theta(Y - h(X))$.

- Data : a random sample $(\boldsymbol{X}_i, Y_i)_{i=1}^n$ with observed values $\mathcal{D}_n = (\boldsymbol{x}_i, y_i)_{i=1}^n$
- **Data-Scientist's role :** given the **data**, choose a **prediction function** h from a class $\mathcal{H}$ that attempts to "minimize" the prediction error for of all possible data (**risk**) $R(h)$, under a **loss** function ℓ measuring the error of predicting Y by $h(X)$.
  - $\hookrightarrow$ minimize the **empirical risk** (data-$\mathcal{D}_n$-driven) $R_n(h)$
  - $\hookrightarrow$ Minimizing $R_n(h)$ always requires an optimization **algorithm** $\mathcal{A}$
- Data-Scientist's "**Toolbox**" **:** {Data, loss, hypothesis, algorithm}

Supervised Learning

- The data are represented by a random pair $(\boldsymbol{X}, Y) \in \mathcal{X} \times \mathcal{Y}$ where $\boldsymbol{X}$ is a vector of descriptors for some variable of interest Y
- The objective is **Prediction**, i.e. to seek for a prediction function $h : \mathcal{X} \rightarrow \mathcal{Y}$ for which $\widehat{y} = h(\boldsymbol{x})$ is a good approximation of the true output y
- Problems : typically $\boldsymbol{X}_i \in \mathbb{R}^p$, $Y \in \mathcal{Y} = \mathbb{R}^d$ for **regression** and $Y \in \mathcal{Y} = \{0, 1\}, \{-1, +1\}$ or $\{1, \cdots, K\}$ for **classification**

↪ We will mainly focus on parametric probabilistic models of the form

$$Y = h(X) + \epsilon, \epsilon \sim p_\theta$$

with the conditional distr. $P(Y|X, h)$ can be computed in terms of $P_\theta(Y - h(X))$.

- Data : a random sample $(\boldsymbol{X}_i, Y_i)_{i=1}^n$ with observed values $\mathcal{D}_n = (\boldsymbol{x}_i, y_i)_{i=1}^n$
- **Data-Scientist's role :** given the **data**, choose a **prediction function** h from a class $\mathcal{H}$ that attempts to "minimize" the prediction error for of all possible data (**risk**) $R(h)$, under a **loss** function ℓ measuring the error of predicting Y by $h(X)$.
  - ↪ minimize the **empirical risk** (data-$\mathcal{D}_n$-driven) $R_n(h)$
  - ↪ Minimizing $R_n(h)$ always requires an optimization **algorithm** $\mathcal{A}$
- Data-Scientist's "**Toolbox**" **:** {Data, loss, hypothesis, algorithm}

Supervised Learning

- The data are represented by a random pair $(\boldsymbol{X}, Y) \in \mathcal{X} \times \mathcal{Y}$ where $\boldsymbol{X}$ is a vector of descriptors for some variable of interest Y
- The objective is **Prediction**, i.e. to seek for a prediction function $h : \mathcal{X} \to \mathcal{Y}$ for which $\widehat{y} = h(\boldsymbol{x})$ is a good approximation of the true output y
- Problems : typically $\boldsymbol{X}_i \in \mathbb{R}^p$, $Y \in \mathcal{Y} = \mathbb{R}^d$ for **regression** and $Y \in \mathcal{Y} = \{0, 1\}, \{-1, +1\}$ or $\{1, \cdots, K\}$ for **classification**

↪ We will mainly focus on parametric probabilistic models of the form

$$Y = h(X) + \epsilon, \epsilon \sim p_\theta$$

with the conditional distr. $P(Y|X, h)$ can be computed in terms of $P_\theta(Y - h(X))$.

- Data : a random sample $(\boldsymbol{X}_i, Y_i)_{i=1}^n$ with observed values $\mathcal{D}_n = (\boldsymbol{x}_i, y_i)_{i=1}^n$
- **Data-Scientist's role :** given the **data**, choose a **prediction function** h from a class $\mathcal{H}$ that attempts to "minimize" the prediction error for of all possible data (**risk**) $R(h)$, under a **loss** function ℓ measuring the error of predicting Y by $h(X)$.
  - ↪ minimize the **empirical risk** (data-$\mathcal{D}_n$-driven) $R_n(h)$
  - ↪ Minimizing $R_n(h)$ always requires an optimization **algorithm** $\mathcal{A}$
- Data-Scientist's "**Toolbox**" **:** {Data, loss, hypothesis, algorithm}

Supervised Learning

- The data are represented by a random pair $(\boldsymbol{X}, Y) \in \mathcal{X} \times \mathcal{Y}$ where $\boldsymbol{X}$ is a vector of descriptors for some variable of interest Y
- The objective is **Prediction**, i.e. to seek for a prediction function $h : \mathcal{X} \rightarrow \mathcal{Y}$ for which $\widehat{y} = h(\boldsymbol{x})$ is a good approximation of the true output y
- Problems : typically $\boldsymbol{X}_i \in \mathbb{R}^p$, $Y \in \mathcal{Y} = \mathbb{R}^d$ for **regression** and $Y \in \mathcal{Y} = \{0, 1\}, \{-1, +1\}$ or $\{1, \cdots, K\}$ for **classification**

↪ We will mainly focus on parametric probabilistic models of the form

$$Y = h(X) + \epsilon, \epsilon \sim p_\theta$$

with the conditional distr. $P(Y|X, h)$ can be computed in terms of $P_\theta(Y - h(X))$.

- Data : a random sample $(\boldsymbol{X}_i, Y_i)_{i=1}^n$ with observed values $\mathcal{D}_n = (\boldsymbol{x}_i, y_i)_{i=1}^n$
- **Data-Scientist's role :** given the **data**, choose a **prediction function** h from a class $\mathcal{H}$ that attempts to "minimize" the prediction error for of all possible data (**risk**) $R(h)$, under a **loss** function ℓ measuring the error of predicting Y by $h(X)$.

  ↪ minimize the **empirical risk** (data-$\mathcal{D}_n$-driven) $R_n(h)$

  ↪ Minimizing $R_n(h)$ always requires an optimization **algorithm** $\mathcal{A}$
- Data-Scientist's "**Toolbox**" **:** {Data, loss, hypothesis, algorithm}

Supervised Learning

- The data are represented by a random pair $(\boldsymbol{X}, Y) \in \mathcal{X} \times \mathcal{Y}$ where $\boldsymbol{X}$ is a vector of descriptors for some variable of interest Y
- The objective is **Prediction**, i.e. to seek for a prediction function $h : \mathcal{X} \rightarrow \mathcal{Y}$ for which $\widehat{y} = h(\boldsymbol{x})$ is a good approximation of the true output y
- Problems : typically $\boldsymbol{X}_i \in \mathbb{R}^p$, $Y \in \mathcal{Y} = \mathbb{R}^d$ for **regression** and $Y \in \mathcal{Y} = \{0, 1\}, \{-1, +1\}$ or $\{1, \cdots, K\}$ for **classification**

$\hookrightarrow$ We will mainly focus on parametric probabilistic models of the form

$$Y = h(X) + \epsilon, \epsilon \sim p_\theta$$

with the conditional distr. $P(Y|X, h)$ can be computed in terms of $P_\theta(Y - h(X))$.

- Data : a random sample $(\boldsymbol{X}_i, Y_i)_{i=1}^n$ with observed values $\mathcal{D}_n = (\boldsymbol{x}_i, y_i)_{i=1}^n$
- **Data-Scientist's role :** given the **data**, choose a **prediction function** h from a class $\mathcal{H}$ that attempts to "minimize" the prediction error for of all possible data (**risk**) $R(h)$, under a **loss** function ℓ measuring the error of predicting Y by $h(X)$.
  - $\hookrightarrow$ minimize the **empirical risk** (data-$\mathcal{D}_n$-driven) $R_n(h)$
  - $\hookrightarrow$ Minimizing $R_n(h)$ always requires an optimization **algorithm** $\mathcal{A}$
- Data-Scientist's "**Toolbox" :** {Data, loss, hypothesis, algorithm}

Supervised Learning

- The data are represented by a random pair $(\boldsymbol{X}, Y) \in \mathcal{X} \times \mathcal{Y}$ where $\boldsymbol{X}$ is a vector of descriptors for some variable of interest Y
- The objective is **Prediction**, i.e. to seek for a prediction function $h : \mathcal{X} \to \mathcal{Y}$ for which $\widehat{y} = h(\boldsymbol{x})$ is a good approximation of the true output y
- Problems : typically $\boldsymbol{X}_i \in \mathbb{R}^p$, $Y \in \mathcal{Y} = \mathbb{R}^d$ for **regression** and $Y \in \mathcal{Y} = \{0, 1\}, \{-1, +1\}$ or $\{1, \cdots, K\}$ for **classification**

↪ We will mainly focus on parametric probabilistic models of the form

$$Y = h(X) + \epsilon, \epsilon \sim p_\theta$$

with the conditional distr. $P(Y|X, h)$ can be computed in terms of $P_\theta(Y - h(X))$.

- Data : a random sample $(\boldsymbol{X}_i, Y_i)_{i=1}^n$ with observed values $\mathcal{D}_n = (\boldsymbol{x}_i, y_i)_{i=1}^n$
- **Data-Scientist's role :** given the **data**, choose a **prediction function** h from a class $\mathcal{H}$ that attempts to "minimize" the prediction error for of all possible data (**risk**) $R(h)$, under a **loss** function ℓ measuring the error of predicting Y by $h(X)$.
  - ↪ minimize the **empirical risk** (data-$\mathcal{D}_n$-driven) $R_n(h)$
  - ↪ Minimizing $R_n(h)$ always requires an optimization **algorithm** $\mathcal{A}$
- Data-Scientist's "**Toolbox**" **:** {Data, loss, hypothesis, algorithm}

Supervised Learning

- The data are represented by a random pair $(\boldsymbol{X}, Y) \in \mathcal{X} \times \mathcal{Y}$ where $\boldsymbol{X}$ is a vector of descriptors for some variable of interest Y
- The objective is **Prediction**, i.e. to seek for a prediction function $h : \mathcal{X} \to \mathcal{Y}$ for which $\widehat{y} = h(\boldsymbol{x})$ is a good approximation of the true output y
- Problems : typically $\boldsymbol{X}_i \in \mathbb{R}^p$, $Y \in \mathcal{Y} = \mathbb{R}^d$ for **regression** and $Y \in \mathcal{Y} = \{0, 1\}, \{-1, +1\}$ or $\{1, \cdots, K\}$ for **classification**

$\hookrightarrow$ We will mainly focus on parametric probabilistic models of the form

$$Y = h(X) + \epsilon, \epsilon \sim p_\theta$$

with the conditional distr. $P(Y|X, h)$ can be computed in terms of $P_\theta(Y - h(X))$.

- Data : a random sample $(\boldsymbol{X}_i, Y_i)_{i=1}^n$ with observed values $\mathcal{D}_n = (\boldsymbol{x}_i, y_i)_{i=1}^n$
- **Data-Scientist's role :** given the **data**, choose a **prediction function** h from a class $\mathcal{H}$ that attempts to "minimize" the prediction error for of all possible data (**risk**) $R(h)$, under a **loss** function ℓ measuring the error of predicting Y by $h(X)$.

  $\hookrightarrow$ minimize the **empirical risk** (data-$\mathcal{D}_n$-driven) $R_n(h)$

  $\hookrightarrow$ Minimizing $R_n(h)$ always requires an optimization **algorithm** $\mathcal{A}$
- Data-Scientist's "**Toolbox**" **:** {Data, loss, hypothesis, algorithm}

Supervised Learning

- The data are represented by a random pair $(\boldsymbol{X}, Y) \in \mathcal{X} \times \mathcal{Y}$ where $\boldsymbol{X}$ is a vector of descriptors for some variable of interest Y
- The objective is **Prediction**, i.e. to seek for a prediction function $h : \mathcal{X} \to \mathcal{Y}$ for which $\widehat{y} = h(\boldsymbol{x})$ is a good approximation of the true output y
- Problems : typically $\boldsymbol{X}_i \in \mathbb{R}^p$, $Y \in \mathcal{Y} = \mathbb{R}^d$ for **regression** and $Y \in \mathcal{Y} = \{0, 1\}, \{-1, +1\}$ or $\{1, \cdots, K\}$ for **classification**

↪ We will mainly focus on parametric probabilistic models of the form

$$Y = h(X) + \epsilon, \epsilon \sim p_\theta$$

with the conditional distr. $P(Y|X, h)$ can be computed in terms of $P_\theta(Y - h(X))$.

- Data : a random sample $(\boldsymbol{X}_i, Y_i)_{i=1}^n$ with observed values $\mathcal{D}_n = (\boldsymbol{x}_i, y_i)_{i=1}^n$
- **Data-Scientist's role :** given the **data**, choose a **prediction function** h from a class $\mathcal{H}$ that attempts to "minimize" the prediction error for of all possible data (**risk**) $R(h)$, under a **loss** function ℓ measuring the error of predicting Y by $h(X)$.

  ↪ minimize the **empirical risk** (data-$\mathcal{D}_n$-driven) $R_n(h)$

  ↪ Minimizing $R_n(h)$ always requires an optimization **algorithm** $\mathcal{A}$
- Data-Scientist's "**Toolbox**" **:** {Data, loss, hypothesis, algorithm}

Supervised Learning

- The data are represented by a random pair $(\boldsymbol{X}, Y) \in \mathcal{X} \times \mathcal{Y}$ where $\boldsymbol{X}$ is a vector of descriptors for some variable of interest Y
- The objective is **Prediction**, i.e. to seek for a prediction function $h : \mathcal{X} \to \mathcal{Y}$ for which $\widehat{y} = h(\boldsymbol{x})$ is a good approximation of the true output y
- Problems : typically $\boldsymbol{X}_i \in \mathbb{R}^p$, $Y \in \mathcal{Y} = \mathbb{R}^d$ for **regression** and $Y \in \mathcal{Y} = \{0, 1\}, \{-1, +1\}$ or $\{1, \cdots, K\}$ for **classification**

↪ We will mainly focus on parametric probabilistic models of the form

$$Y = h(X) + \epsilon, \epsilon \sim p_\theta$$

with the conditional distr. $P(Y|X, h)$ can be computed in terms of $P_\theta(Y - h(X))$.

- Data : a random sample $(\boldsymbol{X}_i, Y_i)_{i=1}^n$ with observed values $\mathcal{D}_n = (\boldsymbol{x}_i, y_i)_{i=1}^n$
- **Data-Scientist's role :** given the **data**, choose a **prediction function** h from a class $\mathcal{H}$ that attempts to "minimize" the prediction error for of all possible data (**risk**) $R(h)$, under a **loss** function ℓ measuring the error of predicting Y by $h(X)$.
  - ↪ minimize the **empirical risk** (data-$\mathcal{D}_n$-driven) $R_n(h)$
  - ↪ Minimizing $R_n(h)$ always requires an optimization **algorithm** $\mathcal{A}$
- Data-Scientist's "**Toolbox" :** {Data, loss, hypothesis, algorithm}

Prediction/decision function

Def. Prediction function

$$h \colon \mathcal{X} \to \mathcal{Y}$$
$$x \mapsto h(x)$$

is a decision/prediction function, parametric or not, linear or not, ...

Example : Linear prediction functions

$$h \colon \mathbb{R}^p \to \mathbb{R}$$
$$x \mapsto \langle x, \theta \rangle = \theta^T x$$

The **predicted** values of Y_i's for new covariates $X_i = x_i$s correspond to

$$\widehat{y}_i = h(x_i)$$

Example : Linear prediction functions (cont.) : $\widehat{y}_i = \langle x_i, \theta \rangle = \theta^T x_i$

Q : How good we are in prediction on a particular pair (x, y) ?

Prediction/decision function

Def. Prediction function

$$\begin{aligned} h\colon \mathcal{X} &\to \mathcal{Y} \\ x &\mapsto h(x) \end{aligned}$$

is a decision/prediction function, parametric or not, linear or not, ...

Example : Linear prediction functions

$$\begin{aligned} h\colon \mathbb{R}^p &\to \mathbb{R} \\ x &\mapsto \langle x, \theta \rangle = \theta^T x \end{aligned}$$

The **predicted** values of Y_i's for new covariates $X_i = x_i$s correspond to

$$\widehat{y}_i = h(x_i)$$

Example : Linear prediction functions (cont.) : $\widehat{y}_i = \langle x_i, \theta \rangle = \theta^T x_i$

Q : How good we are in prediction on a particular pair (x, y) ?

Prediction/decision function

Def. Prediction function

$$h\colon \mathcal{X} \to \mathcal{Y}$$
$$x \mapsto h(x)$$

is a decision/prediction function, parametric or not, linear or not, ...

Example : Linear prediction functions

$$h\colon \mathbb{R}^p \to \mathbb{R}$$
$$x \mapsto \langle x, \theta \rangle = \theta^T x$$

The **predicted** values of Y_i's for new covariates $X_i = x_i$s correspond to

$$\widehat{y}_i = h(x_i)$$

Example : Linear prediction functions (cont.) : $\widehat{y}_i = \langle x_i, \theta \rangle = \theta^T x_i$

Q : How good we are in prediction on a particular pair (x, y) ?

Prediction/decision function

Def. Prediction function

$$h\colon \mathcal{X} \to \mathcal{Y}$$
$$x \mapsto h(x)$$

is a decision/prediction function, parametric or not, linear or not, ...

Example : Linear prediction functions

$$h\colon \mathbb{R}^p \to \mathbb{R}$$
$$x \mapsto \langle x, \theta \rangle = \theta^T x$$

The **predicted** values of Y_i's for new covariates $X_i = x_i$s correspond to

$$\widehat{y}_i = h(x_i)$$

Example : Linear prediction functions (cont.) : $\widehat{y}_i = \langle x_i, \theta \rangle = \theta^T x_i$

Q : How good we are in prediction on a particular pair (x, y) ?

Loss

Def. Loss function

$$\ell \colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$$
$$(y, h(x)) \mapsto \ell(y, h(x))$$

It measures how good we are on a particular pair (x, y).
(We assume that the distribution of the test data is the same as that of the training.)

Examples of loss functions

- Square (ℓ_2)-loss : $\ell(y, h(x)) = (y - h(x))^2$
- Absolute (ℓ_1)-loss : $\ell(y, h(x)) = |y - h(x)|$
- Huber loss : $\ell_\delta(y, h(x)) = \begin{cases} \frac{1}{2}(y - h(x))^2 \text{ if } |y - h(x)| \leq \delta, \\ \delta\left(|y - h(x)| - \frac{1}{2}\delta\right), \text{otherwise.} \end{cases}$
- logarithmic loss : $\ell(y, h_\theta(x)) = -\log(p_\theta(x, y))$
- "0-1" loss : $\ell(y, h(x)) = \mathbb{1}_{h(x) \neq y}$

Denoting $\ell(y, h(x)) = \phi(yh(x))$ and $u = yh(x)$

- Hinge loss $\phi_{\text{hinge}}(u) = (1 - u)_+$
- Logistic loss $\phi_{\text{logistic}}(u) = \log(1 + \exp(-u))$
- Square loss $\phi_{\text{square}}(u) = (1 - u)^2$
- Exponential loss $\phi_{\exp}(u) = \exp(-u)$

Loss

Def. Loss function

$$\ell\colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$$
$$(y, h(x)) \mapsto \ell(y, h(x))$$

It measures how good we are on a particular pair (x, y).
(We assume that the distribution of the test data is the same as that of the training.)

Examples of loss functions

- Square (ℓ_2)-loss : $\ell(y, h(x)) = (y - h(x))^2$
- Absolute (ℓ_1)-loss : $\ell(y, h(x)) = |y - h(x)|$
- Huber loss : $\ell_\delta(y, h(x)) = \begin{cases} \frac{1}{2}(y - h(x))^2 \text{ if } |y - h(x)| \leq \delta, \\ \delta\left(|y - h(x)| - \frac{1}{2}\delta\right), \text{otherwise.} \end{cases}$
- logarithmic loss : $\ell(y, h_\theta(x)) = -\log(p_\theta(x, y))$
- "0-1" loss : $\ell(y, h(x)) = \mathbb{1}_{h(x) \neq y}$

  Denoting $\ell(y, h(x)) = \phi(yh(x))$ and $u = yh(x)$

- Hinge loss $\phi_{\text{hinge}}(u) = (1 - u)_+$
- Logistic loss $\phi_{\text{logistic}}(u) = \log(1 + \exp(-u))$
- Square loss $\phi_{\text{square}}(u) = (1 - u)^2$
- Exponential loss $\phi_{\exp}(u) = \exp(-u)$

Loss

Def. Loss function

$$\ell\colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$$
$$(y, h(x)) \mapsto \ell(y, h(x))$$

It measures how good we are on a particular pair (x, y).
(We assume that the distribution of the test data is the same as that of the training.)

Examples of loss functions

- Square (ℓ_2)-loss :
  $\ell(y, h(x)) = (y - h(x))^2$
- Absolute (ℓ_1)-loss :
  $\ell(y, h(x)) = |y - h(x)|$
- Huber loss : $\ell_\delta(y, h(x)) = \begin{cases} \frac{1}{2}(y - h(x))^2 \text{ if } |y - h(x)| \leq \delta, \\ \delta\left(|y - h(x)| - \frac{1}{2}\delta\right), \text{otherwise.} \end{cases}$
- logarithmic loss :
  $\ell(y, h_\theta(x)) = -\log(p_\theta(x, y))$

- "0-1" loss : $\ell(y, h(x)) = \mathbb{1}_{h(x) \neq y}$

  Denoting $\ell(y, h(x)) = \phi(yh(x))$ and $u = yh(x)$
- Hinge loss $\phi_{\text{hinge}}(u) = (1 - u)_+$
- Logistic loss
  $\phi_{\text{logistic}}(u) = \log(1 + \exp(-u))$
- Square loss $\phi_{\text{square}}(u) = (1 - u)^2$
- Exponential loss $\phi_{\exp}(u) = \exp(-u)$

Loss

Def. Loss function

$$\ell \colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$$
$$(y, h(x)) \mapsto \ell(y, h(x))$$

It measures how good we are on a particular pair (x, y).
(We assume that the distribution of the test data is the same as that of the training.)

Examples of loss functions

- Square (ℓ_2)-loss :
  $\ell(y, h(x)) = (y - h(x))^2$
- Absolute (ℓ_1)-loss :
  $\ell(y, h(x)) = |y - h(x)|$
- Huber loss : $\ell_\delta(y, h(x)) = \begin{cases} \frac{1}{2}(y - h(x))^2 \text{ if } |y - h(x)| \leq \delta, \\ \delta\left(|y - h(x)| - \frac{1}{2}\delta\right), \text{otherwise.} \end{cases}$
- logarithmic loss :
  $\ell(y, h_\theta(x)) = -\log(p_\theta(x, y))$
- "0-1" loss : $\ell(y, h(x)) = \mathbb{1}_{h(x) \neq y}$

  Denoting $\ell(y, h(x)) = \phi(yh(x))$ and $u = yh(x)$
- Hinge loss $\phi_{\text{hinge}}(u) = (1 - u)_+$
- Logistic loss
  $\phi_{\text{logistic}}(u) = \log(1 + \exp(-u))$
- Square loss $\phi_{\text{square}}(u) = (1 - u)^2$
- Exponential loss $\phi_{\exp}(u) = \exp(-u)$

Loss

Def. Loss function

$$\ell\colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$$
$$(y, h(x)) \mapsto \ell(y, h(x))$$

It measures how good we are on a particular pair (x, y).
(We assume that the distribution of the test data is the same as that of the training.)

Examples of loss functions

- Square (ℓ_2)-loss : $\ell(y, h(x)) = (y - h(x))^2$
- Absolute (ℓ_1)-loss : $\ell(y, h(x)) = |y - h(x)|$
- Huber loss : $\ell_\delta(y, h(x)) = \begin{cases} \frac{1}{2}(y - h(x))^2 \text{ if } |y - h(x)| \leq \delta, \\ \delta\left(|y - h(x)| - \frac{1}{2}\delta\right), \text{otherwise.} \end{cases}$
- logarithmic loss : $\ell(y, h_\theta(x)) = -\log(p_\theta(x, y))$
- "0-1" loss : $\ell(y, h(x)) = \mathbb{1}_{h(x) \neq y}$

Denoting $\ell(y, h(x)) = \phi(yh(x))$ and $u = yh(x)$

- Hinge loss $\phi_{\text{hinge}}(u) = (1 - u)_+$
- Logistic loss $\phi_{\text{logistic}}(u) = \log(1 + \exp(-u))$
- Square loss $\phi_{\text{square}}(u) = (1 - u)^2$
- Exponential loss $\phi_{\exp}(u) = \exp(-u)$

Loss

Def. Loss function

$$\ell \colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$$
$$(y, h(x)) \mapsto \ell(y, h(x))$$

It measures how good we are on a particular pair (x, y).
(We assume that the distribution of the test data is the same as that of the training.)

Examples of loss functions

- Square (ℓ_2)-loss :
  $\ell(y, h(x)) = (y - h(x))^2$
- Absolute (ℓ_1)-loss :
  $\ell(y, h(x)) = |y - h(x)|$
- Huber loss : $\ell_\delta(y, h(x)) =$
  $\begin{cases} \frac{1}{2}(y - h(x))^2 & \text{if } |y - h(x)| \leq \delta, \\ \delta\left(|y - h(x)| - \frac{1}{2}\delta\right), & \text{otherwise.} \end{cases}$
- logarithmic loss :
  $\ell(y, h_\theta(x)) = -\log(p_\theta(x, y))$
- "0-1" loss : $\ell(y, h(x)) = \mathbb{1}_{h(x) \neq y}$

  Denoting $\ell(y, h(x)) = \phi(yh(x))$ and $u = yh(x)$
- Hinge loss $\phi_{\text{hinge}}(u) = (1 - u)_+$
- Logistic loss
  $\phi_{\text{logistic}}(u) = \log(1 + \exp(-u))$
- Square loss $\phi_{\text{square}}(u) = (1 - u)^2$
- Exponential loss $\phi_{\exp}(u) = \exp(-u)$

Examples of loss functions

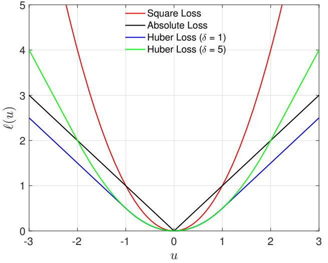

FIGURE – Some loss functions in regression. (curve of $\ell(u)$ for $u = y - h(x)$; $y \in \mathbb{R}$)

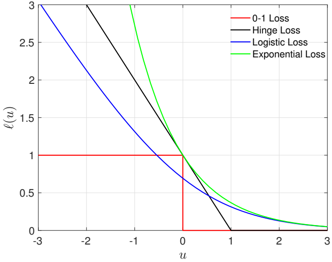

FIGURE – Some loss functions in classification. (curve of $\ell(u)$ for $u = yh(x)$ and $y \in \{-1, +1\}$)

Examples of loss functions in machine learning

- **Squared (ℓ_2)-loss** :

$$\ell(y, h(x)) = (y - h(x))^2$$

used in Ordinary Least Squares (OLS) Also regression with Gaussian noise

- **Absolute (ℓ_1)-loss** : $\ell(y, h(x)) = |y - h(x)|$ used in least absolute deviation (LAD) (Robust) regression (idem Regression with Laplace noise), and in some settings for Lasso regression (for sparsity).

- **Huber loss** : $\ell_\delta(y, h(x)) = \begin{cases} \frac{1}{2}(y - h(x))^2, & |y - h(x)| \leq \delta \\ \delta(|y - h(x)| - \frac{1}{2}\delta), & \text{otherwise} \end{cases}$ used in Robust regression (to mitigate the effect of outliers.).

- **Logarithmic loss** : $\ell(y, h_\theta(x)) = -\log(p_\theta(x, y))$ used in Logistic regression and in many maximum-likelihood estimation problems

- **Hinge loss** :

$$\phi_{\text{hinge}}(u) = (1 - u)_+$$

used in Support Vector Machines

- **Logistic loss** :

$$\phi_{\text{logistic}}(u) = \log(1 + \exp(-u))$$

used in Logistic regression

- **0-1 loss** : $\ell(y, h(x)) = \mathbb{1}_{h(x) \neq y}$ used in theoretical analysis of classifiers (not differentiable) like Bayes

Examples of loss functions in machine learning

- **Squared (ℓ_2)-loss** :

$$\ell(y, h(x)) = (y - h(x))^2$$

used in Ordinary Least Squares (OLS) Also regression with Gaussian noise

- **Absolute (ℓ_1)-loss** : $\ell(y, h(x)) = |y - h(x)|$ used in least absolute deviation (LAD) (Robust) regression (idem Regression with Laplace noise), and in some settings for Lasso regression (for sparsity).

- **Huber loss** : $\ell_\delta(y, h(x)) = \begin{cases} \frac{1}{2}(y - h(x))^2, & |y - h(x)| \leq \delta \\ \delta(|y - h(x)| - \frac{1}{2}\delta), & \text{otherwise} \end{cases}$ used in Robust regression (to mitigate the effect of outliers.).

- **Logarithmic loss** : $\ell(y, h_\theta(x)) = -\log(p_\theta(x, y))$ used in Logistic regression and in many maximum-likelihood estimation problems

- **Hinge loss** :

$$\phi_{\text{hinge}}(u) = (1 - u)_+$$

used in Support Vector Machines

- **Logistic loss** :

$$\phi_{\text{logistic}}(u) = \log(1 + \exp(-u))$$

used in Logistic regression

- **0-1 loss** : $\ell(y, h(x)) = \mathbb{1}_{h(x) \neq y}$ used in theoretical analysis of classifiers (not differentiable) like Bayes

- **Squared (ℓ_2)-loss** :

$$\ell(y, h(x)) = (y - h(x))^2$$

used in Ordinary Least Squares (OLS) Also regression with Gaussian noise

- **Absolute (ℓ_1)-loss** : $\ell(y, h(x)) = |y - h(x)|$ used in least absolute deviation (LAD) (Robust) regression (idem Regression with Laplace noise), and in some settings for Lasso regression (for sparsity).

- **Huber loss** : $\ell_\delta(y, h(x)) = \begin{cases} \frac{1}{2}(y - h(x))^2, & |y - h(x)| \le \delta \\ \delta(|y - h(x)| - \frac{1}{2}\delta), & \text{otherwise} \end{cases}$ used in Robust regression (to mitigate the effect of outliers.).

- **Logarithmic loss** : $\ell(y, h_\theta(x)) = -\log(p_\theta(x, y))$ used in Logistic regression and in many maximum-likelihood estimation problems

- **Hinge loss** :

$$\phi_{\text{hinge}}(u) = (1 - u)_+$$

used in Support Vector Machines

- **Logistic loss** :

$$\phi_{\text{logistic}}(u) = \log(1 + \exp(-u))$$

used in Logistic regression

- **0-1 loss** : $\ell(y, h(x)) = \mathbb{1}_{h(x) \neq y}$ used in theoretical analysis of classifiers (not differentiable) like Bayes

Risk

- **Risk** : Given the pair (X, Y) with (unknown) joint distribution P, the error of approximating Y by $h(X)$ is measured by a chosen loss function $\ell(Y, h(X))$. Then, the *Risk* associated to model/hypothesis h under loss l is the *Expected loss* :

$$R(h) = \mathbb{E}_P[\ell(Y, h(X))] = \int_{\mathcal{X}\times\mathcal{Y}} \ell(y, h(x)) dP(x, y).$$

  $\hookrightarrow$ prediction error that measures the generalization performance of h.

- **Risk Examples**
  - Under the "0-1"-loss $\ell(y, h(x)) = \mathbb{1}_{h(x)\neq y}$:

$$R(h) = \int_{\mathcal{X}\times\mathcal{Y}} \mathbb{1}_{h(x)\neq y} dP(x, y) = \mathbb{E}_P[\mathbb{1}_{h(x)\neq y}] = \mathbb{P}(h(X) \neq Y).$$

    $\hookrightarrow$ This is the most used risk in classification
  - Under the squared loss $\ell(y, h(x)) = (y - h(x))^2$:

$$R(h) = \int_{\mathcal{X}\times\mathcal{Y}} (y - h(x))^2 dP(x, y) = \mathbb{E}_P[(Y - h(X))^2].$$

    $\hookrightarrow$ This is the most used risk in regression

- **Q** : what is the best predictor h ? or equivalently, when the risk $R(h)$ is optimal ?

Risk

- **Risk** : Given the pair (X, Y) with (unknown) joint distribution P, the error of approximating Y by $h(X)$ is measured by a chosen loss function $\ell(Y, h(X))$. Then, the *Risk* associated to model/hypothesis h under loss l is the *Expected loss* :

$$R(h) = \mathbb{E}_P[\ell(Y, h(X))] = \int_{\mathcal{X}\times\mathcal{Y}} \ell(y, h(x)) dP(x, y).$$

$\hookrightarrow$ prediction error that measures the generalization performance of h.

- **Risk Examples**
  - Under the "0-1"-loss $\ell(y, h(x)) = \mathbb{1}_{h(x)\neq y}$:

$$R(h) = \int_{\mathcal{X}\times\mathcal{Y}} \mathbb{1}_{h(x)\neq y} dP(x, y) = \mathbb{E}_P[\mathbb{1}_{h(x)\neq y}] = \mathbb{P}(h(X) \neq Y).$$

$\hookrightarrow$ This is the most used risk in classification

  - Under the squared loss $\ell(y, h(x)) = (y - h(x))^2$:

$$R(h) = \int_{\mathcal{X}\times\mathcal{Y}} (y - h(x))^2 dP(x, y) = \mathbb{E}_P[(Y - h(X))^2].$$

$\hookrightarrow$ This is the most used risk in regression

- **Q** : what is the best predictor h ? or equivalently, when the risk $R(h)$ is optimal ?

Risk

- **Risk** : Given the pair (X, Y) with (unknown) joint distribution P, the error of approximating Y by $h(X)$ is measured by a chosen loss function $\ell(Y, h(X))$. Then, the *Risk* associated to model/hypothesis h under loss l is the *Expected loss* :

$$R(h) = \mathbb{E}_P[\ell(Y, h(X))] = \int_{\mathcal{X}\times\mathcal{Y}} \ell(y, h(x)) dP(x, y).$$

$\hookrightarrow$ prediction error that measures the generalization performance of h.

- **Risk Examples**
  - Under the "0-1"-loss $\ell(y, h(x)) = \mathbb{1}_{h(x)\neq y}$:

$$R(h) = \int_{\mathcal{X}\times\mathcal{Y}} \mathbb{1}_{h(x)\neq y} dP(x, y) = \mathbb{E}_P[\mathbb{1}_{h(x)\neq y}] = \mathbb{P}(h(X) \neq Y).$$

$\hookrightarrow$ This is the most used risk in classification

  - Under the squared loss $\ell(y, h(x)) = (y - h(x))^2$:

$$R(h) = \int_{\mathcal{X}\times\mathcal{Y}} (y - h(x))^2 dP(x, y) = \mathbb{E}_P[(Y - h(X))^2].$$

$\hookrightarrow$ This is the most used risk in regression

- **Q** : what is the best predictor h ? or equivalently, when the risk $R(h)$ is optimal ?

Risk

- **Risk** : Given the pair (X, Y) with (unknown) joint distribution P, the error of approximating Y by $h(X)$ is measured by a chosen loss function $\ell(Y, h(X))$. Then, the *Risk* associated to model/hypothesis h under loss l is the *Expected loss* :

$$R(h) = \mathbb{E}_P[\ell(Y, h(X))] = \int_{\mathcal{X}\times\mathcal{Y}} \ell(y, h(x))dP(x, y).$$

  $\hookrightarrow$ prediction error that measures the generalization performance of h.

- **Risk Examples**
  - Under the "0-1"-loss $\ell(y, h(x)) = \mathbb{1}_{h(x)\neq y}$:

$$R(h) = \int_{\mathcal{X}\times\mathcal{Y}} \mathbb{1}_{h(x)\neq y} dP(x, y) = \mathbb{E}_P[\mathbb{1}_{h(x)\neq y}] = \mathbb{P}(h(X) \neq Y).$$

    $\hookrightarrow$ This is the most used risk in classification
  - Under the squared loss $\ell(y, h(x)) = (y - h(x))^2$:

$$R(h) = \int_{\mathcal{X}\times\mathcal{Y}} (y - h(x))^2 dP(x, y) = \mathbb{E}_P[(Y - h(X))^2].$$

    $\hookrightarrow$ This is the most used risk in regression

- **Q** : what is the best predictor h ? or equivalently, when the risk $R(h)$ is optimal ?

Optimal Risk

- $R(h) = \mathbb{E}[\ell(Y, h(X))]$ is the error of function h under loss ℓ
- **Q** : What is the smallest possible error we can achieve (under loss ℓ) ?

Bayes Risk

- $R(h)$ is minimized at a Bayes decision function $h^* : \mathcal{X} \to \mathcal{Y}$ satisfying

$$\forall x \in \mathcal{X}, h^*(x) \in \arg \min_{h(x) \in \mathcal{Y}} \mathbb{E}[\ell(Y, h(x)) | X = x].$$ [See proof in the next slide]

- The Bayes risk R^* is the risk of all Bayes predictors is equal to

$$R^* = R(h^*) = \inf_h R(h)$$

- The infimum risk R^* (taken for *all possible prediction functions* h) is known as the Bayes risk.
- A Bayes decision function h^* is a function that achieves the minimal risk R^*

- **Excess risk** : The excess risk of h is equal to $R(h) - R^*$ (always non-negative).

Optimal Risk

- $R(h) = \mathbb{E}[\ell(Y, h(X))]$ is the error of function h under loss ℓ
- **Q** : What is the smallest possible error we can achieve (under loss ℓ) ?

Bayes Risk

- $R(h)$ is minimized at a Bayes decision function $h^* : \mathcal{X} \to \mathcal{Y}$ satisfying

$$\forall x \in \mathcal{X}, h^*(x) \in \arg \min_{h(x) \in \mathcal{Y}} \mathbb{E}[\ell(Y, h(x))|X = x].$$ [See proof in the next slide]

- The Bayes risk R^* is the risk of all Bayes predictors is equal to

$$R^* = R(h^*) = \inf_h R(h)$$

- The infimum risk R^* (taken for *all possible prediction functions* h) is known as the Bayes risk.
- A Bayes decision function h^* is a function that achieves the minimal risk R^*

- **Excess risk** : The excess risk of h is equal to $R(h) - R^*$ (always non-negative).

Optimal Risk

- $R(h) = \mathbb{E}[\ell(Y, h(X))]$ is the error of function h under loss ℓ
- **Q** : What is the smallest possible error we can achieve (under loss ℓ) ?

Bayes Risk

- $R(h)$ is minimized at a Bayes decision function $h^* : \mathcal{X} \to \mathcal{Y}$ satisfying

  $\forall x \in \mathcal{X}, h^*(x) \in \arg\min_{h(x) \in \mathcal{Y}} \mathbb{E}[\ell(Y, h(x))|X = x].$ [See proof in the next slide]

- The Bayes risk R^* is the risk of all Bayes predictors is equal to

$$R^* = R(h^*) = \inf_h R(h)$$

- The infimum risk R^* (taken for *all possible prediction functions* h) is known as the Bayes risk.
- A Bayes decision function h^* is a function that achieves the minimal risk R^*

- **Excess risk** : The excess risk of h is equal to $R(h) - R^*$ (always non-negative).

Bayes Risk

- By the law of total expectation we have : $\mathbb{E}[Z] = \mathbb{E}[\mathbb{E}[Z|T]]$. We can then write $R(h) = \mathbb{E}[\mathbb{E}[\ell(Y, h(X))|X]] = \mathbb{E}_{x\sim P_X}[\mathbb{E}[\ell(Y, h(X))|X = x]] = \mathbb{E}_{x\sim P_X}[r(z|x)]$
- if we consider the conditional risk (deterministic function)

$$r(h(x)|x) = \mathbb{E}[\ell(Y, h(x))|X = x],$$

this leads to

$$R(h) = \mathbb{E}[r(h(X)|X)].$$

Bayes risk

Given the distribution $Y|X = x$ for any x, the optimal predictor h^* is known : $R(h)$ is minimized at a Bayes predictor $h^* : \mathcal{X} \to \mathcal{Y}$ satisfying

$$\forall x \in \mathcal{X}, h^*(x) \in \arg\min_{z\in\mathcal{Y}} \mathbb{E}[\ell(Y, z)|X = x].$$

The Bayes risk R^* is the risk of all Bayes predictors and is equal to

$$R^* = R(h^*) = \mathbb{E}_{x\sim P_X} \inf_{h(x)\in\mathcal{Y}} \mathbb{E}[\ell(Y, h(x))|X = x].$$

Bayes Risk

- By the law of total expectation we have : $\mathbb{E}[Z] = \mathbb{E}[\mathbb{E}[Z|T]]$. We can then write $R(h) = \mathbb{E}[\mathbb{E}[\ell(Y, h(X))|X]] = \mathbb{E}_{x\sim P_X}[\mathbb{E}[\ell(Y, h(X))|X = x]] = \mathbb{E}_{x\sim P_X}[r(z|x)]$
- if we consider the conditional risk (deterministic function)

$$r(h(x)|x) = \mathbb{E}[\ell(Y, h(x))|X = x],$$

this leads to

$$R(h) = \mathbb{E}[r(h(X)|X)].$$

Bayes risk

Given the distribution $Y|X = x$ for any x, the optimal predictor h^* is known : $R(h)$ is minimized at a Bayes predictor $h^* : \mathcal{X} \to \mathcal{Y}$ satisfying

$$\forall x \in \mathcal{X}, h^*(x) \in \arg\min_{z\in\mathcal{Y}} \mathbb{E}[\ell(Y, z)|X = x].$$

The Bayes risk R^* is the risk of all Bayes predictors and is equal to

$$R^* = R(h^*) = \mathbb{E}_{x\sim P_X} \inf_{h(x)\in\mathcal{Y}} \mathbb{E}[\ell(Y, h(x))|X = x].$$

Bayes Risk

- By the law of total expectation we have : $\mathbb{E}[Z] = \mathbb{E}[\mathbb{E}[Z|T]]$. We can then write $R(h) = \mathbb{E}[\mathbb{E}[\ell(Y, h(X))|X]] = \mathbb{E}_{x \sim P_X}[\mathbb{E}[\ell(Y, h(X))|X = x]] = \mathbb{E}_{x \sim P_X}[r(z|x)]$
- if we consider the conditional risk (deterministic function)

$$r(h(x)|x) = \mathbb{E}[\ell(Y, h(x))|X = x],$$

this leads to

$$R(h) = \mathbb{E}[r(h(X)|X)].$$

Bayes risk

Given the distribution $Y|X = x$ for any x, the optimal predictor h^* is known : $R(h)$ is minimized at a Bayes predictor $h^* : \mathcal{X} \to \mathcal{Y}$ satisfying

$$\forall x \in \mathcal{X}, h^*(x) \in \arg\min_{z \in \mathcal{Y}} \mathbb{E}[\ell(Y, z)|X = x].$$

The Bayes risk R^* is the risk of all Bayes predictors and is equal to

$$R^* = R(h^*) = \mathbb{E}_{x \sim P_X} \inf_{h(x) \in \mathcal{Y}} \mathbb{E}[\ell(Y, h(x))|X = x].$$

Optimal prediction functions

- **In classification**, i.e $\mathcal{Y} = \{0, 1\}$ or $\mathcal{Y} = \{1, \ldots, K\}$, under the (0-1)-loss, $\ell(y, h(x)) = \mathbb{1}_{h(x) \neq y}$, the best predictor is

$$h^*(x) \in \arg \max_{h(x) \in \mathcal{Y}} \mathbb{P}(Y = h(x)|X = x).$$

We have the conditional risk :

$$\begin{aligned} \mathbb{E}[\ell(Y, h(X))|X = x] &= \mathbb{E}[\mathbb{1}_{h(X) \neq Y}|X = x] \\ &= \mathbb{P}[h(X) \neq Y|X = x] \\ &= 1 - \mathbb{P}[Y = h(X)|X = x], \end{aligned}$$

then

$$\begin{aligned} h^*(x) \quad \in \quad & \arg \min_{h(X) \in \mathcal{Y}} \mathbb{E}[\ell(Y, h(X))|X = x] \\ & = \arg \min_{h(X) \in \mathcal{Y}} \{1 - \mathbb{P}[Y = h(X)|X = x]\} \\ & = \arg \max_{h(X) \in \mathcal{Y}} \mathbb{P}(Y = h(X)|X = x) \end{aligned}$$

- This why under this loss Bayes' risk corresponds to the MAP - Maximum A Posteriori principle

Optimal prediction functions

- **In classification**, i.e $\mathcal{Y} = \{0, 1\}$ or $\mathcal{Y} = \{1, \ldots, K\}$, under the (0-1)-loss, $\ell(y, h(x)) = \mathbb{1}_{h(x) \neq y}$, the best predictor is

$$h^*(x) \in \arg\max_{h(x) \in \mathcal{Y}} \mathbb{P}(Y = h(x)|X = x).$$

We have the conditional risk :

$$\begin{aligned} \mathbb{E}[\ell(Y, h(X))|X = x] &= \mathbb{E}[\mathbb{1}_{h(X) \neq Y}|X = x] \\ &= \mathbb{P}[h(X) \neq Y|X = x] \\ &= 1 - \mathbb{P}[Y = h(X)|X = x], \end{aligned}$$

then

$$\begin{aligned} h^*(x) \quad \in \quad & \arg\min_{h(X) \in \mathcal{Y}} \mathbb{E}[\ell(Y, h(X))|X = x] \\ & = \arg\min_{h(X) \in \mathcal{Y}} \{1 - \mathbb{P}[Y = h(X)|X = x]\} \\ & = \arg\max_{h(X) \in \mathcal{Y}} \mathbb{P}(Y = h(X)|X = x) \end{aligned}$$

- This why under this loss Bayes' risk corresponds to the MAP - Maximum A Posteriori principle

Optimal prediction functions

- **In classification**, i.e $\mathcal{Y} = \{0, 1\}$ or $\mathcal{Y} = \{1, \ldots, K\}$, under the (0-1)-loss, $\ell(y, h(x)) = \mathbb{1}_{h(x)\neq y}$, the best predictor is

$$h^*(x) \in \arg\max_{h(x)\in\mathcal{Y}} \mathbb{P}(Y = h(x)|X = x).$$

We have the conditional risk :

$$\begin{aligned}
\mathbb{E}[\ell(Y, h(X))|X = x] &= \mathbb{E}[\mathbb{1}_{h(X)\neq Y}|X = x] \\
&= \mathbb{P}[h(X) \neq Y|X = x] \\
&= 1 - \mathbb{P}[Y = h(X)|X = x],
\end{aligned}$$

then

$$\begin{aligned}
h^*(x) \in & \arg\min_{h(X)\in\mathcal{Y}} \mathbb{E}[\ell(Y, h(X))|X = x] \\
& = \arg\min_{h(X)\in\mathcal{Y}} \{1 - \mathbb{P}[Y = h(X)|X = x]\} \\
& = \arg\max_{h(X)\in\mathcal{Y}} \mathbb{P}(Y = h(X)|X = x)
\end{aligned}$$

- This why under this loss Bayes' risk corresponds to the MAP - Maximum A Posteriori principle

Optimal prediction functions

- **In regression**, i.e $\mathcal{Y} = \mathbb{R}$, under the square loss, $\ell(y, h(x)) = (h(x) - y)^2$, the best predictor is

$$h^*(x) = \mathbb{E}[Y|X = x]$$

We have the conditional risk :

$$\begin{aligned} R(h(x)) = \mathbb{E}[\ell(Y, h(X))|X = x] &= \mathbb{E}[(h(X) - Y)^2|X = x] \\ &= \int (h(X) - y)^2 p(y|x) dy \end{aligned}$$

optimizing the Risk by differentiating w.r.t $h(x)$ and setting the derivative to 0 :

$$\begin{aligned} \frac{\partial R(h(x))}{\partial h(x)} &= \frac{\partial}{\partial h(x)} \{ \int (h(X) - y)^2 p(y|x) dy \} \\ &= 2 \int (h(x) - y) p(y|x) dy \\ &= 2[h(x) \underbrace{\int p(y|x) dy}_{1} - \int y p(y|x) dy] = 2(h(x) - \mathbb{E}[Y|X = x]) \end{aligned}$$

which is zero at $h(x)^* = \mathbb{E}[Y|X = x]$ then

$h^*(x) = \arg \min_{h(x) \in \mathcal{Y}} \mathbb{E}[\ell(Y, h(X))|X = x] = \mathbb{E}[Y|X = x]$

Optimal prediction functions

- **In regression**, i.e $\mathcal{Y} = \mathbb{R}$, under the square loss, $\ell(y, h(x)) = (h(x) - y)^2$, the best predictor is

$$h^*(x) = \mathbb{E}[Y|X = x]$$

We have the conditional risk :

$$\begin{aligned} R(h(x)) = \mathbb{E}[\ell(Y, h(X))|X = x] &= \mathbb{E}[(h(X) - Y)^2|X = x] \\ &= \int (h(X) - y)^2 p(y|x)dy \end{aligned}$$

optimizing the Risk by differentiating w.r.t $h(x)$ and setting the derivative to 0 :

$$\begin{aligned} \frac{\partial R(h(x))}{\partial h(x)} &= \frac{\partial}{\partial h(x)}\{\int (h(X) - y)^2 p(y|x)dy\} \\ &= 2\int (h(x) - y)p(y|x)dy \\ &= 2\Big[h(x)\underbrace{\int p(y|x)dy}_{1} - \int yp(y|x)dy\Big] = 2\big(h(x) - \mathbb{E}[Y|X = x]\big) \end{aligned}$$

which is zero at $h(x)^* = \mathbb{E}[Y|X = x]$ then
$h^*(x) = \arg\min_{h(x)\in\mathcal{Y}} \mathbb{E}[\ell(Y, h(X))|X = x] = \mathbb{E}[Y|X = x]$

Empirical Risk Minimization

- Then *Expected loss* $R(h)$ depends on the joint distribution P of the pair (X, Y).

× In real situations P is in unknown, as we only have a sample $D_n = (X_i, Y_i)_{1 \leq i \leq n}$,

↪ We attempt to minimize the **Empirical Risk**

$$R_n(h) = \frac{1}{n} \sum_{i=1}^{n} \ell(Y_i, h(X_i))$$

to estimate h (within a family $\mathcal{H}$) :

$$\widehat{h}_n \in \arg \min_{h \in \mathcal{H}} R_n(h).$$

Why this is relevant ? Note : By the Law of Large Numbers, $(\frac{1}{n} \sum_{i=1}^{n} \ell(Y_i, h(X_i))_n \xrightarrow{p} \mathbb{E}[\ell(Y, h(X))]$ (the empirical mean converges to the true mean in probability), then

$$(R_n(h))_n \xrightarrow{p} R(h)$$

Goal : of supervised learning : estimate h^*, knowing only the data $\mathcal{D}_n$ and loss ℓ.

Fitting/Estimation/Learning : The objective is to construct a **fit** (estimate, learning) $\widehat{h}_n$ of the unknown function h to an observed sample (training set) $\mathcal{D}_n$ by minimizing R_n

Empirical Risk Minimization

- Then *Expected loss* $R(h)$ depends on the joint distribution P of the pair (X, Y).

× In real situations P is in unknown, as we only have a sample $D_n = (X_i, Y_i)_{1 \leq i \leq n}$,

↪ We attempt to minimize the **Empirical Risk**

$$R_n(h) = \frac{1}{n} \sum_{i=1}^{n} \ell(Y_i, h(X_i))$$

to estimate h (within a family $\mathcal{H}$) :

$$\widehat{h}_n \in \arg\min_{h \in \mathcal{H}} R_n(h).$$

Why this is relevant ? Note : By the Law of Large Numbers, $(\frac{1}{n}\sum_{i=1}^{n} \ell(Y_i, h(X_i))_n \xrightarrow{P} \mathbb{E}[\ell(Y, h(X))]$ (the empirical mean converges to the true mean in probability), then

$$(R_n(h))_n \xrightarrow{P} R(h)$$

Goal : of supervised learning : estimate h^*, knowing only the data $\mathcal{D}_n$ and loss ℓ.

Fitting/Estimation/Learning : The objective is to construct a **fit** (estimate, learning) $\widehat{h}_n$ of the unknown function h to an observed sample (training set) $\mathcal{D}_n$ by minimizing R_n

Empirical Risk Minimization

- Then *Expected loss* $R(h)$ depends on the joint distribution P of the pair (X, Y).

× In real situations P is in unknown, as we only have a sample $D_n = (X_i, Y_i)_{1 \leq i \leq n}$,

↪ We attempt to minimize the **Empirical Risk**

$$R_n(h) = \frac{1}{n} \sum_{i=1}^{n} \ell(Y_i, h(X_i))$$

to estimate h (within a family $\mathcal{H}$) :

$$\widehat{h}_n \in \arg\min_{h \in \mathcal{H}} R_n(h).$$

Why this is relevant ? Note : By the Law of Large Numbers, $(\frac{1}{n}\sum_{i=1}^{n} \ell(Y_i, h(X_i))_n \xrightarrow{\mathcal{P}} \mathbb{E}[\ell(Y, h(X))]$ (the empirical mean converges to the true mean in probability), then

$$(R_n(h))_n \xrightarrow{\mathcal{P}} R(h)$$

Goal : of supervised learning : estimate h^*, knowing only the data $\mathcal{D}_n$ and loss ℓ.

Fitting/Estimation/Learning : The objective is to construct a **fit** (estimate, learning) $\widehat{h}_n$ of the unknown function h to an observed sample (training set) $\mathcal{D}_n$ by minimizing R_n

Empirical Risk Minimization

- Then *Expected loss* $R(h)$ depends on the joint distribution P of the pair (X, Y).

× In real situations P is in unknown, as we only have a sample $D_n = (X_i, Y_i)_{1 \leq i \leq n}$,

↪ We attempt to minimize the **Empirical Risk**

$$R_n(h) = \frac{1}{n} \sum_{i=1}^{n} \ell(Y_i, h(X_i))$$

to estimate h (within a family $\mathcal{H}$) :

$$\widehat{h}_n \in \arg\min_{h \in \mathcal{H}} R_n(h).$$

Why this is relevant ? Note : By the Law of Large Numbers, $(\frac{1}{n}\sum_{i=1}^{n} \ell(Y_i, h(X_i))_n \xrightarrow{p} \mathbb{E}[\ell(Y, h(X))]$ (the empirical mean converges to the true mean in probability), then

$$(R_n(h))_n \xrightarrow{p} R(h)$$

Goal : of supervised learning : estimate h^*, knowing only the data $\mathcal{D}_n$ and loss ℓ.

Fitting/Estimation/Learning : The objective is to construct a **fit** (estimate, learning) $\widehat{h}_n$ of the unknown function h to an observed sample (training set) $\mathcal{D}_n$ by minimizing R_n

Empirical Risk Minimization

- Then *Expected loss* $R(h)$ depends on the joint distribution P of the pair (X, Y).

× In real situations P is in unknown, as we only have a sample $D_n = (X_i, Y_i)_{1 \leq i \leq n}$,

↪ We attempt to minimize the **Empirical Risk**

$$R_n(h) = \frac{1}{n} \sum_{i=1}^{n} \ell(Y_i, h(X_i))$$

to estimate h (within a family $\mathcal{H}$) :

$$\widehat{h}_n \in \arg\min_{h \in \mathcal{H}} R_n(h).$$

Why this is relevant ? Note : By the Law of Large Numbers, $(\frac{1}{n}\sum_{i=1}^{n} \ell(Y_i, h(X_i))_n \xrightarrow{P} \mathbb{E}[\ell(Y, h(X))]$ (the empirical mean converges to the true mean in probability), then

$$(R_n(h))_n \xrightarrow{P} R(h)$$

Goal : of supervised learning : estimate h^*, knowing only the data $\mathcal{D}_n$ and loss ℓ.

Fitting/Estimation/Learning : The objective is to construct a **fit** (estimate, learning) $\widehat{h}_n$ of the unknown function h to an observed sample (training set) $\mathcal{D}_n$ by minimizing R_n

Empirical Risk Minimization

- Then *Expected loss* $R(h)$ depends on the joint distribution P of the pair (X,Y).

× In real situations P is in unknown, as we only have a sample $D_n = (X_i, Y_i)_{1 \leq i \leq n}$,

↪ We attempt to minimize the **Empirical Risk**

$$R_n(h) = \frac{1}{n}\sum_{i=1}^{n} \ell(Y_i, h(X_i))$$

to estimate h (within a family $\mathcal{H}$) :

$$\widehat{h}_n \in \arg\min_{h \in \mathcal{H}} R_n(h).$$

Why this is relevant ? Note : By the Law of Large Numbers, $(\frac{1}{n}\sum_{i=1}^{n} \ell(Y_i, h(X_i))_n \xrightarrow{P} \mathbb{E}[\ell(Y, h(X))]$ (the empirical mean converges to the true mean in probability), then

$$(R_n(h))_n \xrightarrow{P} R(h)$$

Goal : of supervised learning : estimate h^*, knowing only the data $\mathcal{D}_n$ and loss ℓ.

Fitting/Estimation/Learning : The objective is to construct a **fit** (estimate, learning) $\widehat{h}_n$ of the unknown function h to an observed sample (training set) $\mathcal{D}_n$ by minimizing R_n

Example : Ordinary Least Squares (OLS)

MSE and Ordinary Least Squares (OLS) :

- The standard loss for regression is the squared loss : $\ell_2(x, y, h(x)) = (y - h(x))^2$.
- ERM :

$$\widehat{h}_n \in \arg\min_{h \in \mathcal{H}} R_n(h)$$

where the empirical risk $R_n(h)$ under the square loss is the empirical squared loss [1]

$$R_n(h) = \frac{1}{n} \sum_{i=1}^{n} \|Y_i - h(X_i)\|_2^2$$

- $\widehat{h}_n$ is known as the **Ordinary Least Squares (OLS) Estimator** of h,
- Consider $\mathcal{H} = \{h_\theta(x) = \alpha + \beta^T x\}$, the set of linear functions in x of the form $\theta^T \boldsymbol{x}$ with $\boldsymbol{x} = (1, x^T)^T$, and $\theta = (\alpha, \beta^T)^T$.
- Solution : $\widehat{\theta}_n = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\boldsymbol{Y}$, whenever $\mathbf{X}^T\mathbf{X}$ has full rank. ($\mathbf{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^T$ and $\boldsymbol{Y} = (Y_1, \ldots, Y_n)^T$)

1. also called the Mean Squared Error (MSE), or the mean Residual Squared Sum (RSS) when the ML problem is phrased as an error model $Y = h(X) + \epsilon$, $\epsilon \sim p$

Example : Ordinary Least Squares (OLS)

MSE and Ordinary Least Squares (OLS) :

- The standard loss for regression is the squared loss : $\ell_2(x, y, h(x)) = (y - h(x))^2$.
- ERM :

$$\widehat{h}_n \in \arg\min_{h \in \mathcal{H}} R_n(h)$$

where the empirical risk $R_n(h)$ under the square loss is the empirical squared loss [1]

$$R_n(h) = \frac{1}{n} \sum_{i=1}^{n} \|Y_i - h(X_i)\|_2^2$$

- $\widehat{h}_n$ is known as the **Ordinary Least Squares (OLS) Estimator** of h,
- Consider $\mathcal{H} = \{h_\theta(x) = \alpha + \beta^T x\}$, the set of linear functions in x of the form $\theta^T \boldsymbol{x}$ with $\boldsymbol{x} = (1, x^T)^T$, and $\theta = (\alpha, \beta^T)^T$.
- Solution : $\widehat{\theta}_n = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\boldsymbol{Y}$, whenever $\mathbf{X}^T\mathbf{X}$ has full rank.
  $(\mathbf{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^T$ and $\boldsymbol{Y} = (Y_1, \ldots, Y_n)^T)$

1. also called the Mean Squared Error (MSE), or the mean Residual Squared Sum (RSS) when the ML problem is phrased as an error model $Y = h(X) + \epsilon$, $\epsilon \sim p$

Example : Ordinary Least Squares (OLS)

MSE and Ordinary Least Squares (OLS) :

- The standard loss for regression is the squared loss : $\ell_2(x, y, h(x)) = (y - h(x))^2$.
- ERM :

$$\widehat{h}_n \in \arg\min_{h \in \mathcal{H}} R_n(h)$$

where the empirical risk $R_n(h)$ under the square loss is the empirical squared loss [1]

$$R_n(h) = \frac{1}{n}\sum_{i=1}^{n} \|Y_i - h(X_i)\|_2^2$$

- $\widehat{h}_n$ is known as the **Ordinary Least Squares (OLS) Estimator** of h,
- Consider $\mathcal{H} = \{h_\theta(x) = \alpha + \beta^T x\}$, the set of linear functions in x of the form $\theta^T \boldsymbol{x}$ with $\boldsymbol{x} = (1, x^T)^T$, and $\theta = (\alpha, \beta^T)^T$.
- Solution : $\widehat{\theta}_n = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\boldsymbol{Y}$, whenever $\mathbf{X}^T\mathbf{X}$ has full rank. $(\mathbf{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^T$ and $\boldsymbol{Y} = (Y_1, \ldots, Y_n)^T)$

1. also called the Mean Squared Error (MSE), or the mean Residual Squared Sum (RSS) when the ML problem is phrased as an error model $Y = h(X) + \epsilon$, $\epsilon \sim p$

Example : Ordinary Least Squares (OLS)

MSE and Ordinary Least Squares (OLS) :

- The standard loss for regression is the squared loss : $\ell_2(x, y, h(x)) = (y - h(x))^2$.
- ERM :

$$\widehat{h}_n \in \arg\min_{h \in \mathcal{H}} R_n(h)$$

where the empirical risk $R_n(h)$ under the square loss is the empirical squared loss [1]

$$R_n(h) = \frac{1}{n} \sum_{i=1}^{n} \|Y_i - h(X_i)\|_2^2$$

- $\widehat{h}_n$ is known as the **Ordinary Least Squares (OLS) Estimator** of h,
- Consider $\mathcal{H} = \{h_\theta(x) = \alpha + \beta^T x\}$, the set of linear functions in x of the form $\theta^T \boldsymbol{x}$ with $\boldsymbol{x} = (1, x^T)^T$, and $\theta = (\alpha, \beta^T)^T$.
- Solution : $\widehat{\theta}_n = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\boldsymbol{Y}$, whenever $\mathbf{X}^T\mathbf{X}$ has full rank. $(\mathbf{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^T$ and $\boldsymbol{Y} = (Y_1, \ldots, Y_n)^T)$

1. also called the Mean Squared Error (MSE), or the mean Residual Squared Sum (RSS) when the ML problem is phrased as an error model $Y = h(X) + \epsilon$, $\epsilon \sim p$

Example : Ordinary Least Squares (OLS)

MSE and Ordinary Least Squares (OLS) :

- The standard loss for regression is the squared loss : $\ell_2(x, y, h(x)) = (y - h(x))^2$.
- ERM :

$$\widehat{h}_n \in \arg\min_{h \in \mathcal{H}} R_n(h)$$

where the empirical risk $R_n(h)$ under the square loss is the empirical squared loss [1]

$$R_n(h) = \frac{1}{n}\sum_{i=1}^{n} \|Y_i - h(X_i)\|_2^2$$

- $\widehat{h}_n$ is known as the **Ordinary Least Squares (OLS) Estimator** of h,
- Consider $\mathcal{H} = \{h_\theta(x) = \alpha + \beta^T x\}$, the set of linear functions in x of the form $\theta^T \boldsymbol{x}$ with $\boldsymbol{x} = (1, x^T)^T$, and $\theta = (\alpha, \beta^T)^T$.
- Solution : $\widehat{\theta}_n = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\boldsymbol{Y}$, whenever $\mathbf{X}^T\mathbf{X}$ has full rank. ($\mathbf{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^T$ and $\boldsymbol{Y} = (Y_1, \ldots, Y_n)^T$)

1. also called the Mean Squared Error (MSE), or the mean Residual Squared Sum (RSS) when the ML problem is phrased as an error model $Y = h(X) + \epsilon$, $\epsilon \sim p$

Excess Risk Decomposition

- Suppose that a learning algorithm chooses the predictor from a class $\mathcal{H}$, and define

$$h_{\mathcal{H}} = \arg \inf_{h \in \mathcal{H}} R(h)$$

- Let h^* be the best predictor, i.e which achieves the Bayes risk :

$$R^* = R(h^*) = \inf_{h \in \text{all possible } h} R(h)$$

- Given any $\widehat{h}_n \in \mathcal{H}$, the excess risk of $\widehat{h}_n$, $R(\widehat{h}_n) - R^*$, which compares the risk of $\widehat{h}$ to the Bayes optimal prediction function h^*, can be decomposed as

$$R(\widehat{h}_n) - R^* = \underbrace{R(\widehat{h}_n) - R(h_{\mathcal{H}})}_{\text{Estimation Error}} + \underbrace{R(h_{\mathcal{H}}) - R^*}_{\text{Approximation Error}}$$

- The approximation error : $R(h_{\mathcal{H}}) - R^*$ is deterministic and is caused by the restriction to the class $\mathcal{H}$ rather than all possible functions.

  ↪ It is a property of the class $\mathcal{H}$. Bigger $\mathcal{H}$ implies smaller approximation error.

- The estimation error $R(\widehat{h}_n) - R(h_{\mathcal{H}})$ is caused by the usage of a finite sample that cannot completely represent the underlying distribution. It is random.

  ↪ With smaller $\mathcal{H}$ we expect smaller estimation error.

↪ Trade-off : When the "number" of models in $\mathcal{H}$ (e.g., number of parameters in Θ) grows, the approximation error goes down, while the estimation error goes up, and

Excess Risk Decomposition

- Suppose that a learning algorithm chooses the predictor from a class $\mathcal{H}$, and define

$$h_{\mathcal{H}} = \arg \inf_{h \in \mathcal{H}} R(h)$$

- Let h^* be the best predictor, i.e which achieves the Bayes risk :

$$R^* = R(h^*) = \inf_{h \in \text{all possible } h} R(h)$$

- Given any $\widehat{h}_n \in \mathcal{H}$, the excess risk of $\widehat{h}_n$, $R(\widehat{h}_n) - R^*$, which compares the risk of $\widehat{h}$ to the Bayes optimal prediction function h^*, can be decomposed as

$$R(\widehat{h}_n) - R^* = \underbrace{R(\widehat{h}_n) - R(h_{\mathcal{H}})}_{\text{Estimation Error}} + \underbrace{R(h_{\mathcal{H}}) - R^*}_{\text{Approximation Error}}$$

- The approximation error : $R(h_{\mathcal{H}}) - R^*$ is deterministic and is caused by the restriction to the class $\mathcal{H}$ rather than all possible functions.

  ↪ It is a property of the class $\mathcal{H}$. Bigger $\mathcal{H}$ implies smaller approximation error.

- The estimation error $R(\widehat{h}_n) - R(h_{\mathcal{H}})$ is caused by the usage of a finite sample that cannot completely represent the underlying distribution. It is random.

  ↪ With smaller $\mathcal{H}$ we expect smaller estimation error.

↪ Trade-off : When the "number" of models in $\mathcal{H}$ (e.g., number of parameters in Θ) grows, the approximation error goes down, while the estimation error goes up, and

Excess Risk Decomposition

- Suppose that a learning algorithm chooses the predictor from a class $\mathcal{H}$, and define

$$h_{\mathcal{H}} = \arg \inf_{h \in \mathcal{H}} R(h)$$

- Let h^* be the best predictor, i.e which achieves the Bayes risk :

$$R^* = R(h^*) = \inf_{h \in \text{all possible } h} R(h)$$

- Given any $\widehat{h}_n \in \mathcal{H}$, the excess risk of $\widehat{h}_n$, $R(\widehat{h}_n) - R^*$, which compares the risk of $\widehat{h}$ to the Bayes optimal prediction function h^*, can be decomposed as

$$R(\widehat{h}_n) - R^* = \underbrace{R(\widehat{h}_n) - R(h_{\mathcal{H}})}_{\text{Estimation Error}} + \underbrace{R(h_{\mathcal{H}}) - R^*}_{\text{Approximation Error}}$$

- The approximation error : $R(h_{\mathcal{H}}) - R^*$ is deterministic and is caused by the restriction to the class $\mathcal{H}$ rather than all possible functions.

  $\hookrightarrow$ It is a property of the class $\mathcal{H}$. Bigger $\mathcal{H}$ implies smaller approximation error.

- The estimation error $R(\widehat{h}_n) - R(h_{\mathcal{H}})$ is caused by the usage of a finite sample that cannot completely represent the underlying distribution. It is random.

  $\hookrightarrow$ With smaller $\mathcal{H}$ we expect smaller estimation error.

$\hookrightarrow$ Trade-off : When the "number" of models in $\mathcal{H}$ (e.g., number of parameters in Θ) grows, the approximation error goes down, while the estimation error goes up, and

Excess Risk Decomposition

- Suppose that a learning algorithm chooses the predictor from a class $\mathcal{H}$, and define

$$h_{\mathcal{H}} = \arg \inf_{h \in \mathcal{H}} R(h)$$

- Let h^* be the best predictor, i.e which achieves the Bayes risk :

$$R^* = R(h^*) = \inf_{h \in \text{all possible } h} R(h)$$

- Given any $\widehat{h}_n \in \mathcal{H}$, the excess risk of $\widehat{h}_n$, $R(\widehat{h}_n) - R^*$, which compares the risk of $\widehat{h}$ to the Bayes optimal prediction function h^*, can be decomposed as

$$R(\widehat{h}_n) - R^* = \underbrace{R(\widehat{h}_n) - R(h_{\mathcal{H}})}_{\text{Estimation Error}} + \underbrace{R(h_{\mathcal{H}}) - R^*}_{\text{Approximation Error}}$$

- The approximation error : $R(h_{\mathcal{H}}) - R^*$ is deterministic and is caused by the restriction to the class $\mathcal{H}$ rather than all possible functions.

  $\hookrightarrow$ It is a property of the class $\mathcal{H}$. Bigger $\mathcal{H}$ implies smaller approximation error.

- The estimation error $R(\widehat{h}_n) - R(h_{\mathcal{H}})$ is caused by the usage of a finite sample that cannot completely represent the underlying distribution. It is random.

  $\hookrightarrow$ With smaller $\mathcal{H}$ we expect smaller estimation error.

$\hookrightarrow$ Trade-off : When the "number" of models in $\mathcal{H}$ (e.g., number of parameters in Θ) grows, the approximation error goes down, while the estimation error goes up, and

Excess Risk Decomposition

- Suppose that a learning algorithm chooses the predictor from a class $\mathcal{H}$, and define

$$h_{\mathcal{H}} = \arg\inf_{h \in \mathcal{H}} R(h)$$

- Let h^* be the best predictor, i.e which achieves the Bayes risk :

$$R^* = R(h^*) = \inf_{h \in \text{all possible } h} R(h)$$

- Given any $\widehat{h}_n \in \mathcal{H}$, the excess risk of $\widehat{h}_n$, $R(\widehat{h}_n) - R^*$, which compares the risk of $\widehat{h}$ to the Bayes optimal prediction function h^*, can be decomposed as

$$R(\widehat{h}_n) - R^* = \underbrace{R(\widehat{h}_n) - R(h_{\mathcal{H}})}_{\text{Estimation Error}} + \underbrace{R(h_{\mathcal{H}}) - R^*}_{\text{Approximation Error}}$$

- The approximation error : $R(h_{\mathcal{H}}) - R^*$ is deterministic and is caused by the restriction to the class $\mathcal{H}$ rather than all possible functions.

  ↪ It is a property of the class $\mathcal{H}$. Bigger $\mathcal{H}$ implies smaller approximation error.

- The estimation error $R(\widehat{h}_n) - R(h_{\mathcal{H}})$ is caused by the usage of a finite sample that cannot completely represent the underlying distribution. It is random.

  ↪ With smaller $\mathcal{H}$ we expect smaller estimation error.

↪ Trade-off : When the "number" of models in $\mathcal{H}$ (e.g., number of parameters in Θ) grows, the approximation error goes down, while the estimation error goes up, and

ERM Overview in Practice

- Given a loss function ℓ.
- Choose a hypothesis space $\mathcal{H}$.
- Use an optimization method to find the **Empirical Risk Minimizer (ERM)** :

$$\widehat{h}_n \in \arg\min_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, h(x_i)).$$

$\hookrightarrow$ **Data Scientist's Role :** Choose $\mathcal{H}$ to **balance approximation and estimation error**.

$\hookrightarrow$ As we get more training data, we can use a *larger* $\mathcal{H}$.

Optimization Error :

- In practice, we very often need an **optimization method** to find $\widehat{h}_n \in \mathcal{H}$.
- However, we may not find the exact ERM $\widehat{h}_n$. Instead, we find an approximation $\widetilde{h}_n \in \mathcal{H}$ that is hopefully *good enough*, and in some cases, $\widetilde{h}_n$ may generalize better than $\widehat{h}_n$ (i.e., achieves a lower true risk $R(\widetilde{h}_n) < R(\widehat{h}_n)$) due to regularization or improved numerical stability, early stopping, etc

ERM Overview in Practice

- Given a loss function ℓ.
- Choose a hypothesis space $\mathcal{H}$.
- Use an optimization method to find the **Empirical Risk Minimizer (ERM)** :

$$\widehat{h}_n \in \arg\min_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, h(x_i)).$$

↪ **Data Scientist's Role :** Choose $\mathcal{H}$ to **balance approximation and estimation error**.

↪ As we get more training data, we can use a *larger* $\mathcal{H}$.

Optimization Error :

- In practice, we very often need an **optimization method** to find $\widehat{h}_n \in \mathcal{H}$.
- However, we may not find the exact ERM $\widehat{h}_n$. Instead, we find an approximation $\widetilde{h}_n \in \mathcal{H}$ that is hopefully *good enough*, and in some cases, $\widetilde{h}_n$ may generalize better than $\widehat{h}_n$ (i.e., achieves a lower true risk $R(\widetilde{h}_n) < R(\widehat{h}_n)$) due to regularization or improved numerical stability, early stopping, etc

ERM Overview in Practice

- Given a loss function ℓ.
- Choose a hypothesis space $\mathcal{H}$.
- Use an optimization method to find the **Empirical Risk Minimizer (ERM)** :

$$\widehat{h}_n \in \arg\min_{h\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, h(x_i)).$$

$\hookrightarrow$ **Data Scientist's Role :** Choose $\mathcal{H}$ to **balance approximation and estimation error**.

$\hookrightarrow$ As we get more training data, we can use a *larger* $\mathcal{H}$.

Optimization Error :

- In practice, we very often need an **optimization method** to find $\widehat{h}_n \in \mathcal{H}$.
- However, we may not find the exact ERM $\widehat{h}_n$. Instead, we find an approximation $\widetilde{h}_n \in \mathcal{H}$ that is hopefully *good enough*, and in some cases, $\widetilde{h}_n$ may generalize better than $\widehat{h}_n$ (i.e., achieves a lower true risk $R(\widetilde{h}_n) < R(\widehat{h}_n)$) due to regularization or improved numerical stability, early stopping, etc

ERM Overview in Practice

- Given a loss function ℓ.
- Choose a hypothesis space $\mathcal{H}$.
- Use an optimization method to find the **Empirical Risk Minimizer (ERM)** :

$$\widehat{h}_n \in \arg\min_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, h(x_i)).$$

↪ **Data Scientist's Role :** Choose $\mathcal{H}$ to **balance approximation and estimation error**.

↪ As we get more training data, we can use a *larger* $\mathcal{H}$.

Optimization Error :

- In practice, we very often need an **optimization method** to find $\widehat{h}_n \in \mathcal{H}$.
- However, we may not find the exact ERM $\widehat{h}_n$. Instead, we find an approximation $\widetilde{h}_n \in \mathcal{H}$ that is hopefully *good enough*, and in some cases, $\widetilde{h}_n$ may generalize better than $\widehat{h}_n$ (i.e., achieves a lower true risk $R(\widetilde{h}_n) < R(\widehat{h}_n)$) due to regularization or improved numerical stability, early stopping, etc

ERM Overview in Practice

- Given a loss function ℓ.
- Choose a hypothesis space $\mathcal{H}$.
- Use an optimization method to find the **Empirical Risk Minimizer (ERM)** :

$$\widehat{h}_n \in \arg\min_{h\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, h(x_i)).$$

↪ **Data Scientist's Role :** Choose $\mathcal{H}$ to **balance approximation and estimation error**.

↪ As we get more training data, we can use a *larger* $\mathcal{H}$.

Optimization Error :

- In practice, we very often need an **optimization method** to find $\widehat{h}_n \in \mathcal{H}$.
- However, we may not find the exact ERM $\widehat{h}_n$. Instead, we find an approximation $\widetilde{h}_n \in \mathcal{H}$ that is hopefully *good enough*, and in some cases, $\widetilde{h}_n$ may generalize better than $\widehat{h}_n$ (i.e., achieves a lower true risk $R(\widetilde{h}_n) < R(\widehat{h}_n)$) due to regularization or improved numerical stability, early stopping, etc

ERM Overview in Practice

- Given a loss function ℓ.
- Choose a hypothesis space $\mathcal{H}$.
- Use an optimization method to find the **Empirical Risk Minimizer (ERM)** :

$$\widehat{h}_n \in \arg\min_{h\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n}\ell(y_i, h(x_i)).$$

$\hookrightarrow$ **Data Scientist's Role :** Choose $\mathcal{H}$ to **balance approximation and estimation error**.

$\hookrightarrow$ As we get more training data, we can use a *larger* $\mathcal{H}$.

Optimization Error :

- In practice, we very often need an **optimization method** to find $\widehat{h}_n \in \mathcal{H}$.
- However, we may not find the exact ERM $\widehat{h}_n$. Instead, we find an approximation $\widetilde{h}_n \in \mathcal{H}$ that is hopefully *good enough*, and in some cases, $\widetilde{h}_n$ may generalize better than $\widehat{h}_n$ (i.e., achieves a lower true risk $R(\widetilde{h}_n) < R(\widehat{h}_n)$) due to regularization or improved numerical stability, early stopping, etc

Decomposition

Optimization Error :

- Measures the difference in true risk between the empirical risk minimizer $\widehat{h}_n$ and the function $\widetilde{h}_n$ returned by the *optimization algorithm*.
- Optimization error is defined as :

$$\text{Optimization Error} = R(\widetilde{h}_n) - R(\widehat{h}_n).$$

- This error *can be negative* (if optimization finds a better function than $\widehat{h}_n$ due to regularization or numerical properties as explained in the previous slide).

Excess Risk Decomposition :

- The **excess risk** of $\widetilde{h}_n$ can be decomposed as :

$$\begin{aligned}\text{Excess Risk}(\widetilde{h}_n) &= R(\widetilde{h}_n) - R(h^*)\\ &= \underbrace{R(\widetilde{h}_n) - R(\widehat{h}_n)}_{\text{Optimization Error}} + \underbrace{R(\widehat{h}_n) - R(h_{\mathcal{H}})}_{\text{Estimation Error}} + \underbrace{R(h_{\mathcal{H}}) - R(h^*)}_{\text{Approximation Error}}.\end{aligned}$$

Decomposition

Optimization Error :

- Measures the difference in true risk between the empirical risk minimizer $\widehat{h}_n$ and the function $\widetilde{h}_n$ returned by the *optimization algorithm*.
- Optimization error is defined as :

$$\text{Optimization Error} = R(\widetilde{h}_n) - R(\widehat{h}_n).$$

- This error *can be negative* (if optimization finds a better function than $\widehat{h}_n$ due to regularization or numerical properties as explained in the previous slide).

Excess Risk Decomposition :

- The **excess risk** of $\widetilde{h}_n$ can be decomposed as :

$$\begin{aligned}\text{Excess Risk}(\widetilde{h}_n) &= R(\widetilde{h}_n) - R(h^*) \\ &= \underbrace{R(\widetilde{h}_n) - R(\widehat{h}_n)}_{\text{Optimization Error}} + \underbrace{R(\widehat{h}_n) - R(h_{\mathcal{H}})}_{\text{Estimation Error}} + \underbrace{R(h_{\mathcal{H}}) - R(h^*)}_{\text{Approximation Error}}.\end{aligned}$$

Decomposition

Optimization Error :

- Measures the difference in true risk between the empirical risk minimizer $\widehat{h}_n$ and the function $\widetilde{h}_n$ returned by the *optimization algorithm*.
- Optimization error is defined as :

$$\text{Optimization Error} = R(\widetilde{h}_n) - R(\widehat{h}_n).$$

- This error *can be negative* (if optimization finds a better function than $\widehat{h}_n$ due to regularization or numerical properties as explained in the previous slide).

Excess Risk Decomposition :

- The **excess risk** of $\widetilde{h}_n$ can be decomposed as :

$$\begin{aligned}\text{Excess Risk}(\widetilde{h}_n) &= R(\widetilde{h}_n) - R(h^*) \\ &= \underbrace{R(\widetilde{h}_n) - R(\widehat{h}_n)}_{\text{Optimization Error}} + \underbrace{R(\widehat{h}_n) - R(h_{\mathcal{H}})}_{\text{Estimation Error}} + \underbrace{R(h_{\mathcal{H}}) - R(h^*)}_{\text{Approximation Error}}.\end{aligned}$$

Optimization Error

- **Optimization error can be negative** (but the excess risk is always non-negative) : Optimization does not always return the ERM $\widehat{h}_n$, but sometimes finds a **better function** $\widetilde{h}_n$ that generalizes better (achieving smaller true risk R).
- Example : by **Regularization.** Regularization prevents overfitting and can improve generalization, resulting in a lower true risk R.
- **Example :** We train a **logistic regression** classifier with the log loss :

$$\widehat{h}_n = \arg\min_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, h(x_i))$$

  Instead of attempting to solve this exactly, we use ℓ_2-regularization (Ridge penalty) : $\widetilde{h}_n = \arg\min_{h\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, h(x_i)) + \lambda \|h\|^2$. Then we can get

$$R(\widetilde{h}_n) \leq R(\widehat{h}_n) \quad \text{(if } \lambda \text{ is well-chosen, avoiding underfitting or overfitting)}$$

- This leads to an apparent negative optimization error, but it is due to regularization : Regularization Effect $= R(\widetilde{h}_n) - R(\widehat{h}_n) \leq 0$
- However, this is not always due to optimization – it is due to regularization.

Optimization Error

- **Optimization error can be negative** (but the excess risk is always non-negative) : Optimization does not always return the ERM $\widehat{h}_n$, but sometimes finds a **better function** $\widetilde{h}_n$ that generalizes better (achieving smaller true risk R).
- Example : by **Regularization.** Regularization prevents overfitting and can improve generalization, resulting in a lower true risk R.
- **Example :** We train a **logistic regression** classifier with the log loss :

$$\widehat{h}_n = \arg\min_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, h(x_i))$$

Instead of attempting to solve this exactly, we use ℓ_2-regularization (Ridge penalty) : $\widetilde{h}_n = \arg\min_{h\in\mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, h(x_i)) + \lambda \|h\|^2$. Then we can get

$$R(\widetilde{h}_n) \leq R(\widehat{h}_n) \quad \text{(if } \lambda \text{ is well-chosen, avoiding underfitting or overfitting)}$$

- This leads to an apparent negative optimization error, but it is due to regularization : Regularization Effect $= R(\widetilde{h}_n) - R(\widehat{h}_n) \leq 0$
- However, this is not always due to optimization – it is due to regularization.

Optimization Error

- **Optimization error can be negative** (but the excess risk is always non-negative) : Optimization does not always return the ERM $\widehat{h}_n$, but sometimes finds a **better function** $\widetilde{h}_n$ that generalizes better (achieving smaller true risk R).
- Example : by **Regularization.** Regularization prevents overfitting and can improve generalization, resulting in a lower true risk R.
- **Example :** We train a **logistic regression** classifier with the log loss :

$$\widehat{h}_n = \arg\min_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, h(x_i))$$

Instead of attempting to solve this exactly, we use ℓ_2-regularization (Ridge penalty) : $\widetilde{h}_n = \arg\min_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, h(x_i)) + \lambda \|h\|^2$. Then we can get

$$R(\widetilde{h}_n) \leq R(\widehat{h}_n) \quad \text{(if } \lambda \text{ is well-chosen, avoiding underfitting or overfitting)}$$

- This leads to an apparent negative optimization error, but it is due to regularization : Regularization Effect $= R(\widetilde{h}_n) - R(\widehat{h}_n) \leq 0$
- However, this is not always due to optimization – it is due to regularization.

Optimization Error

Why can regularization improve true risk R?

- Regularization improves generalization by reducing variance.
- Logistic regression without regularization can produce very large coefficients, leading to poor generalization.
- **Avoiding poorly conditioned solutions** helps in optimization stability.
- **SGD/momentum methods** can converge to flatter (less-sharp) minima thus more stable (to small data deviations) that generalize better.
- **Early stopping in neural networks** prevents overfitting by stopping training when validation error increases.

For a reminder on optimization principles and algorithms, see my course : *Optimization for Machine Learning* available at : `https://chamroukhi.com/teaching.php`

Excess Risk and Kullback-Leibler Divergence

- Consider the log-loss : $\ell(y, h_\theta(x)) = -\log(p_\theta(x,y))$
- The risk under this loss is $R(\theta) = \mathbb{E}_P[\ell(Y, h_\theta(X))] = \mathbb{E}_P[-\log p_\theta(X,Y)]$
- The excess risk of θ

$$\begin{aligned} R(\theta) - R^* &= \mathbb{E}_P[-\log p_\theta(X,Y) + \log p_{\theta^*}(X,Y)] \\ &= \mathbb{E}_P[\log \frac{p_{\theta^*}(X,Y)}{p_\theta(X,Y)}] \\ &= \int \log \frac{p_{\theta^*}(x,y)}{p_\theta(x,y)} p_{\theta^*}(x,y)\, dP(x,y) \\ &= \mathsf{KL}(p_{\theta^*} \| p_\theta) \\ &\geq 0 : \end{aligned}$$

which is equal to $\mathrm{KL}(p_{\theta^*} \| p_\theta)$, the Kullback-Leibler divergence between p_θ and p_{θ^*}

- Note : $\mathrm{KL}(p_{\theta^*} \| p_\theta) = 0$ holds if and only if $p_{\theta^*} = p_\theta$.
- Although not a distance measure (not symmetric), the KL-divergence measures the discrepancy between two distributions.

Excess Risk and Kullback-Leibler Divergence

- Consider the log-loss : $\ell(y, h_\theta(x)) = -\log(p_\theta(x, y))$
- The risk under this loss is $R(\theta) = \mathbb{E}_P[\ell(Y, h_\theta(X))] = \mathbb{E}_P[-\log p_\theta(X, Y)]$
- The excess risk of θ

$$\begin{aligned} R(\theta) - R^* &= \mathbb{E}_P[-\log p_\theta(X, Y) + \log p_{\theta^*}(X, Y)] \\ &= \mathbb{E}_P[\log \frac{p_{\theta^*}(X, Y)}{p_\theta(X, Y)}] \\ &= \int \log \frac{p_{\theta^*}(x, y)}{p_\theta(x, y)} p_{\theta^*}(x, y)\, dP(x, y) \\ &= \mathsf{KL}(p_{\theta^*} \| p_\theta) \\ &\geq 0 : \end{aligned}$$

which is equal to $\mathrm{KL}(p_{\theta^*} \| p_\theta)$, the Kullback-Leibler divergence between p_θ and p_{θ^*}

- Note : $\mathrm{KL}(p_{\theta^*} \| p_\theta) = 0$ holds if and only if $p_{\theta^*} = p_\theta$.
- Although not a distance measure (not symmetric), the KL-divergence measures the discrepancy between two distributions.

Excess Risk and Kullback-Leibler Divergence

- Consider the log-loss : $\ell(y, h_\theta(x)) = -\log(p_\theta(x, y))$
- The risk under this loss is $R(\theta) = \mathbb{E}_P[\ell(Y, h_\theta(X))] = \mathbb{E}_P[-\log p_\theta(X, Y)]$
- The excess risk of θ

$$\begin{aligned} R(\theta) - R^* &= \mathbb{E}_P[-\log p_\theta(X,Y) + \log p_{\theta^*}(X,Y)] \\ &= \mathbb{E}_P[\log \frac{p_{\theta^*}(X,Y)}{p_\theta(X,Y)}] \\ &= \int \log \frac{p_{\theta^*}(x,y)}{p_\theta(x,y)} p_{\theta^*}(x,y)\, dP(x,y) \\ &= \mathsf{KL}(p_{\theta^*} \| p_\theta) \\ &\geq 0 : \end{aligned}$$

which is equal to $\mathrm{KL}(p_{\theta^*} \| p_\theta)$, the Kullback-Leibler divergence between p_θ and p_{θ^*}

- Note : $\mathrm{KL}(p_{\theta^*} \| p_\theta) = 0$ holds if and only if $p_{\theta^*} = p_\theta$.
- Although not a distance measure (not symmetric), the KL-divergence measures the discrepancy between two distributions.

Maximum Likelihood Estimation

Maximum Likelihood Estimation

- Def. **Likelihood function** : The likelihood function for model h is the joint pdf of the observed data given h

$$L(h) = P(\mathcal{D}|h) = P(\{(x_i, y_i)_{i=1}^n\}|h)$$

- Def. **The Maximum Likelihood Estimator** : Maximum likelihood estimation seeks for the model $\widehat{h}$ that fits best the data : The Maximum Likelihood Estimator (MLE) is then a maximizer of the likelihood function, i.e :

$$\widehat{h}_n \in \arg\max_{h \in \mathcal{H}} L(h).$$

- **Note** : Since the $\log$ function is strictly increasing, then, the MLE is preferentially performed (for notably numerical reasons, and sums are easier to work with than products) by maximizing the log-likelihood :

$$\widehat{h}_n \in \arg\max_{h \in \mathcal{H}} \log L(h).$$

Maximum Likelihood Estimation

- Def. **Likelihood function** : The likelihood function for model h is the joint pdf of the observed data given h

$$L(h) = P(\mathcal{D}|h) = P(\{(x_i, y_i)_{i=1}^n\}|h)$$

- Def. **The Maximum Likelihood Estimator** : Maximum likelihood estimation seeks for the model $\widehat{h}$ that fits best the data : The Maximum Likelihood Estimator (MLE) is then a maximizer of the likelihood function, i.e :

$$\widehat{h}_n \in \arg\max_{h \in \mathcal{H}} L(h).$$

- **Note** : Since the $\log$ function is strictly increasing, then, the MLE is preferentially performed (for notably numerical reasons, and sums are easier to work with than products) by maximizing the log-likelihood :

$$\widehat{h}_n \in \arg\max_{h \in \mathcal{H}} \log L(h).$$

Maximum Likelihood Estimation

- Def. **Likelihood function** : The likelihood function for model h is the joint pdf of the observed data given h

$$L(h) = P(\mathcal{D}|h) = P(\{(x_i, y_i)_{i=1}^n\}|h)$$

- Def. **The Maximum Likelihood Estimator** : Maximum likelihood estimation seeks for the model $\widehat{h}$ that fits best the data : The Maximum Likelihood Estimator (MLE) is then a maximizer of the likelihood function, i.e :

$$\widehat{h}_n \in \arg\max_{h \in \mathcal{H}} L(h).$$

- **Note** : Since the $\log$ function is strictly increasing, then, the MLE is preferentially performed (for notably numerical reasons, and sums are easier to work with than products) by maximizing the log-likelihood :

$$\widehat{h}_n \in \arg\max_{h \in \mathcal{H}} \log L(h).$$

Parametric models

Def. Parametric model of distributions

A probabilistic model on a data space $\mathcal{X}$ is a family of probability distributions indexed by $\theta \in \Theta$. We denote this as

$$P = \{p_\theta(x); \theta \in \Theta\}$$

where θ is the (vector of) parameter(s) and Θ is the parameter space.

- Bernoulli : $p_\theta(x) = \mathbb{P}_\theta(X = x) = \theta^x(1-\theta)^{1-x}$ with $\mathcal{X} = \{0,1\}$ and $\theta \in \Theta = [0,1]$
- Binomial : $p_{\boldsymbol{\theta}}(x) = \mathbb{P}_{\boldsymbol{\theta}}(X = x) = \binom{N}{x}\nu^x(1-\nu)^{1-x}$ with $\mathcal{X} = \{0,1,...,N\}$ and $\boldsymbol{\theta} = (N,\nu) \in \Theta = \mathbb{N} \times [0,1]$
- Univariate Gaussian : $p_{\boldsymbol{\theta}}(x) = \varphi(x;\mu,\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}(\frac{x-\mu}{\sigma})^2\right)$ with $\mathcal{X} = \mathbb{R}$ and $\boldsymbol{\theta} = (\mu,\sigma^2) \in \Theta = \mathbb{R} \times \mathbb{R}_+$
- multivariate Gaussian : $\phi_d(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{\frac{d}{2}}|\boldsymbol{\Sigma}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$ with $\mathcal{X} = \mathbb{R}^d$ and $\boldsymbol{\theta} = (\boldsymbol{\mu}', \text{vech}(\boldsymbol{\Sigma})')' \in \Theta = \mathbb{R} \times \mathcal{S}^d_{++}$; The set of symmetric positive definite matrices on $\mathbb{R}^d$: $\mathcal{S}^d_{++} = \{\Sigma \in \mathbb{R}^{d\times d} : \Sigma = \Sigma' \text{ and } \Sigma \succ 0\}$

Parametric models

Def. Parametric model of distributions

A probabilistic model on a data space $\mathcal{X}$ is a family of probability distributions indexed by $\theta \in \Theta$. We denote this as

$$P = \{p_\theta(x); \theta \in \Theta\}$$

where θ is the (vector of) parameter(s) and Θ is the parameter space.

- Bernoulli : $p_\theta(x) = \mathbb{P}_\theta(X = x) = \theta^x(1-\theta)^{1-x}$ with $\mathcal{X} = \{0, 1\}$ and $\theta \in \Theta = [0, 1]$
- Binomial : $p_{\boldsymbol{\theta}}(x) = \mathbb{P}_{\boldsymbol{\theta}}(X = x) = \binom{N}{x}\nu^x(1-\nu)^{1-x}$ with $\mathcal{X} = \{0, 1, ..., N\}$ and $\boldsymbol{\theta} = (N, \nu) \in \Theta = \mathbb{N} \times [0, 1]$
- Univariate Gaussian : $p_{\boldsymbol{\theta}}(x) = \varphi(x; \mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}(\frac{x-\mu}{\sigma})^2\right)$ with $\mathcal{X} = \mathbb{R}$ and $\boldsymbol{\theta} = (\mu, \sigma^2) \in \Theta = \mathbb{R} \times \mathbb{R}_+$
- multivariate Gaussian : $\phi_d(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{\frac{d}{2}}|\boldsymbol{\Sigma}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$ with $\mathcal{X} = \mathbb{R}^d$ and $\boldsymbol{\theta} = (\boldsymbol{\mu}', \text{vech}(\boldsymbol{\Sigma})')' \in \Theta = \mathbb{R} \times \mathcal{S}^d_{++}$; The set of symmetric positive definite matrices on $\mathbb{R}^d$: $\mathcal{S}^d_{++} = \{\Sigma \in \mathbb{R}^{d\times d} : \Sigma = \Sigma' \text{ and } \Sigma \succ 0\}$

Examples of MLE

Example : MLE for the Bernoulli

- Bernoulli : $p_\theta(x) = \mathbb{P}(X = x|\theta) = \theta^x(1-\theta)^{1-x}$ with $\mathcal{X} = \{0,1\}$ and $\theta \in \Theta = [0,1]$
- MLE : $\widehat{\theta} = \frac{1}{n}\sum_{i=1}^n X_i$.

MLE : $\widehat{\theta} = \arg\max_\theta \log L(\theta)$. By independence and identical distribution, we have

$$\begin{aligned}
\log L(\theta) &= \log \mathbb{P}(X_1 = x_1, \ldots, X_n = x_n; \theta) = \log \textstyle\prod_{i=1}^n \mathbb{P}(X_i = x_i; \theta) \\
&= \log \textstyle\prod_{i=1}^n \theta^{x_i}(1-\theta)^{1-x_i} \\
&= \textstyle\sum_{i=1}^n x_i \log\theta + \sum_{i=1}^n (1-x_i)\log(1-\theta) \\
\frac{\partial \log L(\theta)}{\partial\theta} &= \frac{1}{\theta}\textstyle\sum_{i=1}^n x_i - \frac{1}{1-\theta}\sum_{i=1}^n (1-x_i), \text{which is zero at}
\end{aligned}$$

$$\begin{aligned}
\frac{1}{\widehat{\theta}}\textstyle\sum_{i=1}^n x_i - \frac{1}{1-\widehat{\theta}}\sum_{i=1}^n (1-x_i) &= 0 \\
(1-\widehat{\theta})\textstyle\sum_{i=1}^n x_i - \widehat{\theta}\sum_{i=1}^n (1-x_i) &= 0 \\
\textstyle\sum_{i=1}^n x_i - n\widehat{\theta} &= 0 \\
\widehat{\theta} &= \frac{1}{n}\textstyle\sum_{i=1}^n X_i.
\end{aligned}$$

Examples of MLE

Example : MLE for the Gaussian mean

- Univariate Gaussian : $p_{\boldsymbol{\theta}}(x) = \phi_1(x; \mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}(\frac{x-\mu}{\sigma})^2\right)$ with $\mathcal{X} = \mathbb{R}$ and $\boldsymbol{\theta} = (\mu, \sigma^2) \in \Theta = \mathbb{R} \times \mathbb{R}_+$
- MLE : $\widehat{\theta} = (\widehat{\mu}, \widehat{\sigma}^2)$ with $\widehat{\mu} = \frac{1}{n}\sum_{i=1}^n X_i$ and $\widehat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^n (X_i - \widehat{\mu})^2$.

MLE : $\widehat{\theta} = \arg\max_\theta \log L(\theta)$.

$$\log L(\mu, \sigma^2) = \log p(X_1 = x_1, \ldots, X_n = x_n; \mu, \sigma^2) = \log \prod_{i=1}^n \frac{1}{\sigma\sqrt{2\,\pi}} \mathrm{e}^{-\frac{1}{2}\left(\frac{x_i-\mu}{\sigma}\right)^2}$$

$$= \sum_{i=1}^n \log \frac{1}{\sigma\sqrt{2\,\pi}} - \frac{1}{2\sigma^2}\sum_{i=1}^n (x_i - \mu)^2 = -\frac{n}{2}\log 2\pi - \frac{n}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\sum_{i=1}^n (x_i - \mu)^2.$$

We have $\frac{\partial L(\mu,\sigma^2)}{\partial \mu} = \frac{1}{\sigma^2}\sum_{i=1}^n (x_i - \mu)$ and $\frac{\partial L(\mu,\sigma^2)}{\partial \sigma^2} = -\frac{n}{2\sigma^2} + \frac{1}{2\sigma^4}\sum_{i=1}^n (x_i - \mu)^2$.
which are zero at

$$\frac{\partial L(\widehat{\mu},\sigma^2)}{\partial \mu} = 0 \Longrightarrow \sum_{i=1}^n (X_i - \widehat{\mu}) = 0 \Longrightarrow \widehat{\mu} = \frac{1}{n}\sum_{i=1}^n X_i$$

$$\frac{\partial L(\mu,\widehat{\sigma}^2)}{\partial \sigma^2} = 0 \Longrightarrow -n\widehat{\sigma}^2 + \sum_{i=1}^n (x_i - \mu)^2 \Longrightarrow \widehat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^n (X_i - \widehat{\mu})^2.$$

When MLE coincides with ERM I

- Consider the parametric setting :
- MLE (density estimation framework) : We seek for an esitmator of the parameters θ of the joint distribution $p_\theta(x, y)$. For an independent and identically distributed (iid) sample $\{(x_i, y_i)_{i=1}^n\}$, the log-likelihood function of θ is :

$$\log L(\theta) = \sum_{i=1}^{n} \log p_\theta(x_i, y_i).$$

- ERM : We seek for a predictor $h_{\boldsymbol{\theta}}$ given a training set $\{(x_i, y_i)_{i=1}^n\}$ from $p_\theta(x, y)$. Consider the log-loss :

$$\ell(y, h_\theta(x)) = -\log(p_\theta(x, y)).$$

The corresponding empirical risk is by definition

$$R_n(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, h_\theta(x_i)) = -\frac{1}{n}\sum_{i=1}^{n} \log p_\theta(x_i, y_i) = -\frac{1}{n}\log L(\theta)$$

$\hookrightarrow$ With the log-loss, ERM coincides with MLE.

Examples :

MLE coincides with OLS (ERM) in Gaussian regression (see later)

MLE coincides with ERM in Logistic regression (see later)

Conditional maximum likelihood risks

- In some situations, we are interested in estimating the conditional distribution $P(Y|X)$, rather than the joint distribution $P(X,Y)$.
- As we'll see it later, this is the case for example in discriminative learning (eg. logistic regression for classification, or Gaussian linear regression with non-random predictors) where we do not need to define a distribution of X.
- In the parametric setting, we therefore have the conditional log-likehood risk

$$R(\theta) = -\mathbb{E}[\log p_\theta(Y|X)]$$

and the corresponding conditional empirical risk

$$R_n(\theta) = -\frac{1}{n}\sum_{i=1}^{n}\log p_\theta(y_i|x_i)$$

which coincides with the conditional log-likehood.

Conditional maximum likelihood risks

- In some situations, we are interested in estimating the conditional distribution $P(Y|X)$, rather than the joint distribution $P(X, Y)$.
- As we'll see it later, this is the case for example in discriminative learning (eg. logistic regression for classification, or Gaussian linear regression with non-random predictors) where we do not need to define a distribution of X.
- In the parametric setting, we therefore have the conditional log-likehood risk

$$R(\theta) = -\mathbb{E}[\log p_\theta(Y|X)]$$

and the corresponding conditional empirical risk

$$R_n(\theta) = -\frac{1}{n}\sum_{i=1}^{n}\log p_\theta(y_i|x_i)$$

which coincides with the conditional log-likehood.

Conditional maximum likelihood risks

- In some situations, we are interested in estimating the conditional distribution $P(Y|X)$, rather than the joint distribution $P(X, Y)$.
- As we'll see it later, this is the case for example in discriminative learning (eg. logistic regression for classification, or Gaussian linear regression with non-random predictors) where we do not need to define a distribution of X.
- In the parametric setting, we therefore have the conditional log-likehood risk

$$R(\theta) = -\mathbb{E}[\log p_\theta(Y|X)]$$

and the corresponding conditional empirical risk

$$R_n(\theta) = -\frac{1}{n}\sum_{i=1}^{n}\log p_\theta(y_i|x_i)$$

which coincides with the conditional log-likehood.

Conditional maximum likelihood risks

Example : Logistic Regression :

- Logistic Regression model : $p_{\boldsymbol{\theta}}(y|\boldsymbol{x}) = \pi_{\boldsymbol{\theta}}(\boldsymbol{x})^y (1 - \pi_{\boldsymbol{\theta}}(\boldsymbol{x}))^{1-y}$ with $y \in \{0, 1\}$, and $\pi_{\boldsymbol{\theta}}(\boldsymbol{x}) = \sigma(\beta_0 + \boldsymbol{\beta}^T \boldsymbol{x}) = \frac{\exp(\beta_0 + \boldsymbol{\beta}^T \boldsymbol{x})}{1 + \exp(\beta_0 + \boldsymbol{\beta}^T \boldsymbol{x})}$ is the logistic function.
- Empirical risk :

$$
\begin{aligned}
R_n(\theta) &= -\frac{1}{n} \sum_{i=1}^{n} \log p_{\boldsymbol{\theta}}(y_i|x_i) \\
&= -\frac{1}{n} \sum_{i=1}^{n} \log[\pi_{\boldsymbol{\theta}}(x_i)^{y_i} (1 - \pi_{\boldsymbol{\theta}}(x_))^{1-y_i}] \\
&= \sum_{i=1}^{n} y_i \log \pi(\boldsymbol{x}_i; \boldsymbol{\theta}) + (1 - y_i) \log (1 - \pi(\boldsymbol{x}_i; \boldsymbol{\theta})) \\
&= -\frac{1}{n} \underbrace{\sum_{i=1}^{n} y_i(\beta_0 + \boldsymbol{\beta}^\top \boldsymbol{x}_i) - \log(1 + \exp(\beta_0 + \boldsymbol{\beta}^\top \boldsymbol{x}_i))}_{\text{Conditional log-likelihood } L(\theta)}
\end{aligned}
$$

↪ Then we have : $\arg\min_\theta R_n(\theta) = \arg\max_\theta \log L(\theta)$.

Conditional maximum likelihood risks

Example : Logistic Regression :

- Logistic Regression model : $p_{\boldsymbol{\theta}}(y|\boldsymbol{x}) = \pi_{\boldsymbol{\theta}}(\boldsymbol{x})^y(1-\pi_{\boldsymbol{\theta}}(\boldsymbol{x}))^{1-y}$ with $y \in \{0,1\}$, and $\pi_{\boldsymbol{\theta}}(\boldsymbol{x}) = \sigma(\beta_0 + \boldsymbol{\beta}^T\boldsymbol{x}) = \frac{\exp(\beta_0+\boldsymbol{\beta}^T\boldsymbol{x})}{1+\exp(\beta_0+\boldsymbol{\beta}^T\boldsymbol{x})}$ is the logistic function.
- Empirical risk :

$$R_n(\theta) = -\frac{1}{n}\sum_{i=1}^{n} \log p_{\boldsymbol{\theta}}(y_i|x_i)$$

$$= -\frac{1}{n}\sum_{i=1}^{n} \log[\pi_{\boldsymbol{\theta}}(x_i)^{y_i}(1-\pi_{\boldsymbol{\theta}}(x_))^{1-y_i}]$$

$$= \sum_{i=1}^{n} y_i \log \pi(\boldsymbol{x}_i;\boldsymbol{\theta}) + (1-y_i)\log(1-\pi(\boldsymbol{x}_i;\boldsymbol{\theta}))$$

$$= -\frac{1}{n}\underbrace{\sum_{i=1}^{n} y_i(\beta_0 + \boldsymbol{\beta}^\top\boldsymbol{x}_i) - \log(1+\exp(\beta_0+\boldsymbol{\beta}^\top\boldsymbol{x}_i))}_{\text{Conditional log-likelihood } L(\theta)}$$

$\hookrightarrow$ Then we have : $\arg\min_\theta R_n(\theta) = \arg\max_\theta \log L(\theta)$.

Conditional maximum likelihood risks

Example : Logistic Regression :

- Logistic Regression model : $p_{\boldsymbol{\theta}}(y|\boldsymbol{x}) = \pi_{\boldsymbol{\theta}}(\boldsymbol{x})^y(1-\pi_{\boldsymbol{\theta}}(\boldsymbol{x}))^{1-y}$ with $y \in \{0,1\}$, and $\pi_{\boldsymbol{\theta}}(\boldsymbol{x}) = \sigma(\beta_0 + \boldsymbol{\beta}^T\boldsymbol{x}) = \frac{\exp(\beta_0+\boldsymbol{\beta}^T\boldsymbol{x})}{1+\exp(\beta_0+\boldsymbol{\beta}^T\boldsymbol{x})}$ is the logistic function.
- Empirical risk :

$$
\begin{aligned}
R_n(\theta) &= -\frac{1}{n}\sum_{i=1}^{n}\log p_{\boldsymbol{\theta}}(y_i|x_i)\\
&= -\frac{1}{n}\sum_{i=1}^{n}\log[\pi_{\boldsymbol{\theta}}(x_i)^{y_i}(1-\pi_{\boldsymbol{\theta}}(x_))^{1-y_i}]\\
&= \sum_{i=1}^{n} y_i\log\pi(\boldsymbol{x}_i;\boldsymbol{\theta}) + (1-y_i)\log(1-\pi(\boldsymbol{x}_i;\boldsymbol{\theta}))\\
&= -\frac{1}{n}\underbrace{\sum_{i=1}^{n} y_i(\beta_0+\boldsymbol{\beta}^\top\boldsymbol{x}_i) - \log(1+\exp(\beta_0+\boldsymbol{\beta}^\top\boldsymbol{x}_i))}_{\text{Conditional log-likelihood } L(\theta)}
\end{aligned}
$$

↪ Then we have : $\arg\min_\theta R_n(\theta) = \arg\max_\theta \log L(\theta)$.

Conditional maximum likelihood risks

Example : Logistic Regression :

- Logistic Regression model : $p_{\boldsymbol{\theta}}(y|\boldsymbol{x}) = \pi_{\boldsymbol{\theta}}(\boldsymbol{x})^y(1-\pi_{\boldsymbol{\theta}}(\boldsymbol{x}))^{1-y}$ with $y \in \{0,1\}$, and $\pi_{\boldsymbol{\theta}}(\boldsymbol{x}) = \sigma(\beta_0 + \boldsymbol{\beta}^T\boldsymbol{x}) = \frac{\exp(\beta_0+\boldsymbol{\beta}^T\boldsymbol{x})}{1+\exp(\beta_0+\boldsymbol{\beta}^T\boldsymbol{x})}$ is the logistic function.
- Empirical risk :

$$
\begin{aligned}
R_n(\theta) &= -\frac{1}{n}\sum_{i=1}^{n}\log p_{\boldsymbol{\theta}}(y_i|x_i)\\
&= -\frac{1}{n}\sum_{i=1}^{n}\log[\pi_{\boldsymbol{\theta}}(x_i)^{y_i}(1-\pi_{\boldsymbol{\theta}}(x_))^{1-y_i}]\\
&= \sum_{i=1}^{n} y_i \log \pi(\boldsymbol{x}_i;\boldsymbol{\theta}) + (1-y_i)\log(1-\pi(\boldsymbol{x}_i;\boldsymbol{\theta}))\\
&= -\frac{1}{n}\underbrace{\sum_{i=1}^{n} y_i(\beta_0 + \boldsymbol{\beta}^\top\boldsymbol{x}_i) - \log(1+\exp(\beta_0+\boldsymbol{\beta}^\top\boldsymbol{x}_i))}_{\text{Conditional log-likelihood } L(\theta)}
\end{aligned}
$$

↪ Then we have : $\arg\min_\theta R_n(\theta) = \arg\max_\theta \log L(\theta)$.

When OLS coincides with MLE

Regression with Gaussian errors

Let $y \in \mathbb{R}$ and $\mathcal{X} = \mathbb{R}^p$ and onsider the following model

$$Y_i = h(\boldsymbol{X}_i; \boldsymbol{\beta}) + \varepsilon_i \quad \text{with} \quad \varepsilon_i | \boldsymbol{X} \underset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2)$$

- Empirical Squared Risk : under the square loss, $R_n(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - h(\boldsymbol{x}_i; \boldsymbol{\beta}))^2$
- Empirical Risk Minimizer : $\widehat{\boldsymbol{\beta}}_n = \arg\min_{\boldsymbol{\beta}} R_n(\boldsymbol{\beta})$
- Conditional Maximum Likelihood Risk

  Data model : $Y_i | \boldsymbol{X}_i \underset{\text{iid}}{\sim} \mathcal{N}(h(\boldsymbol{X}_i; \boldsymbol{\beta}), \sigma^2)$: $p_{\boldsymbol{\theta}}(y_i|\boldsymbol{x}_i) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y - h(\boldsymbol{x}_i; \boldsymbol{\beta})}{\sigma}\right)^2}$

  $$\log L(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log p_{\boldsymbol{\theta}}(y_i|x_i) = -\frac{1}{2\sigma^2} \underbrace{\sum_{i=1}^{n}(y_i - h(\boldsymbol{x}_i; \boldsymbol{\beta}))^2}_{\propto R_n(\boldsymbol{\beta})} - \frac{n}{2}\log\sigma^2 - \frac{n}{2}\log(2\pi)$$

- Conditional MLE : $= \widehat{\boldsymbol{\beta}}_n = \arg\max_{\boldsymbol{\beta}} \log L(\boldsymbol{\theta})$

↪ Then we have : $\arg\min_{\boldsymbol{\beta}} R_n(\boldsymbol{\beta}) = \arg\max_{\boldsymbol{\beta}} \log L(\boldsymbol{\theta})$.

- Remark : For both we can take the sample variance as an estimator of the variance σ^2 : $\widehat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(Y_i - h(\boldsymbol{X}_i, \widehat{\boldsymbol{\beta}}))^2$ which is the Maximum-Likelihood Estimator

When OLS coincides with MLE

Regression with Gaussian errors

Let $y \in \mathbb{R}$ and $\mathcal{X} = \mathbb{R}^p$ and onsider the following model

$$Y_i = h(\boldsymbol{X}_i; \boldsymbol{\beta}) + \varepsilon_i \quad \text{with} \quad \varepsilon_i | \boldsymbol{X} \underset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2)$$

- Empirical Squared Risk : under the square loss, $R_n(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - h(\boldsymbol{x}_i; \boldsymbol{\beta}))^2$
- Empirical Risk Minimizer : $\widehat{\boldsymbol{\beta}}_n = \arg\min_{\boldsymbol{\beta}} R_n(\boldsymbol{\beta})$
- Conditional Maximum Likelihood Risk

  Data model : $Y_i|\boldsymbol{X}_i \underset{\text{iid}}{\sim} \mathcal{N}(h(\boldsymbol{X}_i; \boldsymbol{\beta}), \sigma^2)$: $p_{\boldsymbol{\theta}}(y_i|\boldsymbol{x}_i) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y - h(\boldsymbol{x}_i; \boldsymbol{\beta})}{\sigma}\right)^2}$

  $$\log L(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log p_{\boldsymbol{\theta}}(y_i|x_i) = -\frac{1}{2\sigma^2}\underbrace{\sum_{i=1}^{n}(y_i - h(\boldsymbol{x}_i; \boldsymbol{\beta}))^2}_{\propto R_n(\boldsymbol{\beta})} - \frac{n}{2}\log\sigma^2 - \frac{n}{2}\log(2\pi)$$

- Conditional MLE : $= \widehat{\boldsymbol{\beta}}_n = \arg\max_{\boldsymbol{\beta}} \log L(\boldsymbol{\theta})$

↪ Then we have : $\arg\min_{\boldsymbol{\beta}} R_n(\boldsymbol{\beta}) = \arg\max_{\boldsymbol{\beta}} \log L(\boldsymbol{\theta})$.

- Remark : For both we can take the sample variance as an estimator of the variance σ^2 : $\widehat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(Y_i - h(\boldsymbol{X}_i, \widehat{\boldsymbol{\beta}}))^2$ which is the Maximum-Likelihood Estimator

When OLS coincides with MLE

Regression with Gaussian errors

Let $y \in \mathbb{R}$ and $\mathcal{X} = \mathbb{R}^p$ and onsider the following model

$$Y_i = h(\boldsymbol{X}_i; \boldsymbol{\beta}) + \varepsilon_i \quad \text{with} \quad \varepsilon_i | \boldsymbol{X} \underset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2)$$

- Empirical Squared Risk : under the square loss, $R_n(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^n (y_i - h(\boldsymbol{x}_i; \boldsymbol{\beta}))^2$
- Empirical Risk Minimizer : $\widehat{\boldsymbol{\beta}}_n = \arg\min_{\boldsymbol{\beta}} R_n(\boldsymbol{\beta})$
- Conditional Maximum Likelihood Risk

  Data model : $Y_i | \boldsymbol{X}_i \underset{\text{iid}}{\sim} \mathcal{N}(h(\boldsymbol{X}_i; \boldsymbol{\beta}), \sigma^2)$: $p_{\boldsymbol{\theta}}(y_i | \boldsymbol{x}_i) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y - h(\boldsymbol{x}_i; \boldsymbol{\beta})}{\sigma}\right)^2}$

  $$\log L(\boldsymbol{\theta}) = \sum_{i=1}^n \log p_{\boldsymbol{\theta}}(y_i | x_i) = -\frac{1}{2\sigma^2} \underbrace{\sum_{i=1}^n (y_i - h(\boldsymbol{x}_i; \boldsymbol{\beta}))^2}_{\propto R_n(\boldsymbol{\beta})} - \frac{n}{2}\log\sigma^2 - \frac{n}{2}\log(2\pi)$$

- Conditional MLE : $= \widehat{\beta}_n = \arg\max_{\beta} \log L(\boldsymbol{\theta})$

↪ Then we have : $\arg\min_{\boldsymbol{\beta}} R_n(\boldsymbol{\beta}) = \arg\max_{\boldsymbol{\beta}} \log L(\boldsymbol{\theta})$.

- Remark : For both we can take the sample variance as an estimator of the variance σ^2 : $\widehat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^n (Y_i - h(\boldsymbol{X}_i, \widehat{\beta}))^2$ which is the Maximum-Likelihood Estimator

When OLS coincides with MLE

Regression with Gaussian errors

Let $y \in \mathbb{R}$ and $\mathcal{X} = \mathbb{R}^p$ and onsider the following model

$$Y_i = h(\boldsymbol{X}_i; \boldsymbol{\beta}) + \varepsilon_i \quad \text{with} \quad \varepsilon_i|\boldsymbol{X} \underset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2)$$

- Empirical Squared Risk : under the square loss, $R_n(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^n (y_i - h(\boldsymbol{x}_i; \boldsymbol{\beta}))^2$
- Empirical Risk Minimizer : $\widehat{\boldsymbol{\beta}}_n = \arg\min_{\boldsymbol{\beta}} R_n(\boldsymbol{\beta})$
- Conditional Maximum Likelihood Risk

  Data model : $Y_i|\boldsymbol{X}_i \underset{\text{iid}}{\sim} \mathcal{N}(h(\boldsymbol{X}_i; \boldsymbol{\beta}), \sigma^2)$: $p_{\boldsymbol{\theta}}(y_i|\boldsymbol{x}_i) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y - h(\boldsymbol{x}_i;\boldsymbol{\beta})}{\sigma}\right)^2}$

  $$\log L(\boldsymbol{\theta}) = \sum_{i=1}^n \log p_{\boldsymbol{\theta}}(y_i|x_i) = -\frac{1}{2\sigma^2} \underbrace{\sum_{i=1}^n (y_i - h(\boldsymbol{x}_i; \boldsymbol{\beta}))^2}_{\propto R_n(\boldsymbol{\beta})} - \frac{n}{2}\log\sigma^2 - \frac{n}{2}\log(2\pi)$$

- Conditional MLE : $= \widehat{\boldsymbol{\beta}}_n = \arg\max_{\boldsymbol{\beta}} \log L(\boldsymbol{\theta})$

$\hookrightarrow$ Then we have : $\arg\min_{\boldsymbol{\beta}} R_n(\boldsymbol{\beta}) = \arg\max_{\boldsymbol{\beta}} \log L(\boldsymbol{\theta})$.

- Remark : For both we can take the sample variance as an estimator of the variance σ^2 : $\widehat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^n (Y_i - h(\boldsymbol{X}_i, \widehat{\boldsymbol{\beta}}))^2$ which is the Maximum-Likelihood Estimator

When OLS coincides with MLE

Regression with Gaussian errors

Let $y \in \mathbb{R}$ and $\mathcal{X} = \mathbb{R}^p$ and onsider the following model

$$Y_i = h(\boldsymbol{X}_i;\boldsymbol{\beta}) + \varepsilon_i \quad \text{with} \quad \varepsilon_i|\boldsymbol{X} \underset{\text{iid}}{\sim} \mathcal{N}(0,\sigma^2)$$

- Empirical Squared Risk : under the square loss, $R_n(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^n (y_i - h(\boldsymbol{x}_i;\boldsymbol{\beta}))^2$
- Empirical Risk Minimizer : $\widehat{\boldsymbol{\beta}}_n = \arg\min_{\boldsymbol{\beta}} R_n(\boldsymbol{\beta})$
- Conditional Maximum Likelihood Risk

  Data model : $Y_i|\boldsymbol{X}_i \underset{\text{iid}}{\sim} \mathcal{N}(h(\boldsymbol{X}_i;\boldsymbol{\beta}),\sigma^2)$: $p_{\boldsymbol{\theta}}(y_i|\boldsymbol{x}_i) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y - h(\boldsymbol{x}_i;\boldsymbol{\beta})}{\sigma}\right)^2}$

  $$\log L(\boldsymbol{\theta}) = \sum_{i=1}^n \log p_{\boldsymbol{\theta}}(y_i|x_i) = -\frac{1}{2\sigma^2}\underbrace{\sum_{i=1}^n (y_i - h(\boldsymbol{x}_i;\boldsymbol{\beta}))^2}_{\propto R_n(\boldsymbol{\beta})} - \frac{n}{2}\log\sigma^2 - \frac{n}{2}\log(2\pi)$$

- Conditional MLE : $= \widehat{\boldsymbol{\beta}}_n = \arg\max_{\boldsymbol{\beta}} \log L(\boldsymbol{\theta})$

↪ Then we have : $\arg\min_{\boldsymbol{\beta}} R_n(\boldsymbol{\beta}) = \arg\max_{\boldsymbol{\beta}} \log L(\boldsymbol{\theta})$.

- Remark : For both we can take the sample variance as an estimator of the variance σ^2 : $\widehat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^n (Y_i - h(\boldsymbol{X}_i, \widehat{\boldsymbol{\beta}}))^2$ which is the Maximum-Likelihood Estimator

When OLS coincides with MLE

Regression with Gaussian errors

Let $y \in \mathbb{R}$ and $\mathcal{X} = \mathbb{R}^p$ and onsider the following model

$$Y_i = h(\boldsymbol{X}_i; \boldsymbol{\beta}) + \varepsilon_i \quad \text{with} \quad \varepsilon_i | \boldsymbol{X} \underset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2)$$

- Empirical Squared Risk : under the square loss, $R_n(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^n (y_i - h(\boldsymbol{x}_i; \boldsymbol{\beta}))^2$
- Empirical Risk Minimizer : $\widehat{\boldsymbol{\beta}}_n = \arg\min_{\boldsymbol{\beta}} R_n(\boldsymbol{\beta})$
- Conditional Maximum Likelihood Risk

  Data model : $Y_i|\boldsymbol{X}_i \underset{\text{iid}}{\sim} \mathcal{N}(h(\boldsymbol{X}_i; \boldsymbol{\beta}), \sigma^2)$: $p_{\boldsymbol{\theta}}(y_i|\boldsymbol{x}_i) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y - h(\boldsymbol{x}_i; \boldsymbol{\beta})}{\sigma}\right)^2}$

  $$\log L(\boldsymbol{\theta}) = \sum_{i=1}^n \log p_{\boldsymbol{\theta}}(y_i|x_i) = -\frac{1}{2\sigma^2} \underbrace{\sum_{i=1}^n (y_i - h(\boldsymbol{x}_i; \boldsymbol{\beta}))^2}_{\propto R_n(\boldsymbol{\beta})} - \frac{n}{2}\log\sigma^2 - \frac{n}{2}\log(2\pi)$$

- Conditional MLE : $= \widehat{\boldsymbol{\beta}}_n = \arg\max_{\boldsymbol{\beta}} \log L(\boldsymbol{\theta})$

↪ Then we have : $\arg\min_{\boldsymbol{\beta}} R_n(\boldsymbol{\beta}) = \arg\max_{\boldsymbol{\beta}} \log L(\boldsymbol{\theta})$.

- Remark : For both we can take the sample variance as an estimator of the variance σ^2 : $\widehat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^n (Y_i - h(\boldsymbol{X}_i, \widehat{\boldsymbol{\beta}}))^2$ which is the Maximum-Likelihood Estimator

Overview

- **Data Representation** : A random pair $(X, Y) \in \mathcal{X} \times \mathcal{Y}$, where X contains input features and Y is the target output.
- Supervised learning aims to find a **prediction function** $h : \mathcal{X} \to \mathcal{Y}$ that provides a good approximation of the true output y.
- **Loss Function** $\ell(y, h(x))$: Measures the error in predicting Y using $h(X)$.
- **Risk Function** $R(h) = \mathbb{E}[\ell(Y, h(X))]$: Expected loss over the data distribution. It measures the generalization performance of h.
- **Bayes Risk** : The lowest achievable risk, attained by the optimal prediction function h^*. **Optimal Decision Rules :**
  - **Bayes Classifier** : $h^*(x) = \arg\max_{y \in \mathcal{Y}} \mathbb{P}(Y = y | X = x)$ minimizes classification error under **0-1 loss**.
  - **Optimal Regression Function** : $h^*(x) = \mathbb{E}[Y|X = x]$ provides the best prediction error **under the squared loss**.
- **Empirical Risk Minimization (ERM)** finds h by minimizing the empirical risk : $R_n(h) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, h(x_i))$ using an optimization method.
- The **Excess Risk** $R(\widetilde{h}_n) - R(h^*)$ of a learned model $\widetilde{h}_n$, can be decomposed as sum of an **approximation error**, an**estimation error**, and an **optimization error**.

Overview

Data Scientist's Role :

- Choose a **hypothesis space** $\mathcal{H}$ that balances **approximation** and **estimation error**.
- Adjust $\mathcal{H}$ as more data becomes available to improve approximation.
- **More data implies a larger hypothesis space** $\mathcal{H}$, reducing approximation error.
- Use **optimization algorithms** to minimize empirical risk $R_n(h)$.
- **Regularization and optimization** impact the final model ?s performance.
- **Regularization** (e.g., in logistic regression) prevents overfitting and improves generalization.
- **Optimization can sometimes outperform ERM**, e.g., regularized logistic regression may yield a lower true risk.

Next slides topics

See Later :

- **Bias-Variance Decomposition**
- **Practical illustrations (Risks, Bayes Risk, Bias-Variance Tradeoff, etc)**