



I N S E A







Train and test error

Optimisation faite sur $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)$

“Training” data

L'erreur de prédiction sur ces données est forcément petite.

$$p_i \stackrel{\text{def}}{=} P(y_i = 1 | \mathbf{x}_i) = \text{sigmoid}(\alpha + \beta^\top \mathbf{x}_i)$$

$$\min_{\alpha \in \mathbb{R}, \beta \in \mathbb{R}^d} - \sum_{i=1}^n y_i \log(p_i) + (1 - y_i) \log(1 - p_i)$$

Il faut évaluer la performance du modèle sur des données nouvelles non vues à l'entraînement: "Test data"

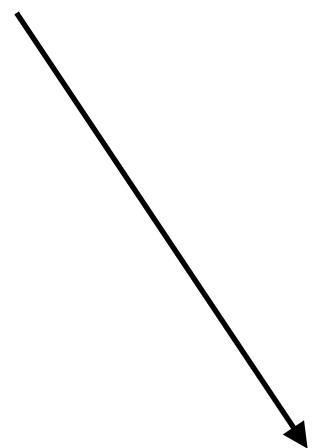
\mathbf{x}_1	y_1
\vdots	\vdots
\mathbf{x}_n	y_n

→ “Training” → “Learned” f^*



predictions	true labels
$f^{\star}(\mathbf{x}_1)$	y_1
\vdots	\vdots
$f^{\star}(\mathbf{x}_n)$	y_n

→ “Train” error



predictions	true labels
$f^*(\mathbf{x}'_1)$	y'_1
\vdots	\vdots
$f^*(\mathbf{x}'_m)$	y'_m

→ “Test” error

Est-ce une bonne manière d'évaluation la performance d'un modèle ?

Machine learning classic: zero-to-hero

$$p_i \stackrel{\text{def}}{=} \mathbb{P}(y_i = 1 | \mathbf{x}_i) = \text{sigmoid}(\alpha + \beta^\top \mathbf{x}_i)$$

Optimisation faite sur $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)$

$$\min_{\alpha \in \mathbb{R}, \beta \in \mathbb{R}^d} - \sum_{i=1}^n y_i \log(p_i) + (1 - y_i) \log(1 - p_i)$$

“Training” data

\mathbf{x}_1	y_1
\vdots	\vdots
\mathbf{x}_n	y_n

→ “Training” → “Learned” f^* →

predictions	true labels
$f^*(\mathbf{x}_1)$	y_1
\vdots	\vdots
$f^*(\mathbf{x}_n)$	y_n

→ “Train” error

Est-ce une bonne manière d’évaluation la performance du modèle ?

L’erreur de prédiction sur ces données est **optimisée**: elle est forcément **petite**.

Il faut évaluer la performance du modèle sur des données nouvelles non vues à l’entraînement: “Test data”

predictions	true labels
$f^*(\mathbf{x}'_1)$	y'_1
\vdots	\vdots
$f^*(\mathbf{x}'_m)$	y'_m

→ “Test” error



Peut-on séparer les classes avec une séparation linéaire dans ces cas ?

$d = 1$

