Note de cours 3

Slide 5. On fixe la partition $\mathcal{A} = \{A_1, \dots, A_M\}$. On définit

$$\hat{c}_{\mathcal{A}} = \underset{c \in \mathcal{F}_{\mathcal{A}}}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}(y_i c(x_i) < 0)$$

$$\frac{1}{n} \sum_{i=1}^{n} \mathbb{1}(y_i c(x_i) < 0) = \frac{1}{n} \sum_{m=1}^{M} \sum_{i : x_i \in A_m} \mathbb{1}(y_i c(x_i) < 0)$$

$$= \frac{1}{n} \sum_{m=1}^{M} \left\{ \sum_{\substack{i : x_i \in A_m \\ y_i = 1}} \mathbb{1}(y_i c(x_i) < 0) + \sum_{\substack{i : x_i \in A_m \\ y_i = -1}} \mathbb{1}(y_i c(x_i) < 0) \right\}$$

On note c_m la valeur que prend c sur A_m.

Si $c_m = 1$ alors $\{\ \} = \#\{i : x_i \in A_m, y_i = -1\}$

$c_m = -1$ — $\{\ \} = \#\{i : x_i \in A_m, y_i = 1\}$

On veut minimiser le nombre d'erreurs on doit poser

$c_m = 1$ si $\#\{i, x_i \in A_m, y_i = 1\} > \#\{i \mid x_i \in A_m, y_i = -1\}$

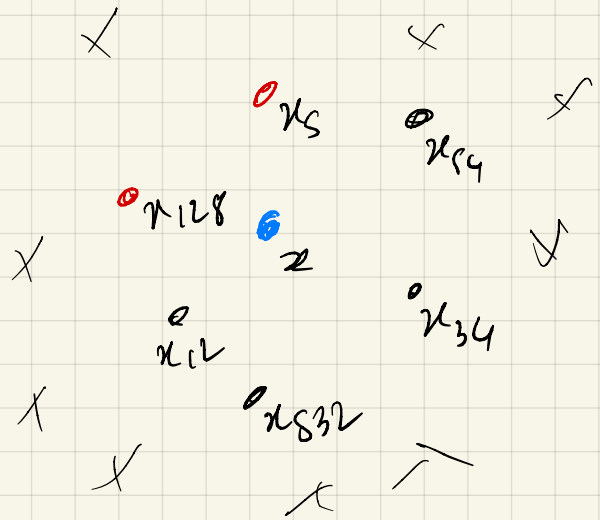
$c_m = -1$ sinon

→ ça correspond donc à un vote à la majorité. → il reste à construire la partition!

Slide 9

$k = 6$.

L'algo k-nn colorie le point bleu en noir.



On note I_x = { des numéros des k + proches voisins de x }

$= \{5, 12, 32, 34, 54, 128\}$.

$= \{i : \|x - x_i\|$ sont les $k = 6$ + petites distances $\}$.

On vote à la majorité :

$$\hat{c}(x) = \underset{\ell \in \{-1,1\}}{\operatorname{argmax}} \{i \in I_x, y_i = \ell\}$$

On remarque I_x est un sous-ensemble à k éléments de $[\![1, n]\!]$, il appartient à $\{\phi^1, \dots, \phi^M\}$ avec $M = \binom{n}{k}$.

On note

$$A_m = \{x \in X, I_x = \phi^m\}.$$

$\rightsquigarrow$ on en déduit que les k-nn sont bien dans la classe des classifieurs construits sur une partition.

Slide 29 ⚠ $\hat{c}_b^*$ sont dépendants (corrélés)

v.a.r. $Z_1, \dots, Z_B$ de même loi tq $\text{cor}(Z_i, Z_j) = \rho$

On a $\mathbb{V}\left(\frac{1}{B}\sum_{b=1}^{B} Z_b\right) = \frac{1}{B^2}\sum_{b=1}^{B}\sum_{b'=1}^{B} \text{cor}(Z_b, Z_{b'})$

$$= \frac{1}{B^2}\left\{\sum_{b=1}^{B} \mathbb{V}(Z_b) + \sum_{b=1}^{B}\sum_{b' \neq b} \text{cor}(Z_b, Z_{b'})\right\}$$

$$= \frac{1}{B^2}\left\{B\,\mathbb{V}(Z_b) + \sum_{b=1}^{B}\sum_{b' \neq b} \mathbb{V}(Z_b)\,\rho\right\}$$

$$= \frac{1}{B^2}\left\{B\,\mathbb{V}(Z_b) + B(B-1)\,\rho\,\mathbb{V}(Z_b)\right\}$$

$$= \frac{1}{B}\mathbb{V}(Z_b) + \frac{(B-1)}{B}\rho\,\mathbb{V}(Z_b)$$

$$= \rho\,\mathbb{V}(Z_b) + \frac{1}{B}(1-\rho)\,\mathbb{V}(Z_b)$$

ici $\text{cor}(Z_b, Z_{b'}) = \dfrac{\text{cov}(Z_b, Z_{b'})}{\sqrt{\mathbb{V}(Z_b)\,\mathbb{V}(Z_{b'})}}$

$\rho = \dfrac{\text{cov}(Z_b, Z_{b'})}{\mathbb{V}(Z_b)}$

il peut arriver que $\mathbb{V}\left(\frac{1}{B}\sum Z_b\right) > \mathbb{V}(Z_b)$

Slide 39. Principe du gradient boosting.

On veut construire $g^{(B)}(x) = \sum_{b=1}^{B} \eta^{(b)} h^{(b)}(x)$ avec $\eta^{(b)} > 0$.

et on veut minimiser la perte empirique

$$\frac{1}{n}\sum_{i=1}^{n} \ell\left(y_i, g^{(B)}(x_i)\right)$$

"Brutalement" on voudrait définir

$$\hat{g}^{(B)}(x) = \sum_{b=1}^{B} \hat{\eta}^{(b)} \hat{h}^{(b)}(x)$$

$$(\underbrace{\hat{\eta}^{(1)}, \dots, \hat{\eta}^{(B)}}_{\in \mathbb{R}_+}, \underbrace{\hat{h}^{(1)}, \dots, \hat{h}^{(B)}}_{\in \mathcal{H}})$$

$$= \underset{\eta^{(1)}, \dots, \eta^{(B)}, h^{(1)}, \dots, h^{(B)}}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} l\left(y_i, \sum_{b=1}^{B} \eta^{(b)} h^{(b)}(x_i)\right)$$

Le problème est qu'il faudrait chercher pour les h dans un ensemble de cardinal $\#(\mathcal{H})^B \longrightarrow$ trop long.... NP dur

On ne fait pas comme ça!

On procède pas-à-pas il y a une sorte de descente de gradient.

A l'itération b: on a $\hat{g}^{(b)}$

Pour b: on cherche $\hat{g}^{(b+1)}$ de la forme $\hat{g}^{(b)} + \eta^{(b+1)} h^{(b+1)}$

il reste à définir $\hat{\eta}^{(b+1)}$ et $\hat{h}^{(b+1)}$.

on pourrait faire:

$$\hat{\eta}^{(b+1)}, \hat{h}^{(b+1)} = \underset{\substack{\eta \in \mathbb{R}_+ \\ h \in \mathcal{H}}}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} l\left(y_i, \hat{g}^{(b)}(x_i) + \eta h(x_i)\right)$$

C'est encore trop long....

On utilise la descente de gradient sur un pas seulement.
On cherche à faire un pas de descente de gradient pour

$$u \mapsto \frac{1}{n} \sum_{i=1}^{n} \ell\left(y_i, \hat{g}^{(b)}(x_i) + \eta u(x_i)\right)$$

en enlevant, pour l'instant la contrainte $u \in \mathcal{H}$.
On remarque que $u : \mathbb{R}^d \to \mathbb{R}$ et qu'on s'intéresse seulement à ses valeurs $\begin{pmatrix} u(x_1) \\ \vdots \\ u(x_n) \end{pmatrix}$ on va identifier la fonction u au vecteur $\begin{pmatrix} u_1 \\ \vdots \\ u_n \end{pmatrix} \in \mathbb{R}^n$.
On cherche à minimiser sur $\mathbb{R}^n$

$$\frac{1}{n} \sum_{i=1}^{n} \ell\left(y_i, \hat{g}^{(b)}(x_i) + \eta u_i\right) \qquad \hat{g}^{(b)} = \hat{g}^{(b)} + 0.$$

On fait un pas de descente de gradient.

$$\nabla_u \left(\frac{1}{n} \sum_{i=1}^{n} \ell\left(y_i, \hat{g}^{(b)}(x_i) + \eta u_i\right) \right)$$

$$= \frac{\eta}{n} \cdot \left(\nabla_{y'} \ell\left(y_i, \hat{g}^{(b)}(x_i) + \eta u_i\right) \right) \in \mathbb{R}^n \qquad (y, y') \to \ell(y, y')$$

Le pas de gradient à considérer est dans la direction

$$\delta^{(b+1)} = \frac{\eta}{n} \begin{pmatrix} \nabla_{y'} \ell\left(y_1, \hat{g}^{(b)}(x_1) + \eta u_1\right) \\ \vdots \\ \nabla_{y'} \ell\left(y_n, \hat{g}^{(b)}(x_n) + \eta u_n\right) \end{pmatrix}_0$$

$$= \frac{\eta}{n} \begin{pmatrix} \nabla_{y'} \ell\left(y_1, \hat{g}^{(b)}(x_1)\right) \\ \vdots \\ \nabla_{y'} \ell\left(y_n, \hat{g}^{(b)}(x_n)\right) \end{pmatrix}.$$

Dans le modèle linéaire

$$\ell(y, y') = \frac{1}{2}(y - y')^2$$

$$\nabla_{y'} \ell(y, y') = -(y - y')$$

dans ce cas, le pas de gradient est donné par

$$\delta_{em}^{(b+1)} = \frac{\eta}{n} \begin{pmatrix} -(y_1 - \hat{g}^{(b)}(x_1)) \\ \vdots \\ -(y_n - \hat{g}^{(b)}(x_n)) \end{pmatrix}$$

↖ on remarque que ce sont des résidus de l'itération b.

Le problème : en faisant comme si on n'avait que le pas de gradient est dans $\mathcal{H}$!

$\hat{g}^{(b)} = \hat{g}^{(b)} + 0$ la valeur courante de "ηh" est $\begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}$

dans la descente de gradient

$$\begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} - \text{gradient} = -\delta^{(b+1)} = -\begin{pmatrix} -\delta^{(b+1)}(1) \\ \vdots \\ -\delta^{(b+1)}(n) \end{pmatrix}$$

On voudrait pour "$\hat{\eta}^{(b+1)} \hat{h}^{(b+1)}$" proportionel à $-\delta^{(b+1)}$

si on fait ça, on n'a aucune assurance que $\hat{h}^{(b+1)}$ est dans $\mathcal{H}$.

On cherche $\hat{h}^{(b+1)}$ comme

$$(\hat{h}^{(b+1)}, \hat{\nu}) = \underset{h \in \mathcal{H},\, \nu \in \mathbb{R}}{\operatorname{argmin}} \frac{2}{n} \sum_{i=1}^{n} \left(\underbrace{\delta^{(b+1)}(i)}_{\text{descente idéale}} - \underbrace{\nu h(x_i)}_{\text{descente contrainte à rester dans } \mathcal{H}} \right)^2$$

Slide 47. En régression linéaire $\delta^{(b+1)}(i) = -(y_i - \hat{g}^{(b)}(x_i))$

on peut résoudre

$$\underset{}{\operatorname{argmin}} \frac{2}{n} \sum_{i=1}^{n} \left(-(y_i - \hat{g}^{(b)}(x_i)) - \nu h(x_i) \right)^2$$

Cela revient à chercher la fonction $h \in \mathcal{H}$ qui "fitte" le mieux les résidus de l'étape b.

Adaboost. Slide 52. $\ell(y,u) = \exp(-yu)$

A l'itération b. $\hat{g}^{(b)}$

A l'itération $b+1$ on cherche η et h tels

$$\hat{\eta}^{(b+1)}, \hat{h}^{(b+1)} = \underset{\eta, h}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} \exp\left(-y_i \left(\hat{g}^{(b)}(x_i) + \eta h(x_i)\right)\right)$$

$$= \frac{1}{n} \sum_{i=1}^{n} \exp\left(-y_i \hat{g}^{(b)}(x_i)\right) \exp\left(-y_i \eta h(x_i)\right)$$

$$(*) = \frac{1}{n} \sum_i w^{(b)}(i) \exp\left(-y_i \eta h(x_i)\right)$$

→ risque empirique associé à la perte exponentielle pondéré par les $w^{(b)}(i)$

$w^{(b)}(i) \propto \exp(-y_i \hat{g}^{(b)}(x_i))$

est grand si on a mal classifié à l'étape b

est petit si on a bien classifié.

On a soit $y_i h(x_i) = 1$

$y_i h(x_i) = -1$

On met la contrainte $\frac{1}{n} \sum w^{(b)}(i) = 1$

$$(*) = \frac{1}{n} \left\{ \sum_{i : y_i h(x_i) = 1} w^{(b)}(i) \exp(-\eta) + \sum_{i : y_i h(x_i) = -1} w^{(b)}(i) \exp(\eta) \right\}$$

$$= \frac{1}{n} \left\{ \sum_{i=1}^{n} w^{(b)}(i) \exp(-\eta) - \sum_{i : y_i h(x_i) = -1} w^{(b)}(i) \exp(-\eta) + \sum_{i : y_i h(x_i) = -1} w^{(b)}(i) \exp(\eta) \right\}$$

$$= \exp(-\eta) + \left(\exp(\eta) - \exp(-\eta)\right) \frac{1}{n} \sum_{i : y_i h(x_i) = -1} w^{(b)}(i)$$

Pour la minimisation en h on va chercher

$$\hat{h}^{(b+1)} = \underset{h}{\operatorname{argmin}} \frac{1}{n} \sum_{i : y_i h(x_i) = -1} w^{(b)}(i) = \underset{h}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} w^{(b)}(i) \mathbb{1}(y_i h(x_i) < 0)$$

Donc $\hat{h}^{(b+1)}$ est le learner classifieur qui minimise la perte 0/1 repondérée.

Si on optimise en η, on trouve :

$$\hat{\eta}^{(b+1)} = \frac{1}{2} \log \left(\frac{1 - \varepsilon(b+1)}{\varepsilon(b+1)} \right)$$

où $\varepsilon(b+1) = \sum_{i=1}^{n} w^{(b)}(i) \, \mathbb{1}_{y_i \hat{h}^{(b+1)}(x_i) < 0}$.