IFT-7002 Fondements de l'apprentissage machine

Amplification ("Boosting") d'un apprenant médiocre

Shai Shalev-Shwartz
The Hebrew University of Jerusalem

Traduit et adapté par Mario Marchand
Université Laval

Hiver 2024

Sommaire

Faiblement apprenable

Définition (γ-faiblement-apprenable)

Une classe $\mathcal{H}$ de classificateurs binaires est γ-faiblement-apprenable s'il existe un algorithme d'apprentissage A et une fonction $m_{\mathcal{H}} : (0,1) \to \mathbb{N}$ tels que pour tout $\delta \in (0,1)$, pour tout $m \geq m_{\mathcal{H}}(\delta)$, pour toute distribution $\mathcal{D}$ sur $\mathcal{X}$ et pour tout $f \in \mathcal{H}$, nous avons

$$\mathcal{D}^m(\{S_x : L_{\mathcal{D},f}(A(S)) \leq 1/2 - \gamma\}) \geq 1 - \delta \ .$$

- Presqu'identique au critère PAC (fortement apprenable), sauf qu'il suffit de satisfaire le critère uniquement pour un $\gamma > 0$ spécifique (et non pas pour une erreur ϵ arbitrairement petite).
- Un algorithme qui permet d'apprendre γ-faiblement est un apprenant *médiocre* ("weak learner") effectuant généralement peu de traitement sur S et produisant une hypothèse simple qui performe juste un peu mieux qu'une prédiction aléatoire.

Motivation et problématique

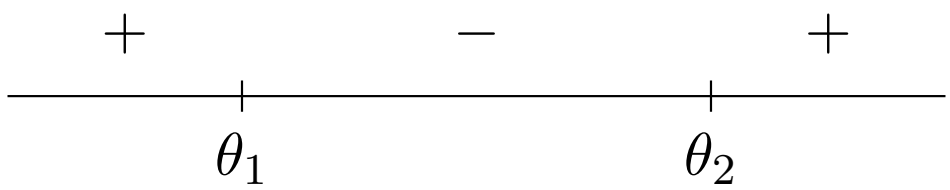
- Si $\mathrm{VCdim}(\mathcal{H}) = d$, la complexité d'échantillon pour apprendre γ-faiblement $\mathcal{H}$ est également donné par le théorème fondamental de l'apprentissage statistique mais avec $\epsilon = 1/2 - \gamma$.
  - Si un algorithme A apprends fortement $\mathcal{H}$, alors A apprends γ-faiblement $\mathcal{H}$.
- Cependant, il existe **possiblement** un algorithme efficace (en temps d'exécution) pouvant apprendre γ-faiblement $\mathcal{H}$ alors qu'il n'existe pas d'algorithme efficace pouvant apprendre fortement $\mathcal{H}$.
- Considérons donc une classe «simple» B d'hypothèses et utilisons l'algorithme ERM_B pour apprendre γ-faiblement un classe $\mathcal{H}$ plus complexe que B. Dans ce cas, B doit satisfaire 2 critères :
  - ERM_B est efficace (s'exécute rapidement).
  - Pour chaque $f \in \mathcal{H}$ et $\mathcal{D}$, $L_{\mathcal{D},f}(\mathrm{ERM}_B(S)) \leq 1/2 - \gamma$ avec probabilité $\geq 1 - \delta$.

Exemple d'un apprenant médiocre ("weak learner")

- Soit $\mathcal{H}$, la classe des 3 régions sur $\mathcal{X} = \mathbb{R}$., e.g.

$$\mathcal{H} \stackrel{\text{def}}{=} \{h_{\theta_1,\theta_2,b} : (\theta_1, \theta_2) \in \mathbb{R}^2, \theta_1 < \theta_2, b \in \{-1, +1\}\}$$

t.q. $\forall\, x \in \mathbb{R}$, $h_{\theta_1,\theta_2,b}(x) = -b$ si $\theta_1 \leq x \leq \theta_2$ et $+b$ autrement.



- Soit $B = \{x \mapsto b \cdot \text{sign}(x - \theta) : \ \theta \in \mathbb{R}, b \in \{\pm 1\}\}$ la classe des souches de décisions (avec 2 directions possibles).
- Affirmation : Cette classe $\mathcal{H}$ est $(1/12)$-faiblement-apprenable à l'aide de $\text{ERM}_B(S)$

Exemple d'un apprenant médiocre ("weak learner")

- **Preuve:**
  - Notez que pour tout $\mathcal{D}$ et pour tout $f_{\theta_1,\theta_2,b} \in \mathcal{H}$, il existe toujours une région R parmi les 3 régions tel que $\mathcal{D}(R) \leq 1/3$.
  - Or, pour tout $f_{\theta_1,\theta_2,b} \in \mathcal{H}$ et pour tout choix de 2 régions, il existe toujours une souche de décision en accord avec ces 2 régions.
  - Donc, pour tout $\mathcal{D}$ et pour tout $f_{\theta_1,\theta_2,b} \in \mathcal{H}$, il existe donc toujours une souche de décision $h \in B$ telle que $L_{\mathcal{D},f}(h) \leq 1/3$.
  - Puisque $\mathrm{VCdim}(B) = d = 2$, le théorème fondamental nous dit qu'il existe $m_B(\epsilon,\delta) \in O(\frac{d+\log(1/\delta)}{\epsilon^2})$ tel que si $m \geq m_B(\epsilon,\delta)$, alors $L_{\mathcal{D},f}(\mathrm{ERM}_B(S)) \leq 1/3 + \epsilon$ avec prob. $\geq 1-\delta$.
  - Donc, si on choisi $\epsilon = 1/12$, on a $1/3 + \epsilon = 5/12 = 1/2 - 1/12$ et, dans ce cas, $L_{\mathcal{D},f}(\mathrm{ERM}_B(S)) \leq 1/2 - 1/12$ avec prob. $\geq 1-\delta$ lorsque $m \geq m_B(\delta) = m_B(1/12,\delta) \in O(\log(1/\delta))$. ■

Sommaire

Le problème fondamental du “boosting”

- Supposons que nous ayons un algorithme *efficace* A pouvant γ-faiblement apprendre une classe $\mathcal{H}$.
- Pouvons-nous alors utiliser A pour obtenir un algorithme *efficace* qui peut PAC-apprendre (fortement) $\mathcal{H}$?
  - i.e., pouvons-nous *amplifier* A pour PAC-apprendre $\mathcal{H}$ efficacement ?
- La réponse est : **oui !**

Historique du problème

Problem raised in 1988 by Kearns and Valiant

Solved in 1990 by Robert Schapire, then a graduate student at MIT

In 1995, Schapire & Freund proposed the AdaBoost algorithm

Sommaire

Stratégie d'apprentissage d'AdaBoost

- AdaBoost exécute un nombre T d'itérations. À chaque itération t, il utilise une distribution $\mathbf{D}^{(t)}$ sur l'échantillon S de m exemples.
  - $D_i^{(t)}$ est le poids sur le i-ème exemple de S. On a $\sum_{i=1}^{m} D_i^{(t)} = 1 \, \forall t$.
- À chaque itération t, AdaBoost exécute un apprenant médiocre WL sur S, pondéré par $\mathbf{D}^{(t)}$, pour obtenir un classificateur h_t.
  - Donc si WL est un apprenant γ-faible (pour un $\mathcal{H}$ complexe), on aura $L_{\mathbf{D}^{(t)},f}(h_t) \leq 1/2 - \gamma$ avec prob. $\geq 1 - \delta \; \forall \, f \in \mathcal{H}$.
- À la fin de chaque itération t, AdaBoost met à jour $\mathbf{D}^{(t)}$ afin que $\mathbf{D}^{(t+1)}$ ait plus de poids sur les exemples mal classifiés par h_t.
  - Il assigne aussi un poids w_t à h_t qu'il utilisera à la fin.
- L'hypothèse finale h_s produite par AdaBoost est un vote de majorité de $h_1, \ldots, h_T$ pondéré par $w_1, \ldots, w_T$.
- On verra que $L_S(h_s)$ diminue exponentiellement en fonction de T si, à chaque itération t, WL retourne h_t t.q. $L_{\mathbf{D}^{(t)},f}(h_t) \leq 1/2 - \gamma$.
- On démontrera ensuite qu'AdaBoost PAC-apprends (fortement) $\mathcal{H}$ si WL est un apprenant γ-faible pour $\mathcal{H}$.

L'algorithme AdaBoost ("adpative boosting")

- **Entrée :** un échantillon $S = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ avec $y_i \in \{-1, +1\}$, un apprenant médiocre WL, un nombre T d'itérations.
- **Initialiser :** $\mathbf{D}^{(1)} = (\frac{1}{m}, \ldots, \frac{1}{m})$
- **pour** $t = 1, \ldots, T$:
  - Appeler l'apprenant médiocre : $h_t = \mathrm{WL}(\mathbf{D}^{(t)}, S)$.
  - Calculer : $\epsilon_t = L_{\mathbf{D}^{(t)}}(h_t) = \sum_{i=1}^{m} D_i^{(t)} \mathbb{1}_{[y_i \neq h_t(x_i)]}$
  - Soit $w_t = \frac{1}{2} \log\left(\frac{1}{\epsilon_t} - 1\right)$
  - Mise à jour : $D_i^{(t+1)} = \frac{D_i^{(t)} \exp(-w_t y_i h_t(x_i))}{\sum_{j=1}^{m} D_j^{(t)} \exp(-w_t y_j h_t(x_j))}, \ \forall\, i = 1, \ldots, m$
- **Sortie :** l'hypothèse $h_s(x) = \mathrm{sign}\left(\sum_{t=1}^{T} w_t h_t(x)\right)$.

Remarques sur AdaBoost

- Notez que $w_t < 0$ lorsque $\epsilon_t > 1/2$.
- Donc AdaBoost donne un poids négatif à un votant h_t très mauvais.
- Cela donne le même effet que de donner un poids positif à $-h_t$ (qui lui à forcément un $\epsilon_t < 1/2$).
- Quoi faire si WL retourne un h_t avec $\epsilon_t = 1/2$?
  - (Notez que s'il $\exists h_t : \epsilon_t > 1/2$, alors $-h_t$ donne $\epsilon_t < 1/2$)
- Réponse : Dans ce cas on aura $w_t = 0$ et $\mathbf{D}^{(t+1)} = \mathbf{D}^{(t)}$. AdaBoost ne pourra donc plus progresser.
  - Si on n'est pas satisfait de h_s, il faut se choisir un autre WL et recommencer.

AdaBoost force WL à se concentrer sur les exemples incorrectement classifiés

- Affirmation : L'erreur de h_t par rapport à $\mathbf{D}^{(t+1)}$ est exactement $1/2$
- **Preuve:** :

$$\sum_{i=1}^{m} D_i^{(t+1)} \mathbb{1}_{[y_i \neq h_t(x_i)]} = \frac{\sum_{i=1}^{m} D_i^{(t)} e^{-w_t y_i h_t(x_i)} \mathbb{1}_{[y_i \neq h_t(x_i)]}}{\sum_{j=1}^{m} D_j^{(t)} e^{-w_t y_j h_t(x_j)}}$$

$$= \frac{e^{w_t} \epsilon_t}{e^{w_t} \epsilon_t + e^{-w_t}(1-\epsilon_t)} = \frac{\epsilon_t}{\epsilon_t + e^{-2w_t}(1-\epsilon_t)}$$

$$= \frac{\epsilon_t}{\epsilon_t + \frac{\epsilon_t}{1-\epsilon_t}(1-\epsilon_t)} = \frac{1}{2} .$$

Théorème (de convergence d'AdaBoost)

Si, à chaque itération t, $WL(\mathbf{D}^{(t)}, S)$ retourne h_t avec $\epsilon_t \leq 1/2 - \gamma$ pour $\gamma > 0$, alors

$$L_S(h_s) = \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}_{[y_i \neq h_s(x_i)]} \leq \exp(-2\,\gamma^2\,T) \quad .$$

Remarques :

- Donc, si $T \geq \frac{\log(1/\epsilon)}{2\gamma^2}$, alors on aura $L_S(h_s) \leq \epsilon$.
- Donc $L_S(h_s) = 0$ lorsque $\epsilon < 1/m$, i.e., lorsque $T > \frac{\log m}{2\gamma^2}$.
- Donc si WL satisfait l'énoncé du théorème pour tout $[m] \stackrel{\text{def}}{=} \{1, \ldots, m\}$, pour tout $\{x_1, \ldots, x_m\}$, pour tout $\mathbf{D}^{(t)}$ sur $[m]$ et pour tout $f \in \mathcal{H}$ étiquetant les m instances, cela signifie que la classe des hypothèses retournées par AdaBoost avec ce WL englobe $\mathcal{H}$.

Remarques (suite)

- Si, contrairement à l'énoncé du théorème, on permet à WL d'échouer avec prob. $\leq \delta_t$ à chaque itération t, cela signifie que pour tout $[m]$, pour tout $S \stackrel{\text{def}}{=} \{z_1, \ldots, z_m\}$, pour tout $\mathbf{D}^{(t)}$ sur $[m]$, on a

$$\mathbb{P}\left(\sum_{i=1}^{m} D_i^{(t)} \mathbb{1}_{[h_t(x_i) \neq y_i]} > 1/2 - \gamma\right) \leq \delta_t \,,$$

  où la probabilité est sur les tirages aléatoires effectués par $\text{WL}(S, \mathbf{D}^{(t)})$ pour retourner h_t.
  - Dans ce cas, WL est forcément stochastique (ex : WL tire m' exemples selon $\mathbf{D}^{(t)}$ et minimise le risque empirique sur ces m' exemples).
- Si WL échoue avec probabilité $\leq \delta_t$, il échouera à chacune des K tentatives avec probabilité $\leq \delta_t^K$.
- Donc, selon la borne de l'union, WL, avec K tentatives, échouera à l'une des T itérations avec probabilité $\leq T\delta_t^K \stackrel{\text{def}}{=} \delta$.
- WL avec K tentatives n'échouera sur aucune des T itérations avec probabilité $\geq 1 - \delta$ (qui est très près de 1 pour K élevé).

Preuve du théorème de convergence d'AdaBoost

Preuve: : Soit $f_t(x) \stackrel{\text{def}}{=} \sum_{p \leq t} w_p h_p(x)$ $\forall t \in \{0, \ldots, T\}$, et $f_0(x) \stackrel{\text{def}}{=} 0$. Alors $h_s(x) = \text{sign}(f_T(x))$.

$$\text{Soit } Z_t \stackrel{\text{def}}{=} \frac{1}{m} \sum_{i=1}^{m} e^{-y_i f_t(x_i)}, \ \forall t \in \{0, \ldots, T\}.$$

Puisque $\mathbb{1}_{[y \neq \text{sign}(f_T(x))]} \leq \mathbb{1}_{[y f_T(x) \leq 0]} \leq e^{-y f_T(x)}$, il suffit de montrer que $Z_T \leq e^{-2\gamma^2 T}$. Or

$$Z_T = \frac{Z_T}{Z_0} = \frac{Z_T}{Z_{T-1}} \cdot \frac{Z_{T-1}}{Z_{T-2}} \cdots \frac{Z_2}{Z_1} \cdot \frac{Z_1}{Z_0}$$

car $Z_0 = 1$. Il suffit alors de montrer que pour tout t, on a

$$\frac{Z_{t+1}}{Z_t} \leq e^{-2\gamma^2}.$$

Pour cela, notez que par induction, nous avons que

$$D_i^{(t+1)} = \frac{e^{-y_i f_t(x_i)}}{\sum_{j=1}^{m} e^{-y_j f_t(x_j)}}.$$

Alors

$$
\begin{aligned}
\frac{Z_{t+1}}{Z_t} &= \frac{\sum_{i=1}^{m} e^{-y_i f_{t+1}(x_i)}}{\sum_{j=1}^{m} e^{-y_j f_t(x_j)}} \\
&= \frac{\sum_{i=1}^{m} e^{-y_i f_t(x_i)} e^{-y_i w_{t+1} h_{t+1}(x_i)}}{\sum_{j=1}^{m} e^{-y_j f_t(x_j)}} \\
&= \sum_{i=1}^{m} D_i^{(t+1)} e^{-y_i w_{t+1} h_{t+1}(x_i)} \\
&= e^{-w_{t+1}} \sum_{i:y_i h_{t+1}(x_i)=1} D_i^{(t+1)} \; + \; e^{w_{t+1}} \sum_{i:y_i h_{t+1}(x_i)=-1} D_i^{(t+1)} \\
&= e^{-w_{t+1}} (1 - \epsilon_{t+1}) \; + \; e^{w_{t+1}} \epsilon_{t+1} \\
&= \sqrt{\frac{\epsilon_{t+1}}{1 - \epsilon_{t+1}}} (1 - \epsilon_{t+1}) \; + \; \sqrt{\frac{1 - \epsilon_{t+1}}{\epsilon_{t+1}}} \epsilon_{t+1} \\
&= 2\sqrt{\epsilon_{t+1}(1 - \epsilon_{t+1})}
\end{aligned}
$$

Puisque $\epsilon_{t+1} \leq 1/2 - \gamma$ et que $x(1-x)$ est monotone croissante pour $x \in [0, 1/2]$, nous avons

$$2\sqrt{\epsilon_{t+1}(1-\epsilon_{t+1})} \;\leq\; 2\sqrt{\left(\frac{1}{2}-\gamma\right)\left(\frac{1}{2}+\gamma\right)} \;=\; \sqrt{1-4\gamma^2}\,.$$

Puisque $1 - x \leq e^{-x}$, nous avons

$$\sqrt{1-4\gamma^2} \;\leq\; e^{-4\gamma^2/2} \;=\; e^{-2\gamma^2}\,.$$

Donc

$$\frac{Z_{t+1}}{Z_t} \leq e^{-2\gamma^2}$$

comme souhaité. ■

Du risque empirique d'AdaBoost vers son vrai risque

- Le *risque empirique* $L_S(h_s)$ de l'hypothèse h_s retourné par AdaBoost tends vers zéro avec T (lorsque WL satisfait la condition du théorème) .
- Par contre, ce qui nous intéresse, c'est son *vrai risque* $L_{\mathcal{D}}(h_s)$.
- Le théorème fondamentale de l'apprentissage nous dit que $L_{\mathcal{D}}(h_s)$ dépends de la VCdim de la classe des fonctions pouvant être produites par AdaBoost.
- AdaBoost construit un demi-espace (ou vote de majorité) sur les prédicteurs h_t produits par l'apprenant médiocre WL qui choisit ses prédicteurs dans une classe B tel que $\mathrm{VCdim}(B)$ est petit.
- Examinons donc comment la richesse de cette classe, et sa VCdim, varie en fonction de B et du nombre T d'itérations.

Sommaire

La classes $L(B,T)$ des hypothèses produites par AdaBoost

- L'apprenant médiocre WL retourne des hypothèses d'une classe B de complexité (i.e., $\mathrm{VCdim}(B) = d'$) limitée.
- AdaBoost, par contre, produit des hypothèses dans la classe $L(B,T)$:

$$L(B,T) \stackrel{\text{def}}{=} \left\{ x \mapsto \operatorname{sign}\left(\sum_{t=1}^{T} w_t h_t(x) \right) : \mathbf{w} \in \mathbb{R}^T, \quad h_t \in B, \ \forall t \right\} .$$

- Puisque WL est utilisé uniquement sur des distributions sur S, on peut supposer s.p.d.g. que $B = \{g_1, \ldots, g_d\}$ pour $d \leq \left(\frac{em}{d'}\right)^{d'}$.
- Soit $\boldsymbol{\psi}(x) \stackrel{\text{def}}{=} (g_1(x), \ldots, g_d(x))$. Alors :

$$L(B,T) = \left\{ x \mapsto \operatorname{sign}\left(\langle \mathbf{w}, \boldsymbol{\psi}(x) \rangle \right) : \mathbf{w} \in \mathbb{R}^d, \ \|\mathbf{w}\|_0 \leq T \right\} ,$$

  où $\|\mathbf{w}\|_0 \stackrel{\text{def}}{=} |\{i : w_i \neq 0\}|$.
- i.e., le demi-espace construit par AdaBoost est représenté par un vecteur $\mathbf{w}$ parcimonieux avec au plus T composantes non nulles sur le vecteur $\boldsymbol{\psi}(x)$ de d composantes avec $d \gg T$.

L'expressivité de $L(B, T)$

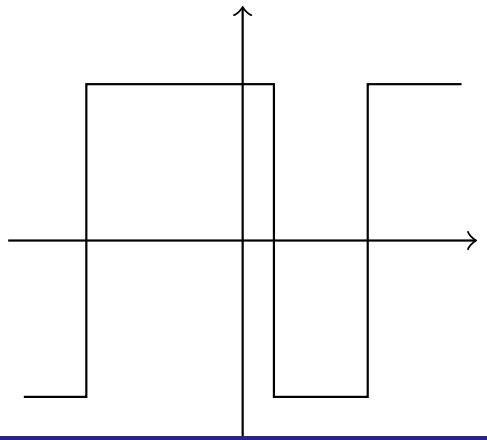
- Considérez $\mathcal{X} = \mathbb{R}$, et $B =$ la classe des souches de décision,

$$B = \left\{ x \mapsto b \cdot \text{sign}(x - \theta) : \ \theta \in \mathbb{R}, b \in \{\pm 1\} \right\} .$$

- Soit $\mathcal{G}_T$, le classe (riche) des fonctions : $\mathbb{R} \to \{-1, +1\}$ qui sont des constantes par morceaux avec T morceaux,

$$\mathcal{G}_T \overset{\text{def}}{=} \left\{ x \mapsto \sum_{i=1}^{T} \alpha_i \mathbb{1}_{[x \in (\theta_{i-1}, \theta_i]]} : \ \alpha_i \in \{\pm 1\} \, \forall i, \right.$$

$$\left. -\infty = \theta_0 < \theta_1 < \ldots < \theta_T = +\infty, \right\} .$$



L'expressivité de $L(B,T)$

- Or, ici on a

$$L(B,T) = \left\{ x \mapsto \text{sign}\left(\sum_{i=1}^{T} w_i \, \text{sign}(x - \theta_i) \right) \right\} \ ,$$

où l'on a remplacé $b_i w_i$ par w_i (w_i pouvant être négatif).

- Affirmation : $\mathcal{G}_T \subseteq L(B,T)$.

Preuve:

- Nous allons montrer que pour toute fonction de $g_{\boldsymbol{\alpha}} \in \mathcal{G}_T$, représentée par un vecteur $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_T)$ et un vecteur $(\theta_1, \ldots, \theta_{T-1})$, il existe une fonction de $f_{\mathbf{w}} \in L(B,T)$ représentée par un vecteur $\mathbf{w} = (w_1, \ldots, w_T)$ et un vecteur $(\theta_1, \ldots, \theta_{T-1})$ identique à celui utilisé par $g_{\boldsymbol{\alpha}}$ telle que $f_{\mathbf{w}}(x) = g_{\boldsymbol{\alpha}}(x) \, \forall x \in \mathbb{R}$.

- L'égalité de ces deux fonctions nous donne le système suivant d'équations à résoudre

$$\begin{aligned}
\text{cas } x \in (-\infty, \theta_1] &: -w_1 - w_2 \ldots - w_T = \alpha_1 \\
\text{cas } x \in (\theta_1, \theta_2] &: +w_1 - w_2 \ldots - w_T = \alpha_2 \\
\ldots &: \ldots \\
\text{cas } x \in (\theta_{T-1}, \theta_T] &: +w_1 + w_2 \ldots - w_T = \alpha_T
\end{aligned}$$

- Nous donnant le système linéaire $A\mathbf{w} = \boldsymbol{\alpha}$ avec

$$A = \begin{pmatrix} -1 & -1 & \ldots & -1 \\ +1 & -1 & \ldots & -1 \\ +1 & +1 & \ldots & -1 \\ \vdots & \vdots & \ddots & \vdots \\ +1 & +1 & \ldots & -1 \end{pmatrix},$$

- qui admet toujours une solution (pour tout $\boldsymbol{\alpha}$) puisque les vecteurs colonnes de A sont linéairement indépendants. ■

La composition de demi-espaces sur une classe simple peut donc donner un ensemble très varié de fonctions !

L'expressivité de $L(B,T)$

Théorème

$$d = \mathrm{VCdim}(B) \implies \mathrm{VCdim}(L(B,T)) \in O\left([T\,d]\log[T\,d]\right)$$

Preuve: :

- Soit $d \stackrel{\text{def}}{=} \mathrm{VCdim}(B)$, $m \stackrel{\text{def}}{=} \mathrm{VCdim}(L(B,T))$, et C un ensemble de m points de $\mathcal{X}$ pulvérisé par $L(B,T)$.
- Chaque fonction de $L(B,T)$ sur C est réalisée en choisissant $h_1,\ldots,h_T$ dans B et en appliquant un demi-espace sur ces T votants.
- Le lemme de Sauer nous dit que la classe B peut réaliser au plus $(em/d)^d$ fonctions Booléennes sur C (lorsque $m \geq d$).
- Il faut donc choisir les T votants parmi ces $(em/d)^d$ fonctions. Il y a donc au plus $(em/d)^{dT}$ ensembles distincts de T votants.
- Pour chaque ensemble de T votants, on peut réaliser au plus $(em/T)^T$ demi-espaces homogènes (car la VCdim des demi-espaces homogènes sur T variables est égale à T).

- $L(B,T)$ peut donc réaliser un nombre de fonctions distinctes sur C qui est au plus de

$$(em/d)^{dT}\,(em/T)^{T} \;<\; m^{(d+1)T}$$

lorsque $T > e$ et $d > e$.

- Puisque C est pulvérisé, ce nombre de fonctions doit être au moins 2^m. Alors on doit avoir $2^m \;\leq\; m^{(d+1)T}$, *i.e.*,

$$m \;\leq\; \frac{(d+1)T}{\log(2)}\log(m)\,.$$

- Or, selon le Lemme A.1 : $m \geq 2a\log(a) \implies m \geq a\log(m)$. Donc pour avoir $m \leq a\log(m)$ il est nécessaire d'avoir $m \leq 2a\log(a)$. Donc, $m = \mathrm{VCdim}(L(B,T))$ doit satisfaire

$$m \;\leq\; \frac{2(d+1)T}{\log(2)}\log\left(\frac{(d+1)T}{\log(2)}\right)\,.$$

La borne de l'intersection

À partir de la prochaine diapositive, j'utiliserai à l'occasion ce théorème.

Théorème (La borne de l'intersection)

Soit A et B deux évènements tels que $\mathbb{P}(A) \geq 1 - \delta_A$ et $\mathbb{P}(B) \geq 1 - \delta_B$. Alors

$$\mathbb{P}(A \cap B) \geq 1 - (\delta_A + \delta_B)\,.$$

Preuve: Selon l'hypothèse du théorème, on a $\mathbb{P}(\overline{A}) \leq \delta_A$ et $\mathbb{P}(\overline{B}) \leq \delta_B$.

Selon la borne de l'union, on a alors $\mathbb{P}(\overline{A} \cup \overline{B}) \leq \delta_A + \delta_B$.

Selon la loi de De Morgan, on a alors $\mathbb{P}(A \cap B) \geq 1 - (\delta_A + \delta_B)$. ■

AdaBoost apprends les classes faiblement apprenables

- Soit WL utilisant B, dont $\mathrm{VCdim}(B) = d$, un apprenant γ-faible pour un $\mathcal{H}$ et que les étiquettes de S sont générées par un $f \in \mathcal{H}$.
- Soit $h_s =$ l'hypothèse retournée par AdaBoost.
- Lorsque $T > \frac{\log m}{2\gamma^2}$, on a vu que $L_S(h_s) = 0$, avec prob. 1 lorsque WL n'échoue jamais.
- Si WL échoue avec probabilité $\leq \delta_t$ à chaque itération $t \in [T]$, remplaçons WL par WL avec K tentatives tel que $T\delta_t^K \leq \delta_1$. On a alors $L_S(h_s) = 0$ avec prob. $\geq 1 - \delta_1$
- De plus, $\mathrm{VCdim}(L(B, T)) \in O(dT \log(dT))$.
- Alors, selon le thm fondamental, $L_{\mathcal{D},f}(h_s) \leq \epsilon$ si $L_S(h_s) = 0$ avec prob. $\geq 1 - \delta_2$ lorsque $m \geq C_2[(dT \log(dT)) + \log(1/\delta_2)]/\epsilon$.
- Alors, avec probabilité $\geq 1 - (\delta_1 + \delta_2) = 1 - \delta$ (pour $\delta_1 = \delta_2 = \delta/2$), on a que $L_S(h_s) = 0$ et $L_{\mathcal{D},f}(h_s) \leq \epsilon$ lorsque $T > \frac{\log m}{2\gamma^2}$ et

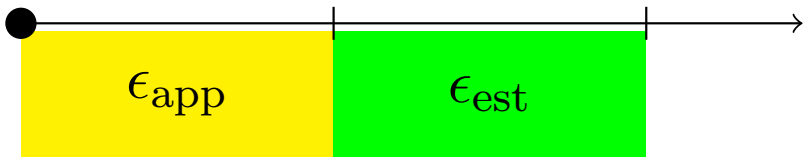
$$m \geq \frac{C_2[(dT \log(dT)) + \log(2/\delta)]}{\epsilon}.$$

- AdaBoost apprends $\mathcal{H}$ au sens PAC avec des demi-espaces sur B.

Sommaire

Compromis biais-complexité

Rappel sur la décomposition de l'erreur : $L_{\mathcal{D}}(h_s) = \epsilon_{\text{app}} + \epsilon_{\text{est}}$



$$\epsilon_{\text{app}} = \min_{h \in L(B,T)} L_D(h) \quad ; \quad \epsilon_{\text{est}} = L_D(h_s) - \epsilon_{\text{app}}$$

- Nous avons montré que $\text{VCdim}(L(B,T)) \in O(dT \log(dT))$.
- Donc, l'erreur d'approximation décroit avec T.
- Et l'erreur d'estimation croît avec T.
- Le paramètre T d'AdaBoost nous permet donc de choisir le bon compromis biais-complexité (afin de trouver un h_s avec $L_{\mathcal{D}}(h_s)$ minimal).
  - Nous pouvons utiliser les méthodes proposées pour la validation et la sélection de modèle afin de trouver T.

Sommaire

Détection de visages

- Prédire si un rectangle dans une image entoure un visage ou non.

Apprenant médiocre pour la détection de visages

Quelques règles approximatives :

- “La région des yeux est souvent plus sombre que celle des joues”
- “Le haut du nez est souvent plus clair que les yeux”

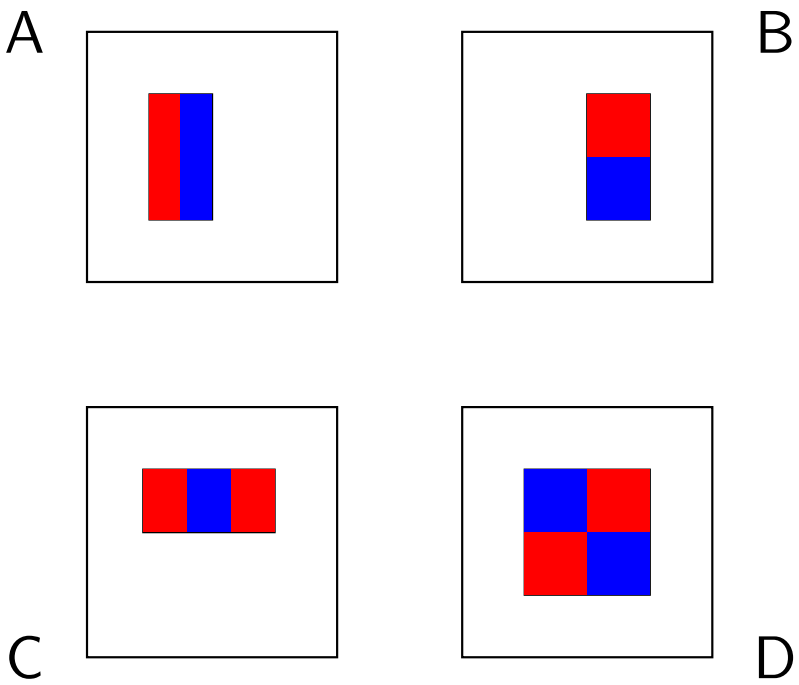
Objectifs :

- Combinons quelques unes de ces règles approximatives pour obtenir un détecteur de visages.
- Le classificateur final devrait être parcimonieux pour qu’il possède une faible erreur d’estimation et qu’il soit rapide à exécuter.

Apprenant médiocre pour la détection de visages

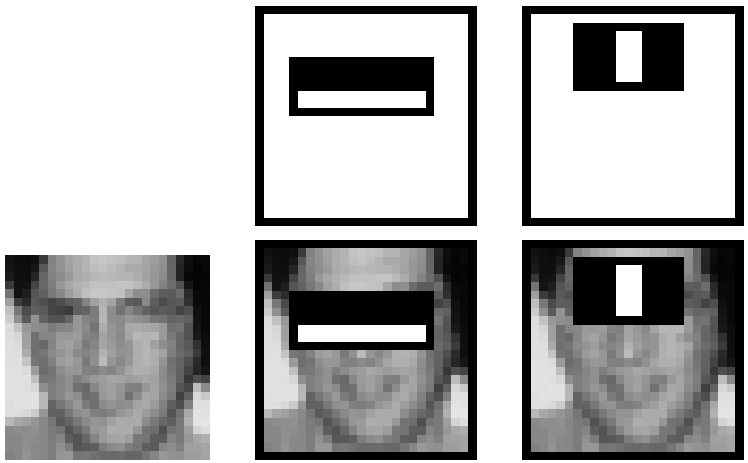
Chaque hypothèse de B est de la forme $h(x) = f(g(x))$, où f est une souche de décision et $g : \mathbb{R}^{24,24} \to \mathbb{R}$ est paramétrisé par :

- Un rectangle R aligné aux axes : Puisque chaque image est constituée de 24×24 pixels, il y a au plus 24^4 rectangles alignés aux axes.
- Un type, $t \in \{A, B, C, D\}$: Chaque type correspond à un *filtre* :
  - $g = \sum$ (pixels région bleu) $- \sum$ (pixels région rouge)



AdaBoost pour la détection de visages

- Cet ensemble de caractéristiques (“features”) a été proposé par Viola et Jones (2001) et a été très utilisée pour la reconnaissance de visages, de personnes, et d’objets.
- Les deux premières règles sélectionnées par AdaBoost capitalisent sur le fait que la région des yeux est habituellement plus foncée que celle des joues et celle du haut du nez.



Résumé

- Apprendre une classe γ-faiblement
- Le problème fondamental du “boosting”
- AdaBoost permet d'apprendre (fortement) les classes faiblement apprenables
- La puissance de composer des demi-espaces avec des classes simples
- AdaBoost fonctionne en pratique pour plusieurs applications !