De la théorie de la compression d'échantillons aux algorithmes de méta-apprentissage

(Adaptation d'une présentation de Pascal Germain)

Benjamin Leblanc
https://benthewhite.github.io/
Université Laval, département d'informatique et de génie logiciel

22 mars 2024

Plan

1 Préambules

2 Théorie de la compression des données
- Définition et exemples
- Borne de généralisation pour classificateurs binaires
- Nouvelle borne de généralisation

3 Deep Reconstruction Machine
- Le paradigme du méta-apprentissage
- Résultats préliminaires - classification binaire

Définitions

Une **observation** $z := (x, y) \in \mathcal{X} \times \mathcal{Y} = \mathcal{Z}$ est une paire **variables explicatives - étiquette**.

Définitions

Une **observation** $z := (x, y) \in \mathcal{X} \times \mathcal{Y} = \mathcal{Z}$ est une paire **variables explicatives - étiquette**.

Distribution génératrice des données

Chaque observation provient d'une **distribution** D sur $\mathcal{Z}$.

Définitions

Une **observation** $z := (x, y) \in \mathcal{X} \times \mathcal{Y} = \mathcal{Z}$ est une paire **variables explicatives - étiquette**.

Distribution génératrice des données

Chaque observation provient d'une **distribution** D sur $\mathcal{Z}$.

Ensemble d'apprentissage

$$S \ := \ \{\ z_1,\ z_2,\ \ldots,\ z_n\ \} \ \sim \ D^n$$

Définitions

Une **observation** $z := (x, y) \in \mathcal{X} \times \mathcal{Y} = \mathcal{Z}$ est une paire **variables explicatives - étiquette**.

Distribution génératrice des données

Chaque observation provient d'une **distribution** D sur $\mathcal{Z}$.

Ensemble d'apprentissage

$$S := \{ z_1, z_2, \ldots, z_n \} \sim D^n$$

Algorithme d'apprentissage

$$A(S) \longrightarrow h$$

Définitions

Une **observation** $z := (x, y) \in \mathcal{X} \times \mathcal{Y} = \mathcal{Z}$ est une paire **variables explicatives - étiquette**.

Distribution génératrice des données

Chaque observation provient d'une **distribution** D sur $\mathcal{Z}$.

Ensemble d'apprentissage

$$S \;:=\; \{\, z_1,\; z_2,\; \ldots,\; z_n \,\} \;\sim\; D^n$$

Algorithme d'apprentissage

$$A(S) \longrightarrow h$$

Prédicteur (où hypothèse)

$$h : \mathcal{X} \to \mathcal{Y}, \quad h \in \mathcal{H}$$

Définitions

Une **observation** $z := (x, y) \in \mathcal{X} \times \mathcal{Y} = \mathcal{Z}$ est une paire **variables explicatives - étiquette**.

Distribution génératrice des données

Chaque observation provient d'une **distribution** D sur $\mathcal{Z}$.

Ensemble d'apprentissage

$$S \;:=\; \{\, z_1,\; z_2,\; \ldots,\; z_n \,\} \;\sim\; D^n$$

Algorithme d'apprentissage

$$A(S) \longrightarrow h$$

Prédicteur (où hypothèse)

$$h : \mathcal{X} \to \mathcal{Y}, \quad h \in \mathcal{H}$$

Fonction de perte

$$\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$$

Exemple

Tâche : prédire le loyer mensuel d'un appartement dans la ville de Québec

Exemple

Tâche : prédire le loyer mensuel d'un appartement dans la ville de Québec

S

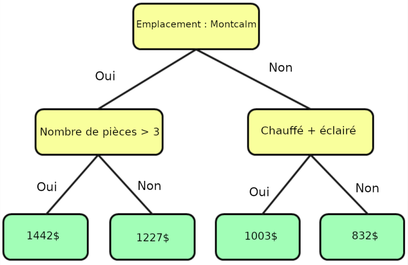
ID	x_1 Emplacement	x_2 Nombre de pièce	x_3 Chauffé + Éclairé	$\dots$	x_d Étage	y Prix
1	Montcalm	3 1/2	Oui	$\dots$	3e	1000
2	Montcalm	4 1/2	Oui	$\dots$	1er	1230
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
n	Sainte-Foy	2 1/2	Non	$\dots$	2e	829

Tâche : prédire le loyer mensuel d'un appartement dans la ville de Québec

S

ID	x_1 Emplacement	x_2 Nombre de pièce	x_3 Chauffé + Éclairé	$\dots$	x_d Étage	y Prix
1	Montcalm	3 1/2	Oui	$\dots$	3e	1000
2	Montcalm	4 1/2	Oui	$\dots$	1er	1230
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
n	Sainte-Foy	2 1/2	Non	$\dots$	2e	829

$$A(S) = h$$



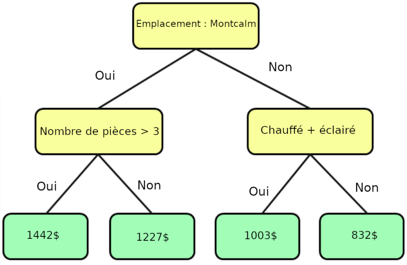
Exemple

Tâche : prédire le loyer mensuel d'un appartement dans la ville de Québec

S

ID	x_1 Emplacement	x_2 Nombre de pièce	x_3 Chauffé + Éclairé	$\cdots$	x_d Étage	y Prix
1	Montcalm	3 1/2	Oui	$\cdots$	3e	1000
2	Montcalm	4 1/2	Oui	$\cdots$	1er	1230
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
n	Sainte-Foy	2 1/2	Non	$\cdots$	2e	829

$A(S) = h$



$\mathcal{H}$: les différents arbres de décision qui existent

Le problème de la généralisation

L'algorithme d'apprentissage se sert uniquement de l'ensemble d'entrainement S :

$$\widehat{\mathcal{L}}_S(h) := \frac{1}{n}\sum_{i=1}^{n} \ell(h(\mathbf{x}_i), y_i)$$

Le problème de la généralisation

L'algorithme d'apprentissage se sert uniquement de l'ensemble d'entrainement S :

$$\widehat{\mathcal{L}}_S(h) := \frac{1}{n}\sum_{i=1}^{n} \ell(h(\mathbf{x}_i), y_i)$$

Le but : Minimiser la perte en moyenne sur D

$$\mathcal{L}_D(h) := \mathop{\mathbf{E}}_{z\sim D} \ell(h(\mathbf{x}_i), y_i)$$

Le problème de la généralisation

L'algorithme d'apprentissage se sert uniquement de l'ensemble d'entrainement S :

$$\widehat{\mathcal{L}}_S(h) := \frac{1}{n}\sum_{i=1}^{n} \ell(h(\mathbf{x}_i), y_i)$$

Le but : Minimiser la perte en moyenne sur D

$$\mathcal{L}_D(h) := \mathop{\mathbf{E}}_{z\sim D} \ell(h(\mathbf{x}_i), y_i)$$

Bornes de généralisation PAC (Probablement Approximativement Correct)

Le problème de la généralisation

L'algorithme d'apprentissage se sert uniquement de l'ensemble d'entrainement S :

$$\widehat{\mathcal{L}}_S(h) := \frac{1}{n}\sum_{i=1}^{n} \ell(h(\mathbf{x}_i), y_i)$$

Le but : Minimiser la perte en moyenne sur D

$$\mathcal{L}_D(h) := \mathop{\mathbf{E}}_{z\sim D} \ell(h(\mathbf{x}_i), y_i)$$

Bornes de généralisation PAC (Probablement Approximativement Correct)

« Avec probabilité au moins "$1-\delta$", la perte de h sera au plus "$\varepsilon(\cdot, \ldots, \cdot)$" »

Le problème de la généralisation

L'algorithme d'apprentissage se sert uniquement de l'ensemble d'entrainement S :

$$\widehat{\mathcal{L}}_S(h) := \frac{1}{n}\sum_{i=1}^{n} \ell(h(\mathbf{x}_i), y_i)$$

Le but : Minimiser la perte en moyenne sur D

$$\mathcal{L}_D(h) := \mathop{\mathbf{E}}_{z\sim D} \ell(h(\mathbf{x}_i), y_i)$$

Bornes de généralisation PAC (Probablement Approximativement Correct)

≪ Avec probabilité au moins "$1-\delta$", la perte de h sera au plus "$\varepsilon(\cdot, \ldots, \cdot)$" ≫

$$\Pr_{S\sim D^n}\Big(\mathcal{L}_D(h) \leq \varepsilon(\widehat{\mathcal{L}}_S(h), n, \delta, \ldots)\Big) \geq 1-\delta$$

Apprentissage PAC - En bref

PAC (Probablement Approximativement Correct)

Avec probabilité au moins "$1-\delta$", la perte de h sera au plus "$\varepsilon(\cdot, \dots, \cdot)$"

$$\Pr_{S\sim D^n}\Big(\mathcal{L}_D(h) \leq \varepsilon(\widehat{\mathcal{L}}_S(h), n, \delta, \dots)\Big) \geq 1-\delta$$

- Hypothèse unique h :

Apprentissage PAC - En bref

PAC (Probablement Approximativement Correct)

Avec probabilité au moins "$1-\delta$", la perte de h sera au plus "$\varepsilon(\cdot, \ldots, \cdot)$"

$$\Pr_{S \sim D^n}\left(\mathcal{L}_D(h) \leq \varepsilon(\widehat{\mathcal{L}}_S(h), n, \delta, \ldots)\right) \geq 1-\delta$$

- Hypothèse unique h :

$$\mathcal{L}_D(h) \leq \widehat{\mathcal{L}}_S(h) + \sqrt{\frac{1}{2n}\log\left(\frac{1}{\delta}\right)}.$$

Apprentissage PAC - En bref

PAC (Probablement Approximativement Correct)

Avec probabilité au moins "$1-\delta$", la perte de h sera au plus "$\varepsilon(\cdot,\ldots,\cdot)$"

$$\Pr_{S\sim D^n}\Big(\mathcal{L}_D(h) \leq \varepsilon(\widehat{\mathcal{L}}_S(h), n, \delta, \ldots)\Big) \geq 1-\delta$$

- Hypothèse unique h :

$$\mathcal{L}_D(h) \leq \widehat{\mathcal{L}}_S(h) + \sqrt{\tfrac{1}{2n}\log\left(\tfrac{1}{\delta}\right)}.$$

- Classe finie d'hypothèses $\mathcal{H}$:

PAC (Probablement Approximativement Correct)

Avec probabilité au moins "$1-\delta$", la perte de h sera au plus "$\varepsilon(\cdot, \ldots, \cdot)$"

$$\Pr_{S \sim D^n}\left(\mathcal{L}_D(h) \leq \varepsilon(\widehat{\mathcal{L}}_S(h), n, \delta, \ldots)\right) \geq 1-\delta$$

- Hypothèse unique h :

$$\mathcal{L}_D(h) \leq \widehat{\mathcal{L}}_S(h) + \sqrt{\frac{1}{2n} \log\left(\frac{1}{\delta}\right)}.$$

- Classe finie d'hypothèses $\mathcal{H}$:

$$\forall h \in \mathcal{H}, \quad \mathcal{L}_D(h) \leq \widehat{\mathcal{L}}_S(h) + \sqrt{\frac{1}{2n} \log\left(\frac{|\mathcal{H}|}{\delta}\right)}$$

Apprentissage PAC - En bref

PAC (Probablement Approximativement Correct)

Avec probabilité au moins "$1-\delta$", la perte de h sera au plus "$\varepsilon(\cdot, \ldots, \cdot)$"

$$\Pr_{S \sim D^n}\left(\mathcal{L}_D(h) \leq \varepsilon(\widehat{\mathcal{L}}_S(h), n, \delta, \ldots)\right) \geq 1-\delta$$

- Hypothèse unique h :

$$\mathcal{L}_D(h) \leq \widehat{\mathcal{L}}_S(h) + \sqrt{\frac{1}{2n} \log\left(\frac{1}{\delta}\right)}.$$

- Classe finie d'hypothèses $\mathcal{H}$:

$$\forall h \in \mathcal{H}, \quad \mathcal{L}_D(h) \leq \widehat{\mathcal{L}}_S(h) + \sqrt{\frac{1}{2n} \log\left(\frac{|\mathcal{H}|}{\delta}\right)}$$

- Classe dénombrable d'hypothèses h_i, avec avec probabilité *a priori* $p(h_i)$:

Apprentissage PAC - En bref

PAC (Probablement Approximativement Correct)

Avec probabilité au moins "$1-\delta$", la perte de h sera au plus "$\varepsilon(\cdot,\ldots,\cdot)$"

$$\Pr_{S\sim D^n}\left(\mathcal{L}_D(h) \leq \varepsilon(\widehat{\mathcal{L}}_S(h), n, \delta, \ldots)\right) \geq 1-\delta$$

- Hypothèse unique h :

$$\mathcal{L}_D(h) \leq \widehat{\mathcal{L}}_S(h) + \sqrt{\frac{1}{2n}\log\left(\frac{1}{\delta}\right)}.$$

- Classe finie d'hypothèses $\mathcal{H}$:

$$\forall h \in \mathcal{H}, \quad \mathcal{L}_D(h) \leq \widehat{\mathcal{L}}_S(h) + \sqrt{\frac{1}{2n}\log\left(\frac{|\mathcal{H}|}{\delta}\right)}$$

- Classe dénombrable d'hypothèses h_i, avec avec probabilité *a priori* $p(h_i)$:

$$\forall h \in \mathcal{H}, \quad \mathcal{L}_D(h) \leq \widehat{\mathcal{L}}_S(h) + \sqrt{\frac{1}{2n}\log\left(\frac{1}{p(h)\delta}\right)}$$

PAC (Probablement Approximativement Correct)

Avec probabilité au moins "$1-\delta$", la perte de h sera au plus "$\varepsilon(\cdot,\ldots,\cdot)$"

$$\Pr_{S\sim D^n}\left(\mathcal{L}_D(h) \leq \varepsilon(\widehat{\mathcal{L}}_S(h), n, \delta, \ldots)\right) \geq 1-\delta$$

- Hypothèse unique h :
$$\mathcal{L}_D(h) \leq \widehat{\mathcal{L}}_S(h) + \sqrt{\frac{1}{2n}\log\left(\frac{1}{\delta}\right)}.$$
- Classe finie d'hypothèses $\mathcal{H}$:
$$\forall h \in \mathcal{H},\quad \mathcal{L}_D(h) \leq \widehat{\mathcal{L}}_S(h) + \sqrt{\frac{1}{2n}\log\left(\frac{|\mathcal{H}|}{\delta}\right)}$$
- Classe dénombrable d'hypothèses h_i, avec avec probabilité *a priori* $p(h_i)$:
$$\forall h \in \mathcal{H},\quad \mathcal{L}_D(h) \leq \widehat{\mathcal{L}}_S(h) + \sqrt{\frac{1}{2n}\log\left(\frac{1}{p(h)\delta}\right)}$$
- Classe indénombrable d'hypothèses : Dimension VC, Complexité de Rademacher, PAC-Bayes...

Plan

1 Préambules

2 Théorie de la compression des données
- Définition et exemples
- Borne de généralisation pour classificateurs binaires
- Nouvelle borne de généralisation

3 Deep Reconstruction Machine
- Le paradigme du méta-apprentissage
- Résultats préliminaires - classification binaire

Plan

1 Préambules

2 Théorie de la compression des données
- Définition et exemples
- Borne de généralisation pour classificateurs binaires
- Nouvelle borne de généralisation

3 Deep Reconstruction Machine
- Le paradigme du méta-apprentissage
- Résultats préliminaires - classification binaire

Généalogie

Initialisation : Littlestone et Warmuth 1986 : "Relating Data Compression and Learnability"

The Set Covering Machine (SCM) et ses variants

Marchand et Shawe-Taylor 2002 : "The Set Covering Machine"
Marchand et Sokolova 2005 : "Learning with Decision Lists of Data-Dependent Features"
Laviolette et al. 2005 : "Margin-Sparsity Trade-Off for the Set Covering Machine"
Drouin et al. 2014 : "Learning interpretable models of phenotypes from whole genome sequences with the Set Covering Machine"
Godon et al. 2022 : "RandomSCM: interpretable ensembles of sparse classifiers tailored for omics data"

Autres

Campi et Garatti 2023 : "Compression, Generalization and Learning"
Paccagnan et al. 2023 : "The Pick-to-Learn Algorithm: Empowering Compression for Tight Generalization Bounds and Improved Post-training Performance"

Motivations

Motivations

- Classe indénombrable d'hypothèses : nouvel angle d'attaque

Motivations

- Classe indénombrable d'hypothèses : nouvel angle d'attaque
- Classe dénombrable d'hypothèses : le terme $|\mathcal{H}|$ peut être très pénalisant

Motivations

- Classe indénombrable d'hypothèses : nouvel angle d'attaque
- Classe dénombrable d'hypothèses : le terme $|\mathcal{H}|$ peut être très pénalisant

$$\forall h \in \mathcal{H}, \quad \mathcal{L}_D(h) \leq \widehat{\mathcal{L}}_S(h) + \sqrt{\frac{1}{2n} \log\left(\frac{|\mathcal{H}|}{\delta}\right)}$$

Motivations

- Classe indénombrable d'hypothèses : nouvel angle d'attaque
- Classe dénombrable d'hypothèses : le terme $|\mathcal{H}|$ peut être très pénalisant
- Permet de n'utiliser qu'un sous-ensemble des observations

$$\forall h \in \mathcal{H}, \quad \mathcal{L}_D(h) \leq \widehat{\mathcal{L}}_S(h) + \sqrt{\frac{1}{2n} \log\left(\frac{|\mathcal{H}|}{\delta}\right)}$$

Définitions

Un prédicteur h peut être **compressé** $h_{\mathbf{i}}^{\mu}$ si obtenu par un nouvel algorithme R dépendant de deux sources d'informations complémentaires :

Définitions

Un prédicteur h peut être **compressé** $h_{\mathbf{i}}^{\mu}$ si obtenu par un nouvel algorithme R dépendant de deux sources d'informations complémentaires :

- Un **ensemble de compression** $S_{\mathbf{i}}$, un sous-ensemble de S :

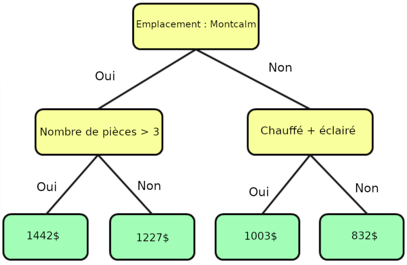
Exemple

Tâche : prédire le loyer mensuel d'un appartement dans la ville de Québec

S

ID	x_1 Emplacement	x_2 Nombre de pièce	x_3 Chauffé + Éclairé	$\ldots$ $\ldots$	x_d Étage	y Prix
1	Montcalm	3 1/2	Oui	$\ldots$	3e	1000
2	Montcalm	4 1/2	Oui	$\ldots$	1er	1230
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
n	Sainte-Foy	2 1/2	Non	$\ldots$	2e	829

$$A(S) = h$$

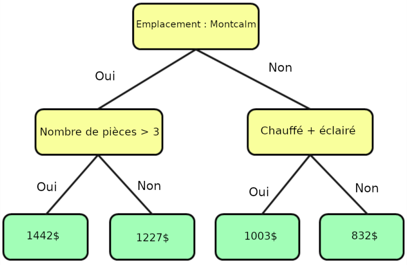


Tâche : prédire le loyer mensuel d'un appartement dans la ville de Québec

$$S_{\mathbf{i}} = S_{\{1,29,263,1902\}}$$

ID	x_1 Emplacement	x_2 Nombre de pièce	x_3 Chauffé + Éclairé	...	x_d Étage	y Prix
1	Montcalm	3 1/2	Oui	...	3e	1442
29	Montcalm	4 1/2	Oui	...	1er	1227
263	Montcalm	4 1/2	Oui	...	1er	1003
1902	Sainte-Foy	2 1/2	Non	...	2e	832

$$A(S) = h$$



Definitions

Un **prédicteur compressé** $h_{\mathbf{i}}^{\mu}$ est un prédicteur obtenu par un algorithme R dépendant de deux sources d'informations complémentaires :

- Un **ensemble de compression** $S_{\mathbf{i}}$, un sous-ensemble de S :

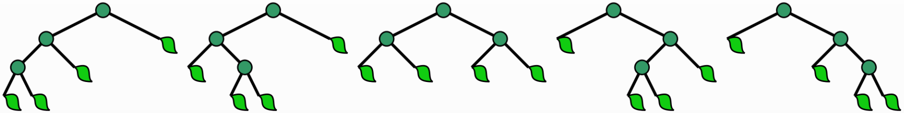
Definitions

Un **prédicteur compressé** $h_{\mathbf{i}}^{\mu}$ est un prédicteur obtenu par un algorithme R dépendant de deux sources d'informations complémentaires :

- Un **ensemble de compression** $S_{\mathbf{i}}$, un sous-ensemble de S :
- Un **message** $\mu \in \mathcal{M}_{\mathbf{i}}$ qui contient de l'information additionnelle pour décrire $h_{\mathbf{i}}^{\mu}$.

Profondeur ?

Nombre de feuilles ?



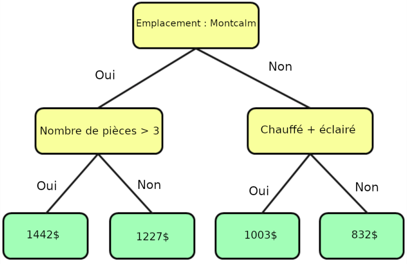
Exemple

Tâche : prédire le loyer mensuel d'un appartement dans la ville de Québec

$$S_{\mathbf{i}} = S_{\{1,29,263,1902\}}$$

ID	x_1 Emplacement	x_2 Nombre de pièce	x_3 Chauffé + Éclairé	... ...	x_d Étage	y Prix
1	Montcalm	3 1/2	Oui	...	3e	1442
29	Montcalm	4 1/2	Oui	...	1er	1227
263	Montcalm	4 1/2	Oui	...	1er	1003
1902	Sainte-Foy	2 1/2	Non	...	2e	832

$$A(S) = h$$



Definitions

Un **prédicteur compressé** $h_{\mathbf{i}}^{\mu}$ est un prédicteur obtenu par un algorithme R dépendant de deux sources d'informations complémentaires :

- Un **ensemble de compression** $S_{\mathbf{i}}$, un sous-ensemble de S :
- Un **message** $\mu \in \mathcal{M}_{\mathbf{i}}$ qui contient de l'information additionnelle pour décrire $h_{\mathbf{i}}^{\mu}$.

Definitions

Un **prédicteur compressé** $h_{\mathbf{i}}^{\mu}$ est un prédicteur obtenu par un algorithme R dépendant de deux sources d'informations complémentaires :

- Un **ensemble de compression** $S_{\mathbf{i}}$, un sous-ensemble de S :
- Un **message** $\mu \in \mathcal{M}_{\mathbf{i}}$ qui contient de l'information additionnelle pour décrire $h_{\mathbf{i}}^{\mu}$.

Étant donné $S_{\mathbf{i}} \in \mathcal{Z}^{|\mathbf{i}|}$ et $\mu \in \mathcal{M}_{\mathbf{i}}$, une **fonction de reconstruction** $\mathcal{R}$ donne un prédicteur :

$$h_{\mathbf{i}}^{\mu} = \mathcal{R}(S_{\mathbf{i}}, \mu) \,.$$

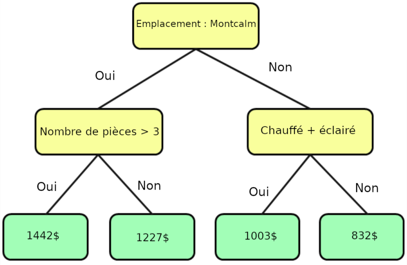
Exemple

Tâche : prédire le loyer mensuel d'un appartement dans la ville de Québec

$$A(S) = h$$
$$R(S_{\mathbf{i}}, \mu) = h$$

$$S_{\mathbf{i}} = S_{\{1,29,263,1902\}}$$

ID	x_1 Emplacement	x_2 Nombre de pièce	x_3 Chauffé + Éclairé	... ...	x_d Étage	y Prix
1	Montcalm	3 1/2	Oui	...	3e	1442
29	Montcalm	4 1/2	Oui	...	1er	1227
263	Montcalm	4 1/2	Oui	...	1er	1003
1902	Sainte-Foy	2 1/2	Non	...	2e	832



Examples de prédicteurs facile à compresser

SVM : *Support Vector Machine* (marge rigide)

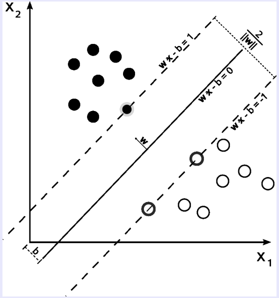


Image : Wikipedia

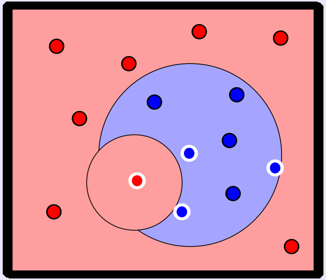
L'algorithme d'apprentissage du SVM agit comme sa propre fonction de reconstruction

$$\mathrm{SVM}(S) = h_{\mathbf{i}}^{\mu} = \mathrm{SVM}(S_{\mathbf{i}})$$

with $S_{\mathbf{i}} = \{\text{vecteurs de support}\}$
and $\mu = \emptyset$

Examples de prédicteurs facile à compresser

SCM : Set Covering Machine (Marchand et Shawe-Taylor 2002)
$\rightarrow$ conjonction variables explicatives booléennes créées



Le SCM apprend des *features* $b_{i,j} : \mathcal{X} \to \{-1,+1\}$

Chaque variable explicative est une boule $b_{i,j} \in \mathcal{B}$ définie par un centre (x_i, y_i) et une bordure (x_j, y_j) :

$$b_{i,j}(x) \quad := \quad \begin{cases} +y_i \text{ if } \|x - x_i\| \leq \|x_i - x_j\| + \epsilon \cdot y_i, \\ -y_i \text{ sinon.} \end{cases}$$

Le SCM apprend une conjonction comme classifieur :

$$h_{\mathbf{i}}^{\mu}(x) \quad := \quad \bigwedge_{b_{i,j} \in \mathcal{B}} b_{i,j}(x) \quad (\text{où } +1 \equiv \texttt{Vrai} \ -1 \equiv \texttt{Faux})$$

avec $S_{\mathbf{i}} = \{$les points « centre » et « bordure »$\}$
and $\mu = \{$les indices parmi $S_{\mathbf{i}}$ $\}$

Plan

1 Préambules

2 Théorie de la compression des données
- Définition et exemples
- Borne de généralisation pour classificateurs binaires
- Nouvelle borne de généralisation

3 Deep Reconstruction Machine
- Le paradigme du méta-apprentissage
- Résultats préliminaires - classification binaire

Bornes de généralisation pour des classifieurs binaires compressés

Étant donné $h^{\mu}_{\mathbf{i}} : \mathcal{X} \to \{-1, +1\}$, et $\mathcal{L}^{01}_{D}(h^{\mu}_{\mathbf{i}}) = \mathop{\mathbf{E}}_{(x,y)\sim D} I(h^{\mu}_{\mathbf{i}}(x) \neq y)$ la perte zéro-un.

Bornes de généralisation pour des classifieurs binaires compressés

Étant donné $h^{\mu}_{\mathbf{i}} : \mathcal{X} \to \{-1, +1\}$, et $\mathcal{L}^{01}_{D}(h^{\mu}_{\mathbf{i}}) = \mathop{\mathbf{E}}_{(x,y)\sim D} I(h^{\mu}_{\mathbf{i}}(x) \neq y)$ la perte zéro-un.

Théorème (Marchand et Sokolova 2005 ; Laviolette et al. 2005)

Bornes de généralisation pour des classifieurs binaires compressés

Étant donné $h_{\mathbf{i}}^{\mu} : \mathcal{X} \to \{-1, +1\}$, et $\mathcal{L}_D^{01}(h_{\mathbf{i}}^{\mu}) = \mathop{\mathbf{E}}_{(x,y)\sim D} I(h_{\mathbf{i}}^{\mu}(x) \neq y)$ la perte zéro-un.

Théorème (Marchand et Sokolova 2005 ; Laviolette et al. 2005)

Soit $\mathcal{R}$ une fonction de reconstruction, $P_{\mathcal{M}_{\mathbf{i}}}$ une distribution sur les messages, et $\delta \in (0, 1]$.
Avec forte probabilité ($\geq 1 - \delta$) sur $S \sim D^n$:

Bornes de généralisation pour des classifieurs binaires compressés

Étant donné $h_{\mathbf{i}}^{\mu} : \mathcal{X} \to \{-1,+1\}$, et $\mathcal{L}_D^{01}(h_{\mathbf{i}}^{\mu}) = \underset{(x,y)\sim D}{\mathbf{E}} I(h_{\mathbf{i}}^{\mu}(x) \neq y)$ la perte zéro-un.

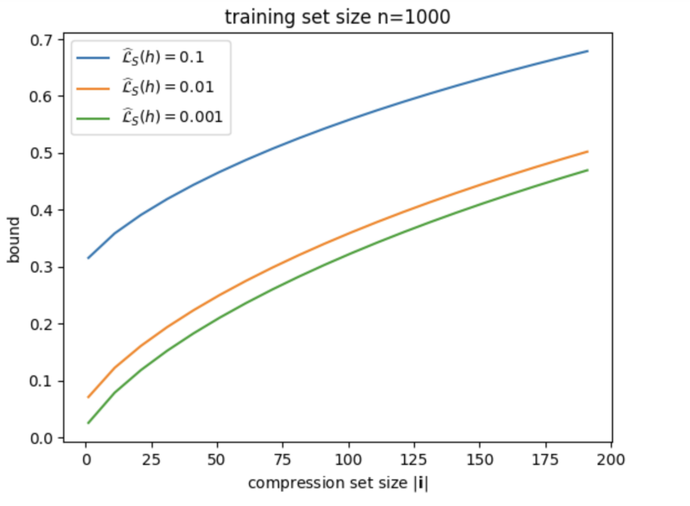
Théorème (Marchand et Sokolova 2005 ; Laviolette et al. 2005)

Soit $\mathcal{R}$ une fonction de reconstruction, $P_{\mathcal{M}_{\mathbf{i}}}$ une distribution sur les messages, et $\delta \in (0,1]$.
Avec forte probabilité ($\geq 1-\delta$) sur $S \sim D^n$: $\forall \mathbf{i} \in \mathcal{I}_n, \mu \in \mathcal{M}_{\mathbf{i}}$:

$$\mathcal{L}_D^{01}(h_{\mathbf{i}}^{\mu}) \leq 1 - \exp\left(\frac{-1}{n-|\mathbf{i}|-k_{S_{\mathbf{i}^c}}}\left[\ln\binom{n-|\mathbf{i}|}{k_{S_{\mathbf{i}^c}}} + \ln\binom{n}{|\mathbf{i}|} + \ln\left(\frac{1}{P_{\mathcal{M}_{\mathbf{i}}}(\mu)\cdot\xi(|\mathbf{i}|)\cdot\delta}\right)\right]\right)$$

où $k_{S_{\mathbf{i}^c}} := |\mathbf{i}^c|\widehat{\mathcal{L}}_{S_{\mathbf{i}^c}}^{01}(h_{\mathbf{i}}^{\mu})$ est le nombre d'erreurs sur $S_{\mathbf{i}^c} := S \setminus S_{\mathbf{i}}$ et $\xi(a) := \frac{6}{\pi^2}(a+1)^{-2}$.

Bornes non-triviales !



Plan

1 Préambules

2 Théorie de la compression des données
- Définition et exemples
- Borne de généralisation pour classificateurs binaires
- Nouvelle borne de généralisation

3 Deep Reconstruction Machine
- Le paradigme du méta-apprentissage
- Résultats préliminaires - classification binaire

Motivation

La question

Peut-on apprendre une fonction de reconstruction $\mathcal{R}(\mathbf{i}, \mu)$ pour construire un réseau de neurones ?

Motivation

La question

Peut-on apprendre une fonction de reconstruction $\mathcal{R}(\mathbf{i}, \mu)$ pour construire un réseau de neurones ?

Il faudrait alors adapter nos résultats de telle sorte que :

Motivation

La question

Peut-on apprendre une fonction de reconstruction $\mathcal{R}(\mathbf{i}, \mu)$ pour construire un réseau de neurones ?

Il faudrait alors adapter nos résultats de telle sorte que :

1. Les pertes à valeurs continues sont considérées

Motivation

La question

Peut-on apprendre une fonction de reconstruction $\mathcal{R}(\mathbf{i}, \mu)$ pour construire un réseau de neurones ?

Il faudrait alors adapter nos résultats de telle sorte que :

1. Les pertes à valeurs continues sont considérées
2. Une quantité indénombrable de message

Nouvelle borne pour une quantité indénombrable de mesages

PAC-Bayes à la rescousse !

On considère maintenant une distribution *a posteriori* $Q_{\mathcal{M}}$ sur l'ensemble (potentiellement continu) des messages $\mathcal{M}$.

Nouvelle borne pour une quantité indénombrable de mesages

PAC-Bayes à la rescousse !

On considère maintenant une distribution *a posteriori* $Q_{\mathcal{M}}$ sur l'ensemble (potentiellement continu) des messages $\mathcal{M}$.

Théorème

Soit $\mathcal{R}$, une fonction de reconstruction, $P_{\mathcal{M}}$ une distribution *a priori* sur les messages, et $\delta \in (0, 1]$. Avec forte probabilité ($\geq 1 - \delta$) sur $S \sim D^n$:

Nouvelle borne pour une quantité indénombrable de mesages

PAC-Bayes à la rescousse !

On considère maintenant une distribution *a posteriori* $Q_{\mathcal{M}}$ sur l'ensemble (potentiellement continu) des messages $\mathcal{M}$.

Théorème

Soit $\mathcal{R}$, une fonction de reconstruction, $P_{\mathcal{M}}$ une distribution *a priori* sur les messages, et $\delta \in (0,1]$. Avec forte probabilité ($\geq 1-\delta$) sur $S \sim D^n$:

$\forall \mathbf{i} \in \mathcal{I}_n, Q_{\mathcal{M}}$ sur $\mathcal{M}$:

$$\mathop{\mathbf{E}}_{\mu \sim Q_{\mathcal{M}}} \mathcal{L}_D(h_{\mathbf{i}}^{\mu}) \leq \mathop{\mathbf{E}}_{\mu \sim Q_{\mathcal{M}}} \widehat{\mathcal{L}}_{S_{\mathbf{i}^c}}(h_{\mathbf{i}}^{\mu}) + \frac{1}{\sqrt{n-|\mathbf{i}|}} \left[\mathrm{KL}(Q_{\mathcal{M}} \| P_{\mathcal{M}}) + \frac{\sigma^2}{2} + \ln \binom{n}{|\mathbf{i}|} + \ln \left(\frac{1}{\xi(|\mathbf{i}|) \cdot \delta} \right) \right].$$

Plan

1 Préambules

2 Théorie de la compression des données
- Définition et exemples
- Borne de généralisation pour classificateurs binaires
- Nouvelle borne de généralisation

3 Deep Reconstruction Machine
- Le paradigme du méta-apprentissage
- Résultats préliminaires - classification binaire

Plan

1 Préambules

2 Théorie de la compression des données
- Définition et exemples
- Borne de généralisation pour classificateurs binaires
- Nouvelle borne de généralisation

3 Deep Reconstruction Machine
- Le paradigme du méta-apprentissage
- Résultats préliminaires - classification binaire

Le méta-apprentissage ?

Idée

Le méta-apprentissage ?

Idée

- Considérer plusieurs tâches simultanément

Le méta-apprentissage ?

Idée

- Considérer plusieurs tâches simultanément

Motivations

Le méta-apprentissage ?

Idée

- Considérer plusieurs tâches simultanément

Motivations

- Partage de connaissance entre les tâches

Le méta-apprentissage ?

Idée

- Considérer plusieurs tâches simultanément

Motivations

- Partage de connaissance entre les tâches
- Davantage d'exemple par entrainement

Définitions

Méta-entrainement

$$\mathcal{S} := \{S^{(1)}, S^{(2)}, \ldots, S^{(m)}\} \text{ tel que } S^{(i)} \sim \left(D^{(i)}\right)^{n_i}.$$

Définitions

Méta-entrainement

$$\boldsymbol{\mathcal{S}} \coloneqq \{S^{(1)}, S^{(2)}, \ldots, S^{(m)}\} \text{ tel que } S^{(i)} \sim \left(D^{(i)}\right)^{n_i}.$$

Algorithme

$$\boldsymbol{\mathcal{A}}(\boldsymbol{\mathcal{S}}) \longrightarrow (\mathcal{C}, \mathcal{R})$$

Définitions

Méta-entrainement

$$\boldsymbol{\mathcal{S}} := \{S^{(1)}, S^{(2)}, \ldots, S^{(m)}\} \text{ tel que } S^{(i)} \sim \left(D^{(i)}\right)^{n_i}.$$

Algorithme

$$\boldsymbol{\mathcal{A}}(\boldsymbol{S}) \longrightarrow (\mathcal{C}, \mathcal{R})$$

Fonction de compression

$$\mathcal{C}(S) \longrightarrow (\mathbf{i}, \mu)$$

Définitions

Méta-entrainement

$$\boldsymbol{\mathcal{S}} := \{S^{(1)}, S^{(2)}, \ldots, S^{(m)}\} \text{ tel que } S^{(i)} \sim \left(D^{(i)}\right)^{n_i}.$$

Algorithme

$$\boldsymbol{\mathcal{A}}(\boldsymbol{S}) \longrightarrow (\mathcal{C}, \mathcal{R})$$

Fonction de compression

$$\mathcal{C}(S) \longrightarrow (\mathbf{i}, \mu)$$

Fonction de reconstruction

$$\mathcal{R}(\mathbf{i}, \mu) \longrightarrow \theta$$

Définitions

Méta-entrainement

$$\boldsymbol{\mathcal{S}} := \{S^{(1)}, S^{(2)}, \ldots, S^{(m)}\} \text{ tel que } S^{(i)} \sim \left(D^{(i)}\right)^{n_i}.$$

Algorithme

$$\boldsymbol{\mathcal{A}}(\boldsymbol{\mathcal{S}}) \longrightarrow (\mathcal{C}, \mathcal{R})$$

Fonction de compression

$$\mathcal{C}(S) \longrightarrow (\mathbf{i}, \mu)$$

Fonction de reconstruction

$$\mathcal{R}(\mathbf{i}, \mu) \longrightarrow \theta$$

Prédicteur

$$h_\theta(\mathbf{x}) \longrightarrow y$$

Définitions

Méta-entrainement

$$\boldsymbol{\mathcal{S}} := \{S^{(1)}, S^{(2)}, \ldots, S^{(m)}\} \text{ tel que } S^{(i)} \sim \left(D^{(i)}\right)^{n_i}.$$

Algorithme

$$\boldsymbol{\mathcal{A}}(\boldsymbol{\mathcal{S}}) \longrightarrow (\mathcal{C}, \mathcal{R})$$

Fonction de compression

$$\mathcal{C}(S) \longrightarrow (\mathbf{i}, \mu)$$

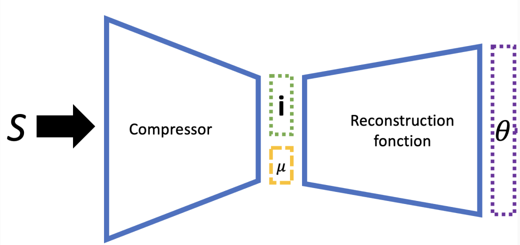
Fonction de reconstruction

$$\mathcal{R}(\mathbf{i}, \mu) \longrightarrow \theta$$

Prédicteur

$$h_\theta(\mathbf{x}) \longrightarrow y$$

Combiner meta-learning et sample compression :



Plan

1 Préambules

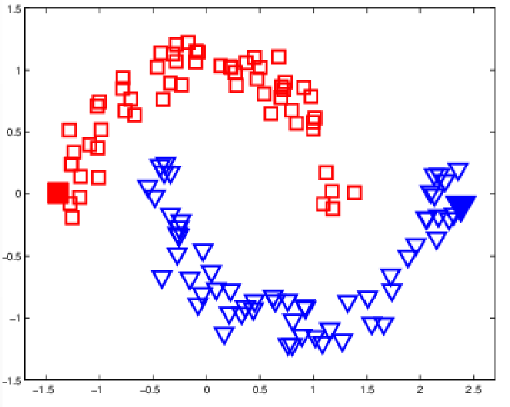
2 Théorie de la compression des données

- Définition et exemples
- Borne de généralisation pour classificateurs binaires
- Nouvelle borne de généralisation

3 Deep Reconstruction Machine

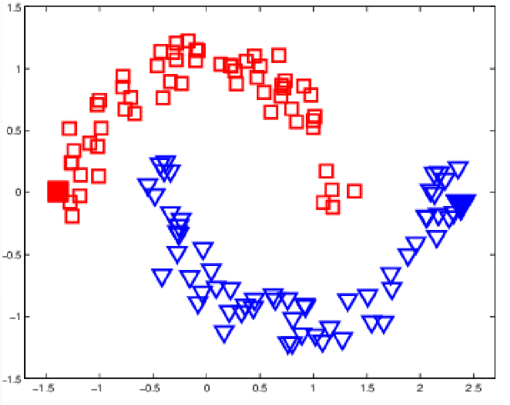
- Le paradigme du méta-apprentissage
- Résultats préliminaires - classification binaire

Résultats préliminaires



Tâche #1

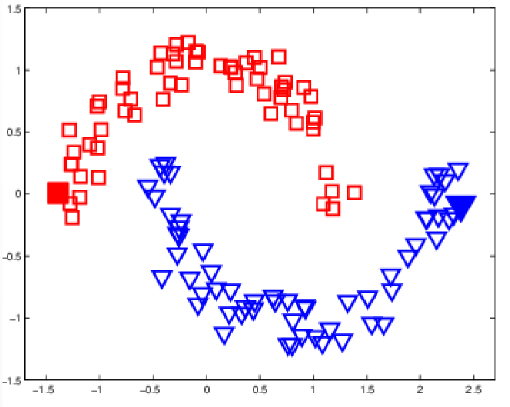
Résultats préliminaires



Tâche #1

- Non-linéairement séparable

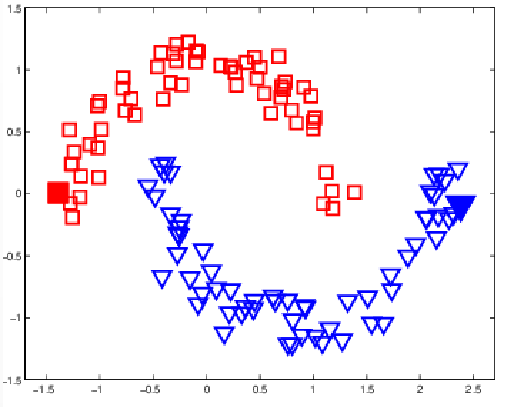
Résultats préliminaires



Tâche #1

- Non-linéairement séparable
- Translations / rotations / homothéties

Résultats préliminaires



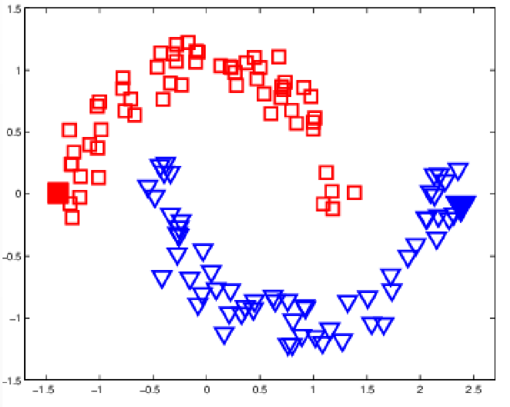
Tâche #1

- Non-linéairement séparable
- Translations / rotations / homothéties

Le prédicteur

- h : réseau de neurones simple

Résultats préliminaires



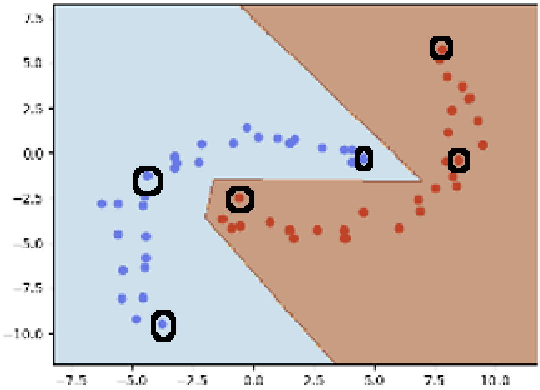
Tâche #1

- Non-linéairement séparable
- Translations / rotations / homothéties

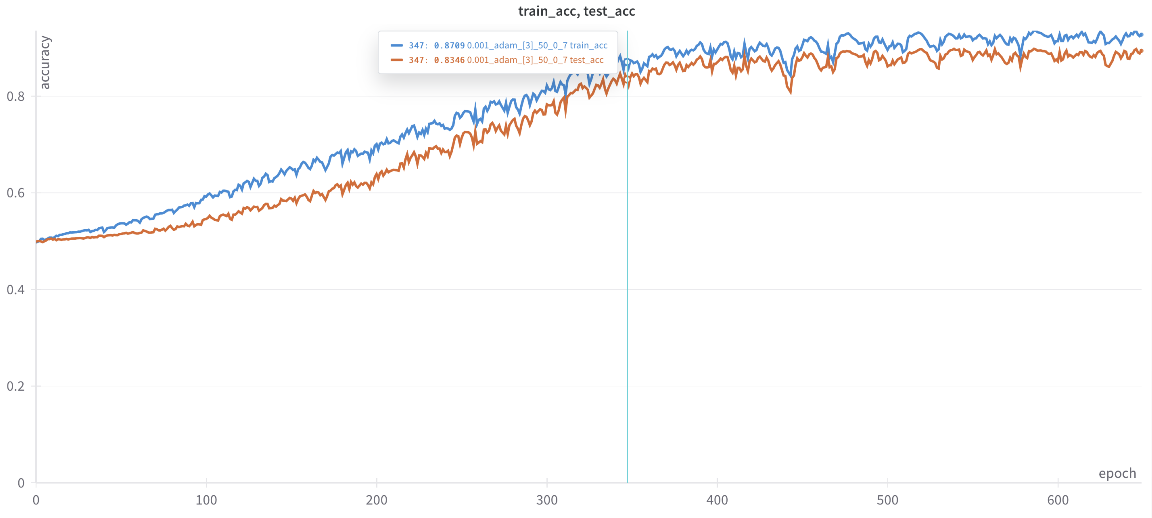
Le prédicteur

- h : réseau de neurones simple
- $\theta = (\mathbf{w}, b)$

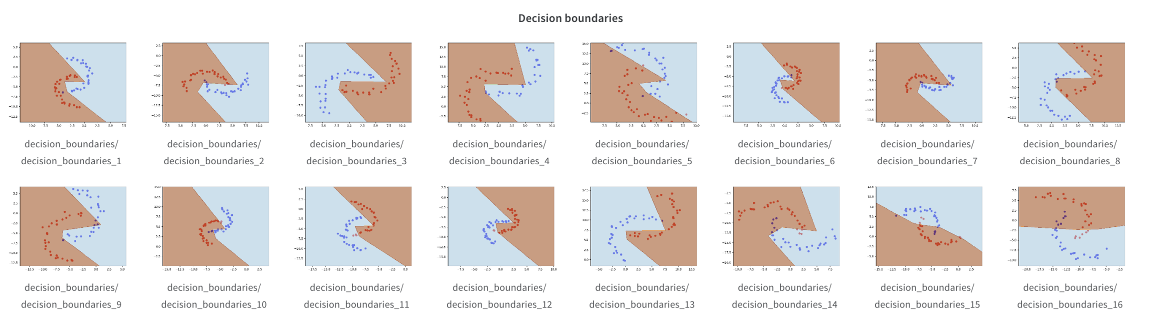
Résultats préliminaires



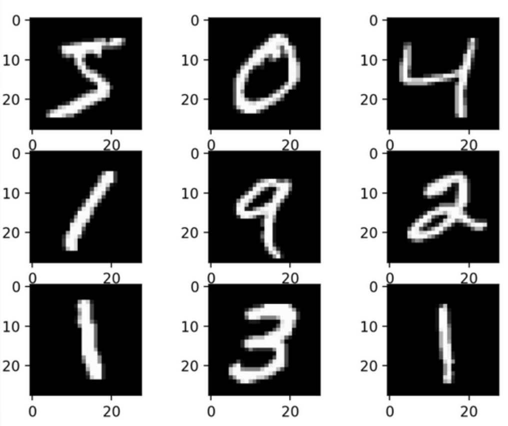
Résultats préliminaires



Résultats préliminaires

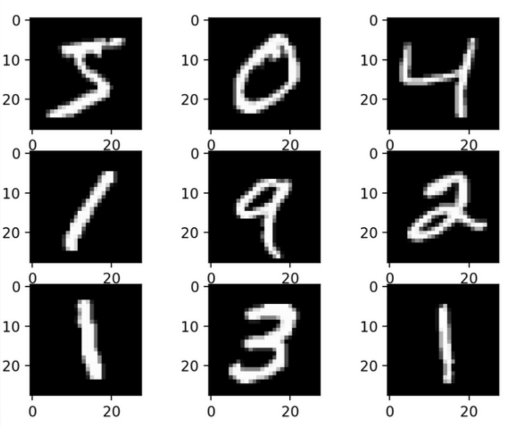


Résultats préliminaires



Tâche #2

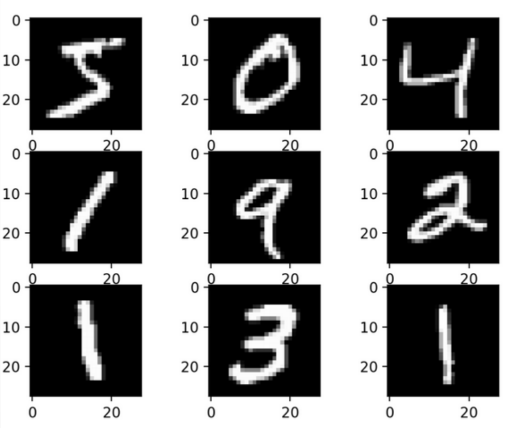
Résultats préliminaires



Tâche #2

- Entrée de dimension 28x28

Résultats préliminaires



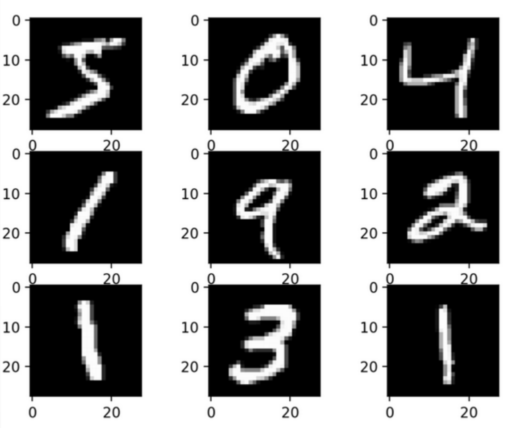
Tâche #2

- Entrée de dimension 28x28

Le prédicteur

- h : séparateur linéaire
- $h(\mathbf{x}) = \text{sgn}(\mathbf{w} \cdot \mathbf{x} + b)$

Résultats préliminaires



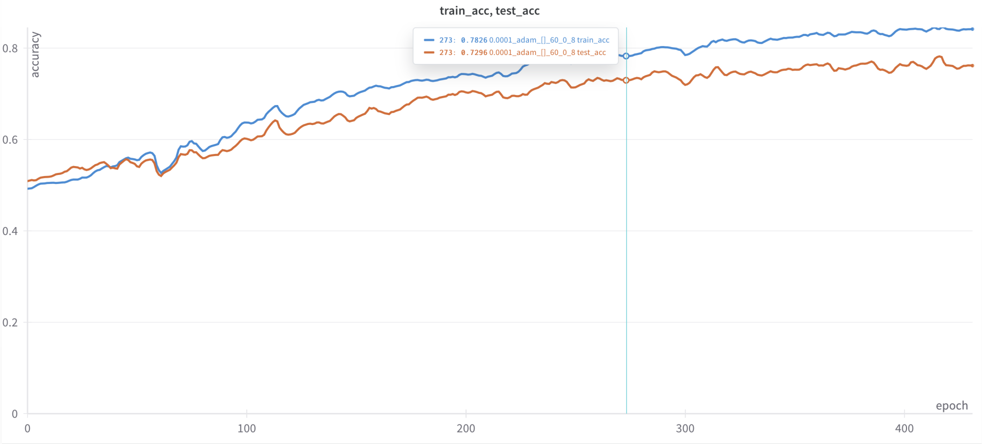
Tâche #2

- Entrée de dimension 28x28

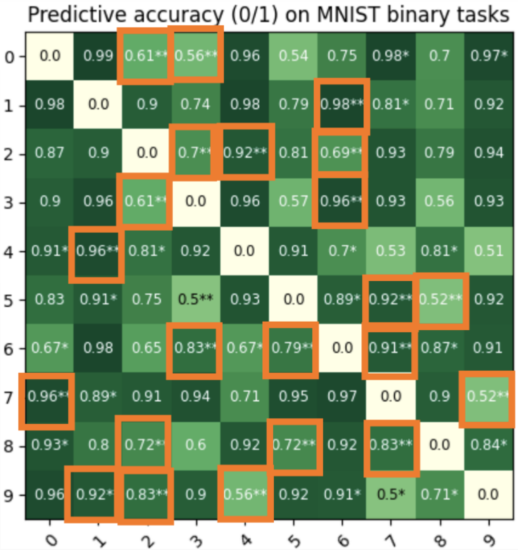
Le prédicteur

- h : séparateur linéaire
- $h(\mathbf{x}) = \mathrm{sgn}(\mathbf{w} \cdot \mathbf{x} + b)$
- $\theta = (\mathbf{W}_i, \mathbf{b}_i)_{i=1}^{2}$

Résultats préliminaires



Résultats préliminaires



Merci de votre écoute :) !

References I

Campi, Marco C. et Simone Garatti (2023). "Compression, Generalization and Learning". In : *JMLR* abs/2301.12767.

Drouin, Alexandre, Sébastien Giguere, Vladana Sagatovich, Maxime Déraspe, François Laviolette, Mario Marchand et Jacques Corbeil (2014). "Learning interpretable models of phenotypes from whole genome sequences with the Set Covering Machine". In : *arXiv preprint arXiv:1412.1074*.

Godon, Thibaud, Pier-Luc Plante, Baptiste Bauvin, Elina Francovic-Fontaine, Alexandre Drouin et Jacques Corbeil (2022). "RandomSCM: interpretable ensembles of sparse classifiers tailored for omics data". In : *CoRR* abs/2208.06436. doi : 10.48550/ARXIV.2208.06436. arXiv : 2208.06436. url : https://doi.org/10.48550/arXiv.2208.06436.

Laviolette, François, Mario Marchand et Mohak Shah (2005). "Margin-Sparsity Trade-Off for the Set Covering Machine". In : *ECML*. T. 3720. Lecture Notes in Computer Science. Springer, p. 206-217.

Littlestone, Nick et Manfred K. Warmuth (1986). "Relating Data Compression and Learnability". In : *Technical Report*.

Marchand, Mario et John Shawe-Taylor (2002). "The Set Covering Machine". In : *JMLR* 3.

Marchand, Mario et Marina Sokolova (2005). "Learning with Decision Lists of Data-Dependent Features". In : *JMLR* 6.

Paccagnan, Dario, Marco C. Campi et Simone Garatti (2023). "The Pick-to-Learn Algorithm: Empowering Compression for Tight Generalization Bounds and Improved Post-training Performance". In : *NeurIPS*.