

Un algorithme d'apprentissage pour l'agrégation de réseaux de neurones à activations binaires

Louis Fortier-Dubois

Pascal Germain

Gaël Letarte

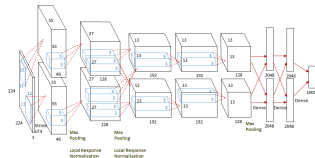
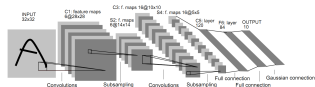
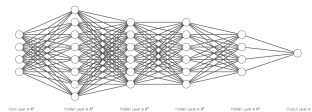
François Laviolette

Benjamin Leblanc

4 avril 2022

Introduction

- Les réseaux de neurones : pourquoi sont-ils si difficile à analyser?
- La simplification comme solution
- Distribution de probabilité sur les poids du réseau
- Basé sur les travaux de *Letarte et al.*[1]



[1] Gaël Letarte, Pascal Germain, Benjamin Guedj, and François Laviolette. Dichotomize and generalize: PAC-Bayesian binary activated deep neural networks. NeurIPS, 2019.

Plan de la présentation

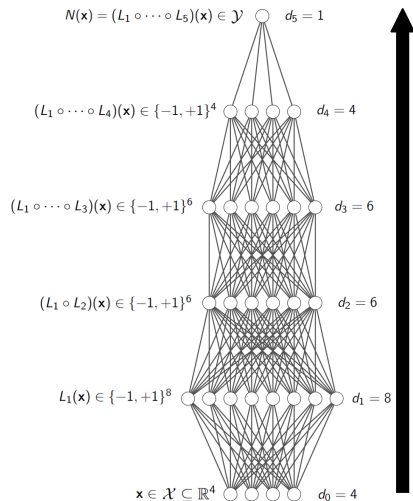
- 1 Notions préliminaires
- 2 Agrégation de réseaux de neurones
- 3 Expérimentations
- 4 Période de questions



Notions préliminaires

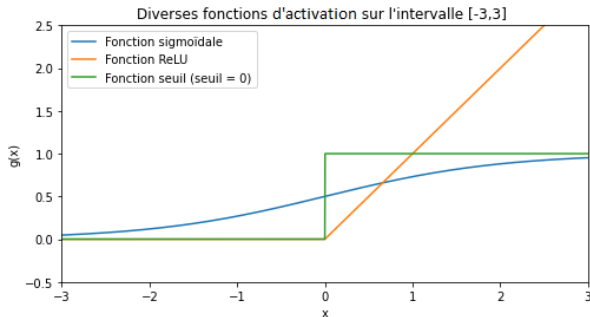
Les réseaux de neurones

- Réseaux pleinement connectés seulement
- Tâches de classification binaire
- $F(\mathbf{x}) = (L_1 \circ \dots \circ L_l)(\mathbf{x})$
- $L_k(\mathbf{x}) = g_k(\mathbf{W}_k \mathbf{x} + \mathbf{b}_k) \quad \forall k \in \{1, \dots, l\}$



Pourquoi?

- Gains de temps de calculs
- Gains en mémoire requise
- Réseaux moins prompts au surapprentissage [1] [2]
- Possibilité d'interprétation



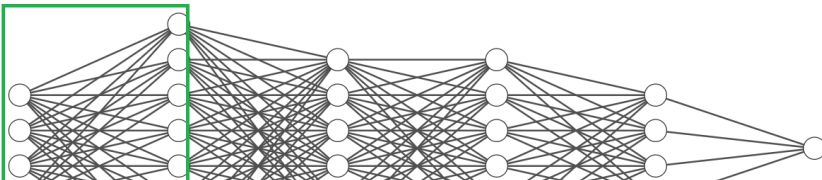
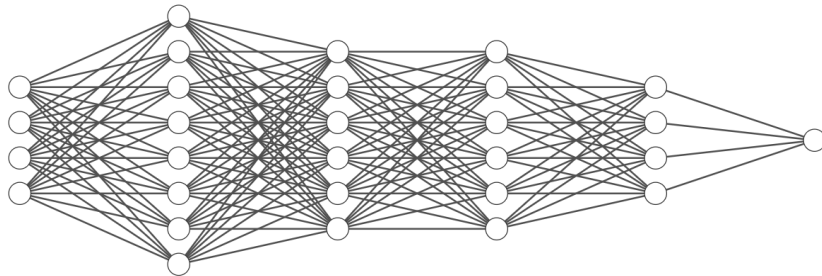
- [1] J. Lin, C. Gan, and S. Han, "Defensive quantization: When efficiency meets robustness". CoRR, 2019
- [2] M. Courbariaux, I. Hubara, D. Soudry, R. El-Yaniv, and Y. Bengio, "Binarized neural networks: Training deep neural networks with weights and activations constrained to $+1$ or -1 ". NIPS, 2016.

Étude des réseaux à activations binaires : rôle de chaque partie du réseau

Définition : les parties d'un réseau à activations binaires

- La première partie du réseau représente la première couche L_1
- La seconde partie du réseau représente le reste du réseau ($L_2 \circ \dots \circ L_l$)

Activations binaires - Visualisation du problème



Étude des réseaux à activations binaires : rôle de chaque partie du réseau

Définition : les parties d'un réseau binaire

- La première partie du réseau représente la première couche L_1 et sert à assigner aux exemples reçus un *groupe*
- La seconde partie du réseau représente le reste du réseau ($L_2 \circ \dots \circ L_I$) et sert à assigner aux différents groupes une *prédiction*

Constatation : on a $L_1 : \mathcal{X} \rightarrow \{-1, +1\}^{d_1}$ et $(L_2 \circ \dots \circ L_I) : \{-1, +1\}^{d_1} \rightarrow \mathcal{Y}$

Agrégation de réseaux de neurones

Aspect Bayésien

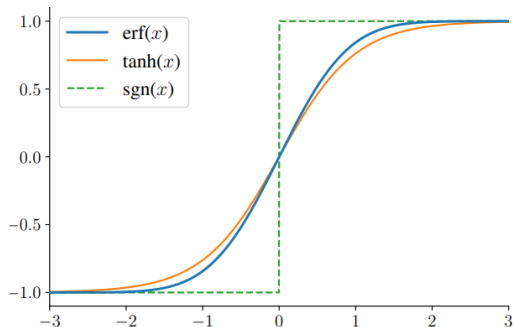
On suppose tous les poids $B = \langle \mathbf{W}_k \rangle_{k=1}^l$ d'une agrégation de réseaux suivant des distributions de probabilité Gaussienne, centrée en B et à matrice de covariance isotropique.

La sortie d'une neurone de l'agrégat

La sortie espérée d'une fonction signe avec distribution Gaussienne :

$$\mathbf{E}_{\mathbf{v} \sim \mathcal{N}(\mathbf{w}, \mathbf{I})} \text{sgn}(\mathbf{v} \cdot \mathbf{a}) = \text{erf} \left(\frac{\mathbf{w} \cdot \mathbf{a}}{\sqrt{2} \|\mathbf{a}\|} \right),$$

avec $\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$ étant la fonction d'erreur gaussienne.



Probabilité d'observer une sortie donnée pour un neurone donnée

L'activation binaire agit à la façon d'une distribution de Bernoulli; on connaît son espérance. La probabilité d'observer $s_i \in \{-1, +1\}$ est donnée par

$$\Pr(L_{k,i}(\mathbf{a}) = s) = \frac{1}{2} + \frac{s}{2} \operatorname{erf}\left(\frac{\mathbf{w}_{k,i} \cdot \mathbf{a}}{\sqrt{2}\|\mathbf{a}\|}\right), i \in \{1, \dots, d_k\}, k \in \{1, \dots, l\}.$$

Probabilité d'observer une sortie donnée pour une couche donnée

Soit $\Pr((L_1 \circ \dots \circ L_k)(\mathbf{a}) = \mathbf{s}) = L_{1,k}^{\mathbf{s}}(\mathbf{a})$. De la même manière que pour la sortie d'une couche, la probabilité d'observer une certaine sortie chez une couche peut être calculée de façon itérative :

$$\Pr(L_{1,k}^{\mathbf{s}}(\mathbf{a})) = \begin{cases} \prod_{i=1}^{d_1} \Pr(L_{1,i}^{s_i}(\mathbf{a})) & \text{si } k = 1, \\ \sum_{\bar{\mathbf{s}}} \Pr(L_{1,k}^{\mathbf{s}}(\mathbf{a}) \mid L_{1,k-1}^{\bar{\mathbf{s}}}(\mathbf{a})) \Pr(L_{1,k-1}^{\bar{\mathbf{s}}}(\mathbf{a})) & \text{sinon.} \end{cases}$$

Le calcul exact de ABNet

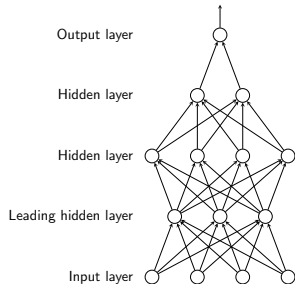
Méthode de programmation dynamique

$$\text{Première couche : } \mathbf{P}_1(\mathbf{x}) = \left[\Pr(L_1^s(\mathbf{a})) \right]_{s \in R_1}$$

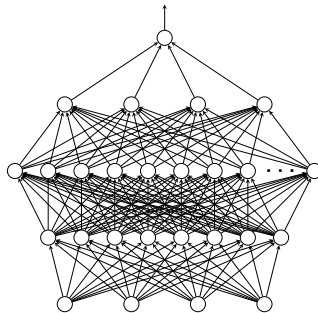
$$\begin{aligned} \text{Couches subséquentes : } \mathbf{P}_k &= \left[\Pr(L_{1,k-1}^s(\mathbf{a})) \right]_{s \in R_k} \\ &= \boldsymbol{\Psi}_k \cdot \mathbf{P}_{k-1} , \end{aligned}$$

$$\text{avec } \boldsymbol{\Psi}_k = \left[\Pr(L_{1,k}^s(\mathbf{a}) \mid L_{1,k-1}^{\bar{s}}(\mathbf{a})) \right]_{s \in R_k, \bar{s} \in R_{k-1}}$$

Agrégation de réseaux de neurones



(a) Un réseau de neurone typique.

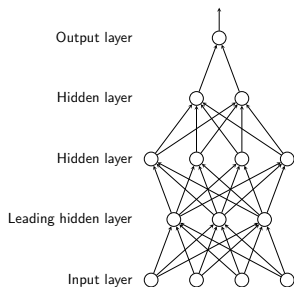


(b) L'architecture d'**ABNet**, avec comme réseau sous-tendu celui présenté en (a).

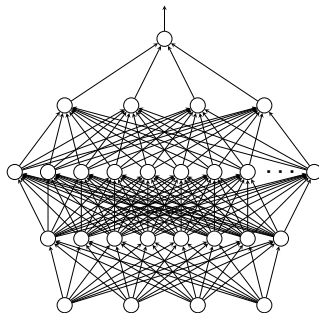
ABNet compact

$$\begin{aligned}\Pr(L_{1,I}^s(\mathbf{a})) &= s \cdot (\Psi_I(\Psi_{I-1}(\dots \Psi_3(\Psi_2 \mathbf{P}_1(\mathbf{x})) \dots))) \\ &= s \cdot \underbrace{\Psi_I \cdot \Psi_{I-1} \cdot \dots \cdot \Psi_3 \cdot \Psi_2}_{H_s} \cdot \mathbf{P}_1(\mathbf{x})\end{aligned}$$

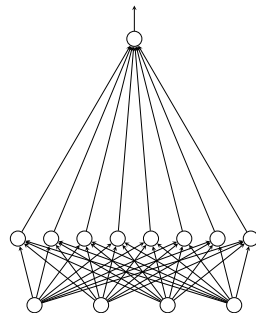
Agrégation de réseaux de neurones



(a) Un réseau de neurone typique.



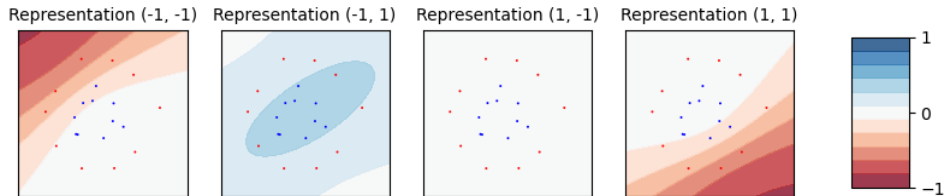
(b) L'architecture d'**ABNet**, avec comme réseau sous-tendu celui présenté en (a).



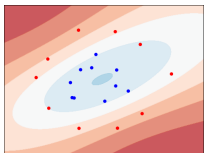
(c) La structure d'**ABNet compact**, obtenue à partir du ABNet en (b).

Expérimentations

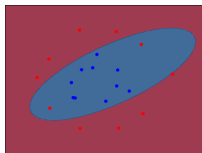
ABNet en pratique



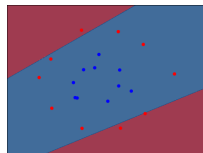
(a) Pour diverses valeurs de $\mathbf{x}$, les probabilités d'avoir une certaine représentation en sortie de la première couche cachée, obtenues à partir de $\mathbf{P}_1(\mathbf{x})$, multipliées par $\mathbf{H}$.



(b) La sortie de l'agrégation $\Pr(L_{1,l})$.



(c) ABNet comme classifieur.



(d) Réseau unique, centré sur les moyennes *a posteriori* des lois Gaussiennes.

Expérimentations

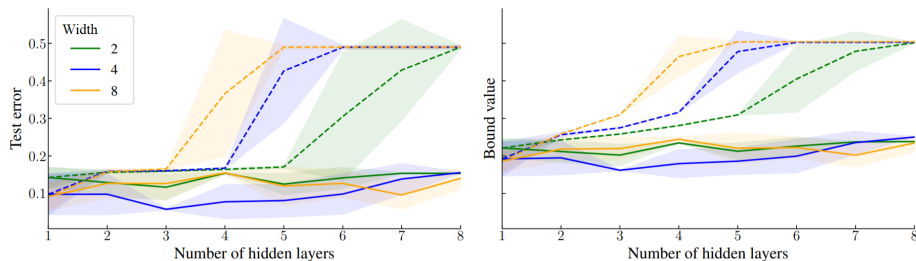


Figure: Impact de la profondeur pour **PBGNet (pointillé)** et **ABNet (continu)** sur l'erreur en test et la valeur de la borne selon la largeur pour le jeu de données mnistLH. 5 répétitions effectuées.

Expérimentations

Dataset	Model	$L-1$	d	Bound	Error _S	Error _T
ads	PBGNet	3	2	0.192 ± 0.004	0.140 ± 0.004	0.141 ± 0.012
	ABNet	3	2	0.192 ± 0.004	0.140 ± 0.004	0.141 ± 0.012
	PBGNet _ℓ	3	4	1.000 ± 0.001	0.018 ± 0.005	0.026 ± 0.004
	ABNet _ℓ	3	4	0.887 ± 0.064	0.015 ± 0.003	0.026 ± 0.003
	EBP	2	2	—	0.003 ± 0.002	0.040 ± 0.008
	BC	1	4	—	0.025 ± 0.005	0.031 ± 0.004
	BNN	1	8	—	0.037 ± 0.002	0.038 ± 0.004
mnistLH	PBGNet	1	8	0.186 ± 0.028	0.091 ± 0.037	0.092 ± 0.036
	ABNet	3	4	0.162 ± 0.001	0.056 ± 0.001	0.058 ± 0.002
	PBGNet _ℓ	3	8	1.000 ± 0.000	0.018 ± 0.003	0.038 ± 0.002
	ABNet _ℓ	2	8	0.998 ± 0.003	0.025 ± 0.008	0.042 ± 0.006
	EBP	3	8	—	0.016 ± 0.002	0.043 ± 0.002
	BC	2	8	—	0.023 ± 0.002	0.035 ± 0.001
	BNN	1	2	—	0.123 ± 0.005	0.133 ± 0.004

Questions / commentaires

Merci pour votre attention :) !