S. Gerchinovitz

VC-dimension des réseaux de neurones feed-forward
Contrôle de l'erreur de généralisation

Intro

Sur les 3 termes : erreur d'approximation / de généralisation / d'optimisation, on va s'intéresser au 2ème.

Plus précisément : on se restreint au problème de classification binaire :

- A partir d'un échantillon $(X_1,Y_1),\dots,(X_m,Y_m) \overset{iid}{\sim} P_{X,Y}$ où $P_{X,Y}$ est une loi sur $\mathcal{X}\times\{0,1\}$ proposer un classifieur $\hat{h}_m : \mathcal{X} \to \{0,1\}$ tel que $\mathbb{P}\left(\hat{h}_m(X)\neq Y\right)$ soit le plus petit possible.
- On va étudier des classifieurs $\hat{h}_m$ de la forme $\hat{h}_m = \mathbb{1}_{\{\hat{f}_m>0\}} = \operatorname{sgn}(\hat{f}_m)$ où $\hat{f}_m$ est un réseau de neurones feed-forward.

  On cherchera à contrôler l'excès de risque

$$\mathbb{P}\left(\hat{h}_m(X)\neq Y\right) - \inf_{h\in H} \mathbb{P}(h(X)\neq Y)$$

où $H = \operatorname{sgn} F = \{\operatorname{sgn}(f) : f\in F\}$

avec F = toutes les fonctions qu'on peut obtenir à partir d'une architecture de réseau feed-forward fixée (on fait juste varier les biais et les poids sur les arêtes reliant les neurones).

1/ VC-dimension et borne de risque (quelques rappels)

Cf cours de l'X de C. Giraud "Fondements mathématiques de l'apprentissage statistique".

Déf : Soit H un ensemble de classifieurs $h : \mathcal{X} \to \{0,1\}$.

On appelle "coefficient d'éclatement" (ou : coefficient de pulvérisation, shattering coefficient, growth function) la quantité :

$$\pi_H(m) = \max_{x_1,\dots,x_m \in \mathcal{X}} \operatorname{card} \left\{ \underbrace{(h(x_1),\dots,h(x_m))}_{\in \{0,1\}^m} : h \in H \right\} \qquad (m \in \mathbb{N}^*)$$

Rem : $\pi_H(m) \leq 2^m$ et $\pi_H(m) \leq \operatorname{card} H$ si H fini.

Déf : On appelle <u>dimension de Vapnik-Chervonenkis de H</u>, notée $VCdim(H)$, la quantité (avec la convention $\pi_H(0) = 1$) :

$$VCdim(H) := \sup \{ m \in \mathbb{N} : \pi_H(m) = 2^m \} \in \mathbb{N} \cup \{+\infty\}.$$

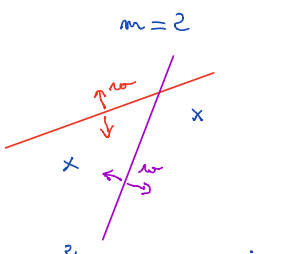
$$\pi_H(m) = 2^m \iff \exists x_1,\dots,x_m \in \mathcal{X},\ \forall \sigma \in \{0,1\}^m,\ \exists h \in H,\ \forall i \in \{1,\dots,m\},\ h(x_i) = \sigma_i.$$

Interprétation : $VCdim(H)$ est la taille du plus grand échantillon $(x_1,\dots,x_m)$ que H peut "éclater", i.e., tel qu'on puisse obtenir les 2^m vecteurs de signes possibles en appliquant des fonctions $h \in H$ à l'échantillon $(x_1,\dots,x_m)$.

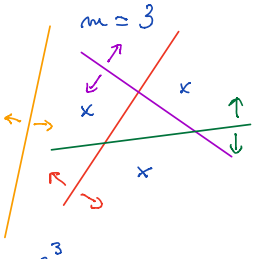
Ex : $H := \{ x \in \mathbb{R}^d \mapsto \operatorname{sgn}(\langle w, x \rangle + b) \ ; \ w \in \mathbb{R}^d, b \in \mathbb{R} \}$ "perceptron"

$VCdim(H) = d+1$

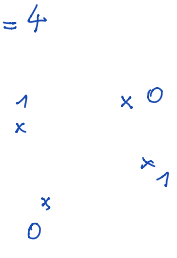
<u>Schéma pour $d=2$:</u>



2^2 vecteurs de signes possibles



2^3 vecteurs de signes possibles



Ce vecteur de signes n'est pas atteignable

Lemme de Sauer : Soit H tel que $0 < V := \mathrm{VCdim}(H) < +\infty$. $\forall m \geq 1$,

$$\pi_H(m) \leq \sum_{i=0}^{V} \binom{m}{i}$$

$$\begin{cases} = 2^m & \text{si } m \leq V \\ \leq \left(\frac{me}{V}\right)^V & \text{si } m \geq V \quad \text{(transition exp} \rightsquigarrow \text{poly en } m) \end{cases}$$

Preuve par récurrence.

Proposition 1 (majoration du risque de classification de l'ERM)

Soit $H \subset \{0,1\}^{\mathcal{X}}$ tel que $V := \mathrm{VCdim}(H) \in \mathbb{N}^*$.

Alors :

(a) $$\mathbb{E}_{\text{sur } (X_i, Y_i)_{1 \leq i \leq m}} \left[\sup_{h \in H} \left| \mathbb{P}(h(X) \neq Y) - \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}_{h(X_i) \neq Y_i} \right| \right] \leq 2 \sqrt{\frac{2 \log(2 \pi_H(m))}{m}}$$

Cela contrôle les déviations uniformes (sur H) du risque empirique autour du vrai risque.

(b) Le minimiseur $\hat{h}_m$ du risque empirique (ERM) : $\hat{h}_m \in \underset{h \in H}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}_{Y_i \neq h(X_i)}$

vérifie :

$$\mathbb{P}(\hat{h}_m(X) \neq Y) - \inf_{h \in H} \mathbb{P}(h(X) \neq Y) \leq 4 \sqrt{\frac{2 \log(2 \pi_H(m))}{m}}$$

Rem :

- $\forall m \geq V$, $\log \pi_H(m) \overset{\text{Sauer}}{\leq} V \log\left(\frac{em}{V}\right)$ donc la borne de risque est en $\sqrt{\frac{V \log(em/V)}{m}}$ (on pourrait enlever ici le facteur log avec la technique de "chaînage").
- En combinant la preuve avec l'inégalité de Mc Diarmid, on obtient une borne en grande proba en $\sqrt{\frac{V \log(em/V)}{m}} + \sqrt{\frac{\log(1/\delta)}{m}}$ avec proba $\geq 1 - \delta$.

Proposition 2 (minoration de l'excès de risque dans le pire des cas, pour tout classifieur) [cf. annexe]

Soit $H \subset \{0,1\}^{\mathcal{X}}$ de VC-dimension $V \in \mathbb{N}^*$.

Alors : $\forall m \geq c_1 V$,

$$\inf_{\hat{g}_m} \sup_{P \in \mathcal{M}_1^+(\mathcal{X} \times \{0,1\})} \left\{ \mathbb{P}\left(\hat{g}_m\left((X_i,Y_i)_{1\leq i\leq m}, X\right) \neq Y\right) - \inf_{g \in H} \mathbb{P}(g(X) \neq Y) \right\} \geq c_2 \sqrt{\frac{V}{m}}$$

où $c_1, c_2 > 0$ sont des constantes absolues et où l'infimum est pris sur tous les classifieurs $\hat{g}_m : \underbrace{(\mathcal{X} \times \{0,1\})^m}_{\text{pour l'échantillon}} \times \mathcal{X} \to \{0,1\}$.

Cette proposition signifie que, quel que soit le classifieur $\hat{g}_m$ considéré, il existe une loi jointe P sur $\mathcal{X} \times \{0,1\}$ qui rend l'excès de risque de $\hat{g}_m$ au moins de l'ordre de $\sqrt{V/m}$. Cette borne inférieure s'appelle une "borne inférieure minimax". Elle permet d'identifier l'ordre de grandeur de l'excès de risque dans le pire des cas du meilleur classifieur.

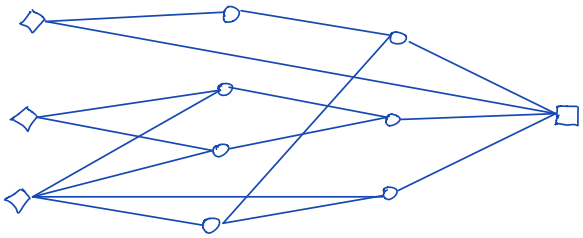
2/ Contrôle de la VC-dimension d'un réseau de neurones feed-forward.

"Nearly-tight VC-dimension and pseudodimension bounds for piecewise linear neural networks", Bartlett, Harvey, Liaw and Mehrabian, COLT 2017.

On va majorer la VC-dimension d'un réseau de neurones feed-forward (architecture fixée) en fonction de
- L : nb de couches (layers)
- U : nb de neurones (computation units)
- W : nb de poids (weights)

Définition formelle d'un réseau de neurones feed-forward :
- fonction d'activation $\sigma : \mathbb{R} \to \mathbb{R}$
- graphe orienté acyclique (DAG) $G = (S, A)$

- vecteur de poids : un réel par arête et un réel par neurone



Le graphe est orienté de gauche à droite.

nœuds d'entrée | 1ère couche cachée | 2e couche cachée | 1 seul nœud de sortie

U neurones

couche 0 | couche 1 | couche 2 | couche $L=3$

$L :=$ longueur maximale d'un chemin dans G.

$W = |A| + U$

$|A|$: réels sur les arêtes ; U : biais pour chaque neurone

Sur l'exemple ci-dessus : $L=3$, $U=8$, $W=24$

Les numéros de couche sont définis formellement ainsi :

- couche 0 $=$ { nœuds de degré entrant nul }
- couche l $=$ { nœuds qui admettent au moins un prédécesseur de la couche $l-1$, éventuellement d'autres prédécesseurs des couches $0, 1, \dots, l-2$; et aucun autre prédécesseur }

$l = 1, \dots, L$

On suppose qu'il existe un unique nœud de degré sortant nul (nœud de sortie, couche L).

Rem : on autorise des connexions entre couches non-consécutives.

Fonctionnement de chacun des $U-1$ neurones cachés : en entrée : $x \in \mathbb{R}^{\text{degré entrant}}$

en sortie : $\sigma(\langle w, x \rangle + b)$ (b : biais ; w : poids sur les arcs entrants)

Neurone de sortie : $x \mapsto \langle w, x \rangle + b$

On va se restreindre au cas "polynomial par morceaux" : on suppose que $\sigma : \mathbb{R} \to \mathbb{R}$ est polynomiale sur chacun de $p+1$ morceaux ($p \geq 1$) $I_1, I_2, \dots, I_{p+1}$ intervalles d'intérieurs non-vides, 2 à 2 disjoints, d'union $\mathbb{R}$.

Ex : $\sigma = \mathrm{ReLU}$ $\quad I_1 = (-\infty, 0)$ $\quad I_2 = [0, +\infty)$

$$\sigma(x) = \begin{cases} 0 & \text{sur } I_1 \\ x & \text{sur } I_2 \end{cases}$$

Théorème 1 (Bartlett, Harvey, Liaw et Mehrabian, COLT 2017)

Soit $L \geq 1$, $U \geq 3$, $d \geq 0$, $p \geq 1$ et $W \geq U \geq L$.

Soit un réseau feed-forward avec W paramètres, U neurones, L couches tel que décrit ci-dessus. On note k_i le nb de neurones sur la i-ème couche ($i = 1, \dots, L$). On suppose que les neurones cachés ont une fonction d'activation σ polynomiale sur $p+1$ morceaux ($p \geq 1$) de degré majoré par $d \in \mathbb{N}$. En sortie : fonction d'activation = identité

On pose : pour $i \in \{1, \dots, L\}$,

- Si $d = 0$, W_i = nb de paramètres utiles au calcul de tous les neurones de la couche i
  $= (\text{nb d'arcs entrants vers la couche } i) + k_i$ (↙ poids sur les arcs, ↙ biais)
- Si $d \geq 1$, W_i = nb de paramètres (poids et biais) utiles au calcul de tous les neurones des couches 1 à i.

On pose $\bar{L} := \frac{1}{W} \sum_{i=1}^{L} W_i \in [1; L]$

→ égal à 1 si $d = 0$

→ proche de L si $d \geq 1$ et si les neurones sont concentrés sur les premières couches (ou même uniformément répartis entre les couches)

et $R := \sum_{i=1}^{L} k_i \left(1+(i-1)\,d^{i-1}\right) \leq U + U(L-1)\,d^{L-1}$ $\begin{cases} = U & \text{si } d=0 \\ \leq UL\,d^{L-1} & \text{si } d \geq 1 \end{cases}$

Alors : la classe $\mathcal{F}$ de toutes les fonctions $f_{a \in \mathbb{R}^W} : \mathbb{R}^{\text{(entrée)}} \to \mathbb{R}$ (a : paramètres du réseau) vérifie : $\forall m \geq W$,

$$\Pi_{\text{sgn}(\mathcal{F})}(m) \leq \prod_{i=1}^{L} 2\left(\frac{2em\,k_i\,p\,(1+(i-1)\,d^{i-1})}{W_i}\right)^{W_i} \qquad (1)$$

$$\leq \left(4em\,p\left(1+(L-1)\,d^{L-1}\right)\right)^{\sum_{i=1}^{L} W_i} \qquad (2)$$

Par ailleurs :

$$\text{VCdim}(\text{sgn}(\mathcal{F})) \leq L + \tilde{L}W \log_2\left(4epR \underbrace{\log_2(2epR)}_{\text{négligeable}}\right) \qquad (3)$$

$$= O\left(\tilde{L}W \log(pU) + \tilde{L}LW \log d\right) \quad \text{si } d \geq 1$$

En particulier :

- Si $d=0$, $\text{VCdim}(\text{sgn}(\mathcal{F})) \leq L + W\log_2\left(4epU \log_2(2epU)\right)$
  $= O\left(W \log(pU)\right) = O\left(W \log(pW)\right)$
- Si $d=1$, $\text{VCdim}(\text{sgn}(\mathcal{F})) = \underbrace{O\left(\tilde{L}W \log(pU)\right)}_{\text{meilleur que la borne}}$

$O\left(\min\left\{W^2, WL\log W + \underbrace{L^2W}_{\text{inutile quand } d=1}\right\}\right)$ connue jusqu'alors.

NB : Cette borne est presque optimale dans le fin des cas : pour $d=1$, $W \geq cL$ et $L \geq c$ il existe un réseau ReLU à $\leq L$ couches et $\leq W$ paramètres tel que $\text{VCdim}(\text{sgn}(\mathcal{F})) \geq WL \log\left(\frac{W}{L}\right)/c$.
($c>0$ constante absolue).

Preuve : repose sur le résultat de géométrie algébrique suivant :

Lemme 1 (prouvé par ex dans Anthony et Bartlett '99, theorem 8.3)
Soit $p_1, \dots, p_m$ des polynômes de degré au plus $d \geq 1$ en $n \leq m$ variables.
$K := \operatorname{card}\left\{ (\operatorname{sgn}(p_1(x)), \dots, \operatorname{sgn}(p_m(x))) : x \in \mathbb{R}^n \right\}$
le nb de vecteurs de signes possibles.
Alors : $K \leq 2\left(2em d/n\right)^n$.

Notons $f(x, a)$ la sortie du réseau pour l'entrée $x \in \mathcal{X}$ et le vecteur de paramètres $a \in \mathbb{R}^W$.

$\mathcal{X} = \mathbb{R}^{\text{dim des entrées}}$

Soit $x_1, \dots, x_m \in \mathcal{X}$. Afin de majorer $\Pi_{\operatorname{sgn}(\mathcal{F})}(m)$, majorons

$$\operatorname{card}\left\{ (\operatorname{sgn}(f(x_1, a)), \dots, \operatorname{sgn}(f(x_m, a))) : a \in \mathbb{R}^W \right\}$$

$$\leq \sum_{i=1}^{N} \operatorname{card}\left\{ (\operatorname{sgn}(f(x_1, a)), \dots, \operatorname{sgn}(f(x_m, a))) : a \in P_i \right\}$$

où $P_1, \dots, P_N$ est une partition de $\mathbb{R}^W$ qui sera choisie de sorte que les m fonctions $a \mapsto f(x_j, a)$ soient polynomiales sur chaque cellule P_i. Il suffira alors d'appliquer le lemme 1.

Tout l'exercice se réduit à la construction d'une bonne partition.

Partitions construites par récurrence : $\mathcal{S}_0, \mathcal{S}_1, \dots, \mathcal{S}_{L-1}$ partitions de $\mathbb{R}^W$ telles que :

partition finale : $\mathcal{S}_{L-1} = \{P_1, \dots, P_N\}$

(a) les partitions sont emboîtées : chaque $S \in \mathcal{S}_i$ est une réunion d'un ou plusieurs $S' \in \mathcal{S}_{i+1}$ $(0 \le i \le L-2)$

(b) $\mathrm{card}(\mathcal{S}_0) = 1$ $(\mathcal{S}_0 = \{\mathbb{R}^W\})$ et : $\forall i \in \{1, \dots, L-1\}$,

$$\frac{|\mathcal{S}_i|}{|\mathcal{S}_{i-1}|} \le 2 \left(\frac{2em k_i p (1 + (i-1) d^{i-1})}{W_i} \right)^{W_i}$$

(c) Pour tout $i \in \{0, \dots, L-1\}$, tout $S \in \mathcal{S}_i$, tout $j \in \{1, \dots, m\}$, la sortie d'un neurone de la i-ième couche (relatif à l'entrée x_j) est une fonction polynomiale de W_i variables de $a \in S$, de degré $\le i d^i$.

W_i : si $d=0$: les paramètres des neurones de la couche i ; si $d \ge 1$: ——— des couches $1, \dots, i$

Récurrence :

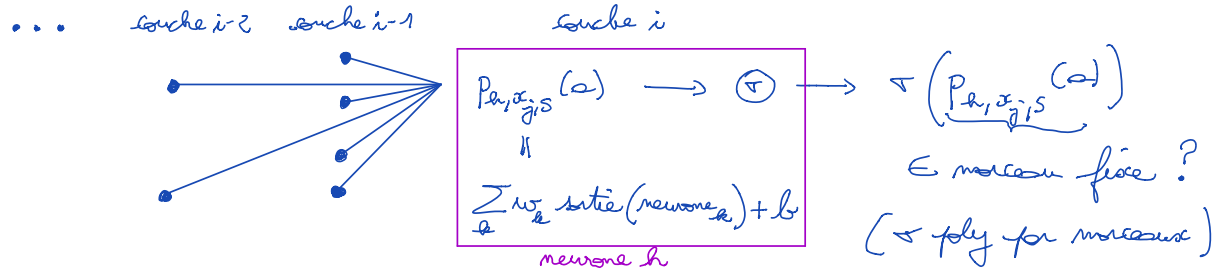
- $i = 0$: $\mathcal{S}_0 = \{\mathbb{R}^W\}$ — propriété (c) OK (en posant $W_0 = 0$)
  sortie d'un neurone d'entrée = fonction constante en $a \in \mathbb{R}^W$
- Soit $1 \le i \le L-1$. Supposons avoir construit des partitions emboîtées $\mathcal{S}_0, \mathcal{S}_1, \dots, \mathcal{S}_{i-1}$ vérifiant (b) et (c). Construisons $\mathcal{S}_i$.

On note $p_{k, x_j, S}(a)$ l'entrée (juste avant σ) du k-ième neurone $(k = 1, \dots, k_i)$ de la couche i, pour l'entrée x_j, comme fonction de $a \in S$ avec $S \in \mathcal{S}_{i-1}$.

D'après l'hyp de récurrence (c), puisque $p_{k, x_j, S}(a)$ est de la forme $\sum_k w_k \, \mathrm{sortie}(\mathrm{neurone}_k) + b$, et puisque les partitions sont emboîtées, on a : $p_{k, x_j, S}$ est polynomiale sur S, de degré $\le 1 + (i-1) d^{i-1}$, et dépend d'au plus W_i variables.

NB : OK si $d = 0$.

À cause de σ, la sortie du neurone h est polynomiale par morceaux sur S. On va découper S en sous-cellules pour que la sortie soit polynomiale sur chacune des sous-cellules, et ce $\forall$ neurone h, $\forall$ entrée x_j.



Soit $t_1 < t_2 < \dots < t_p$ les coupures des morceaux $I_1, \dots, I_{p+1}$ de σ.
Considérons les polynômes $\pm\left(P_{h,x_j,S}(a) - t_r\right)$, $h \in \{1,\dots,k_i\}$, $j \in \{1,\dots,m\}$, $r \in \{1,\dots,p\}$

($\mathrm{sgn}(u) = \mathbb{1}_{u>0}$) + si I_{r+1} est ouvert en t_r ; − si I_{r+1} est fermé en t_r

D'après le lemme 1, cet ensemble de polynômes sur $\mathbb{R}^W$ atteint au plus

$$\pi := 2\left(2e\,(k_i\, m\, p)\,(1+(i-1)d^{i-1})/W_i\right)^{W_i} \qquad (\text{NB} : m \geq W \geq W_i)$$

(k_i : nb de polynômes ; $1+(i-1)d^{i-1}$: degré ; W_i : nb de variables effectives)

vecteurs de signes différents $\left(\mathrm{sgn}\left[\pm\left(P_{h,x_j,S}(a) - t_r\right)\right]\right)_{h,j,r}$ quand $a \in \mathbb{R}^W$ et donc quand $a \in S$. On peut donc partitionner S en $\leq \pi$ sous-cellules de sorte que, sur chacune de ces sous-cellules, les $P_{h,x_j,S}(a)$ ne changent pas de morceau (de σ) lorsque a varie dans cette sous-cellule.

NB : Ces $\leq \pi$ sous-cellules de S sont les mêmes pour tous les neurones h et toutes les entrées x_j (utile pour la récurrence).

$\Rightarrow$ On obtient une nouvelle partition $\mathcal{I}_i$ de cardinal $\leq \pi \cdot \mathrm{card}(\mathcal{I}_{i-1})$

La propriété (b) est vérifiée.

Propriété (c) ?

$\forall S' \in \mathcal{J}_{i-1}$, la sortie du neurone $h \in \{1, \dots, k_i\}$:

$a \in S' \mapsto \sigma\left(P_{h,x_j,S}(a)\right)$ est polynomiale de W_i variables

de degré $\leq d\left(1+(i-1)d^{i-1}\right) \leq i d^i$

multiplication du degré en entrée par le degré de σ.

donc propriété (c) OK.

- Cela clôt la récurrence $\Rightarrow$ partitions emboîtées $\mathcal{J}_0, \mathcal{J}_1, \dots, \mathcal{J}_{L-1}$ vérifiant (b) et (c).

En particulier, $\mathcal{J}_{L-1}$ est une partition de $\mathbb{R}^W$ telle que la sortie de chaque neurone des couches $0, \dots, L-1$ est polynomiale de degré $\leq (L-1)d^{L-1}$ sur chaque $S \in \mathcal{J}_{L-1}$, pour tout $j \in \{1, \dots, m\}$.

car les $\mathcal{J}_i$ sont emboîtées

Ainsi, pour chaque cellule $S \in \mathcal{J}_{L-1}$ et chaque entrée $x_j \in \mathcal{X}$, la fonction $a \in S \mapsto f(x_j, a)$ en sortie de réseau est polynomiale de degré $\leq 1 + (L-1)d^{L-1}$ et ne dépend que de W_L variables.

à cause de la combinaison linéaire calculée par le neurone de sortie

D'où, d'après le lemme 1 :

$$\operatorname{card}\left\{\left(\operatorname{sgn}(f(x_1,a)), \dots, \operatorname{sgn}(f(x_m,a))\right) : a \in S\right\}$$

$$\leq 2\left(\frac{2em\left(1+(L-1)d^{L-1}\right)}{W_L}\right)^{W_L} \qquad \left(\text{NB : } m \geq W \geq W_L\right)$$

Dès lors :

$$\operatorname{card}\left\{\left(\operatorname{sgn}(f(x_1,a)), \dots, \operatorname{sgn}(f(x_m,a))\right) : a \in \mathbb{R}^W\right\}$$

$$\leq \sum_{S \in \mathcal{J}_{L-1}} \operatorname{card}\left\{\left(\operatorname{sgn}(f(x_1,a)), \dots, \operatorname{sgn}(f(x_m,a))\right) : a \in S\right\}$$

$$\leq \mathrm{card}\left(\mathcal{Y}_{L-1}\right) \times 2^{L}\left(\frac{2emp\left(1+(L-1)d^{L-1}\right)}{W_L}\right)^{W_L} \qquad (4)$$

Or, d'après la propriété (6) :

$$\mathrm{card}\left(\mathcal{Y}_{L-1}\right) \leq \prod_{i=1}^{L-1} 2^{}\left(\frac{2emk_i p\left(1+(i-1)d^{i-1}\right)}{W_i}\right)^{W_i}$$

et donc (car (4) est valable pour tous $x_1, \dots, x_m \in \mathcal{X}$) :

$$\Pi_{sgn(\mathcal{F})}(m) \leq \prod_{i=1}^{L} 2^{}\left(\frac{2emk_i p\left(1+(i-1)d^{i-1}\right)}{W_i}\right)^{W_i} \quad \rightsquigarrow (1) \text{ OK}$$

Rappel (moyenne géométrique ≤ moyenne arithmétique)

- $\forall y_1, \dots, y_k > 0$, $\forall \alpha_1, \dots, \alpha_k \geq 0$ tq $\sum_{i=1}^{k} \alpha_i = 1$,
  $\prod_{i=1}^{k} y_i^{\alpha_i} \leq \sum_{i=1}^{k} \alpha_i y_i$ (preuve par Jensen)
- D'où : $\forall y_1, \dots, y_k > 0$, $\forall a_1, \dots, a_k \geq 0$ tq $\sum_{i=1}^{k} a_i > 0$,
  $\prod_{i=1}^{k} y_i^{a_i} \leq \left(\frac{\sum_{i=1}^{k} a_i y_i}{\sum_{i=1}^{k} a_i}\right)^{\sum_{i=1}^{k} a_i}$

$$\Pi_{sgn(\mathcal{F})}(m) \leq 2^{L}\left(\frac{2emp\sum_{i=1}^{L} k_i\left(1+(i-1)d^{i-1}\right)}{\sum_{i=1}^{L} W_i}\right)^{\sum_{i=1}^{L} W_i}$$

$$= 2^{L}\left(\frac{2empR}{\sum_{i=1}^{L} W_i}\right)^{\sum_{i=1}^{L} W_i} \text{ par définition de } R \qquad (5)$$

$$\leq \left(\frac{4emp\left(1+(L-1)d^{L-1}\right)\sum_{i=1}^{L} k_i}{\sum_{i=1}^{L} W_i}\right)^{\sum_{i=1}^{L} W_i} \text{ car } L \leq \sum_{i=1}^{L} W_i$$

$$\leq \left(4emp\left(1+(L-1)d^{L-1}\right)\right)^{\sum_{i=1}^{L} W_i} \text{ car } \sum_{i=1}^{L} k_i \leq \sum_{i=1}^{L} W_i \quad \rightsquigarrow (2) \text{ OK}$$

Pour prouver la borne (3) sur $VCdim(sgn(\mathcal{F}))$, on va combiner (5) et le lemme suivant :

Lemme 2 : Soit $r \geq 16$ et $w \geq t > 0$

Alors, pour tout $m > t + w \log_2(2r \log_2 r) =: x_0$, on a

on a $2^m > 2^t \left(\frac{mr}{w}\right)^w$

Suite de la preuve du théorème 1 : d'après (5) et le lemme 2 avec $t = L$, $w = \sum_{i=1}^{L} W_i$ et $r = 2epR \geq 2eU \overset{U \geq 3}{\geq} 16$, on a :

$$\forall m > V := L + \left(\sum_{i=1}^{L} W_i\right) \log_2\left(4epR \log_2(2epR)\right) \geq W, \quad \pi_H(m) < 2^m$$

et donc $VCdim(sgn(\mathcal{F})) \leq V$ par définition de la VC-dimension, ce qui prouve (3). ∎

Preuve du lemme 2 : Soit $x_0 := t + w \log_2(2r \log_2 r)$.

Montrons que, pour tout $x > x_0$,

$$2^x > 2^t \left(\frac{xr}{w}\right)^w \iff \underbrace{x - t - w \log_2\left(\frac{xr}{w}\right)}_{f(x) :=} > 0$$

Il suffit de montrer que $f(x_0) \geq 0$ et $f'(x) > 0$ $\forall x \geq x_0$.

- $f(x_0) \geq 0 \iff x_0 - t - w \log_2\left(\frac{x_0 r}{w}\right) \geq 0$
  $\iff w \log_2(2r \log_2 r) \geq w \log_2\left(\frac{x_0 r}{w}\right)$ (déf de x_0)

$$\Leftrightarrow \quad 2\log_2(r) \geq \frac{x_0}{w} = \frac{t}{w} + \log_2(2r\log_2 r)$$
$$\Leftrightarrow \quad \log_2\left(\frac{r^2}{2r\log_2 r}\right) \geq \frac{t}{w}$$
$$\Leftrightarrow \quad \frac{r}{2\log_2 r} \geq 2^{t/w}$$

ce qui est vrai car $\frac{r}{2\log_2 r} \geq 2$ car $r \geq 16$ et $\frac{t}{w} \leq 1$.

- Pour tout $x > x_0$, $f'(x) = 1 - \frac{w}{x \log 2}$

d'où $f'(x) > 0 \Leftrightarrow x > \frac{w}{\log 2}$

ce qui est vrai car $x_0 \geq w\log_2(2r\log_2 r) \overset{r \geq 16}{>} w/\log 2$

On en déduit que $f(x) > 0$ pour tout $x > x_0$. ∎

Remarque : cette borne de complexité, quasi-optimale dans le pire des cas, n'explique vraisemblablement pas les bonnes performances pratiques des réseaux de neurones, car elles sont pessimistes.

Ex de papiers proposant des bornes dépendant plus finement du réseau considéré :

- "Spectrally-normalized margin bounds for neural networks", Bartlett, Foster, Telgarsky, NIPS 2017.
- "Size-independent sample complexity of neural networks", Golowich, Rakhlin, Shamir, COLT 2018.

La recherche est encore très active dans le domaine ; cf. par ex :

- "Uniform convergence may be unable to explain generalization in deep learning", Nagarajan and Kolter, NeurIPS 2019.