TER M1 :
NN and RKHS

Matthieu Denis

17 juin 2021

Table des matières

1 Reproducing Kernel Hilbert space (RKHS) et leurs applications **3**
1.1 Cadre théorique 3
1.2 Applications des RKHS en machine learning 6
1.2.1 Representer theorem 6
1.2.2 Exemple 1 : Kernel Ridge Regression 6
1.2.3 Exemple 2 : Support Vector Machine (SVM) 7
1.2.4 Le Kernel Trick 8

2 Introduction : Réseau de neurone simple **8**
2.1 Lois suivant la largeur des couches m 8
2.2 Gradients 9
2.3 Descente de gradient 10
2.3.1 Choix de η 10
2.3.2 Ordres de grandeur des écarts relatifs 10
2.3.3 Choix de α 11

1 Reproducing Kernel Hilbert space (RKHS) et leurs applications

1.1 Cadre théorique

Nous allons dans cette section s'intéresser aux RKHS, des espaces de Hilbert réels qui satisfont certaines propriétés, et qui ont des applications intéressantes en machine learning. Par exemple, on montrera un théorème qui nous permet de simplifier un problème de minimisation de risque empirique de dimension infini à un problème en dimension fini. Ou encore des applications dans plusieurs algorithmes de ML, les transformant d'algorithmes linéaires à non-linéaires à très faible prix, et d'autres encore.

Soit X un ensemble quelconque, H un espace de Hilbert de fonctions réelles sur X, muni de l'addition point par point ainsi que de la multiplication par scalaire point par point. On introduit aussi une forme linéaire qui à chaque fonction de H l'évalue en un certain point $x \in X$,

$$L_x : f \mapsto f(x) \; \forall f \in H$$

Définition 1 (RKHS)**.** On dit d'un espace de Hilbert qu'il est un RKHS si $\forall x \in X$, L_x est continue sur H, ou encore si L_x est bornée sur H, i.e

$$\forall x \in X, \; \exists M_x > 0, \; \forall f \in H \text{ t.q } |L_x(f)| \coloneqq |f(x)| \leq M_x ||f||_H$$

Dans ce qui suit, H sera un RKHS.

Propriété 1 (Convergence en norme dans un RKHS implique pointwise convergence)**.** Soient f_n, $f \in H$. Si $f_n \overset{H}{\to} f$, alors $\forall x \in X, \; f_n(x) \to f(x)$.

Démonstration.

$$\forall x \in X, \; |f_n(x) - f(x)| = |L_x(f_n) - L_x(f)| = |L_x(f_n - f)| \leq M_x ||f_n - f||_H$$

■

Définition 2 (Noyau / Kernel)**.** Une fonction $k : X \times X \to \mathbb{R}$ est un noyau si

$$\exists \phi : X \to H \text{ t.q } k(x, y) = \langle \phi(x), \phi(y) \rangle_H \; \forall x, y \in X$$

Propriété 2. Tout noyau k est symétrique défini positif.

Démonstration.
- Symétrie : découle de la symétrie du produit scalaire.
- Défini positif :
  $\forall x_1, \cdots, x_n \in X, \forall c_1, \cdots, c_n \in X$

$$\sum_{i,j=1}^{n} c_i c_j k(x_i, x_j) = \left\langle \sum_{i=1}^{n} c_i \phi(x_i), \sum_{j=1}^{n} c_j \phi(x_j) \right\rangle_H = \left\| \sum_{i=1}^{n} c_i \phi(x_i) \right\|_H^2 \geq 0$$

■

Définition 3 (Noyau reproduisant / Reproducing kernel)**.** Une fonction $k : X \times X \to \mathbb{R}$ est un noyau reproduisant de H si $\forall x \in X$, $f \in H$:
- $k(\cdot, x) \in H$
- $f(x) = \langle f, k(\cdot, x) \rangle_H$

Propriété 3. Tout noyau reproduisant k est un noyau.

Démonstration. En prenant $f = k(\cdot, y) \in H$ pour un certain $y \in X$ dans la définition, on a en particulier $\forall x, y \in X\ k(x, y) = \langle k(\cdot, x), k(\cdot, y)\rangle_H$. Ici $\phi(u) = k(\cdot, u)\ \forall u \in X$. ■

Théorème 1 (Théorème de représentation de Riesz)**.** Soient :

- H un espace de Hilbert réel, muni de son produit scalaire $\langle \cdot, \cdot \rangle_H$
- $L \in H'$ une forme linéaire continue sur H.

Alors

$$\exists!\, g \in H,\ \forall f \in H,\ L(f) = \langle f, g\rangle_H$$

Propriété 4 (Existence et Unicité)**.** Si H est un RKHS, alors il existe un unique noyau reproduisant de H.

Démonstration. Montrons l'existence puis l'unicité :

- On applique le théorème de Riesz à L_x :

  $$\forall x \in X,\ \exists!\, k_x \in H,\ \forall f \in H,\ f(x) = L_x(f) = \langle f, k_x\rangle_H$$

  Soit $k(y, x) \coloneqq k_x(y)\ \forall x, y \in X$, alors $\forall x \in X,\ f \in H,\ k(\cdot, x) \in H$ et $f(x) = \langle f, k(\cdot, x)\rangle_H$.
  Donc k est un noyau reproduisant de H.
- Soient k_1 et k_2 deux noyau reproduisant de H. Alors $\forall x \in X,\ f \in H$

  $$\langle f, k_1(\cdot, x) - k_2(\cdot, x)\rangle_H = f(x) - f(x) = 0$$

  En prenant $f = k_1(\cdot, x) - k_2(\cdot, x)$, on a :

  $$||k_1(\cdot, x) - k_2(\cdot, x)||_H^2 = 0$$

  C'est-à-dire que $k_1 = k_2$: le noyau reproduisant de H est unique.

■

Théorème 2 (Lien des deux visions)**.** H est un RKHS si et seulement si il existe un unique noyau reproduisant de H.

Démonstration.
- ($\Rightarrow$) Donné par 4
- ($\Leftarrow$) $\forall x \in X,\ f \in H$

  $$|L_x(f)| = |f(x)| = |\langle f, k(\cdot, x)\rangle_H| \leq ||f||_H ||k(\cdot, x)||_H = \underbrace{\sqrt{k(x, x)}}_{M_x} ||f||_H$$

  M_x ne dépendant pas de f, on a bien l'inégalité $\forall x \in X,\ \exists M_x > 0,\ \forall f \in H$, i.e H est un RKHS.

■

Théorème 3 (Théorème de Moore-Aronszajn)**.** Soit K un kernel. Alors il existe un unique espace de Hilbert H de fonctions sur X pour lequel K est un noyau reproduisant.

Démonstration. $\forall x \in X$, on pose $K_x(\cdot) \coloneqq K(x, \cdot)$. Soit H_0 le sous-espace vectoriel engendré par $\{K_x : x \in X\}$. Sous couvert d'existence de H, la reproducing property de K nous assure que $\forall x, y \in X,\ K(x, y) = \langle K_x, K_y\rangle_H$. On a de même :

$$\left\langle \sum_{i=1}^{n} a_i K_{x_i}, \sum_{j=1}^{m} b_j K_{x_j} \right\rangle_H = \sum_{i=1}^{n}\sum_{j=1}^{m} a_i b_j K(x_i, x_j)$$

Il est alors naturel de définir pour toutes fonctions de H_0 : $f \coloneqq \sum_{i=1}^{n} a_i K_{x_i}$ et $g \coloneqq \sum_{j=1}^{m} b_i K_{y_j}$

$$\langle f, g \rangle_{H_0} \coloneqq \sum_{i=1}^{n} \sum_{j=1}^{m} a_i b_j K(x_i, y_j) = \sum_{i=1}^{n} a_i g(x_i) = \sum_{j=1}^{m} b_j f(y_j)$$

Les deux dernières égalités nous assure que $\langle \cdot, \cdot \rangle_{H_0}$ est bien défini. De plus, il est billinéaire, symétrique et positif, et on a la reproducing property :

$$\forall x \in X,\ f \in H_0,\ \langle f, K_x \rangle_{H_0} = \sum_{i=1}^{n} a_i K_x(x_i) = \sum_{i=1}^{n} a_i K_{x_i}(x) = f(x)$$

Il reste à montrer que $||f||_{H_0} = 0 \Rightarrow f = 0$ pour que ca soit un produit scalaire : **CAUCHY SCHWARZ POUR SEMI INNER PRODUCT**

$$|f(x)| = |\langle f, K_x \rangle_{H_0}| \leq ||f||_{H_0} ||K_x||_{H_0} = ||f||_{H_0} \sqrt{K(x, x)}$$

Donc $\langle \cdot, \cdot \rangle_{H_0}$ est un produit scalaire.

On complète H_0 avec le produit scalaire $\langle \cdot, \cdot \rangle_{H_0}$ en H, et toutes les propriétés sont gardées.

Unicité : supposons G un autre espace de Hilbert pour lequel K est un noyau reproduisant.

- $(\subseteq)$:
  On a
  $$\forall x, y \in X,\ \langle K_x, K_y \rangle_H = K(x, y) = \langle K_x, K_y \rangle_G$$
  Par linéarité, $\langle \cdot, \cdot \rangle_H = \langle \cdot, \cdot \rangle_G$ sur tout H_0, i.e $H_0 \subseteq G$. Donc $H \subseteq G$ car G est complet.
- $(\supseteq)$:
  Soit $f \in G$. Comme H est un sous-espace fermé de G, on peut écrire $f = f_H + f_{H^\perp}$ par le théorème de décomposition orthogonal. De plus comme K est un noyau reproductif de G et H on a $\forall x \in X$:
  $$f(x) = \langle K_x, f \rangle_G = \langle K_x, f_H \rangle_G + \langle K_x, f_{H^\perp} \rangle_G = \langle K_x, f_H \rangle_G = \langle K_x, f_H \rangle_H = f_H(x)$$
  Ainsi $f \in H$.

■

On a montré avec le le théorème 2 que si l'on a un RKHS H, alors il existe un unique noyau reproduisant k pour H, celui-ci étant aussi un noyau par la propriété 3. Et on a ensuite montré par le théorème 3 que si l'on a un noyau k, alors c'est un noyau reproduisant d'un RKHS H. Entre autre, on a montré l'équivalence entre plusieurs représentations pour un RKHS.

Regardons quelques kernels classiques :

- $k(x, y) \coloneqq \langle x, y \rangle$, son RKHS est $H = \{\langle \cdot, \beta \rangle : ||\langle \cdot, \beta \rangle||_H^2 = ||\beta||^2\}$
- $k(x, y) \coloneqq (\alpha \langle x, y \rangle + 1)^d,\ \alpha \in \mathbb{R},\ d \in \mathbb{N}$
- $k(x, y) \coloneqq \exp(||x - y||^2 / (2\sigma^2)),\ \sigma > 0$
- $k(x, y) \coloneqq \exp(||x - y|| / \sigma),\ \sigma > 0$
- $k(x, y) \coloneqq \sin(a(x - y))/\pi(x - y)$, son RKHS correspond aux fonctions de $L^2(\mathbb{R})$ dont la transformée de fourier est à support dans $[-a, a]$.

1.2 Applications des RKHS en machine learning

Une application très importante des RKHS est le Representer theorem, un théorème qui permet de transformer des problèmes d'optimisation de dimension infinie en dimension finie, qu'on pourra alors résoudre avec les méthodes d'optimisation numérique. On verra son application avec le Kernel Ridge Regression et avec les SVM.

1.2.1 Representer theorem

Théorème 4 (Representer theorem)**.** Soit k un kernel sur X et soit H sa RKHS associée. Posons $x_1, \cdots, x_n \in X$ notre training sample. Regardons le problème d'optimisation suivant :

$$\min_{f \in H} J(f) \coloneqq E(f(x_1), \cdots, f(x_n)) + P(||f||_H^2)$$

Où P est une fonction croissante.

Alors si ce problème d'optimisation a (au-moins) une solution, il y a (au-moins) une solution de la forme

$$f = \sum_{i=1}^{n} \alpha_i \cdot k(\cdot, x_i)$$

.

De plus, si P est strictement croissante, alors toute solution a cette forme.

Démonstration. Soit H_0 le sous espace engendré par $\{k(\cdot, x_i) : i \in 1, \cdots, n\}$. Comme $H_0 \in H$, H_0 est fermé car de dimension finie. Alors le théorème de décomposition orthogonale nous dit que : $\forall f \in H,\ f = f_{H_0} + f_{{H_0}^\perp}$.

De plus, comme k est un noyau reproductif de H, on a $\forall x_i$:

$$f(x_i) = \langle k(\cdot, x_i), f\rangle_H = \langle k(\cdot, x_i), f_{H_0}\rangle_H + \langle k(\cdot, x_i), f_{{H_0}^\perp}\rangle_H = \langle k(\cdot, x_i), f_{H_0}\rangle_H = f_{H_0}(x_i)$$

Alors :

$$\begin{aligned}
J(f) &\coloneqq E(f(x_1), \cdots, f(x_n)) + P(||f||_H^2) \\
&= E(f_{H_0}(x_1), \cdots, f_{H_0}(x_n)) + P(||f||_H^2) \\
&\geq E(f_{H_0}(x_1), \cdots, f_{H_0}(x_n)) + P(||f_{H_0}||_H^2) \text{ par croissance de } P \text{ car } ||f||_H^2 = ||f_{H_0}||_H^2 + ||f_{{H_0}^\perp}||_H^2 \\
&= J(f_{H_0})
\end{aligned}$$

Donc si f est un minimiseur de J, alors $f_{H_0} \coloneqq \sum_{i=1}^{n} \alpha_i \cdot k(\cdot, x_i)$ aussi. Si P est strictement croissante, alors l'inégalité est stricte et si l'on veut un minimiseur de J il faut nécessairement qu'il soit de cette forme. ■

1.2.2 Exemple 1 : Kernel Ridge Regression

Ici, $J(f) \coloneqq \sum_{i=1}^{n}(y_i - f(x_i))^2 + \lambda||f||_H^2$, prenons k un kernel sur X. Le representer theorem nous dit que la solution de ce problème (sous couvert d'existence) est nécessairement de la forme

$$f = \sum_{i=1}^{n} \alpha_i \cdot k(\cdot, x_i)$$

Rappelons que :

$$||f||_H^2 = \left\langle \sum_{i=1}^n a_i k(\cdot, x_i), \sum_{j=1}^n a_j k(\cdot, x_j) \right\rangle_H = \sum_{i=1}^n \sum_{j=1}^n a_i a_j K(x_i, x_j)$$

Le problème d'optimisation

$$\min_{f \in H} J(f) \coloneqq \sum_{i=1}^n (y_i - f(x_i))^2 + \lambda ||f||_H^2$$

est alors équivalent à

$$\min_{\alpha \in \mathbb{R}^n} \sum_{i=1}^n (y_i - \sum_{j=1}^n \alpha_j \cdot k(x_i, x_j))^2 + \lambda \sum_{i=1}^n \sum_{j=1}^n a_i a_j K(x_i, x_j)$$

ou encore avec $(K)_{ij} = k(x_i, x_j)$ (qui est symétrique)

$$\min_{\alpha \in \mathbb{R}^n} F(\alpha) \coloneqq ||y - K\alpha||_2^2 + \lambda \alpha^T K \alpha$$

Reste à résoudre ce problème de la même manière qu'une régression ridge simple :

$$\nabla_\alpha F(\alpha) = -2K(y - K\alpha) + 2\lambda K\alpha$$

$$\nabla_\alpha F(\alpha) = 0 \Leftrightarrow \alpha = (K + \lambda Id_n)^{-1} y$$

Notre prédicteur sera ainsi

$$f = \sum_{i=1}^n ((K + \lambda Id_n)^{-1} y)_i \cdot k(\cdot, x_i)$$

Ce qui est mieux par rapport à la régression pénalisée, c'est que notre prédicteur est une fonction non-linéaire de x, nous permettant d'aller chercher dans la classe des fonctions de la RKHS associée à k (et donc potentiellement avoir de meilleures prédictions).

1.2.3 Exemple 2 : Support Vector Machine (SVM)

Dans le cadre d'une SVM, $J(f) \coloneqq \frac{1}{n} \sum_{i=1}^n max(0, 1 - y_i f(x_i)) + \frac{\lambda}{2} ||f||_H^2$. Prenons encore une fois un kernel k sur X. Encore une fois, le representer theorem nous dit que la seule solution (si elle existe) est sous la forme

$$f = \sum_{i=1}^n \alpha_i \cdot k(\cdot, x_i)$$

On cherche alors à résoudre le problème suivant :

$$\min_{\alpha \in \mathbb{R}^n} \sum_{i=1}^n max(0, 1 - y_i \sum_{j=1}^n \alpha_j \cdot k(x_i, x_j)) + \frac{\lambda}{2} \sum_{i=1}^n \sum_{j=1}^n a_i a_j K(x_i, x_j)$$

On peut montrer que le dual de ce problème est :

$$\min_{\gamma \in \mathbb{R}^n} -\sum_{i=1}^n \gamma_i + \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \gamma_i \gamma_j y_i y_j k(x_i, x_j) \text{ t.q } 0 \leq \gamma_i \leq \frac{1}{n\lambda} \; \forall i \in \{1, \cdots, n\}$$

avec $\alpha_i = y_i\gamma_i \ \forall i \in \{1, \cdots, n\}$. On trouve la la solution du dual par des algorithmes d'optimisation quadratique.

Le prédicteur est donc :

$$f = \sum_{i=1}^{n} y_i\gamma_i \cdot k(\cdot, x_i)$$

On a remplacé l'estimateur linéaire de la SVM par un estimateur non-linéaire de x, sans changer la complexité de l'algorithme.

1.2.4 Le Kernel Trick

Plus généralement, à tout algorithme qui utilise des produits scalaires, on peut les remplacer par un kernel. Ainsi on peut transformer rendre non-linéaires les algorithmes, en manipulant des vecteurs de dimensions infinie sans que cela pose problème, comme leur produit scalaire est égal au kernel. C'est ce qu'on appelle le "kernel trick", et il a une très grande importance dans les applications pratiques. Par exemple, ce que l'on a fait plus haut avec peut aussi être fait avec le kernel trick dans la preuve de la construction de la solution linéaire en remplacant $\langle \cdot, \cdot \rangle$ par $\langle \phi(\cdot), \phi(\cdot) \rangle = k(\cdot, \cdot)$.

2 Introduction : Réseau de neurone simple

INTRODUIRE LA NOTATION ASYMP POUR LES ORDRES DE GRANDEUR

Commencons par étudier un NN très simple : une fonction $\Phi : (\mathbb{R}^m \times \mathbb{R}^{m\times m} \times \mathbb{R}^m) \times \mathbb{R} \to \mathbb{R}$ combinaison d'applications linéaires, sans non linéarités intermédiaires :

$$\Phi((\beta, A, u), x) \coloneqq \frac{1}{m^\alpha}\beta^T \left(\frac{1}{m^\gamma}A\right) ux$$

On initialise $\theta^0 \coloneqq (\beta^0, A^0, u^0)$ de manière standarde : $\forall i, j \in \{1, \cdots, m\},\ u_i^0, A_{ij}^0, \beta_i^0 \sim_{iid} N(0, 1)$

Nous montrerons quelques propriétés asymptotiques en la largeur des couches, et sur l'évolution des paramètres lors du premier pas de la descente de gradient.

2.1 Lois suivant la largeur des couches m

- Loi de $||u^0||_2^2$ pour m grand

  Comme $||u^0||_2^2 = \sum_{i=1}^m (u_i^0)^2 \sim \chi^2(m)$ et $u_i^0 \sim \chi^2(1)$, en appliquant le TCL aux u_i^0, on a :

  $$\frac{||u^0||_2^2 - m}{\sqrt{2m}} \sim_{m\to\infty} N(0, 1)$$

  En particulier, $||u^0||_2^2 \sim m$ pour m grand.

- Loi de $\left(\frac{1}{m^\gamma}A^0\right) u^0 x$ sachant u^0

  $(A^0u^0)_i = \sum_{j=1}^m A_{ij}^0 u_j^0$. En sachant u^0, comme $A_{i\cdot}^0$ est un vecteur gaussien, $(A^0u^0)_i \sim N(0, ||u^0||_2^2)$. De même, part indépendance des A_{ij}^0, les $(A^0u^0)_i$ sont indépendants et (conditionnellement à u_0) $A^0u^0 \sim N(0_m, ||u^0||_2^2\ Id_m)$.

Ainsi, la loi de $\left(\frac{1}{m^\gamma}A^0\right)u^0x$ sachant u^0 est $N(0_m, \left(\frac{x}{m^\gamma}\right)^2 ||u^0||_2^2\, Id_m)$.

- Loi de $\left(\frac{1}{m^\gamma}A^0\right)u^0x$ pour m grand

  On aura besoin du lemme suivant :

  $$X_n \sim N(\mu_n, \sigma_n) \text{ avec } \mu_n \to \mu \text{ et } \sigma_n \to \sigma, \text{ alors } (X_n) \text{ converge en loi vers } X_\infty \sim N(\mu, \sigma)$$

  Regardons la variance de $\left(\left(\frac{1}{m^\gamma}A^0\right)u^0x\right)_i$ sachant u^0 pour m grand : on utilise $||u^0||_2^2 \sim m$ et elle est donc environ égale à $x^2 m^{1-2\gamma}$. Si $\gamma < 1/2$ cette variance diverge vers l'infini pour $m \to \infty$. Si $\gamma > 1/2$, elle tendra vers 0 pour $m \to \infty$ et sa loi sera constante égale à 0. Seul le choix $\gamma = 1/2$ permet de stabiliser la variance pour $m \to \infty$. Dans ce cas là, on applique le lemme : on a $\left(\left(\frac{1}{m^\gamma}A^0\right)u^0x\right)_i$ sachant u^0 qui converge vers une $N(0, x^2)$. Comme cette loi est indépendante de u^0, on a donc que $\left(\left(\frac{1}{m^\gamma}A^0\right)u^0x\right)_i$ converge en loi vers $N(0, x^2)$.

- Loi de $|| \left(\frac{1}{m^\gamma}A^0\right)u^0x ||_2^2$ pour m grand

  On a $\left(\left(\frac{1}{m^\gamma}A^0\right)u^0x\right)_i \sim N(0, x^2)$ pour m grand.
  De plus, $\left(\left(\frac{1}{m^\gamma}A^0\right)u^0x\right)_i^2 \sim x^2 \cdot \chi^2(1)$. En appliquant la LGN, $|| \left(\frac{1}{m^\gamma}A^0\right)u^0x ||_2^2 \sim mx^2$.

- [•] Loi de $\frac{1}{m^\alpha}(\beta^0)^T x_2$ sachant x_2, avec $x_2 = \left(\frac{1}{m^\gamma}A^0\right)u^0x$

  On a $\frac{1}{m^\alpha}(\beta^0)^T x_2 | x_2 \sim N(0, \frac{1}{m^{2\alpha}}||x_2||_2^2)$

- [•] Loi de $\frac{1}{m^\alpha}(\beta^0)^T x_2$ pour m grand

  On a **JUSTIFIER MEME CHOSE QU AU DESSUS** $\frac{1}{m^\alpha}(\beta^0)^T x_2 \sim N(0, x^2 m^{1-2\alpha})$

2.2 Gradients

Trivialement,

$$\nabla_\beta \Phi = \frac{x}{m^{\alpha+1/2}} Au$$

$$\nabla_u \Phi = \frac{x}{m^{\alpha+1/2}} A^T \beta$$

$$\nabla_A \Phi = \frac{x}{m^{\alpha+1/2}} \beta u^T$$

Etudions les ordres de grandeur des normes correspondantes à l'initialisation pour m grand :
On va simplement utiliser les approximations données par la loi des grands nombres : $||u^0|| \simeq \sqrt{m}$, et comme vu plus haut, $||A^0 u^0|| \simeq \sqrt{m}||u^0|| \simeq m$. Ainsi $||\nabla_\beta \Phi(\theta^0, x)|| \sim m^{-\alpha-1/2} \cdot m = m^{1/2-\alpha}$.

$$||\nabla_\beta \Phi(\theta^0, x)|| \sim m^{1/2-\alpha}$$

Exactement de la même manière, on aboutit à :

$$||\nabla_u \Phi(\theta^0, x)|| \sim m^{1/2-\alpha}$$

Pour A, on prend la norme de Frobenius : la LGN nous dit que $||\beta^0(u^0)^T||_F^2 \simeq m^2$ (car il y a m^2 lois du $\chi^2(1)$ dans $\beta^0(u^0)^T$). Ainsi $||\beta^0(u^0)^T||_F \simeq m$ et donc :

$$||\nabla_A \Phi(\theta^0, x)||_F \sim m^{1/2-\alpha}$$

2.3 Descente de gradient

On va étudier ici le premier pas de descente de gradient.

Posons une fonction de perte $F : \mathbb{R} \to \mathbb{R}$ t.q $F'(0) \neq 0$ et $\Delta F \coloneqq F(\Phi(\theta^1, x)) - F(\Phi(\theta^0, x))$, avec $\theta^1 \coloneqq \theta^0 - \eta \nabla_\theta F(\Phi(\theta^0, x))$

Il semble honnête de prendre η dépendant de m, le produit scalaire final ayant plus de chance d'exploser en grande dimension. Prenons $\eta \coloneqq m^a$, $a \in \mathbb{R}$

2.3.1 Choix de η

On veut que ΔF ne diverge pas ni ne tende vers 0 lorsque m tend vers l'infini.

Pour cela, on utilise l'approximation $\Delta F \simeq < \Delta\theta, \nabla_\theta F(\Phi(\theta^0, x)) >$.

On a

$$\Delta F \simeq < -\eta \nabla_\theta F(\Phi(\theta^0, x)), \nabla_\theta F(\Phi(\theta^0, x)) >= -\eta ||\nabla_\theta F(\Phi(\theta^0, x))||^2$$

$$\left\{ \begin{array}{c} \nabla_\theta F(\Phi(\theta^0, x)) = \\ F'(\Phi(\theta^0, x))_{\text{constante en } m} \cdot \nabla_\theta \Phi(\theta^0, x) \end{array} \right\}$$

Or $||\nabla_\theta \Phi(\theta^0, x)||^2 = ||\nabla_\beta \Phi(\theta^0, x)||^2 + ||\nabla_u \Phi(\theta^0, x)||^2 + ||\nabla_A \Phi(\theta^0, x)||^2$

Donc pour m grand :

$$\begin{aligned} \Delta F &\simeq -\eta ||\nabla_\theta F(\Phi(\theta^0, x))||^2 \\ &\sim \eta ||\nabla_\theta \Phi(\theta^0, x)||^2 \\ &\simeq \eta (3 \cdot (m^{1/2-\alpha})^2) \\ &\simeq m^a \cdot m^{1-2\alpha} \end{aligned}$$

Ce qui impose le choix $a = 2\alpha - 1$

2.3.2 Ordres de grandeur des écarts relatifs

Pour cela introduisons $\Delta\theta \coloneqq \theta^1 - \theta^0$. Remarquons que $\Delta\theta = -\eta \nabla_\theta F(\Phi(\theta^0, x))$.

$$\begin{aligned} ||\Delta u|| &\sim \eta ||\nabla_u \Phi(\theta^0, x)|| && (1) \\ &\sim m^{2\alpha-1} \cdot m^{1/2-\alpha} && (2) \\ &\sim m^{\alpha-1/2} && (3) \end{aligned}$$

Ce qui nous donne un ordre de grandeur de l'écart relatif :

$$\frac{||\Delta u||}{||u^0||} \sim m^{\alpha-1}$$

On a le même résultat pour l'écart relatif de β^0 :

$$\frac{||\Delta\beta||}{||\beta^0||} \sim m^{\alpha-1}$$

Pour A, la LGN nous donne $||A^0||_F \simeq m$ pour m grand, on a alors par les mêmes calculs :

$$\frac{||\Delta A||_F}{||A^0||_F} \sim m^{\alpha-3/2}$$

Concernant l'écart entrywise de A, on a $|\Delta A_{ij}| \sim \eta|(\nabla_A \Phi(\theta^0, x))_{ij}| \sim m^{2\alpha-1} \cdot m^{-1/2-\alpha} \cdot 1 \sim m^{\alpha-3/2}$ car $|\beta^0(u^0)^T| \sim 1$. $|A_{ij}| \sim 1$, donc :

$$\frac{|\Delta A_{ij}|}{|A_{ij}|} \sim m^{\alpha-3/2}$$

Maintenant avec la norme opétateur : le corollaire 7.9 du cours de MIA2 nous donne une majoration sur $|A^0|_{op}$: $|A^0|_{op} \leq \sqrt{m} + 7\sqrt{m+\xi} = (\sqrt{m})$, avec $\xi \sim Exp(1)$.

De plus, on peut trouver la la norme opérateur de ΔA comme suit : tout le travail est de trouver $|\beta^0(u^0)^T|_{op} \coloneqq \sup\{||\beta^0(u^0)^T x|| \text{ avec } ||x|| = 1\}$.

$$(\beta^0(u^0)^T)_{ij} = \beta_i^0 u_j^0 \tag{4}$$

$$(\beta^0(u^0)^T x)_i = \sum_{j=1}^{m} (\beta^0(u^0)^T)_{ij} \cdot x_j \tag{5}$$

$$= \beta_i^0 < u^0, x > \tag{6}$$

$$||\beta^0(u^0)^T x|| = | < u^0, x > | \cdot ||\beta^0|| \tag{7}$$

Donc le sup est bien atteint en $x = u/||u^0||$ et est égal à $||u^0|| \cdot ||\beta^0||$. En utilisant les approximations précédentes, on a donc $|\beta^0(u^0)^T|_{op} \simeq m$. Ainsi on a $|\Delta A|_{op} \sim m^{2\alpha-1} \cdot m^{-1/2-\alpha} \cdot m \sim m^{\alpha-1/2}$. et :

$$\frac{|\Delta A|_{op}}{|A^0|_{op}} \sim m^{\alpha-1}$$

2.3.3 Choix de α

- $\alpha < 1$

  Dans ce cas là, tous les écarts relatifs d'ordre $m^{\alpha-1} \xrightarrow[m\to\infty]{} 0$. On a donc pour m grand :

  $$||\Delta u|| \ll ||u^0||$$

  $$||\Delta\beta|| \ll ||\beta^0||$$

  $$|\Delta A|_{op} \ll |A^0|_{op}$$

  Regardons maintenant les Δ des gradients en u pour la première itération :

$$\Delta\nabla_u\Phi \coloneqq \nabla_u\Phi(\theta^1, x) - \nabla_u\Phi(\theta^0, x) \tag{8}$$
$$\sim m^{-\alpha-1/2}\underbrace{((\beta^1)^T A^1 - (\beta^0)^T A^0)}_{(\star)} \tag{9}$$
$$(\star) = (\beta^0 - \eta\nabla_\beta F(\Phi(\theta^0, x)))^T(A^0 - \eta\nabla_A F(\Phi(\theta^0, x))) - (\beta^0)^T A^0 \tag{10}$$
$$= \eta^2[\nabla_\beta F(\Phi)]^T[\nabla_A F(\Phi)] - \eta[\nabla_\beta F(\Phi)]A^0 - \eta\beta^0[\nabla_A F(\Phi)] \tag{11}$$
$$= (\Delta\beta)^T(\Delta A) + (\Delta\beta)A^0 + \beta^0(\Delta A) \tag{12}$$

Donc :

$$||\Delta\nabla_u\Phi|| \sim m^{-\alpha-1/2}||(\Delta\beta)^T(\Delta A) + (\Delta\beta)A^0 + \beta^0(\Delta A)|| \tag{13}$$
$$\lesssim m^{-\alpha-1/2}(||\Delta\beta|| \cdot |\Delta A|_{op} + ||\Delta\beta|| \cdot |A^0|_{op} + ||\beta^0|| \cdot |\Delta A|_{op}) \tag{14}$$
$$\lesssim m^{-\alpha-1/2}(m^{\alpha-1/2}m^{\alpha-1/2} + m^{\alpha-1/2}m^{1/2} + m^{1/2}m^{\alpha-1/2}) \tag{15}$$
$$\lesssim m^{-\alpha-1/2}(m^{\alpha-1/2}m^{1/2} + m^{\alpha-1/2}m^{1/2} + m^{1/2}m^{\alpha-1/2}) \tag{16}$$
$$\lesssim m^{-1/2} \tag{17}$$

C'est-à-dire que $||\Delta\nabla_u\Phi|| = \mathcal{O}(m^{-1/2})$ pour m grand.

Par la même démarche on trouve $||\Delta\nabla_\beta\Phi|| = \mathcal{O}(m^{-1/2})$ et $||\Delta\nabla_A\Phi||_F = \mathcal{O}(m^{-1/2})$ pour m grand.

Ainsi :

$$||\Delta\nabla_{\beta/u/A}\Phi|| = \mathcal{O}(m^{-1/2}) \overset{\alpha<1}{\ll} m^{1/2-\alpha} \sim ||\nabla_{\beta/u/A}\Phi||$$

Cela traduit un comportement linéaire lorsque $\alpha < 1$ pour m grand.

On peut aussi remarquer que

$$\Delta\nabla_\theta F(\Phi) = F'(\Phi) \cdot \Delta\nabla_\theta\Phi$$

On a aussi :

$$||\Delta\nabla_{\beta/u/A}F(\Phi)|| = \mathcal{O}(m^{-1/2}) \overset{\alpha<1}{\ll} m^{1/2-\alpha} \sim ||\nabla_{\beta/u/A}F(\Phi)||$$

ce qu'il faut discuter, c'est la chose suivante : on voit qu'après une étape de descente de gradient, $\Delta F \asymp 1$ mais les variations relatives des paramètres et gradients tendent vers 0. Donc à l'étape suivante, les évaluations précédentes restent valables et le même phénomène se reproduira. In fine, après un nombre fini d'étape de gradient, on aura toujours $F(\Phi(\theta^t, x)) = F(\Phi(\theta^0, x)) + \langle\theta^t - \theta^0, \nabla_\theta F(\Phi(\theta^0, x))\rangle + O(m^{-1/2})$. Cela signifie donc qu'on apprend un modèle linéaire relatif aux features $\nabla_\theta F(\Phi(\theta^0, x))$. Autrement dit on fait un apprentissage dans un RKHS de noyau

$$k(x, y) = \langle\nabla_\theta F(\Phi(\theta^0, x)), \nabla_\theta F(\Phi(\theta^0, y))\rangle \overset{LGN}{=} \mathbb{E}[\langle\nabla_\theta F(\Phi(\theta^0, x)), \nabla_\theta F(\Phi(\theta^0, y))\rangle].$$

En particulier, le noyau ne dépend que de l'architecture du réseau de neurones et de l'initialisation, pas des données (no feature learning).

Tu peux aussi mettre a remarque que pour $\alpha = 1$ cela n'est plus vrai, il y a bien feature learning. Ici on a fait le calcul dans le cas le plus simple possible pour voir apparaitre le phénomène. Dans la suite on va : voir que le calcul reste valable pour $\sigma(x) \neq x$ et $x \in R^d$. Puis on dérivera proprement le résultat.