Projet d'appentissage statistique
Facteur de régulation des ResNets profonds

SHEN Pingya, VIN Charles

December 26, 2023

Contents

Chapter 1

Introduction

Avant l'introduction de ResNet en 2015 par He et al., l'architecture GoogLeNet était le dernier gagnant des challenges de vision par ordinateur. Cette architecture avait été développée pour pallier les problèmes d'apprentissage liés à l'augmentation de la profondeur de VGG, une autre architecture proéminente.

Un réseau plus profond peut offrir de meilleures performances dans certaines conditions, mais il est aussi sujet à des problèmes tels que l'explosion ou l'évanouissement du gradient de la fonction de coût. Durant la rétropropagation, les grandes ou petites valeurs de gradient peuvent s'amplifier à travers les couches du réseau, entraînant un gradient bien plus grand ou plus petit dans les dernières couches par rapport aux premières. Cet effet est multiplicatif et dépend donc de la profondeur du réseau.

Pour un réseau d'une profondeur L, on modélise ces états cachés de chaque couche de dimension d par une séquence $(h_k)_{1\leq k\leq L}$ avec $h_k \in \mathbb{R}^d, \forall 0 \leq k \leq L$. L'explosion du gradient peut être décrite mathématiquement par, avec une forte probabilité, $\|\frac{\partial \mathscr{L}}{\partial h_0}\| \gg \|\frac{\partial \mathscr{L}}{\partial h_L}\|$, où $\mathscr{L}$ représente la loss et $\|\cdot\|$ la norme euclidienne.

GoogLeNet, bien qu'offrant une légère amélioration des performances par rapport à VGG, était encore relativement complexe et sa profondeur comparable à celle de VGG, passant de 22 à 16 couches. En 2015, Microsoft introduit ResNet, un modèle allant jusqu'à 152 couches et divisant par deux le nombre d'erreurs de GoogLeNet. Son innovation réside dans l'intégration de *skip connections* entre les couches successives, facilitant le passage du gradient au sein du réseau. Mathématiquement, cela donne la relation récurrente suivante pour la séquence $(h_k)_{1\leq k\leq L}$:

$$h_{k+1} = h_k + f(h_k, \theta_{k+1}).$$

où $f(\cdot, \theta_{k+1})$ représente les transformations effectuées par la couche k et paramétrées par $\theta_{k+1} \in \mathbb{R}^p$.

Les ResNets sont devenus la base de nombreux modèles d'apprentissage profond de pointe, s'étendant au-delà du traitement d'images pour inclure des domaines tels que le traitement du langage naturel et l'apprentissage par renforcement. L'idée des *skip connections* a inspiré de nombreuses autres architectures et est devenue une pratique courante dans la conception des réseaux neuronaux profonds.

Figure 1.1: Example d'architecture pour la vision par ordinateur. **Bas** : VGG-19 (19.6 billion FLOPs). **Millieu** : un réseau classique (3.6 billion FLOPs). **Haut** : Un réseau résiduel (3.6 billion FLOPs) avec la présence de *skip connections* dans chaque bloc. Figure extraite de l'article original de ResNet (He et al., 2016)

Malgré ces avancées, ResNet rencontre toujours des problèmes de gradient durant l'apprentissage. La méthode traditionnelle pour contrer cela est la normalisation des états cachés après chaque couche (*batch normalization*). Cependant, cette approche a un coût computationnel et dépend fortement de la taille du *batch*. Une alternative est d'incorporer un facteur d'échelle α_L devant le terme résiduel, conduisant au modèle suivant :

$$h_{k+1} = h_k + \alpha_L f(h_k, \theta_{k+1}) \tag{1.1}$$

Le choix de α_L est crucial et dépend naturellement de la profondeur L du réseau. Il assure que la variance du signal reste stable lors de sa propagation à travers les couches. Cependant, il n'existe actuellement ni preuve formelle ni justification mathématique solide pour le choix de ce facteur de régularisation.

Dans ce cours, nous examinerons les fondements mathématiques pour choisir la valeur de α_L en fonction de L et de la distribution initiale des poids, dans le but d'éviter les problèmes d'apprentissage. Deux axes principaux d'étude seront abordés :

1. Le facteur α_L à l'initialisation : L'initialisation des paramètres est cruciale pour la phase d'apprentissage d'un modèle et influe même sur ses capacités de généralisation. Une mauvaise initialisation peut entraîner une divergence ou une disparition rapide du gradient, voire un blocage dans l'apprentissage. L'étude du rôle de α_L lors de l'initialisation est donc pertinente. Nous considérerons que, à l'initialisation, les poids de chaque couche $(\theta_k)_{1\leq k\leq L}$ sont choisis de manière indépendante et identique selon une loi, typiquement gaussienne ou uniforme sur $\mathbb{R}^p$.

2. L'approche continue : Dans les réseaux neuronaux composés de nombreuses couches, l'ensemble peut être considéré comme une fonction continue, particulièrement lorsque le nombre de neurones est élevé. Cette méthode envisage le réseau non pas comme une série de couches discrètes, mais plutôt comme un système continu. En adoptant cette vue, chaque couche est perçue comme une évolution marginale de la précédente, similaire à l'idée derrière les connexions résiduelles de ResNet. On peut imaginer que l'entrée de chaque couche suivante résulte de l'intégration des ajustements minimes apportés par la couche précédente, une approche semblable à la modélisation du mouvement en physique, où la position est intégrée sur le temps.
   Les EDN représentent une avancée significative dans la théorie et la pratique de l'apprentissage profond, trouvant des applications dans la classification d'images, l'analyse de séries temporelles et d'autres domaines. Il est intéressant de noter que tout réseau de deep learning doté de connexions résiduelles peut être approximativement exprimé par des équations différentielles neuronales.

Chapter 2

Le facteur α_L à l'initialisation

Dans cette section, notre objectif est d'examiner comment le facteur de mise à l'échelle α_L affecte la stabilité des ResNets lors de leur initialisation, en supposant que les poids sont des variables aléatoires indépendantes et identiquement distribuées (i.i.d.). Dans un premier temps, nous nous concentrerons sur l'analyse de la modélisation, l'initialisation des paramètres et les hypothèses requises pour notre étude. Puis par la suite, nous examinerons les limites probabilistes relatives aux valeurs des états cachés et des gradients.

2.1 Modèle et hypothèses

Modèle

Le modèle repose sur un ensemble de données composé de n paires $(x_i, y_i)_{1\leqslant i\leqslant n}$, où chaque $x_i \in \mathbb{R}^{n_{\text{in}}}$ représente un vecteur d'entrée et chaque $y_i \in \mathbb{R}^{n_{\text{out}}}$ un vecteur de sortie à prédire. Ces sorties peuvent être sous forme de valeurs continues ou codées en format *one-hot*. Soit $F_\pi(x) \in \mathbb{R}^{n_{\text{out}}}, x \in \mathbb{R}^{n_{\text{in}}}$ la sortie du ResNet définie par

$$\begin{aligned} h_0 &= Ax, \\ h_{k+1} &= h_k + \alpha_L V_{k+1} g(h_k, \theta_k), \quad 0 \leqslant k \leqslant L-1, \\ F_\pi(x) &= Bh_L, \end{aligned} \tag{2.1}$$

où $\pi = (A, B, (\theta_k)_{k\leqslant L}, (V_k)_{1\leqslant k\leqslant L})$ sont les paramètres du modèle avec $A \in \mathbb{R}^{d\times n_{\text{in}}}, B \in \mathbb{R}^{n_{\text{out}}\times d}, \theta_k \in \mathbb{R}^p$ et $V_k \in \mathbb{R}^{d\times d}$ pour $k = 1, \dots, L$. La fonction $g : \mathbb{R}^d \times \mathbb{R}^p \to \mathbb{R}^d$ représente le choix de l'architecture d'un bloc du ResNet. Nous nous intéressons principalement à la suite des états cachés $(h_k)_{0\leqslant k\leqslant L}$ et non aux changements de dimension permis par les matrices A et B. Finalement, on définit $l : \mathbb{R}^{n_{\text{out}}} \times \mathbb{R}^{n_{\text{out}}} \to \mathbb{R}_+$ comme la fonction de coût, différentiable par rapport à son premier paramètre. Ce coût peut être une perte quadratique ou une entropie croisée. L'objectif de l'apprentissage est de trouver le paramètre optimal π qui minimise le risque empirique $\mathscr{L}(\pi) = \sum_{i=1}^n l(F_\pi(x_i), y_i)$ à travers une descente de gradient stochastique ou l'une de ses variantes.

Durant ce cours, nous nous focaliserons spécifiquement sur trois architectures classiques de ResNet, détaillées dans la Table 2.1 ci-dessous. Il serait également pertinent d'explorer une quatrième architecture, qui intégrerait plusieurs couches linéaires ou convolutives.

Nom	Récurrence	Paramètres
res-1	$h_{k+1} = h_k + \alpha_L V_{k+1}\sigma(h_k)$	$\theta_{k+1} = \emptyset$
res-2	$h_{k+1} = h_k + \alpha_L V_{k+1}\sigma(W_{k+1}h_k)$	$\theta_{k+1} = W_{k+1}$
res-3	$h_{k+1} = h_k + \alpha_L V_{k+1}\text{ReLU}(W_{k+1}h_k)$	$\theta_{k+1} = W_{k+1}$

Table 2.1: Exemples d'architectures ResNet considérées dans l'article. Dans les deux premiers cas, la fonction d'activation σ est telle que, pour tout $x \in \mathbb{R}$, $a|x| \leqslant |\sigma(x)| \leqslant b|x|$, avec $1/\sqrt{2} \leqslant a < b \leqslant 1$. Dans les deux derniers cas, $W_{k+1} \in \mathbb{R}^{d\times d}$.

Initialisation des paramètres

Nous rappelons que $\theta_k \in \mathbb{R}^p$ et $V_k \in \mathbb{R}^{d\times d}$ sont les paramètres des couches cachées de notre modèle pour tout $k \in [\![1, L]\!]$. Ces paramètres sont choisis à l'initialisation comme la réalisation de variables aléatoires i.i.d., généralement suivant une distribution uniforme ou gaussienne. Cette initialisation est

indépendante de L et donc du modèle représenté par g, permettant de considérer plusieurs architectures différentes dans notre étude. D'autres auteurs choisissent d'étudier le choix de α_L comme un problème de variance à l'initialisation, rendant l'analyse dépendante de l'architecture g (par exemple Yang and Schoenholz (2017) ou Wang et al. (2022)).

Hypothèses

Pour notre étude, certaines hypothèses concernant le choix de l'architecture et de l'initialisation du réseau sont nécessaires. Avant de commencer, nous avons besoin de la définition suivante :

Définition 1 (Variable aléatoire s^2 sub-gaussienne) *En théorie des probabilités, une distribution s^2 sub-gaussienne est une distribution de probabilité caractérisée par une décroissance rapide des queues de distribution. Bien qu'il existe de nombreuses définitions et propriétés, nous retiendrons dans ce cours la suivante : soit X une variable aléatoire réelle,*

$$\forall \lambda \in \mathbb{R}, \mathbb{E}(\exp(\lambda X)) \leqslant \exp\left(\frac{\lambda^2 s^2}{2}\right).$$

De manière informelle, les queues d'une distribution sub-gaussienne sont dominées par celles d'une distribution gaussienne, c'est-à-dire qu'elles décroissent au moins aussi rapidement.

Avec cette définition en tête, passons maintenant aux hypothèses. Pour tout $1 \leqslant k \leqslant L$

Hypothèse 1 *Pour un certain $s \geqslant 1$, les entrées de $\sqrt{d}V_k$ sont des variables aléatoires symétriques i.i.d., s^2 sub-gaussiennes, indépendantes de d et L et de variance unitaire.*

Hypothèse 2 *Pour un certain $C > 0$, indépendant de d et L, et pour tout $h \in \mathbb{R}^D$*

$$\frac{\|h\|^2}{2} \leqslant \mathbb{E}(\|g(h, \theta_k)\|^2) \leqslant \|h\|^2.$$

and

$$\mathbb{E}(\|g(h, \theta_k)\|^8) \leqslant C\|h\|^8.$$

Note 1 *L'Hypothèse 1 est en pratique satisfaite par toutes les initialisations, en particulier celle par défaut dans les paquets Keras (Chollet et al., 2015) et Torch Vision (maintainers and contributors, 2016).*

La première partie de l'Hypothèse 2 assure que $g(\cdot, \theta_{k+1})$ se comporte en moyenne approximativement comme une isométrie, c'est-à-dire qu'elle préserve les longueurs et les mesures d'angles entre son espace de départ et son espace d'arrivée.

La deuxième partie de l'Hypothèse 2 vise à limiter les variations excessives de la norme de $g(h_k, \theta_{k+1})$.

Proposition 1 (Admis) *Soit les modèles `res-1`, `res-2`, `res-3` décrit dans la Table 2.1, on a*

(i) L'Hypothèse 2 est valide pour l'architecture `res-1`.

(ii) L'Hypothèse 2 est valide pour les architectures `res-2` et `res-3` dès lors que les entrées de $\sqrt{d}W_{k+1}, 0 \leqslant k \leqslant L-1$ sont des variables aléatoire de variance unitaire, i.i.d., symétriques, sub-gaussienne et indépendante de d et L.

Lemme 1 (Admis) *Considérons un ResNet (1.1) tel que les Hypothèses 1 et 2 soient satisfaites. Alors*

$$((1 + \frac{\alpha_L^2}{2})^L - 1) \leqslant \mathbb{E}(\frac{\|h_L - h_0\|^2}{\|h_0\|^2}) \leqslant ((1 + \alpha_L^2)^L - 1).$$

Ce lemme nous servira en particulier dans la preuve de la Proposition 2.

2.2 Limite probabilistique de la norme des états cachés

Dans cette section, nous nous intéressons à la quantité $\|h_L - h_0\|/\|h_0\|$. Cette mesure permet d'analyser la valeur des états cachés entre le début et la fin du réseau. Si $\|h_L - h_0\| \ll \|h_0\|$, cela suggère que le réseau agit presque comme une fonction identité. À l'inverse, un ratio $\|h_L - h_0\| \gg \|h_0\|$ indique une explosion des valeurs des états cachés. Une situation équilibrée serait représentée par $\|h_L - h_0\| \approx \|h_0\|$.

Nous appliquerons un raisonnement similaire aux gradients dans la Section 2.3 avec la quantité $\|\frac{\partial \mathscr{L}}{\partial h_0} - \frac{\partial \mathscr{L}}{\partial h_L}\|/\|\frac{\partial \mathscr{L}}{\partial h_L}\|$. En raison de la propagation rétroactive du gradient qui commence à partir de la fin du réseau, cette mesure est comparée à la dernière valeur du gradient $\partial \mathscr{L}/\partial h_L$.

Les propositions et corollaires suivants décriront comment le rapport $\|h_L - h_0\|/\|h_0\|$ se comporte en fonction de $L\alpha_L$, en établissant différentes bornes supérieures et inférieures.

Proposition 2 *Considérons un ResNet (1.1) tel que les hypothèses 1 et 2 soient satisfaites. Si* $L\alpha_L^2 \leqslant 1$, *alors, pour tout* $\delta \in (0,1)$, *avec une probabilité d'au moins* $1-\delta$,

$$\frac{\|h_L - h_0\|^2}{\|h_0\|^2} \leqslant \frac{2L\alpha_L^2}{\delta}.$$

La proposition 2 par sa borne supérieur petite indique que le réseau se comporte comme une fonction identité dans le cas où $L\alpha_L^2 \ll 1$.

Preuve 1 (Proposition 2) *En se basant sur le Lemme 1, on a*

$$\mathbb{E}\left(\frac{\|h_L - h_0\|^2}{\|h_0\|^2}\right) \leqslant ((1+\alpha_L^2)^L - 1).$$

Considérons le cas où $L\alpha_L^2 \leqslant 1$ *(valeur faible) et* L *tend vers de grandes valeurs. Dans ce contexte,* $(1+\alpha_L^2)^L$ *est une bonne approximation de* $\exp(L\alpha_L^2)$ *par définition, tout en restant inférieur ou égal à celui-ci en raison de la croissance exponentielle de* exp. *En effet,* $(1+\alpha_L^2)^L$ *se rapproche de* $1+L\alpha_L^2$ *selon la formule du binôme de Newton, et correspond aux premiers termes du développement en série de Taylor de l'exponentielle. Finalement, on a obtiens*

$$(1+\alpha_L^2)^L - 1 \leqslant \exp(L\alpha_L^2) - 1.$$

En poursuivant avec le développement de Taylor, nous obtenons une majoration plus précise.

$$(1+\alpha_L^2)^L - 1 \leqslant \exp(L\alpha_L^2) - 1) \leqslant L\alpha_L^2 \leqslant 2L\alpha_L^2.$$

Ainsi on obtient

$$\mathbb{E}\left(\frac{\|h_L - h_0\|^2}{\|h_0\|^2}\right) \leqslant 2L\alpha_L^2.$$

En appliquant l'inégalité de Markov, nous parvenons au résultat souhaité de la proposition 2.

Proposition 3 (Admise) *Considérons un ResNet (1.1) tel que les hypothèses 1 et 2 soient satisfaites.*

(i) Supposons que $d \geqslant 64$ *et* $\alpha_L^2 \leqslant \frac{2}{(\sqrt{C}s^4+4\sqrt{C}+16s^4)d}$. *Alors, pour tout* $\delta \in (0,1)$, *avec une probabilité d'au moins* $1-\delta$,

$$\frac{\|h_L - h_0\|^2}{\|h_0\|^2} > \exp\left(\frac{3L\alpha_L^2}{8} - \sqrt{\frac{11L\alpha_L^2}{d\delta}}\right) - 1,$$

à condition que

$$2L\exp\left(-\frac{d}{64\alpha_L^2 s^2}\right) \leqslant \frac{\delta}{11}.$$

(ii) Supposons que $\alpha_L^2 \leqslant \frac{1}{\sqrt{C}(d+128s^4)}$. *Alors, pour tout* $\delta \in (0,1)$, *avec une probabilité d'au moins* $1-\delta$,

$$\frac{\|h_L - h_0\|^2}{\|h_0\|^2} < \exp\left(L\alpha_L^2 + \sqrt{\frac{5L\alpha_L^2}{d\delta}}\right) + 1.$$

La Proposition 3 aborde les deux cas restants : $L\alpha_L^2 \gg 1$ et $L\alpha_L^2 \approx 1$. Dans la partie *(i)*, la borne inférieure indique une explosion très probable du gradient lorsque $L\alpha_L^2 \gg 1$. La partie *(ii)* traite du cas où $L\alpha_L^2 \approx 1$, avec une borne supérieure qui, en combinaison avec celle de *(i)*, suggère que h_L fluctue aléatoirement autour de h_0, borné des deux côtés.

La Proposition 3 peut présenter des hypothèses qui semblent atypiques, mais elles sont en réalité souvent vérifiées dans la majorité des ResNets profonds. En effet, il est courant de trouver des ResNets avec une profondeur $L \geqslant 100$, pour lesquels on définit généralement $\alpha_L = 1/L^\beta$ avec $\beta > 0$. De plus, la dimension des états cachés atteint fréquemment des valeurs telles que $d \geqslant 100$.

Les conséquences des Propositions 2 et 3 vont devenir plus clair en fixant $\alpha_L = 1/L^\beta$ comme montré dans le corollaire suivant.

Corollaire 1 *Considérons un ResNet (1.1) tel que les hypothèses 1 et 2 soient satisfaites. Soit* $\alpha_L = 1/L^\beta$, *avec* $\beta > 0$.

(i) Si $\beta > 1/2$, *alors*

$$\frac{\|h_L - h_0\|}{\|h_0\|} \xrightarrow[L\to\infty]{\mathbb{P}} 0.$$

(ii) Si $\beta < 1/2$ *et* $d \geqslant 9$*, alors*

$$\frac{\|h_L - h_0\|}{\|h_0\|} \xrightarrow[L\to\infty]{\mathbb{P}} \infty.$$

(iii) Si $\beta = 1/2$*,* $d \geqslant 64$*,* $L \geqslant (\frac{1}{2}\sqrt{C}s^4 + 2\sqrt{C} + 8s^4)d + 96\sqrt{C}s^4$*, alors, pour tout* $\delta \in (0,1)$*, avec une probabilité d'au moins* $1-\delta$*,*

$$\exp\left(\frac{3}{8} - \sqrt{\frac{22}{d\delta}}\right) - 1 < \frac{\|h_L - h_0\|^2}{\|h_0\|^2} < \exp\left(1 + \sqrt{\frac{10}{d\delta}}\right) + 1,$$

à condition que

$$2L\exp\left(-\frac{Ld}{64s^2}\right) \leqslant \frac{\delta}{11}.$$

Preuve 2 (Corollaire 1) *Pour chaque partie du corollaire, on a*

- *L'affirmation (i) est une conséquence de la Proposition 2.*
- *L'affirmation (ii) est une conséquence de la Proposition 3. En effet, cette dernière est valide sous deux conditions vérifiées dans notre cas :*
  - *La contrainte* $d \geqslant 64$ *peut être, dans notre cas, relaché à* $d \geqslant 9$ *en observant la preuve de la Proposition 3 non décrite ici.*
  - *La majoration* $\alpha_L \leqslant \frac{2}{(\sqrt{C}s^4+4\sqrt{C}+16s^4)d}$ *est automatiquement satisfaite pour* L *assez grand.*
- *Pour prouver l'affirmation (iii), nous utilisons l'union des deux affirmations de la Proposition 3.*

Le corollaire 4 précise le comportement de notre dernier état caché $\|h_L\|$ en fonction de β.

- Lorsque $\beta > 1/2$, la distance entre h_L et h_0 tend vers zéro lorsque L augmente indéfiniment. Cela indique que le réseau fonctionne essentiellement comme une fonction identité.
- Lorsque $\beta < 1/2$, la norme de h_L tend à l'explosion avec la valeur de L .
- Lorsque $\beta = 1/2$, h_L fluctue autour de h_0, indépendamment de la longueur du réseau L.

En conséquence, fixer $\beta = 1/2$ est la seule manière d'assurer une distribution adéquate des valeurs de h_L. La Figure 2.1 illustre ce comportement.

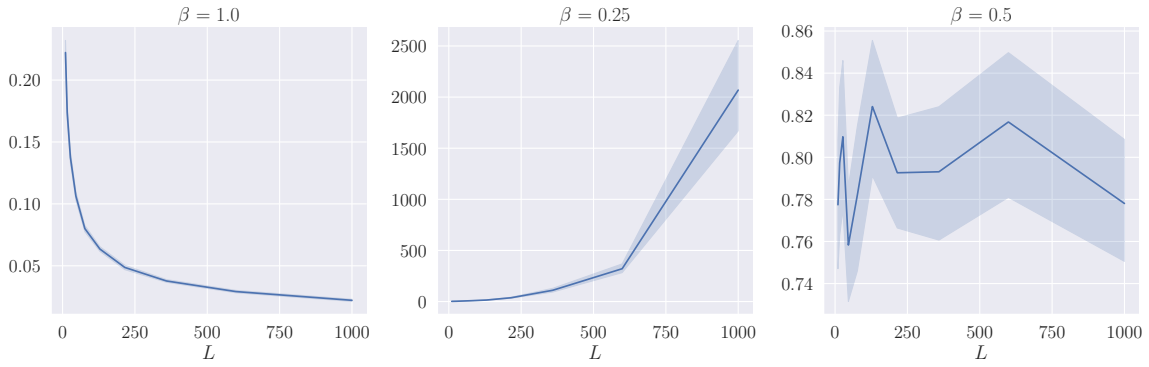

Figure 2.1: Illustration du Corollaire 1. Évolution de $\|h_L - h_0\| / \|h_0\|$ en fonction de L pour différente valeur de β.

2.3 Limite probabilistique des gradients

Dans la section précédente, nous avons étudié le comportement de la sortie du réseau pour des valeurs élevées de L. Cependant, cette analyse ne nous renseigne pas sur le comportement du gradient du coût $p_k = \frac{\partial \mathscr{L}}{\partial h_k} \in \mathbb{R}^d$ durant la rétropropagation, qui est pourtant une donnée cruciale pour évaluer la capacité d'entraînement du réseau à l'initialisation. Par conséquent, cette section se consacrera à l'étude des variations de $\|p_0 - p_L\| / \|p_L\|$, toujours dans un contexte où la profondeur du réseau L est grande. Il est important de noter que, en raison de la nature de la rétropropagation, notre attention se porte désormais sur p_0 plutôt que sur p_L. Nous définissons donc la séquence $(p_k)_{0 \leqslant k \leqslant L}$ comme suit :

$$p_k = p_{k+1} + \alpha_L \frac{\partial g(h_k, \theta_{k+1})^T}{\partial h} V_{k+1}^T p_{k+1}$$

Bien que cette récurrence soit similaire à celle des états cachés (eq. (2.1)), nous ne pouvons pas appliquer les mêmes techniques de preuve utilisées précédement. Cette différence s'explique par la dépendance de $\frac{\partial g(h_k, \theta_{k+1})}{\partial h}$ à h_k, et donc à $\theta_1, V_1, \ldots, \theta_k, V_k$. Supposer l'indépendance de ces deux quantités serait une hypothèse assez forte, non vérifiée par de nombreuses architectures, comme notre modèle `res-1`. Ainsi, nous adopterons une autre approche, basée sur la dérivation automatique. Cette méthode nécessite des hypothèses moins contraignantes, mais les résultats obtenus seront exprimés en termes d'espérance plutôt qu'avec une forte probabilité.

Soit $z \in \mathbb{R}^d$. Pour tout $0 \leqslant i, j \leqslant L$, définissons $\frac{\partial h_j}{\partial h_i} \in \mathbb{R}^{d \times d}$ comme la matrice Jacobienne de h_j par rapport à h_i. Pour rappel, l'élément en position (m, n) de la Jacobienne représente la dérivée de la m-ème coordonnée de h_j par rapport à la n-ème coordonnée de h_i. Soit $q_k(z) = \frac{\partial h_k}{\partial h_0} z$, en appliquant la règle de la chaîne, nous obtenons

$$q_{k+1}(z) = \frac{\partial h_{k+1}}{\partial h_k} q_k(z) = q_k(z) + \alpha_L V_{k+1} \frac{\partial g(h_k, \theta_{k+1})}{\partial h} q_k(z) \tag{2.2}$$

Nous observons une récurrence similaire à celle présentée dans l'Equation (1.1). En considérant maintenant que z est une variable aléatoire suivant une distribution gaussienne, il devient possible de décrire la valeur de $\|p_0\| / \|p_L\|$ en fonction du dernier vecteur $q_L(z)$.

$$\frac{\|p_0\|^2}{\|p_L\|^2} = \frac{1}{\|p_L\|^2} \mathbb{E}_{z \sim \mathcal{N}(0, I_d)} \left(\left| p_0^T z \right|^2 \right) = \mathbb{E}_{z \sim \mathcal{N}(0, I_d)} \left(\left| \left(\frac{p_L}{\|p_L\|} \right)^T q_L(z) \right|^2 \right) \tag{2.3}$$

avec I_d la matrice identité sur $\mathbb{R}^d$. On retrouve la seconde égalité comme conséquence de

$$p_0^T = \left(\frac{\partial \mathscr{L}}{\partial h_0} \right)^T, z = \left(\frac{\partial \mathscr{L}}{\partial h_0} \right)^T \frac{\partial h_L}{\partial h_0} z = p_L^T q_L(z).$$

En résumé, la récurrence énoncée dans l'équation (2.2) va nous permettre de déterminer les bornes de $\|q_L(z)\|$. Ces résultats pourront ensuite être transposés à $\|p_0\| / \|p_L\|$ en utilisant l'Equation (2.3). Pour ce faire, il nous faudra établir certaines hypothèses sur le rapport $p_L / \|p_L\|$.

Hypothèses

Hypothèse 3 *Soit $b = p_L / \|p_L\|$, alors $\mathbb{E}[b|h_L] = 0$ et $\mathbb{E}(b^T b|h_L) = I_d/d$*

Note 2 *Cette hypothèse est facilement vérifiée. Par exemple, en considérant $n_{out} = 1$ et en utilisant l'entropie croisée pour une classification binaire, ou bien l'erreur quadratique dans le cadre d'une régression.*

L'hypothèse suivante est l'équivalent pour les gradients de ce qui est énoncé dans Hypothèse 2.

Hypothèse 4 *On a, presque surement,*

$$\frac{\|q_k\|^2}{2} \leqslant \mathbb{E} \left(\left\| \frac{\partial g(h_k, \theta_{k+1})}{\partial h} q_k \right\|^2 | h_k, q_k \right) \leqslant \|q_k\|^2 .$$

L'Hypothèse 4 s'applique à toutes les architectures répertoriées dans la Table 2.1, comme le démontre la proposition suivante.

Proposition 4 *Soit les réseaux* `res-1`, `res-2` *et* `res-3` *définit dans la Table 2.1. Supposons l'Hypothèse 1 satisfaite et* σ *dérivable presque partout avec* $a \leqslant \sigma' \leqslant b$. *Alors*

(i) L'Hypothèse 4 est valide pour `res-1`

(ii) L'Hypothèse 4 est valide pour `res-2` *et* `res-3` *dès lors que les entrées de* $\sqrt{d}W_{k+1}, 0 \leqslant k \leqslant L-1$ *sont des variables aléatoires de variance unitaire, i.i.d, symétriques et indépendante de* d *et* L.

Les propositions Proposition 5 et 6 sont les homologues des Proposition 2 et 3 pour l'analyse du gradient du coût à l'initialisation.

Proposition 5 *Soit un ResNet (2.1) tel que les Hypothèse 1 à 4 sont satisfaites. Si* $L\alpha_L^2 \leqslant 1$ *alors, pour tout* $\delta \in (0,1)$ *avec une probabilité d'au moins* $1-\delta$

$$\frac{\|p_0 - p_L\|^2}{\|p_l\|^2} \leqslant \frac{2L\alpha_L^2}{\delta}.$$

Proposition 6 *Soit un ResNet (2.1) tel que les Hypothèse 1 à 4 sont satisfaites. Alors*

$$(1+\frac{1}{2}\alpha_L^2)^L - 1 \leqslant \mathbb{E}\left(\frac{\|p_0 - p_L\|^2}{\|p_l\|^2}\right) \leqslant (1+\alpha_L^2)^L - 1.$$

De façon similaire au Corollaire 1, le Corollaire 2 est déduit à partir des Proposition 5 et 6.

Corollaire 2 *Soit un ResNet (2.1) tel que les Hypothèses 1 à 4 sont satisfaites. En posant* $\alpha_L = 1/L^\beta$ *avec* $B > 0$, *on obtient*

(i) Si $\beta > 1/2$, *alors*

$$\frac{\|p_0 - p_L\|}{\|p_l\|} \xrightarrow[L\to\infty]{\mathbb{P}} 0.$$

(ii) Si $\beta < 1/2$ *alors*

$$\mathbb{E}\left(\frac{\|p_0 - p_L\|^2}{\|p_l\|^2}\right) \xrightarrow[L\to\infty]{\mathbb{P}} \infty.$$

(iii) Si $\beta = 1/2$ *alors*

$$\exp(\frac{1}{2}) - 1 \leqslant \mathbb{E}\left(\frac{\|p_0 - p_L\|^2}{\|p_l\|^2}\right) \leqslant \exp(4) - 1.$$

Ce corrolaire est illustré dans la Figure 2.2.

export la figure du pdf avec Windows

Figure 2.2: Illustration du corollaire 2. Évolution de $\|p_0 - p_L\| / \|p_L\|$ en fonction de L pour différente valeur de β.

2.4 Conclusion

Dans ce chapitre, nous avons examiné comment le comportement des gradients et des états cachés est influencé par la valeur de β, qui se décline en trois cas distincts :

- $\beta < 1/2$: Une explosion qui empêche l'entraînement du réseau.
- $\beta > 1/2$: Un effet d'identité qui diminue les performances du réseau.
- $\beta = 1/2$: Une limite non-dégénérée favorable.

Il est intéressant de noter que la valeur $\beta = 1/2$ joue un rôle clé. De manière surprenante, cette valeur trouve une interprétation spécifique dans l'étude des ResNet dans un cadre continu, sujet que nous aborderons dans le chapitre suivant.

Chapter 3

Équation différentielle ordinaire et stochastique

L'interprétation en temps continu des ResNets fournit un cadre puissant pour comprendre leur comportement, notamment dans le contexte d'architectures d'apprentissage profond comportant un grand nombre de couches.

L'une des principales conclusions est la similarité formelle entre les ResNets mis à l'échelle et les équations différentielles. Comme la profondeur L tend vers l'infini, le comportement des ResNets peut être approché par un processus continu. Ceci est mathématiquement exprimé comme une transition des mises à jour discrètes par couche dans ResNets vers un système dynamique en temps continu. Plus précisément, l'évolution des états cachés dans un ResNet peut être considérée comme une discrétisation d'une équation différentielle, ce qui constitue une réalisation profonde pour comprendre ce type de modèles d'apprentissage profond.

3.1 Équation différentielle ordinaire

Une équation différentielle ordinaire (EDO) est une équation différentielle dans laquelle la fonction inconnue est fonction d'une variable et les dérivées de l'équation dépendent uniquement de cette variable. Formellement, EDO peut être exprimé comme $\frac{dy}{dk} = f(k, y)$ où y est fonction de k. Le but de la résolution d'une équation différentielle du premier ordre est de trouver une fonction y qui satisfait l'équation. Cependant, nous ne pouvons pas calculer directement y . Au lieu de cela, nous savons comment la fonction y change avec le temps k, ce qui est représenté par la dérivée $\frac{dy}{dk}$.

3.1.1 EDO neuronale

Dans l'apprentissage profond, en particulier lors de la conception de structures de réseau, les équations différentielles ordinaires (EDO) peuvent être utilisées pour décrire les changements dynamiques continus entre les différentes couches du réseau ou les fonctions d'activation.

La motivation des équations différentielles régulières divines vient de ResNet. ResNet possède généraleme divines ?
une couche non linéaire avec des connexions résiduelles. Nous pouvons résumer cela en une fonction qui représente une fonction non linéaire, une matrice de poids, un biais et une connexion résiduelle.

$$h_{k+1} = h_k + f(h_k, \theta_{k+1}) \tag{3.1}$$

En poussant l'espacement entre les couches du réseau à une valeur infinitésimale, nous pouvons convertir ResNet en un réseau neuronal continu, ce qui est exactement ce que fait l'EDO neuronale. En faisant cela, nous pouvons comparer les couches discrètes de ResNet à sa représentation continue de réseau neuronal. Nous pouvons voir que le taux de changement de l'état sous-jacent dans un réseau neuronal continu est déterminé par une fonction non linéaire, et cette fonction ne change pas dans le temps, un peu comme la

forme d'une EDO.

$$\begin{aligned}
& h_{k+1} = h_k + f(h_k, \theta_{k+1}) \\
\Leftrightarrow \quad & h_{k+1} - h_k = f(h_k, \theta_{k+1}) \\
\Leftrightarrow \quad & \frac{h_{k+1} - h_k}{1} = f(h_k, \theta_{k+1}) \\
\Leftrightarrow \quad & \frac{h_{k+\Delta} - h_k}{\Delta}_{|\Delta=1} = f(h_k, \theta_{k+1}) \\
\Leftrightarrow \quad & \lim_{\Delta \to 0} \frac{h_{k+\Delta} - h_k}{\Delta}_{|\Delta=1} = f(h_k, \theta, k) \\
\Leftrightarrow \quad & \frac{dh(k)}{dt} = f(h_k, \theta, k)
\end{aligned}$$

Ici, les couches de réseaux neuronaux traditionnelles sont considérées comme des échantillons discrets d'un système dynamique en temps continu.

3.1.2 Convergence vers une EDO

Il est nécessaire de se poser la question de savoir si l'utilisation de méthodes de répartition de poids alternatives lors de l'initialisation et de la mise à l'échelle pourrait conduire à une équation différentielle ordinaire neuronale conventionnelle. Nous supposons que les poids $(V_k)_{1 \leqslant k \leqslant L}$ et $(\theta_k)_{1 \leqslant k \leqslant L}$ sont des discrétisations de fonctions lisses $\mathcal{V} : [0,1] \to \mathcal{R}^{d \times d}$ et $\Theta : [0,1] \to \mathcal{R}^p$. On considère alors l'itération générale avant avec $\alpha_L = \frac{1}{L}$, soit:

$$h_0 = Ax, \; h_{k+1} = h_k + \frac{1}{L} V_{k+1} g(h_k, \theta_{k+1}), \; 0 \leqslant k \leqslant L-1 \tag{3.2}$$

avec $V_k = \mathcal{V}_{k/L}$ et $\theta_k = \Theta_{k/L}$. Combiner avec l'idée d'EDO neuronale précédente, on considère $(V_k)_{1 \leqslant k \leqslant L}$ et $(\theta_k)_{1 \leqslant k \leqslant L}$ sont variables aléatoires en mettant $(\mathcal{V}_t)_{t \in [0,1]}$ et $(\Theta_t)_{t \in [0,1]}$ sont les temps continus processus stochastiques. Dans ce modèle, nous aurons besoin des hypothèses suivantes. Ces hypothèses constituent la base d'une dérivation théorique ultérieure et garantissent que les outils et méthodes mathématiques utilisés sont adaptés à l'analyse des modèles neuronale.

Hypothèse 5 *Pour chaque $0 \leqslant k \leqslant L-1$, les processus stochastique $\mathcal{V}$ et Θ sont presque sûrement Lipschitziens continus et bornés.*

Plus précisément, il existe presque sûrement $K_{\mathscr{V}}, K_\Theta, C_{\mathscr{V}}, C_\Theta > 0$, tel que, pour tous $s, t \in [0,1]$

$$\begin{aligned}
\|\mathscr{V}_t - \mathscr{V}_s\| &\leqslant K_{\mathscr{V}} |t-s| & \|\mathscr{V}_t\| &\leqslant C_{\mathscr{V}} \\
\|\Theta_t - \Theta_s\| &\leqslant K_\Theta |t-s| & \|\Theta_t\| &\leqslant C_\Theta
\end{aligned}$$

Hypothèse 6 *La fonction g est Lipschitzienne continue sur les ensembles compacts, dans le sens où pour tout compact $\mathscr{P} \subseteq \mathbb{R}^p$, il existe $K_{\mathscr{P}} > 0$ tel que, pour tous $h, h' \in \mathbb{R}^d, \theta \in \mathscr{P}$,*

$$\|g(h,\theta) - g(h',\theta)\| \leqslant K_{\mathscr{P}} \|h - h'\|,$$

et pour tous compact $\mathscr{D} \subseteq \mathbb{R}^d$, il existe $K_{\mathscr{D},\mathscr{P}} > 0$ tel que, pour tous $h \in \mathscr{D}, \theta, \theta' \in \mathscr{P}$,

$$\|g(h,\theta) - g(h,\theta')\| \leqslant K_{\mathscr{D},\mathscr{P}} \|\theta - \theta'\|.$$

Sous les Hypothèse 5 et 6, la récurrence 3.2 converge presque sûrement vers l'EDO neuronale donnée par

$$H_0 = Ax, \quad dH_t = \mathscr{V}_t g(H_t, \Theta_t)\, dt, \quad t \in [0,1] \tag{3.3}$$

comme la proposition ci-dessous.

Proposition 7 *Considérons le modèle (3.2) tel que les Hypothèse 5 et 6 sont satisfaites. Alors l'EDO (3.3) a une solution unique H, et, presque sûrement, il existe un $c > 0$ tel que, pour tout $0 \leqslant k \leqslant L$,*

$$\left\|H_{k/L} - h_k\right\| \leqslant \frac{c}{L} \tag{3.4}$$

En supposant que les poids du réseau sont des discrétisations d'une fonction lisse (Hypothèse 5), il est possible d'obtenir des résultats de stabilité, en fonction de la valeur de β.

Nous montrons ci-dessous que β est une valeur critique, en examinant les états cachés. Nous avons la proposition suivant.

Proposition 8 *Sous les hypothèses 5 et 6, soit $\alpha_L = \frac{1}{L^\beta}$, avec $\beta > 0$.*

(i) Si $\beta > 1$, alors presque sûrement,

$$\frac{\|h_L - h_0\|}{\|h_0\|} \xrightarrow{L\to\infty} 0.$$

(ii) Si $\beta = 1$, alors presque sûrement,il existe un $c > 0$ tel que

$$\frac{\|h_L - h_0\|}{\|h_0\|} \leqslant c.$$

(iii) Si $\beta < 1$, Le cas de l'explosion est plus délicat à traiter, nous n'en discuteros pas ici.

Preuve 3 (Proposition 8) *En appliquant l'Hypothèse 6, nous pouvons aisément déterminer l'existence de constantes C_1 et C_2, dont les valeurs dépendent des réalisations de $\mathcal{V}$ et Θ, telles que*

$$\|h_{k+1}\| \leqslant (1 + C_1\alpha_L) \|h_k\| + C_2\alpha_L$$

Par récurrence,

$$\|h_{k+1}\| \leqslant (1 + C_1\alpha_L)^k \left(\|h_0\| + \frac{C_2}{C_1} \right).$$

Puis, en utilisant $\alpha_L \leqslant 1/L$,

$$\|h_{k+1}\| \leqslant \exp(C_1) \left(\|h_0\| + \frac{C_2}{C_1} \right).$$

Car g est lipschitzien continu en ensemble compact, il est délimité sur chaque boule de $\mathbb{R}^d \times \mathbb{R}^p$. Le résultat est alors une conséquence de l'identité

$$h_L - h_0 = \alpha_L \sum_{k=0}^{L-1} V_{k+1} g(h_k, \theta_{k+1})$$

puisque nous avons montré que chaque terme de la somme est borné par une constante $C_3 > 0$, indépendante de L et k. Nous avons donc

$$\|h_L - h_0\| \leqslant C_3 L \alpha_L = C_3 L^{1-\beta},$$

donnant les résultats en fonction de la valeur de β.

Proposition 9 *Considèrons le modèle `res-1` ($h_{k+1} = h_k + \alpha_L V_{k+1}\sigma(h_k)$), en prenant σ comme fonction d'identité. Supposons que l'Hypothèse 5 soit satisfaite et que $\mathcal{V}_0^T$ ait une valeur propre positive. Soit $\alpha_L = \frac{1}{L^\beta}$, avec $\beta \in (0, 1)$. Alors,*

$$\max_k \frac{||h_k - h_0||}{||h_0||} \xrightarrow{L\to\infty} \infty.$$

Dans ce contexte, nous pouvons observer expérimentalement une évolution du comportement de la sortie et des gradients lorsque la valeur de L augmente, similaire à celle explorée dans la section précédente. Cependant, le point remarquable est que la séparation se produit pour $\beta = 1$ ici.

3.2 Équation différentielle stochastique

Les équations différentielles stochastiques (EDS) étendent les équations différentielles ordinaires (EDO) en incluant un terme aléatoire, souvent utilisé pour modéliser l'impact de processus aléatoires ou de bruit. Formellement, une EDS s'exprime sous la forme $dy = f(k, y)dk + g(k, y)dB$, où B représente un mouvement brownien ou un processus de Wiener.

En apprentissage profond, les EDS trouvent des applications pratiques pour simuler des systèmes avec un élément aléatoire, comme le bruit durant l'entraînement ou l'initialisation aléatoire des poids. Ces modèles aident à comprendre le comportement des réseaux face à des perturbations aléatoires.

3.2.1 EDS neuronale

Les EDS neuronales se distinguent des EDO neuronales par l'intégration d'un aspect aléatoire, ce qui permet une meilleure gestion de l'incertitude et du bruit dans les données. Le mouvement brownien, B est un modèle mathématique pour décrire le chemin d'une marche aléatoire. Dans le contexte des réseaux de neurones profonds, il peut modéliser les fluctuations aléatoires, telles que les variations dans les mises à jour de poids ou les valeurs d'activation, avec un impact particulièrement marqué dans les architectures multicouches où ces fluctuations peuvent s'accumuler.

Définition 2 *Le mouvement brownien unidimensionnel $(B_t)_{t\geqslant 0}$ est un processus stochastique continu,avec des incréments indépendants, dépendant du temps t et vérifiant : $B_0 = 0$ et pour tous $0 \leqslant s \leq t \leqslant 1, B_t - B_s \sim \mathcal{N}(0, t-s)$.*

L'un des principaux messages du Chapitre 2 est que l'initialisation standard avec les paramètres i.i.d. conduit à un modèle non dégénéré pour les grandes valeurs de L seulement lorsque $L\alpha_L^2 \approx 1$. C'est à dire pour $\beta = 1/2$ avec $\alpha_L = \frac{1}{L^\beta}$.

(Par les propositions et corollaires d'avance). De manière remarquable, il convient de noter que ce régime correspond à la discrétisation d'une EDS dans la limite du temps continu. Pour étayer cette affirmation, prenons en compte, à des fins de simplification, le modèle ResNet `res-1` discret :

$$h_{k+1} = h_k + \frac{1}{\sqrt{L}} V_{k+1}\sigma(h_k), 0 \leqslant k \leqslant L-1 \tag{3.5}$$

où les entrées de V_{k+1} sont supposées être i.i.d et suivant une loi normale $\mathcal{N}(0, 2/d)$. Maintenant, on pose $\mathbf{B} : [0,1] \to \mathbb{R}^{d\times d}$ comme un movement brownien de dimension $(d \times d)$, ainsi on retrouve $(B_{ij})_{1\leqslant i,j\leqslant d}$ comme un mouvement brownien unidimensionnel. Maintenant, pour tous $0 \leqslant k \leqslant L-1$ et tous $1 \leqslant i, j \leqslant d$, on a

$$\mathbf{B}_{(k+1)/L,i,j} - \mathbf{B}_{k/L,i,j} \sim \mathcal{N}\left(0, \frac{1}{L}\right).$$

et les incréments pour différentes valeurs de (i, j, k) sont indépendants. En conséquence, l'Equation (3.5) est équivalente en distribution à la récurrence

$$h_{k+1}^\top = h_k^\top + \sqrt{\frac{2}{d}}\sigma\left(h_k^\top\right)\left(\mathbf{B}_{(k+1)/L} - \mathbf{B}_{k/L}\right), \quad 0 \leqslant k \leqslant L-1.$$

cat V_{k+1} a même distribution de $V_{k+1}^\top$. On peut obtenir que $\{k/L, 0 \leqslant k \leqslant L\}$

$$dH_t^\top = \sqrt{\frac{2}{d}}\sigma\left(H_t^\top\right) d\mathbf{B}_t, \quad t \in [0,1] \tag{3.6}$$

où la sortie du réseau est désormais fonction de la valeur finale de H, c'est-à-dire $H1$. Le lien entre le ResNet discret (3.5) et l'EDS (3.6) est formalisé dans la proposition suivante.

Proposition 10 *Considérons le modèle `res-1`, où les entrées de V_{k+1} sont des variables aléatoires i.i.d., gaussiennes $\mathcal{N}(0, 2/d)$. Supposons que la fonction d'activation σ soit lipschitzienne. Alors l'EDS (3.6) a une unique solution H et, pour tout $0 \leqslant k \leqslant L$,*

$$\mathbb{E}\left(\left\|H_{k/L} - h_k\right\|\right) \leqslant \frac{c}{\sqrt{L}},$$

pour un $c > 0$.

Preuve 4 (Proposition 10) *La proposition est une conséquence de Kloeden and Platen (1992, Théorèmes 4.5.3 et 10.2.2) pour les EDS*

$$dH_t^\top = \sqrt{\frac{d}{2}}\sigma\left(H_t^\top\right) dB_t$$

Supposons $a(h,t) = 0$ et $b(h,t) = \sqrt{\frac{d}{2}}\sigma(h)$,On doit vérifier les hypothèses suivantes:

(H_1) Les fonctions $a(\cdot,\cdot)$ et $b(\cdot,\cdot)$ sont jointe mesurables en $\mathbb{R}^d \times [0,1]$.

(H_2) Il existe un constante $C_1 > 0$ tel que, pour tous $x, y \in \mathbb{R}^d, t \in [0,1]$,

$$\|a(x,t) - a(y,t)\| + \|b(x,t) - b(y,t)\| \leqslant C_1\|x - y\|.$$

(H_3) *Il existe un constante* $C_2 > 0$ *tel que, pour tous* $x \in \mathbb{R}^d, t \in [0,1]$,

$$\|a(x,t)\| + \|b(x,t)\| \leqslant C_2(1 + \|x\|).$$

(H_4) $\mathbb{E}\left(|\ H_0\|^2\right) < \infty$.

(H_5) *Il existe un constante* $C_3 > 0$ *tel que, pour tous* $x \in \mathbb{R}^d, s, t \in [0,1]$,

$$\|a(x,t) - a(x,s)\| + \|b(x,t) - b(x,s)\| \leqslant C_3(1 + \|x\|)|t - s|^{1/2}.$$

Les Hypothèses $(H_1), (H_4)$, *et* (H_5) *découlent facilement des définitions. L'Hypothèse* (H_2) *est vrai car* σ *est lipschitzienne, et* (H_3) *découle de*

$$\|\sigma(x)\| \leqslant b\|x\| \leqslant \|x\| \leqslant 1 + \|x\|.$$

Fixer le facteur d'échelle β à $1/2$ ne se limite pas à générer un comportement non trivial à l'initialisation comme vu dans le Chapter 2; cela correspond aussi à un modèle de diffusion particulièrement "simple" dans l'approche en temps continu. Cette observation suggère que les réseaux de neurones très profonds peuvent être considérés comme équivalents à la solution d'une équation différentielle stochastique (EDS) lorsqu'une initialisation de poids indépendante et identiquement distribuée (i.i.d.) est utilisée.

3.3 Conclusion

La plupart des fonctions d'activation classiques (telles que ReLU) sont lipschitzienne. Cela indique que ces fonctions ont certaines limites en termes de taux de changement, ce qui est important pour la stabilité et la prévisibilité du réseau.

1. Lorsque le facteur d'échelle $\beta = 1$ ($\alpha = \frac{1}{L}$) et que l'initialisation des poids n'est pas i.i.d., le modèle correspondant tend vers une EDO. Parce que les poids ne sont pas identifiés, le comportement du réseau est plus déterministe et peut être affecté par une stratégie d'initialisation ou une distribution de poids spécifique. Dans ce cas, il est approprié d'utiliser des EDO pour simuler le comportement du réseau, car les EDO fournissent un moyen d'analyser les systèmes dynamiques dans un cadre déterministe.

2. Lorsque le facteur d'échelle $\beta = 1/2$ ($\alpha = \frac{1}{\sqrt{L}}$) et que l'initialisation des poids est i.i.d., le modèle correspondant tend vers une EDS.

Dans l'ensemble, le choix d'utiliser les EDS ou les EDO dépend de la nature des poids dans le modèle (i.i.d. ou non-i.i.d.) et du comportement du réseau que nous souhaitons capturer (stochastique ou déterministe).

Ce modèle est intéressant dans la mesure où le mouvement brownien des EDS est $(\frac{1}{2} - \epsilon)$-Holder, un processus stochastique (EDO) est 1-Holder.

Il est important de noter que le choix de la mise à l'échelle d'un ResNet semble être étroitement lié à la régularité des poids en fonction de la couche. Plus précisément, dans tous les régimes, le facteur d'échelle critique entre l'explosion et l'identité semble être étroitement lié à la régularité des poids en fonction de la couche. Ces résultats ont une interprétation naturelle en termes de régularité (Holder) du processus stochastique en temps continu sous-jacent.

Ces modèles en temps continu, à la fois équations différentielles ordinaires (EDO) et équations différentielles stochastiques (EDS), offrent un cadre exhaustif pour l'analyse et l'interprétation du comportement des réseaux de neurones résiduels (ResNets) profonds. Ils établissent ainsi un lien entre les architectures de deep learning discrètes et la théorie approfondie des équations différentielles.

Chapter 4

Conclusion

Ce tutoriel présente une analyse complète des défis liés à la mise à l'échelle dans les ResNets profonds. Il met en évidence l'importance du facteur de mise à l'échelle α_L, de l'analyse probabiliste et des informations fournies par les modèles en temps continu. En combinant une analyse théorique et des preuves empiriques, on parvient à une compréhension approfondie des mécanismes impliqués dans la formation des ResNets profonds. Cette approche ouvre la voie à la création d'architectures d'apprentissage profond plus efficaces et efficientes à l'avenir.

Premièrement, il convient de mentionner que les Resnets ont été un véritable exploit dans le domaine de l'apprentissage automatique complexe. Ils ont été les premiers modèles de réseaux neuronaux profonds à être entraînés avec succès avec un grand nombre de couches, ce qui a considérablement amélioré les performances. Étant donné son application étendue et son importance, il est essentiel de réfléchir à la manière de construire un cadre "parfait" afin d'éviter tout problème de disparition ou d'explosion de gradient lors de l'entraînement en profondeur, qui pourrait conduire à de mauvais résultats.

Deuxièmement, nous avons constaté, à travers de nombreuses expériences, que la répartition des valeurs initiales (les poids $(V_k)_{1\leqslant k\leqslant L}$ et $(\theta_k)_{1\leqslant k\leqslant L}$)joue un rôle crucial dans les résultats de l'entraînement. Par conséquent, il est impératif d'examiner attentivement la régularité de l'évolution des poids dans le processus de descente de gradient et son impact sur la dynamique de l'entraînement.

La relation entre le taux d'apprentissage et le facteur de mise à l'échelle α_L joue un rôle essentiel dans l'évaluation plus précise de la corrélation entre les performances du réseau formé et la mise à l'échelle.

Bibliography

François Chollet et al. Keras. https://keras.io, 2015.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. doi: 10.1109/CVPR.2016.90.

Peter E. Kloeden and Eckhard Platen. *Stochastic Differential Equations*. Springer, 1992. ISBN 978-3-662-12616-5. doi: 10.1007/978-3-662-12616-5. URL http://dx.doi.org/10.1007/978-3-662-12616-5_4.

TorchVision maintainers and contributors. Torchvision: Pytorch's computer vision library. https://github.com/pytorch/vision, 2016.

Hongyu Wang, Shuming Ma, Li Dong, Shaohan Huang, Dongdong Zhang, and Furu Wei. Deepnet: Scaling transformers to 1,000 layers, 2022.

Greg Yang and Samuel S. Schoenholz. Mean field residual networks: On the edge of chaos. *CoRR*, abs/1712.08969, 2017. URL http://arxiv.org/abs/1712.08969.