

Modélisation supervisée de données fonctionnelles par perceptron multi-couches

Brieuc Conan-Guez

▶ **To cite this version:**

Brieuc Conan-Guez. Modélisation supervisée de données fonctionnelles par perceptron multi-couches. Mathématiques [math]. Université Paris Dauphine - Paris IX, 2002. Français. NNT : . tel-00178892

HAL Id: tel-00178892
https://theses.hal.science/tel-00178892

Submitted on 12 Oct 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Université Paris IX Dauphine

Thèse

présentée par **M. Brieuc CONAN-GUEZ**

pour obtenir le grade de docteur de l'Université Paris IX Dauphine

spécialité : mathématiques appliquées

Modélisation supervisée de données fonctionnelles par perceptron multi-couches

Soutenue le 18 décembre 2002 devant la Commission d'examen composée de :

Besse	Philippe	Examinateur
Cottrell	Marie	Rapporteur
Diday	Edwin	Directeur de thèse
Ferré	Louis	Rapporteur
Lechevallier	Yves	Examinateur
Rossi	Fabrice	Directeur de thèse
Verleysen	Michel	Examinateur

Thèse préparée au sein du Centre de Recherche de Mathématiques de la Décision (Université Paris-IX Dauphine) et de l'INRIA (Projet AXIS, Centre de Rocquencourt)

Remerciements

J'aimerais par ces quelques lignes remercier très vivement Fabrice ROSSI, Maître de conférences à l'université Paris IX-Dauphine. Outre la confiance qu'il m'a accordée en me proposant d'effectuer une thèse sous sa direction, il m'a permis de progresser continuellement dans ma formation d'enseignant et de chercheur. La qualité de l'encadrement scientifique qu'il procure, son enthousiasme ajoutés à son soutien permanent ont été prépondérants dans l'aboutissement de ce travail.

Un grand merci également à Yves LECHEVALLIER, Directeur de Recherches à l'INRIA, pour m'avoir accueilli au sein de son équipe. Sa connaissance très profonde de l'analyse de données et de la statistique, sa disponibilité, et sa patience en ont fait pour moi un interlocuteur privilégié durant ces quelques années.

Table des matières

1 Introduction **7**

2 Analyse de Données Fonctionnelles **11**
2.1 Introduction 11
2.2 Analyse de Données Fonctionnelles 12
2.3 Quelques méthodes de l'ADF 15
2.3.1 La représentation des fonctions 16
2.3.2 Les analyses factorielles 19
2.3.3 Les modèles linéaires fonctionnels 23
2.3.4 Autres méthodes 28
2.4 Conclusion 31

3 Le perceptron multi-couches fonctionnel **33**
3.1 Introduction 33
3.1.1 Les différents modèles 33
3.1.2 Le cadre fonctionnel 35
3.2 Neurone Fonctionnel 36
3.2.1 Neurone général 36
3.2.2 Neurone dans un espace de Banach 37
3.2.3 Neurone dans les espaces L^p 37
3.3 Perceptron multi-couches fonctionnel 39
3.4 Entrées multiples 39
3.5 Approche paramétrique 41
3.6 Apprentissage 43
3.6.1 Calcul du gradient 43
3.6.2 Rétro-propagation 44
3.7 Conclusion 44

4 Approximation universelle **45**
4.1 Introduction 45

4.2 Définitions 46
4.2.1 Espaces fonctionnels et distances associées 46
4.2.2 Perceptron à une couche cachée 47
4.3 Résultats existants 48
4.4 Corollaires pour le perceptron multi-couches fonctionnel 49
4.5 Discussion 51
4.6 Entrées multiples 52
4.7 Conclusion 52

5 Cadre Probabiliste 55
5.1 Introduction 55
5.2 Loi forte des grands nombres uniforme 56
5.3 Connaissance parfaite des fonctions 58
5.3.1 Cadre probabiliste 58
5.3.2 Consistance 59
5.3.3 Discussion 62
5.4 Connaisance limitée des fonctions 63
5.4.1 Cadre probabiliste 63
5.4.2 Conséquences pour l'approximation universelle 64
5.4.3 Consistance 65
5.4.4 Discussion 71
5.5 Mise en oeuvre pratique 72
5.5.1 Introduction 72
5.5.2 Cas général 72
5.5.3 Régresseurs paramétriques pseudo-linéaires 74
5.5.4 Liens avec des travaux antérieurs 77
5.6 Conclusion 77

6 Approche par projection 79
6.1 Introduction 79
6.2 Approche par projection 81
6.2.1 Etape de projection 81
6.2.2 Perceptron Multi-couches Fonctionnel basé sur une étape de projection 81
6.2.3 Approche paramétrique 82
6.2.4 Liens avec le perceptron multi-couches numériques 84
6.2.5 Coût algorithmique 85
6.2.6 Régression des fonctions d'entrée 86
6.3 Approximation universelle 87
6.3.1 Définitions 87

6.3.2 Approximation universelle 87
6.4 Cadre probabiliste 88
6.4.1 Connaissance parfaite des fonctions 89
6.4.2 Connaissance limitée des fonctions 91
6.5 Conclusion 97

7 Perceptron multi-couches fonctionnel à valeurs fonctionnelles 99
7.1 Introduction 99
7.2 Perceptron fonctionnel à valeurs fonctionnelles 100
7.2.1 Présentation du modèle 100
7.2.2 Liens avec un modèle existant 101
7.3 Approximation universelle 102
7.3.1 Définitions 102
7.3.2 Approximation universelle 103
7.4 Cadre probabiliste 105
7.4.1 Connaissance parfaite des fonctions 105
7.4.2 Connaissance limitée des fonctions 108
7.5 Conclusion 112

8 Simulations 113
8.1 Introduction 113
8.2 Initialisation 114
8.2.1 Initialisation géométrique 114
8.2.2 Adaptation aux modèles fonctionnels 116
8.3 Les fonctions sinus 117
8.3.1 Les fonctions d'entrée 117
8.3.2 Les différents modèles 119
8.3.3 Résultats 120
8.4 Les cercles 123
8.4.1 Les fonctions d'entrée 123
8.4.2 Modèles 124
8.4.3 Résultats 126
8.5 Conclusion 128

9 Conclusions et Perspectives 129

Chapitre 1

Introduction

A l'ère du tout numérique, la multiplication des systèmes d'information est la cause d'un accroissement constant de la masse de données à traiter. Cette inflation toujours plus importante révèle les limites des techniques d'exploitation et d'analyse actuelles, et impose le développement puis la mise en œuvre d'outils nouveaux, adaptés à cette profusion de données.

L'Analyse de Données Fonctionnelles (ADF) est un domaine de la statistique qui apporte entre autres plusieurs éléments de réponse au problème soulevé ci-dessus. Comme le soulignent Ramsey et Silverman dans leur livre [63], qui constitue une excellente introduction au domaine, ce champ de recherche a trouvé un réel écho auprès de la communauté des statisticiens, et a donc fait l'objet de nombreux travaux, tant théoriques que pratiques (les auteurs donnent une liste importante d'exemples qui montrent le large potentiel applicatif des différentes méthodes liées à l'Analyse de Données Fonctionnelles).

L'hypothèse sur laquelle repose l'Analyse de Données Fonctionnelles est que les données à traiter possèdent une structure sous-jacente plus ou moins apparente, et que l'identification et la prise en compte explicite de cette structure peut être utilisée afin d'étendre efficacement les techniques de l'analyse de données traditionnelles. Plus précisément, et comme son nom l'indique, l'Analyse de Données Fonctionnelles s'applique aux données dont la structure est représentée correctement par une ou plusieurs fonctions. Cette modélisation est particulièrement fructueuse dans le cas où les données présentent par exemple une variabilité temporelle : Ramsay et Silverman [63] illustrent ce cas en s'intéressant à la croissance d'un groupe d'enfants au cours du temps. Ils montrent que la description de chaque individu par une fonction régulière (la taille de chaque enfant en fonction de l'âge) est particulièrement bien adaptée à la nature du problème. De plus, dans ce cas précis, cette approche fonctionnelle permet de mettre en évidence des caractéristiques non décelables par des techniques traditionnelles :

par exemple l'identification des phases de croissance grâce à l'étude de la dérivée de la fonction. A la lumière de cet exemple, on constate que le traitement de données volumineuses n'est qu'une des possibilités offertes par la modélisation fonctionnelle. Comme on le verra dans le chapitre 2, la prise en compte de la structure des données a en effet de nombreux avantages comme la possibilité de tenir compte d'informations *a priori* sur les données (périodicité, régularité, etc), d'éviter les conséquences néfastes de la redondance des informations (dans le cas du modèle linéaire par exemple), etc.

Bien que de nombreuses méthodes classiques aient été adaptées avec succès à l'Analyse de Données Fonctionnelles (par exemple, les modèles à noyaux ou le modèle linéaire), on note cependant que les méthodes neuronales semblent absentes du panorama fonctionnel. Or comme nous le rappellerons dans les chapitres suivants, le Perceptron Multi-Couches est un modèle semi-paramétrique très intéressant, car il permet une modélisation parcimonieuse d'un ensemble de données (c'est-à-dire que le modèle a besoin de peu de paramètres pour réaliser une modélisation fine des données). Il offre donc en ce sens une alternative intéressante aux modèles non paramétriques et aux modèles semi-paramétriques pseudo-linéaires.

Le but de cette thèse est donc d'effectuer la jonction entre le domaine de l'Analyse de Données Fonctionnelles et celui des techniques neuronales classiques. On s'attachera notamment au cours des chapitres suivants à montrer que le perceptron multi-couches peut être aisément étendu au cadre fonctionnel, ce qui permet d'appliquer une transformation non-linéaire à des données fonctionnelles. On montre de plus que cette extension est réalisée en conservant les propriétés théoriques importantes de ce modèle (approximation universelle et estimation consistante des paramètres).

Le plan de la thèse s'organise de la manière suivante :

- le chapitre deux est consacrée à la présentation de l'Analyse de Données Fonctionnelles. Nous avons cherché à une modeste échelle à compléter le livre de Ramsay et Silverman [63], en rappelant les résultats théoriques existants et en dressant un panorama partiel des avancées les plus récentes.
- le troisième chapitre est consacrée à l'extension du perceptron multi-couches au cadre fonctionnel. Le modèle proposé s'appuie sur les travaux antérieurs menés par Sandberg [69], Sandberg et Xu [70], Chen [19] et Stinchcombe [75]. Il est important de noter que ces différentes études ont été menées d'un point de vue purement théorique, et que donc la mise en œuvre pratique de ce modèle n'a pas été abordée par ces auteurs. Dans cette thèse, une attention particulière a été accordée à la mise en œuvre pratique ainsi qu'aux problèmes théoriques soulevés par ce nouveau modèle ;

– dans le chapitre 4, on démontre un résultat théorique important, qui s'appuie sur le travail de Stinchcombe [75] : le perceptron multi-couches fonctionnel est un approximateur universel. Ce résultat est la justification théorique de l'utilisation du perceptron multi-couches fonctionnel pour tout problème d'approximation de fonctions (par exemple, dans le cas d'un problème de régression où l'on cherche à approcher la fonction de régression $E(Y|X)$) ;
– le chapitre 5 montre comment on peut adapter en pratique le perceptron multi-couches fonctionnel afin de prendre en compte la discrétisation des fonctions d'entrée (approche par traitement direct des fonctions d'entrée). Dans la suite du chapitre, on énonce deux résultats de consistance différents : dans le premier, on fait l'hypothèse que les fonctions d'entrée sont connues de manière parfaite, alors que dans le second, l'échantillonnage des fonctions d'entrée est pris en compte ;
– dans le chapitre 6, on propose une approche alternative à celle étudiée dans le chapitre 5. Cette méthode est basée sur une étape de projection préalable des fonctions d'entrée, qui est une technique classique en Analyse de Données Fonctionnelles. Cette phase de projection présente deux avantages par rapport à l'approche directe (chapitre 5) : premièrement, elle permet un débruitage des fonctions d'entrée, ce qui facilite la phase d'estimation du modèle. De plus, on verra que dans certains cas, l'étape de projection permet aussi une réduction du temps d'apprentissage. Les deux propriétés théoriques fondamentales (approximation universelle et consistance) sont démontrées dans ce nouveau cadre ;
– dans le chapitre 7, on montre que le perceptron multi-couches fonctionnel peut être adapté afin d'obtenir une réponse fonctionnelle. Cette adaptation est dans la pratique très intéressante, car elle permet par exemple de modéliser des processus fonctionnels à temps discret. On démontre pour ce nouveau modèle la propriété d'approximation universelle, ainsi que celle de consistance (sous l'hypothèse d'indépendance des individus) ;
– dans le dernier chapitre, on propose l'adaptation d'une technique d'initialisation issue de l'approche classique au cadre du perceptron multi-couches fonctionnel. Cette technique permet une réduction du temps de calcul nécessaire à l'optimisation du modèle, ainsi qu'une meilleure utilisation des ressources du réseau. On termine ce chapitre en comparant les différentes approches fonctionnelles grâce à deux expériences distinctes.

Chapitre 2

Analyse de Données Fonctionnelles

2.1 Introduction

L'informatisation d'un nombre croissant d'activités humaines produit un volume très important de données dont l'analyse et l'exploitation posent des problèmes complexes, comme par exemple :

le volume des données : il est fréquent en météorologie par exemple d'avoir à traiter des tera-octets de données *tous les jours*. Dans ces conditions, une approche naïve (traiter directement les données) est très difficilement envisageable car même des algorithmes de complexité linéaire peuvent demander trop de temps calcul pour être utilisables ;

la redondance dans les observations : comme l'acquisition et le stockage de l'information sont devenus beaucoup plus faciles, il n'est pas rare de multiplier les mesures. Au lieu d'avoir par exemple une image satellite comportant 6 canaux, on peut obtenir aisément 50 canaux. Les informations contenues dans les différents canaux peuvent être fortement corrélées, ce qui n'est pas toujours compatible avec les méthodes statistiques classiques (par exemple pour les méthodes de régression linéaire, cf la section 2.3.3) ;

la structure des données : pour prolonger le point précédent, on constate que la redondance des observations est un cas particulier de structure dans celles-ci. On sait par exemple que le climat est grossièrement périodique. Quand on mesure des grandeurs physiques liées au climat, on s'attend donc à observer une forme de périodicité.

De nombreux autres problèmes se posent, comme tout simplement celui d'un stockage intelligent (dans des entrepôts de données), d'une visualisation simple, etc.

L'Analyse de Données Fonctionnelles (ADF), que nous présentons dans les sections suivantes, apporte des éléments de réponse à certains des problèmes présentés.

2.2 Analyse de Données Fonctionnelles

Le principe de base de l'Analyse de Données Fonctionnelles (ADF), Ramsay et Silverman [63], est de considérer que les individus étudiés sont décrits par des fonctions plutôt que par des vecteurs de $\mathbb{R}^n$. En d'autres termes, chaque variable observée est à valeurs fonctionnelles plutôt que réelles.

Ce principe est naturel dans de nombreux cas, comme l'illustrent les exemples suivants :

- l'étude du climat est un domaine dans lequel la modélisation fonctionnelle est particulièrement adaptée. En effet, les "individus" observés présentent systématiquement une variabilité temporelle et/ou spatiale.
  L'exemple le plus simple est celui de l'étude du climat dans un pays donné, par l'intermédiaire des variations de la température mensuelle moyenne au cours de l'année en différents points du pays. Chaque station météo est alors un individu qui est décrit par une fonction qui au mois associe la température moyenne observée (cf Ramsay et Silverman [63]).
  On peut aussi observer une zone géographique avec deux échelles temporelles. On peut par exemple étudier l'évolution de diverses mesures géophysiques sur une zone donnée pendant un an. Chaque année d'observations correspond à un individu et on étudie la série temporelle des individus, comme dans la modélisation auto-régressive fonctionnelle du phénomène *El niño* proposée dans Besse et al. [6] ;
- à partir des exemples climatiques, on constate que l'ADF est bien adaptée aux problèmes dans lesquels les individus étudiés présentent une variabilité temporelle. En dehors de la météo, on peut citer un des exemples de Ramsay et Silverman [63], à savoir l'étude de la croissance d'enfants : chaque enfant est représenté par une fonction qui à l'age associe la taille. Un autre exemple est donné dans Abraham et al. [1] qui étudie l'évolution du pH lors du processus d'acidification des fromages (chaque fromage est décrit par l'évolution de son pH au cours du temps). On peut aussi citer Rice et Wu [66] (entre autres), article qui présente l'exemple de l'évolution du nombre de lymphocytes T4 dans le sang de malades atteints du SIDA.

En fait, la variabilité temporelle est extrêmement fréquente dans les données réelles (en particulier quand on possède des observations à haute résolution d'un phénomène quelconque), et l'approche fonctionnelle permet d'éviter d'avoir à simplifier radicalement les observations en les remplaçant par une moyenne par exemple. Elle permet aussi de traiter des données pour lesquelles les instants d'observation sont différents pour chaque individu, ce qui est très difficile dans l'approche classique (problème de données non directement comparables et de données manquantes) ;

– l'ADF permet aussi de prendre en compte la variabilité géographique, par exemple lors de l'étude de l'évolution au cours du temps d'une ou plusieurs grandeurs observées en plusieurs emplacements. On pense bien entendu à des applications climatiques, comme l'étude de l'évolution de la pluviométrie sur une zone donnée : un individu est décrit par la fonction qui à des coordonnées géographiques (latitude et longitude par exemple) associe la hauteur de pluie reçue sur une période de temps donnée. Chaque individu correspond donc à une période de temps différente.
– l'ADF est bien entendu très naturelle dans tous les domaines où on sait représenter une variabilité dans les observations qui caractérisent un individu sous la forme d'une ou plusieurs fonctions. On peut songer par exemple à des mesures spectroscopiques (par exemple des sources lumineuses caractérisées par leur spectre d'émission), à des courbes de réponse (allongement d'un ressort en fonction de l'effort exercé), etc.
Un exemple important est celui des images : on peut considérer une image comme une fonction qui à des coordonnées dans $\mathbb{R}^2$ associe une couleur. Ce type de représentation a été très fructueux en traitement d'images (e.g., Morel et Solimini [57]).

De façon générale, l'ADF offre une réponse aux problèmes évoqués dans la section précédente. L'ADF permet souvent de résumer les données en remplaçant les observations réelles par une modélisation fonctionnelle simple. Comme l'illustrent Ramsay et Silverman [63], il est effet très classique d'approcher les fonctions qui décrivent chaque individu par une représentation régulière, par exemple en projetant sur une base de splines ou d'ondelettes. Cette technique permet à la fois de réduire le volume des données (puisqu'on peut travailler sur les coordonnées du projeté à la place des données brutes) mais aussi de prendre en compte la structure des données. Si on sait par exemple que les données sont périodiques, il est intéressant d'utiliser une représentation périodique sous forme de séries de Fourier. De façon générale, on peut tenir compte d'informations *a priori* sur les individus pendant la phase de modélisation. La redondance dans les données, et plus généralement les liens entre les observations peuvent souvent être pris en compte sous forme de contraintes de régularité sur les fonctions

manipulées. La phase de modélisation permet justement d'introduire ce type de contrainte : on peut par exemple représenter les individus par des fonctions C^2 (des splines par exemple).

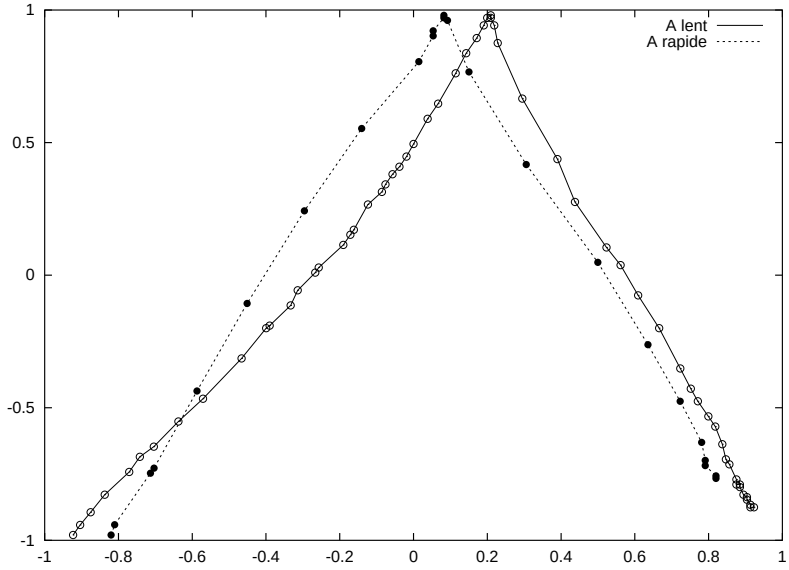


FIG. 2.1 – *Graffiti* pour la lettre 'A'

De plus, l'ADF permet de traiter des données difficiles à soumettre à des méthode classiques, même en négligeant les problèmes de volume ou de prise en compte de la structure. En effet, il n'est pas rare d'observer des individus par l'intermédiaire de mesures impossibles à comparer directement. Considérons l'exemple simple de l'écriture *Graffiti* des assistants personnels de type *Palm Pilot*. L'utilisateur dessine d'un seul trait (le levé du stylet n'est pas autorisé) une courbe qui est ensuite reconnue et associée à une des 26 lettres de l'alphabet. L'écran tactile possède une certaine résolution temporelle, c'est-à-dire qu'il note la position du stylet toute les τ microsecondes. Cependant, on peut difficilement imposer à l'utilisateur de tracer les caractères toujours à la même vitesse. De ce fait, la courbe correspondant à la lettre A (par exemple) peut très bien être définie par 24 positions du stylet comme par 62 positions (voir la figure 2.1). Le système de reconnaissance procède donc à un recalage temporel du tracé réalisé par un utilisateur. Si on recale toutes les courbes sur l'intervalle temporel $[0, 1]$, un tracé à 24 positions correspond à 24 mesures avec un pas de $\frac{1}{23}$ entre deux mesures. Par contre, un tracé à 62 positions correspond à 62 mesures avec un pas de $\frac{1}{61}$. Non seulement le nombre d'observations dépend de l'individu, mais de plus, les observations ne sont pas directement comparables. Par exemple, la

deuxième mesure de la courbe à 24 positions correspond à l'instant $\frac{1}{23}$ (après recalage), alors qu'elle correspond à l'instant $\frac{1}{61}$ pour la courbe à 62 positions (voir les figures 2.2 et 2.3). Il est très difficile de faire une analyse classique de ce genre de données, car le nombre de variables qui décrit chaque individu dépend de l'individu, et, pire encore, le sens à accorder à chaque variable dépend aussi de l'individu. Au contraire, en ADF, on compare deux courbes, ce qui ne pose pas vraiment de problème, en particulier après représentation régulière (ce qui induit une interpolation).

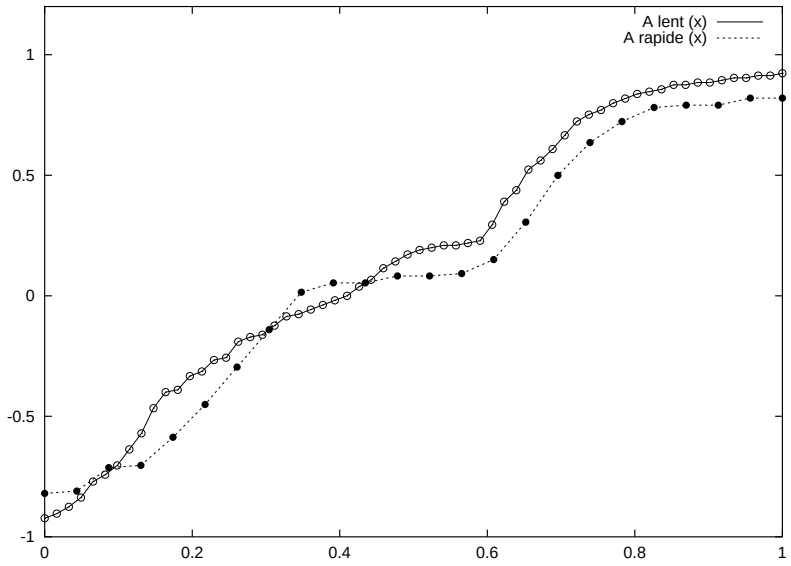


FIG. 2.2 – *Graffiti* pour la lettre 'A' : évolution de l'abscisse du stylet

2.3 Quelques méthodes de l'ADF

Le livre de Ramsay et Silverman [63] présente de façon assez synthétique et complète les méthodes les plus classiques en ADF. Cependant, cet ouvrage est avant tout orienté vers la mise en œuvre pratique des algorithmes et contient très peu de résultats théoriques. De plus, il ne contient bien entendu pas les développements les plus récents de l'ADF. En complément de Ramsay et Silverman [63], nous proposons donc dans les sections suivantes un panorama des méthodes de l'ADF, en mentionnant les résultats théoriques importants. Bien entendu, l'ADF forme un domaine trop large et trop actif pour prétendre à une

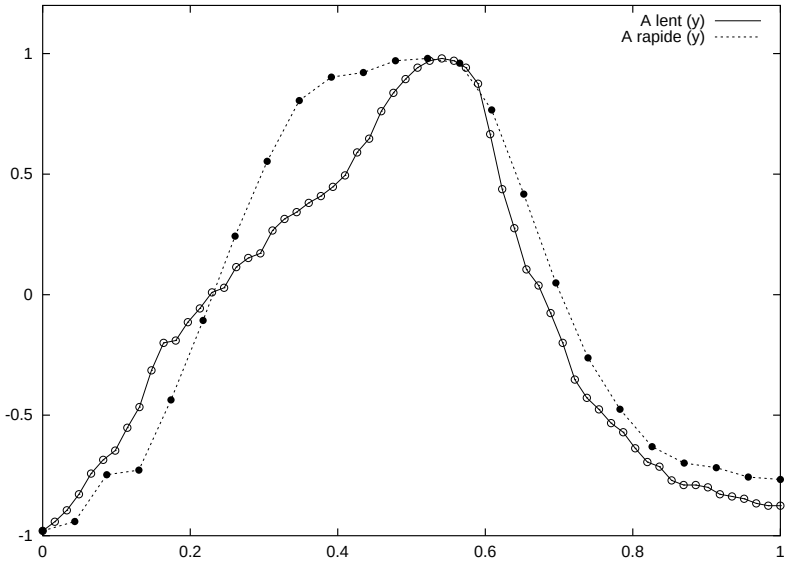


FIG. 2.3 – *Graffiti* pour la lettre 'A' : évolution de l'ordonnée du stylet

quelconque exhaustivité[1].

2.3.1 La représentation des fonctions

Une des principales difficultés de l'ADF est bien entendu la manipulation des fonctions. Quand on traite des données classiques numériques, on se contente de manipuler des vecteur de $\mathbb{R}^n$, sans que cela ne pose de problème. Dès qu'on quitte ce cadre simple, on rencontre des difficultés. On sait par exemple qu'il ne faut pas recoder trop naïvement les données nominales pour éviter d'introduire une distorsion dans les analyses effectuées Lebart et al. [50]. En ADF, le problème devient particulièrement délicat car il est très difficile de manipuler informatiquement des fonctions arbitraires.

La situation la plus favorable est celle où les fonctions sont toutes évaluées aux mêmes points. Si on dispose par exemple de m évaluations pour n fonctions, on étudie en fait un tableau de données à m lignes et n colonnes, les $f_i(x_j)$ ($1 \leq i \leq n$ et $1 \leq j \leq m$). Cette situation simplifie énormément les problèmes car les "fonctions" sont directement comparables. On pourrait même faire une analyse classique en "oubliant" l'aspect fonctionnel et en considérant qu'on observe des données dans $\mathbb{R}^m$. Bien entendu, cette dernière solution n'est

[1]On trouvera des informations complémentaires dans les recueils de résumés des exposés donnés dans le cadre du groupe STAPH, STAPH [73] et STAPH [74]

pas très satisfaisante (et pose des problèmes dus aux données fonctionnelles, cf par exemple la section 2.3.3), mais elle montre que du point de vue pratique, le cas des données directement comparables reste relativement simple. En gardant l'aspect fonctionnel, tout se passe comme si on travaillait avec comme espace de départ pour les fonctions l'ensemble fini $D = \{x_1, \ldots, x_m\}$. On peut définir sur l'ensemble des fonctions de D dans $\mathbb{R}$ différentes normes classiques (comme la norme uniforme et la norme quadratique), la notion de produit scalaire, et, plus généralement, toutes opérations numériques utiles pour les méthodes de l'ADF. On peut aussi effectuer un traitement plus élaboré, par exemple en introduisant de la régularisation, par l'intermédiaire d'une représentation par splines d'interpolation (ou de lissage). Comme les fonctions sont évaluées aux mêmes points, on utilise les mêmes noeuds pour les splines de toutes les fonctions. On peut calculer exactement différentes opérations fonctionnelles sur les splines qui représentent les fonctions étudiées.

Malheureusement, il est fréquent d'obtenir des fonctions qui ne sont pas régulièrement échantillonnées, comme l'illustre l'exemple du *Graffiti* présenté dans la section précédente. Dans ce cas, la situation se complique radicalement. Dans toutes les méthodes de l'ADF, on est amené à comparer des fonctions entre elles. On peut distinguer deux grandes catégories de comparaisons :

1. Les comparaisons entre une fonction observée (une donnée) et une fonction contrôlée par la méthode : cette situation est la plus simple. En effet, les fonctions qui sont produites par les méthodes de l'ADF peuvent parfaitement être représentées par une des très nombreuses méthodes classiques d'approximation de fonctions (représentation polynomiale, par splines, par séries de Fourier, par réseaux de neurones, etc.). Avec une telle représentation, il est possible d'évaluer les fonctions produites par la méthode en n'importe quel point et donc de les comparer avec les fonctions observées. Si on souhaite calculer par exemple une distance quadratique entre une fonction observée (i.e., les $(x_j, f(x_j))$) et une fonction contrôlée par la méthode, i.e., g, on calcule simplement

$$\|f - g\|_2^2 \simeq \frac{1}{m} \sum_{j=1}^{m} \|f(x_j) - g(x_j)\|^2$$

   En fait, on contourne le problème de l'hétérogénéité des fonctions observées en les comparant à des fonctions parfaitement connues. Pour chaque fonction observée, on est dans la situation du cas simple.

2. Les comparaisons entre fonctions observées : cette situation est la plus délicate car les fonctions observées sont, par hypothèse, impossible à comparer directement. Il faut donc réaliser une forme d'interpolation pour

pouvoir comparer ces fonctions.

Dans les deux cas, on voit que certaines fonctions seront représentées par une méthode classique d'approximation de fonctions. Même si la représentation des fonctions observées n'est pas toujours indispensable pour certaines méthodes, elle peut être très utile. Il est par exemple classique de supposer que les fonctions observées sont en fait éléments d'un ensemble $\mathcal{F}$ de fonctions régulières exactement représentables (par exemple l'espace vectoriel engendré par les premiers éléments d'une base topologique dénombrable de l'espace fonctionnel considéré, i.e., $vect(\phi_1, \ldots, \phi_q)$). On suppose alors que les observations de la fonction f sont bruitées, i.e., qu'on connaît des couples $(x_j, f(x_j) + \varepsilon_j)$, puis on estime f par une méthode d'approximation adaptée à l'ensemble $\mathcal{F}$. Par exemple dans le cas Hilbertien, on cherche les coefficients γ_i qui minimisent

$$\mathcal{D}(\gamma_1, \ldots, \gamma_q) = \sum_{j=1}^{m} \left\| f(x_j) + \varepsilon_j - \sum_{k=1}^{q} \gamma_k \phi_k(x_j) \right\|^2$$

On peut ajouter à cette distorsion une pénalité sur les γ_k qui impose une plus forte régularité sur la représentation obtenue pour f. Cette approche permet de gagner sur différents points :

- la régularisation permet de s'affranchir en partie du bruit sur chaque fonction observée ;
- la manipulation des fonctions représentées ne pose plus de problème ;
- la représentation permet dans certains cas de réduire considérablement la taille des données, en particulier quand on dispose au départ d'observations à haute résolution de fonctions très régulières, ou encore quand des informations *a priori* permettent de proposer un modèle (paramétrique) adapté aux fonctions observées ;
- de plus, les opérations sur les fonctions représentées peuvent souvent être implantées de façon très efficace. Si on travaille par exemple par projection sur une base Hilbertienne, les produits scalaires et les calculs de normes sont de simples opérations linéaires ou quadratiques.

En résumé, les techniques d'approximation de fonctions sont au centre des méthodes d'ADF (les chapitres 3 et 4 de Ramsay et Silverman [63] sont consacrés à ce problème). En général, elles permettent la représentation des fonctions produites par l'ADF (par exemple les fonctions propres de l'analyse en composantes principales fonctionnelle). De plus, elles sont très utiles (et dans certains cas indispensables) pour représenter les fonctions observées. Notons que l'utilisation de fonctions régulières est dans certains cas indispensable pour donner un sens au résultat : pour le modèle linéaire à variables fonctionnelles par exemple (cf section 2.3.3), on obtient un problème mal posé si on n'introduit pas des

contraintes de régularisation. C'est le cas aussi pour l'analyse canonique fonctionnelle pour laquelle on obtient des résultats de consistance seulement en présence de régularisation (cf section 2.3.2).

Nous n'avons pas abordé un dernier point important dans la représentation des fonctions : le recalage. Dans certaines situations en effet, on peut ne pas avoir une confiance absolue en la localisation des points de discrétisation des fonctions observées. On peut aussi avoir de bonnes raisons pour transformer l'espace de départ des fonctions. L'exemple du *Graffiti* donné à la section 2.2 illustre parfaitement cette dernière motivation : il est illusoire de vouloir comparer directement deux dessins car la vitesse de tracé d'un caractère peut dépendre des conditions du tracé, sans pour autant que le dessin ne soit pas reconnaissable. On est donc amené dans certains circonstances à traiter des fonctions recalées, c'est-à-dire $f_i^*(t) = f_i(\tau_i(t))$. Nous renvoyons le lecteur au chapitre 5 de Ramsay et Silverman [63] pour la description de quelques techniques de recalage.

2.3.2 Les analyses factorielles

L'Analyse en Composantes Principales

Comme en analyse de données classique, un des premiers problèmes qu'on rencontre en ADF est celui de la visualisation des données. L'Analyse en Composantes Principales (ACP) qui permet une réduction de la dimension d'un ensemble de données (i.e., la réduction du nombre de variables) a de ce fait été une des premières méthodes classiques adaptée aux données fonctionnelles. Comme dans le cas classique, l'ACP fonctionnelle correspond à une représentation linéaire optimale (au sens des moindres carrés, i.e., de la variance) d'un ensemble de données fonctionnelles dans un espace de dimension finie. Si on dispose de n observations fonctionnelles dans L^2, $f_1, \ldots, f_n$, on cherche ainsi q fonctions de L^2, $\psi_1, \ldots, \psi_q$, orthogonales, et telles que la projection des f_i sur l'espace vectoriel engendré par les ψ_j engendre le moins de perte possible (au sens des moindres carrés), i.e., que la distorsion donnée par l'équation suivante soit minimale :

$$\mathcal{E}(\psi_1, \ldots, \psi_q) = \sum_{i=1}^{n} \left\| f_i - \sum_{j=1}^{q} \alpha_j(f_i)\psi_j \right\|_2^2 \tag{2.1}$$

Dans cette équation, $\|.\|_2$ désigne la norme de L^2 et $\alpha_j(f_i) = \langle f_i, \psi_j \rangle$ est le produit scalaire dans L^2 de f_i et ψ_j, c'est-à-dire la coordonnée selon ψ_j de la projection orthogonale de f_i sur l'espace engendré par les ψ_j.

Comme pour l'ACP classique, l'ACP fonctionnelle s'effectue en recherchant le spectre d'un opérateur compact. Cet opérateur est défini à partir de la fonction de covariance donnée par :

$$\nu(s,t) = \frac{1}{n}\sum_{i=1}^{n} \left(f_i(s) - \mu(s)\right)\left(f_i(t) - \mu(t)\right), \tag{2.2}$$

où μ désigne la fonction moyenne des f_i. Réaliser l'ACP des f_i revient à chercher les valeurs propres de l'opérateur défini par :

$$\Gamma(g) = \langle \nu(s,.), g \rangle$$

Les fonctions propres associées aux valeurs propres sont alors les ψ_j. On peut montrer que la fonction propre associée à la plus grand valeur propre, ψ_1, est solution du problème d'optimisation sous contrainte suivant :

$$\max_{\|\psi\|_2=1} \langle \Gamma(\psi), \psi \rangle \tag{2.3}$$

On trouvera aux chapitres 6 et 7 de Ramsay et Silverman [63] une présentation de techniques permettant la mise en œuvre pratique de l'ACP fonctionnelle. Les difficultés rencontrées sont celles évoquées dans la section 2.3.1, à savoir la représentation des fonctions observées et celle des fonctions propres. Deux stratégies sont utilisées en pratique :

1. Par discrétisation de toutes les fonctions : on approche alors les intégrales (les produits scalaires) par des techniques classiques de quadrature. Si les fonctions ne sont pas échantillonnées aux mêmes points, on commence par les interpoler.
2. Par représentation régulière : on cherche une représentation des fonctions propres sur les premiers éléments d'une base topologique. Pour simplifier les calculs, on commence en général par représenter les fonctions observées sur la même base.

Résultats théoriques

Du point de vue théorique, on suppose que les f_i (les fonctions observées) sont des réalisations d'une suite de variables aléatoires fonctionnelles (v.a.f.) i.i.d. $(F_i)_{i\in\mathbb{N}}$, et on remplace les moyennes empiriques par des espérances. Le principe reste cependant le même et conduit toujours à un problème de calcul du spectre d'un opérateur linéaire. La principale question théorique à résoudre est celle de la consistance : peut-on estimer les fonctions propres à partir d'un ensemble fini d'exemples fonctionnels sans commettre d'erreur systématique, i.e.,

de sorte que les fonctions obtenues convergent vers les fonctions théoriques ? Cette question est étudiée par exemple dans Deville [26], Dauxois et Pousse [23] et Dauxois et al. [24]. Notons que ces travaux généralisent le problème au cas des espaces de Hilbert séparables et non plus simplement à un de ces espaces, L^2. Deville [26] établit la consistance de l'estimation de l'opérateur de covariance dans le cas de fonctions parfaitement connues. Il en déduit la consistance (en moyenne quadratique) de l'estimation des valeurs et des fonctions propres de cet opérateur. Dauxois et Pousse [23] démontrent un résultat de consistance plus fort (convergence presque sûre uniforme) pour l'estimation de l'opérateur de covariance (et donc pour les valeurs propres et fonctions propres). Dauxois et al. [24] donnent les lois asymptotiques des éléments propres, alors que Bosq [11] donne des vitesses de convergence en se détachant de plus de l'hypothèse d'indépendance des observations fonctionnelles (ce qui permet de la modélisation auto-régressive fonctionnelle, par exemple, cf la section 2.3.3).

Le cadre fonctionnel complique la question de la consistance car il faut prendre en compte l'approximation des fonctions manipulées, à la fois pour les fonctions propres, mais aussi pour les fonctions observées (ces fonctions sont échantillonnées). Deville [26] montre qu'on peut approcher les fonctions observées et les fonctions propres par projection sur un sous-espace de dimension finie (l'article propose une représentation par fonctions polynomiales ou constante par morceaux). Si l'espace de projection est bien choisi, n'avoir que des fonctions échantillonnées ne pose pas de problème (on peut quand même calculer exactement les projetés). Dauxois et Pousse [23] proposent aussi des résultats de convergence dans le cas où les fonctions sont discrétisées (toutes au même endroit). Dans les deux cas, les deux parties de l'approximation sont séparées : soit on suppose que les fonctions sont en nombre fini mais parfaitement connues, soit on suppose que les fonctions sont échantillonnées, mais qu'on peut calculer exactement l'opérateur de covariance (i.e., qu'on connaît "toutes" les fonctions). On trouve dans Besse [4] un résultat de consistance en présence d'observations discrétisées, avec approximation par splines des fonctions observées et des fonctions propres (l'échantillonnage est uniforme). En fait, les résultats proposés sont très généraux car ils correspondent à de l'approximation spline dans des espaces de Hilbert généraux au lieu de se limiter à L^2. Plus récemment, Cardot [15] donne des vitesses de convergence pour les estimations quand les fonctions observées et les fonctions propres sont représentées sur une base de B-splines (ce qui permet d'avoir un échantillonnage différent pour chaque courbe). Les vitesses sont données pour le cas d'un échantillonnage identique pour chaque courbe, mais la consistance reste acquise dans le cas d'échantillonnages distincts. Notons que la discrétisation est supposée déterministe.

Extensions

Quand on réalise une ACP en optimisant la distortion donnée par l'équation 2.1, les fonctions propres obtenues n'ont aucun raison d'être particulièrement régulières. L'optimisation étant conduite dans L^2, on peut obtenir des fonctions propres très irrégulières, ce qui ne facilite pas l'interprétation des résultats. C'est pourquoi certains auteurs ont proposé de rechercher des fonctions propres régulières. Par exemple, Besse et Ramsay [7] représentent les fonctions propres par des splines. D'autres approches sont possibles, comme l'intégration d'un terme de pénalisation dans le problème d'optimisation (2.3) (ou dans le critère 2.1). Si J est un critère de pénalisation, comme par exemple $J(\psi) = \|\psi''\|_2^2$ (ce qui impose de chercher des fonctions propres de classe C^2), Rice et Silverman [65] proposent de remplacer 2.3 par le problème :

$$\max_{\|\psi\|_2=1} \langle \Gamma(\psi), \psi \rangle - \rho J(\psi), \tag{2.4}$$

Pezzulli et Silverman [59] démontrent la consistance des estimateurs obtenus par cette méthode (en cas de connaissance parfaite des fonctions). Silverman [72] remplace 2.3 par :

$$\max_{\|\psi\|_2+\rho J(\psi)=1} \langle \Gamma(\psi), \psi \rangle \tag{2.5}$$

et démontre la convergence des estimateurs lissés en cas de connaissance parfaite des fonctions. On peut montrer (cf Pezzulli et Silverman [59], Silverman [72] et Cardot [15]) que le lissage améliore l'estimation des fonctions propres (en cas de connaissance parfaite des fonctions).

Dans le cas de fonctions échantillonnées de façon irrégulière, Besse et al. [5] proposent une estimation lisse basée sur des splines hybrides, qu'on peut voir comme une régularisation de la représentation des fonctions observées et des fonctions propres sur une base de B-spline. Cette technique est assez subtile puisqu'elle consiste à d'abord représenter les fonctions observées sur une base de B-spline, puis à remplacer chaque fonction par une version plus lisse de sorte que l'ensemble de ces nouvelles fonctions vérifie une contrainte de rang (c'est-à-dire de dimension de l'espace vectoriel auquel ces fonctions appartiennent). Cardot [15] montre la consistance de cette méthode, donne des vitesses de convergence et montre que le lissage améliore l'estimation des fonctions propres.

D'autres variantes et extensions de l'ACP fonctionnelle ont été présentées et étudiées, comme par exemple Silverman [71] qui propose de traiter conjointement le problème du recalage des courbes étudiées et de l'ACP, ou encore Rice et Wu [66] et James et al. [48] qui proposent un modèle linéaire mixte dont les éléments sont estimés par un algorithme de type EM. Ces modèles mixtes permettent de traiter des fonctions discrétisées de façon différente pour chaque

fonction sans pour autant passer par une étape de représentation régulière, ce qui améliore le traitement des données dont les observations sont très irrégulièrement réparties, au prix d'un algorithme d'estimation plus lourd que dans les autres méthodes.

Autres méthodes factorielles

D'autres méthodes liées à l'ACP ont été étendues au cas fonctionnel. On peut citer par exemple l'analyse canonique, proposée dans le cadre hilbertien dans Dauxois et Pousse [23] et plus spécifiquement dans le cadre fonctionnel dans Leurgans et al. [52]. L'analyse canonique d'un couple de variables aléatoires fonctionnelles consiste à trouver une transformation linéaire de ces variables qui maximise la corrélation des fonctions transformées. Il est intéressant de noter que pour obtenir la consistance des estimateurs, on doit introduire une forme de régularisation qui n'a donc plus seulement un rôle d'amélioration des résultats (cf Leurgans et al. [52]). En fait, en l'absence de régularisation, Leurgans et al. [52] montrent qu'il existe toujours une transformation linéaire de deux variables fonctionnelles telle que les transformées ont une corrélation de 1, ce qui signifie clairement que la transformation en question n'a pas beaucoup d'intérêt. C'est pourquoi d'ailleurs Dauxois et Pousse [23] font des hypothèses restrictives sur les variables aléatoires hilbertiennes pour lesquelles ils proposent une extension de l'analyse canonique.

L'analyse factorielle discriminante étant un cas particulier de l'analyse canonique, elle s'applique naturellement (et avec le même besoin de régularisation) aux données fonctionnelles. On trouve un exemple de mise en œuvre dans Hastie et al. [38]. Dans le cas de fonctions très irrégulièrement discrétisées, James et Hastie [47] adaptent leurs travaux sur l'ACP (James et al. [48]) à l'analyse discriminante en proposant de nouveau un modèle linéaire mixte estimé par l'algorithme EM.

2.3.3 Les modèles linéaires fonctionnels

Introduction

Les méthodes factorielles sont très utiles pour visualiser, et plus généralement pour explorer, des données. Cependant, dans certaines applications, l'exploration n'est pas l'aspect le plus important : on cherche plutôt à modéliser une relation entre certains groupes de variables utilisées pour décrire les individus étudiés. Plus précisément, on cherche à expliquer une variable ou plusieurs variables grâce aux valeurs d'autres variables. Cette formulation générale recouvre des techniques de type régression où on cherche à prédire les valeurs des

variables à expliquer grâce aux variables explicatives et des techniques de type analyse de la variance où on cherche à décomposer les variables à expliquer sous forme d'une combinaison d'effets, chaque effet étant caractéristique d'un groupe d'individus (l'appartenance d'un individu à un groupe est une variable explicative nominale). La façon la plus simple d'introduire un lien entre différentes variables est de faire une hypothèse de linéarité. Dans la régression linéaire, les variables à expliquer sont obtenues comme combinaisons linéaires des variables explicatives, alors qu'en analyse de la variance, la combinaison des effets est une superposition, c'est-à-dire une somme.

L'extension des modèles linéaires au cadre fonctionnel peut se faire dans différentes directions, selon qu'on considère des variables explicatives fonctionnelles, une réponse fonctionnelle ou encore les deux. Nous allons présenter brièvement dans la suite de cette section différents modèles linéaires utilisables dans le cadre fonctionnel.

Analyse de la variance fonctionnelle

Il s'agit de généraliser l'analyse de la variance au cas d'une variable fonctionnelle. Dans Ramsay et Silverman [63] (chapitre 9), les auteurs donnent l'exemple de la décomposition de la courbe de température annuelle observée dans une station météo sous la forme suivante :

$$\mathrm{Temp}_{kg}(t) = \mu(t) + \alpha_g(t) + \varepsilon_{kg}(t)$$

Dans cette formule, g désigne le groupe de la station météo et k le numéro d'ordre de cette station dans le groupe. La fonction $\mu(t)$ est la courbe moyenne de température sur l'ensemble des stations, $\alpha_g(t)$ est l'effet du groupe g et $\varepsilon_{kg}(t)$ est le résidu non expliqué par l'appartenance de la station k au groupe g.

Quelques résultats théoriques ont été obtenus. Par exemple Fan et Lin [27] établissent les propriétés théoriques d'un test de différence entre des groupes de fonctions, quand les fonctions sont toutes observées aux mêmes points. Fan et Zhang [28] proposent une méthode d'estimation des effets dans l'analyse de la variance fonctionnelle basée sur un calcul direct suivi d'un lissage des fonctions obtenues, et donnent des résultats sur le biais asymptotique des estimateurs dans certains cas particuliers. Comme dans Brumback et Rice [12], l'analyse proposée est basée sur le modèle linéaire à coefficients variables, présenté à la section 2.3.3. Notons de plus que, comme on peut voir l'analyse de la variance fonctionnelle comme un cas particulier du modèle linéaire fonctionnel, les résultats obtenus pour le cas général s'appliquent au cas particulier.

Modèle linéaire avec réponse scalaire

L'extension la plus naturelle du modèle linéaire au cas fonctionnel est celle qui consiste à considérer une variable réelle à expliquer grâce à une variable fonctionnelle. En effet, le modèle linéaire dans $\mathbb{R}^n$ se traduit naturellement en modèle linéaire sur un espace fonctionnel. Plus formellement, on écrit l'espérance conditionnelle de Y (la variable à prédire, supposée centrée) sachant X (la fonction observée) sous la forme suivante (Ramsay et Silverman [63], chapitre 10) :

$$E(Y|X) = \int \alpha(t)X(t)dt \tag{2.6}$$

On peut généraliser au cas de X à valeurs dans un espace hilbertien séparable (Cardot et al. [17]) :

$$E(Y|X) = A(X), \tag{2.7}$$

où A désigne un opérateur linéaire continu sur l'espace hilbertien considéré. Dans les deux cas, le problème est donc d'estimer l'opérateur linéaire. Du point de vue pratique, cela pose des problèmes, même quand les fonctions sont toutes observées aux mêmes points. Comme le rappellent Frank et Friedman [36], estimer un modèle linéaire demande l'inversion de l'opérateur de covariance. Or, dans le cadre fonctionnel discrétisé, cet opérateur est souvent très mal conditionné car la régularité des fonctions observées induit une quasi-colinéarité des observations. De ce fait, on est amené à utiliser diverses techniques pour améliorer le conditionnement de l'opérateur de covariance, par exemple en travaillant dans le sous-espace vectoriel engendré par les plus grandes valeurs propres de l'opérateur ou encore en introduisant un terme de régularisation. Dans cette dernière approche, on cherche α qui minimise

$$E\left(\left(Y - \int \alpha(t)X(t)dt\right)^2\right) + \rho J(\alpha),$$

où J est un critère de pénalisation qui favorise les fonctions régulières. Pour mettre en œuvre cette technique, on peut utiliser des splines d'interpolation (si les fonctions sont discrétisées toutes aux mêmes points), comme dans Hastie et Mallows [39]. On peut aussi chercher une représentation de α sur une base de B-splines (Ramsay et Silverman [63], chapitre 10) ou combiner la recherche sur une base de B-splines avec l'utilisation d'un terme de régularisation (splines hybrides, ou P-splines, Marx et Eilers [55] et Cardot et al. [18]).

Dans le cas où les fonctions sont parfaitement connues, le problème reste le même. Comme l'expliquent Cardot et al. [17], l'opérateur de covariance n'est pas inversible en général, et quand l'inverse existe, il n'est pas continu. La technique

proposée dans Cardot et al. [17] consiste à estimer l'opérateur A grâce à une suite d'opérateurs obtenus en restreignant l'opérateur de covariance à l'espace engendré par ses premiers vecteurs propres. La subtilité est que le nombre de vecteurs propres pris en compte croit avec le nombre de réalisations fonctionnelles considérées. Sous certaines hypothèses sur les valeurs propres de l'opérateur de covariance et sur la vitesse de croissance de la dimension de l'espace de projection, Cardot et al. [17] donnent des résultats de convergence en probabilité et presque sûre.

Comme nous l'avons dit plus haut, une autre solution pour estimer le modèle linéaire consiste à représenter α sur une base de B-splines, en incluant un terme de pénalisation. Cardot et al. [18] proposent une analyse théorique de cette technique, en donnant en particulier des vitesses de convergence pour la méthode, à la fois dans le cas où les fonctions sont parfaitement connues et dans celui d'une discrétisation (supposée non aléatoire). Comme dans la méthode précédente, la convergence de l'estimateur de α demande une augmentation progressive de la précision de la représentation, c'est-à-dire du nombre de B-splines considérés.

Pour conclure sur le modèle linéaire à réponse scalaire, on peut noter que Cardot et al. [16] proposent un test d'hypothèses (de la forme $\Phi = \Phi_0$ contre $\Phi \neq \Phi_0$) pour le modèle linéaire fonctionnel et établissent sa consistance. D'autre part, James [46] propose une généralisation au cas fonctionnel du modèle linéaire généralisé, en utilisant une représentation régularisée des fonctions observées et en estimant les paramètres du modèle par un algorithme de type EM.

Modèle linéaire avec réponse fonctionnelle

On peut aller plus loin dans la généralisation du modèle linéaire en supposant que la variable à expliquer est elle aussi une fonction. Si les variables explicatives sont classiques, on se retrouve avec un modèle de type analyse de la variance fonctionnelle, présenté à la section 2.3.3. Le cas le plus général est celui où les variables explicatives sont elles aussi fonctionnelles. Il s'agit donc dans le cas général de trouver un modèle de $E(Y|X)$ où Y est une fonction à valeurs réelles et X une fonction à valeurs réelles ou vectorielles.

Modèle linéaire à coefficients variables

Hoover et al. [41] proposent une généralisation fonctionnelle des modèles linéaires à coefficients variables introduits dans Hastie et Tibshirani [40]. L'idée est modéliser Y par :

$$Y(t) = X^T(t)\beta(t) + \varepsilon(t), \tag{2.8}$$

où $X^T(t)$ est le vecteur ligne obtenu par transposition de la valeur vectorielle de la fonction X au point t. On cherche alors à estimer la fonction vectorielle

β. En tout t, β est donnée théoriquement par la formule suivante (si on utilise un critère de moindres carrés) :

$$\beta(t) = \left(E(X(t)X^T(t))\right)^{-1} E(X(t)Y(t))$$

Différentes méthodes sont proposées et étudiées dans Hoover et al. [41]. Une première méthode consiste à représenter β dans une base de B-splines en intégrant une pénalité pour obtenir une solution régulière. Une autre méthode consiste à interpoler X et Y grâce à une méthode d'estimateur à noyaux. Hoover et al. [41] donnent des résultats de vitesse de convergence des estimateurs dans le cas de cette deuxième méthode.

Fan et Zhang [28] proposent une méthode d'estimation de β (déjà évoquée à la section 2.3.3) basée sur un mécanisme à deux étapes : dans un premier temps, on obtient une estimation discrétisée de β (les points de discrétisation sont ceux pour lesquels on connaît les courbes étudiées). Ensuite, l'estimation est lissée, par exemple grâce à un lissage localement polynomial. L'avantage de cette technique sur celles proposées dans Hoover et al. [41] est son temps de calcul en général très inférieur. Fan et Zhang [28] réalisent une étude du biais et de la variance des estimateurs proposés.

Modèle intégral

Ramsay et Silverman [63] proposent (chapitre 11) une autre approche du modèle de régression avec réponse fonctionnelle, en écrivant $E(Y|X)$ sous la forme suivante (dans le cas centré) :

$$E(Y|X)(s) = \int \beta(s,t)X(t)dt \tag{2.9}$$

Dans cette équation, X désigne une variable aléatoire fonctionnelle dont les valeurs sont des fonctions à valeurs réelles. La généralisation au cas de plusieurs variables fonctionnelles ne pose pas de problème. Pour estimer β, Ramsay et Silverman [63] représentent $X(t)$ grâce à une base (les ϕ_j) et $Y(s)$ grâce à une autre base (les ψ_i). On représente alors β sur la base des produits $\psi_i(s)\phi_j(t)$. Comme souvent en ADF, il est possible (et même recommandé) d'ajouter un terme de pénalisation.

Modèle auto-régressif

Dans l'étude des séries temporelles, il est assez naturel de considérer le modèle fonctionnel auto-régressif et même le modèle auto-régressif hilbertien. On suppose ainsi qu'on observe une suite (h_i) d'éléments d'un espace hilbertien séparable qu'on modélise par (dans le cas d'un processus d'ordre 1 centré) :

$$h_{i+1} = A(h_i) + \varepsilon_i \tag{2.10}$$

On cherche alors à estimer l'opérateur linéaire continu A. Bosq [11] propose une stratégie d'estimation basée sur une projection des fonctions observées sur l'espace vectoriel engendré par les plus grandes valeurs propres de l'opérateur de covariance (on retient un nombre fini de valeurs propres). On procède ensuite par inversion de la restriction de l'opérateur à l'espace considéré, comme dans le cas du modèle linéaire à réponse scalaire (cf la section 2.3.3). Bosq [11] donne des résultats de convergence des estimateurs proposés.

Pumo [62] propose une interpolation linéaire des observations discrétisées et montre la convergence des estimateurs obtenus. Besse et Cardot [8] proposent une amélioration de la méthode basée sur une représentation par splines de lissage des fonctions observées à laquelle on ajoute une contrainte de rang. La méthode combine deux avantages : elle permet une meilleure représentation des fonctions observées, tout en choisissant les représentants dans un sous-espace fonctionnel de dimension finie, ce qui permet d'anticiper la restriction de l'opérateur de covariance nécessaire à son inversion. L'utilisation de splines hybrides à la place des splines de lissage (comme dans Besse et al. [5]) permet de plus de prendre en compte une discrétisation différente pour chaque courbe. D'autres techniques sont évoquées par Besse et Cardot [8], comme par exemple un représentation des fonctions observées par des splines de lissage sans introduction de contrainte de rang, ou encore une recherche de fonctions propres lisses de l'opérateur de covariance. Les expériences conduites dans Besse et Cardot [8] semblent montrer que l'approche proposée par l'article donne des résultats plus satisfaisant que les autres. Cardot [14] étudie d'un point de vue théorique la technique qui consiste à représenter les fonctions par des splines de lissage (sans contrainte de rang et en considérant une même discrétisation pour toute les fonctions observées) et donne des vitesses de convergence pour les estimateurs ainsi obtenus.

2.3.4 Autres méthodes

Les méthodes présentées jusqu'à présent sont essentiellement des méthodes linéaires, ce qui est relativement restrictif. Certaines méthodes non linéaires ont cependant été adaptées au cas fonctionnel, comme nous l'indiquons dans les sections suivantes.

Régression inverse par tranche

Ferré et Yao [29][22] proposent une généralisation du modèle de régression inverse par tranche qui est un modèle semi-paramétrique non linéaire. Le principe du SIR (Sliced Inverse Regression, Li [53]) est de chercher Y (la variable à

expliquer) comme une fonction de X (la variable explicative à valeurs dans $\mathbb{R}^p$) sous la forme :

$$Y = f(\beta_1^T X, \ldots, \beta_k^T X, \varepsilon) \tag{2.11}$$

Dans cette équation, les β_i sont k vecteurs linéairement indépendants de $\mathbb{R}^p$, ε est un bruit centré (à valeurs réelles) indépendant de X et f une fonction de $\mathbb{R}^{k+1}$ dans $\mathbb{R}$. La généralisation au cas hilbertien est très naturelle et consiste à chercher Y sous la forme

$$Y = f(\langle \beta_1, X \rangle, \ldots, \langle \beta_k, X \rangle, \varepsilon) \tag{2.12}$$

Dans cette nouvelle équation, $\langle ., . \rangle$ désigne le produit scalaire d'un espace hilbertien séparable et X est une variable aléatoire à valeurs dans cet espace.

La principale difficulté du SIR est l'estimation de k et des β_i, car f est obtenue par une méthode de régression non paramétrique classique. Quand k est fixé, on obtient les β_i en effectuant l'analyse spectrale de l'opérateur $\Gamma^{-1}\Gamma_e$, où Γ désigne comme toujours l'opérateur de covariance de X, alors que Γ_e est l'opérateur de covariance de $E(X|Y)$. Le problème est que, comme nous l'avons déjà indiqué, Γ^{-1} n'est pas un opérateur borné, ce qui complique singulièrement l'estimation. La technique proposée dans **?**] est celle dont nous avons déjà régulièrement parlé, à savoir l'inversion de la restriction de Γ à l'espace vectoriel engendré par les vecteurs propres associés à ses plus grandes valeurs propres. **?**] montrent la consistance des estimateurs obtenus par cette méthode, en présence de fonctions parfaitement connues.

Régression non paramétrique

Ferraty et Vieu [33] proposent une généralisation du modèle de régression non paramétrique par noyau au cas des espaces vectoriels semi-normés. Si on observe n réalisations du couple de variables aléatoires (Y, X) (à valeur dans $\mathbb{R} \times \mathcal{H}$, les (Y_i, X_i), on définit l'estimateur $E(Y|X = x) = \Phi(x)$ par

$$\widehat{\Phi}_n(x) = \sum_{i=1}^{n} w_i(x) Y_i, \tag{2.13}$$

avec

$$w_i(x) = \frac{K\left(\frac{\|X_i - x\|}{h_n}\right)}{\sum_{j=1}^{n} K\left(\frac{\|X_j - x\|}{h_n}\right)} \tag{2.14}$$

Dans cette équation, $\|.\|$ désigne la semi-norme de l'espace vectoriel $\mathcal{H}$, K est un noyau, c'est-à-dire une fonction de $\mathbb{R}^+$ dans $\mathbb{R}^+$, Lipschitzienne de rapport 1,

à support compact et d'intégrale 1 sur son support. Enfin, h_n est la largeur de la fenêtre de l'estimateur. On constate que l'estimateur proposé est l'extension très naturelle de l'estimateur de Nadaraya-Watson au cas où la distance entre une observation et un x quelconque est calculée à partir d'une semi-norme.

Ferraty et Vieu [33] établissent la consistance de l'estimateur $\widehat{\Phi}_n$ sous des hypothèses assez classiques (continuité de Φ, hypothèses techniques sur h_n, observations i.i.d., Y bornée), excepté une hypothèse sur la variable aléatoire X qui peut s'interpréter comme une condition sur la dimension fractale de P_X. Avec des hypothèses un peu plus fortes (Φ Hölderienne ainsi qu'une hypothèse plus forte sur P_X), Ferraty et Vieu [33] obtiennent la vitesse de convergence de $\widehat{\Phi}_n$ vers Φ. On trouve dans Ferraty et Vieu [34] (chapitre 5) et dans Ferraty et al. [32] des résultats plus complets, en particulier la convergence uniforme sur un compact ainsi que le cas où la suite de réalisations $(Y_i, X_i)_{i\in\mathbb{N}}$ n'est plus i.i.d. mais α-mélangeante. Dans tous les cas, les résultats sont obtenus en supposant les fonctions observées parfaitement connues. On trouve dans Ferraty [31] une application de la méthode proposée au cas de la discrimination de courbes. L'article s'intéresse en particulier au choix pratique de la semi-norme sur $\mathcal{H}$ et propose d'utiliser la norme de la projection de $X_i - x$ sur l'espace vectoriel engendré par les q premiers facteurs fonctionnels de l'ACP des X_i.

Besse et al. [6] adaptent le modèle décrit dans cette section pour réaliser une estimation non paramétrique d'un processus auto-régressif fonctionnel. L'adaptation ne pose pas de problème conceptuel car l'estimateur $\widehat{\Phi}_n$ donné par l'équation 2.13 est parfaitement bien défini si Y est à valeurs dans un espace vectoriel quelconque et donc en particulier dans L^2. Du point de vue pratique, la discrétisation des fonctions introduit une difficulté contournée par l'utilisation d'une représentation régulière des fonctions considérées.

Segmentation

Dans les méthodes non paramétriques on peut citer la segmentation par arbre de décision qui consiste à construire un arbre dont les décisions sont effectuées sur des variables explicatives, avec comme but d'obtenir des feuilles homogènes pour la variable à expliquer. Michel-Briand et Escouffier [56] proposent une adaptation des algorithmes classiques de segmentation au cas où la variable à expliquer est fonctionnelle (les variables explicatives restent numériques). Le principe de l'adaptation consiste à utiliser une semi-norme sur l'espace fonctionnel considéré afin de juger de l'homogénéité d'une feuille. Yu et Lambert [77] travaillent sur le même type de modèle en choisissant de comparer les fonctions par l'intermédiaire de leurs coordonnées dans une base de splines ou leurs cordonnées factorielles.

Classification

Une des méthodes les plus classiques pour la classification non supervisée est celle des k-moyennes. Abraham et al. [1] proposent une adaptation de cet algorithme au cas des données fonctionnelles, plus précisément des fonctions de $L^2[a, b]$. La solution proposée consiste à représenter chaque fonction observée sur une base de B-splines, puis d'effectuer une classification de type k-moyennes sur les coordonnées des courbes dans la base. Abraham et al. [1] montrent la consistance des estimateurs ainsi construit, en tenant compte d'une discrétisation différente pour chaque fonction. De plus, la discrétisation est aléatoire, ce qui complique singulièrement le problème.

Une autre approche est proposée dans James et Sugar [49]. L'article propose une généralisation du modèle de mélange (gaussien) au cas fonctionnel. En fait, James et Sugar [49] modélisent la fonction aléatoire observée comme un vecteur aléatoire de dimension finie par l'intermédiaire d'une base de B-splines. Le vecteur aléatoire est supposé engendré par un mélange gaussien. Les paramètres du modèle sont estimés par maximum de vraisemblance avec un algorithme de type EM. Un aspect original du travail est que les fonctions observées ne sont pas projetées sur la base de B-spline. En fait, on modélise les observations sous la forme $y_{ij} = f_i(t_j) + \varepsilon_{ij}$ (où f_i désigne une réalisation de la variable aléatoire fonctionnelle étudiée), ce qui permet de calculer directement la vraisemblance des observations d'origine et pas celle des coordonnées dans la base de B-splines.

2.4 Conclusion

Comme nous venons de le voir, l'Analyse de Données Fonctionnelles propose un cadre général permettant le traitement d'une classe très large de données pour lesquelles les méthodes classiques ne sont pas adaptées. De nombreuses méthodes de l'Analyse de Données ont été adaptées avec succès aux données fonctionnelles, tant du point de vue théorique que pratique. On note cependant, et c'est toute la motivation de cette thèse, que les méthodes neuronales semblent absentes du panorama fonctionnel. Or, comme nous le rappellerons dans les chapitres suivants, le Perceptron Multi-Couches est un modèle semi-paramétrique très intéressant dans la pratique, essentiellement grâce à son aspect parcimonieux : il peut fournir un modèle fidèle d'un ensemble de données en utilisant peu de paramètres. Il offre en ce sens un alternative intéressante aux modèles non paramétrique et aux modèles semi-paramétriques pseudo-linéaires. Nous allons nous attacher dans les chapitres suivant à montrer qu'on peut étendre le Perceptron Multi-Couches de manière à lui permettre de traiter des données fonctionnelles, en nous intéressant aux propriétés théoriques du modèle proposé

et en illustrant son fonctionnement sur des données simulées.

Chapitre 3

Le perceptron multi-couches fonctionnel

3.1 Introduction

3.1.1 Les différents modèles

L'approximation de fonctions joue un rôle fondamental dans les problèmes de modélisation statistique tels que les problèmes de régression ou de discrimination. Dans le premier cas, on cherche en effet à approcher la fonction de régression afin d'obtenir une relation déterministe entre les variables explicatives et les variables à prédire. Dans le cas de la discrimination, le calcul des probabilités *a posteriori* peut s'exprimer sous la forme d'un problème de régression classique, et peut donc être ramené au problème de l'approximation de la fonction de régression.

Dans les deux cas, le choix du modèle (paramétrique, semi-paramétrique, non paramétrique) dépend entièrement de la nature du problème à traiter. L'utilisation de modèles paramétriques suppose une bonne compréhension du phénomène à modéliser, ce qui restreint leurs champs d'application à des problèmes spécifiques. A contrario, l'utilisation de modèles non-paramétriques, tels que les K plus proches voisins ou les méthodes à noyaux, ne nécessite pas de connaissance *a priori*. Cependant, ce type d'approches présente dans la pratique un inconvénient majeur : ces méthodes sont peu adaptées aux problèmes nécessitant un important volume de données, car elles imposent le stockage et le traitement de l'ensemble des données lors de chaque évaluation du modèle. Finalement, les modèles semi-paramétriques présentent les avantages des deux précédentes approches sans leurs inconvénients : la classe de fonctions réalisables par ces modèles est très large, ce qui autorise leur utilisation dans de nombreux problèmes,

de plus contrairement aux méthodes non-paramétriques, seule la phase d'estimation (apprentissage) nécessite la disponibilité de l'ensemble des données. Par la suite, l'évaluation du modèle est indépendante des données initiales[1].

Dans le cas semi-paramétrique, on pourrait être tenté d'utiliser toute famille assez "large" de fonctions paramétrées pour résoudre un problème de régression (ou de discrimination). Cependant, sans connaissance *a priori* sur le phénomène à modéliser, on ne peut pas faire d'hypothèses trop restrictives sur la nature de la fonction à approcher. Ceci conduit alors à ne considérer que des modèles ayant la capacité d'approcher une classe relativement importante de fonctions[2]. Cette propriété d'approximation universelle est fondamentale et justifie d'un point de vue théorique le choix de certains modèles de préférence à d'autres.

De nombreuses familles de fonctions paramétrées possèdent la propriété d'approximation universelle (dont le perceptron multi-couches comme on pourra le voir dans le chapitre suivant). Le choix d'un modèle plutôt qu'un autre semble donc être de peu d'importance, et pourrait être motivé par des considérations purement pragmatiques : la réduction des ressources informatiques nécessaires à l'obtention et à la manipulation du modèle. En suivant ce raisonnement, il semble légitime de s'interroger sur l'utilité des modèles non-linéaires, tels que les perceptrons multi-couches en regard des modèles linéaires généralisés. En effet, dans la pratique, l'étape d'estimation des paramètres d'un modèle non-linéaire est une tâche difficile et coûteuse : l'optimisation non-linéaire implique l'utilisation d'algorithmes nécessitant des ressources importantes.

En fait, la seule propriété d'approximation universelle n'est pas suffisante pour justifier le choix d'une classe de régresseurs paramétriques : le problème de la complexité du modèle doit aussi être pris en considération. On sait en effet que le nombre de paramètres ajustables d'un modèle doit toujours être petit devant la taille de l'ensemble d'apprentissage. Dans le cas contraire, le modèle aura tendance à sur-apprendre la base d'exemples (overfitting), et n'aura donc pas la capacité de généraliser lors de la présentation de nouveaux individus. C'est la raison pour laquelle le choix de modèles parcimonieux (i.e. réalisant une même qualité d'approximation avec un nombre réduit de paramètres) est dans la pratique préféré.

Un résultat important, démontré par Barron [3], montre que les perceptrons multi-couches sont des approximateurs parcimonieux. Plus précisément, ce résultat montre que pour des problèmes dont la dimension d'entrée est supérieure ou égale à 3, les perceptrons multi-couches réalisent une approximation plus fine

[1] les paramètres du modèles sont une sorte de "résumé statistique" de l'ensemble des données.

[2] par exemple, un ensemble des fonctions régulières ou un ensemble de fonctions intégrables ($L^2(\mu)$,...).

que les modèles linéaires généralisés à nombre de paramètres égal. Ce résultat est la justification théorique de l'utilisation des perceptrons multi-couches pour des problèmes en dimension élevée.

3.1.2 Le cadre fonctionnel

Comme on a pu le voir dans le chapitre précédent, l'Analyse de Données Fonctionnelles est un domaine récent de la statistique qui a fait l'objet de nombreux travaux, tant pratiques que théoriques. En effet, comme le souligne [30], l'extension au cadre fonctionnel des techniques classiques d'estimation, tels que les modèles à noyaux, ou le modèle linéaire, a permis de développer des outils théoriques performants qui offrent un potentiel important en terme d'applications (voir le chapitre 2 pour quelques exemples). Le travail présenté ici s'inscrit pleinement dans ce courant, et tente de faire la jonction entre le domaine de la statistique sur données fonctionnelles, et celui des techniques "neuronales" classiques.

La nécessité d'utiliser des modèles parcimonieux dans tout problème d'estimation suggère naturellement l'adaptation du perceptron multi-couches au cadre fonctionnel. En effet, comme on a pu le voir précédemment, le perceptron multi-couches réalise une approximation d'autant plus parcimonieuse par rapport aux modèles linéaires généralisés que la dimension d'entrée est élevée. Le postulat sur lequel s'appuie l'ensemble de ce travail est que cette propriété reste valable dans le cas d'espaces de dimension infinie, et plus particulièrement dans le cas des espaces fonctionnels.

L'extension du perceptron multi-couches numérique à des espaces topologiques arbitraires, présentée dans la première partie de ce chapitre, s'appuie sur les travaux successifs de Sandberg [69], Sandberg et Xu [70], Chen [19], Stinchcombe [75]. L'idée commune à tous ces travaux consiste à modifier la définition du neurone numérique classique, en remplaçant la partie linéaire du neurone (le produit scalaire entre le vecteur d'entrée et le vecteur de poids) par une fonction arbitraire. Cette fonction, définie sur l'espace d'entrée du neurone et à valeurs réelles, est appelée "*la fonction de poids*".

Cette définition générale peut aisément être adaptée au cas particulier des espaces L^p (l'espace des fonctions mesurables de puissance p intégrable), qui offrent un cadre naturel permettant le traitement de données fonctionnelles. L'avantage majeur des espaces L^p par rapport à des espaces plus généraux est qu'ils permettent une caractérisation simple du neurone fonctionnel. Cette caractérisation impose cependant des hypothèses supplémentaires sur la nature des fonctions de poids : ces fonctions doivent être linéaires et continues.

Grâce à cette définition du neurone fonctionnel, l'extension du percep-

tron multi-couches au cadre fonctionnel s'effectue naturellement. En effet, la construction d'un perceptron multi-couches fonctionnel est réalisée de manière similaire à celle d'un perceptron numérique classique. On combine pour cela des neurones fonctionnels et des neurones numériques selon une organisation par couches.

Pour permettre la mise en œuvre pratique des perceptrons multi-couches fonctionnels, on remplace les fonctions de poids arbitraires par une représentation paramétrique. On obtient ainsi un perceptron multi-couches fonctionnel paramétré par un nombre fini de paramètres réels. On montre alors que l'on peut appliquer les algorithmes classiques d'optimisation (basés sur le calcul du gradient de l'erreur commise par le réseau) grâce à une adaptation de l'algorithme de rétro-propagation. On montre de plus qu'une extension de ce dernier algorithme permet d'accélérer certains calculs intermédiaires.

3.2 Neurone Fonctionnel

3.2.1 Neurone général

L'extension du neurone numérique à des espaces arbitraires a été proposée dans les travaux de Sandberg [69], Sandberg et Xu [70], Chen [19], [75]. Cette extension s'effectue de manière naturelle comme on va le voir dans la suite du texte.

Le neurone numérique, dont les entrées appartiennent à $\mathbb{R}^n$, est caractérisé par une fonction d'activation T (une fonction de $\mathbb{R}$ dans $\mathbb{R}$), un vecteur de poids $w \in \mathbb{R}^n$, et par un biais $b \in \mathbb{R}$. Si l'on considère le vecteur d'entrée $x \in \mathbb{R}^n$, le neurone numérique calcule alors la fonction suivante :

$$N(x) = T(w.x + b)$$

De manière similaire à Sandberg [69], Sandberg et Xu [70], Chen [19], Stinchcombe [75], on remplace dans la définition du neurone numérique le produit scalaire $w.x$ par une fonction arbitraire. Plus précisément, on propose la définition suivante :

Définition 1. Soit E est un espace arbitraire, et M un ensemble de fonctions de E dans $\mathbb{R}$.

Un neurone général défini sur E est caractérisé par une fonction d'activation T (de $\mathbb{R}$ dans $\mathbb{R}$), une fonction $\omega \in M$ (*la fonction de poids*) et par un biais $b \in \mathbb{R}$. Etant donnée une entrée $x \in E$, le neurone général calcule alors la fonction suivante :

$$N(x) = T(\omega(x) + b) \tag{3.1}$$

3.2.2 Neurone dans un espace de Banach

La manipulation des éléments de M (les fonctions de poids) n'étant pas aisée, une première restriction dans la définition du neurone général est nécessaire afin de permettre sa mise en œuvre pratique : comme pour un neurone numérique, on impose à la fonction de poids d'être linéaire et continue. Plus précisément, comme Sandberg et Xu [70], on impose dans la définition 1 à E d'être un espace de Banach (espace vectoriel normé complet) et à M d'être un sous-ensemble du dual topologique de E, E^* (l'espace des formes linéaires continues de E dans $\mathbb{R}$).

On voit que cette nouvelle définition (comme la première) englobe parfaitement la définition du neurone numérique classique. En effet, si l'on considère le cas particulier où $E = \mathbb{R}^n$, alors tout élément ω du dual topologique de $\mathbb{R}^n$ s'écrit sous la forme suivante : $\omega(x) = w.x$, où $w \in \mathbb{R}^n$ et $x \in \mathbb{R}^n$.

3.2.3 Neurone dans les espaces L^p

On a vu dans la section précédente que la définition du neurone numérique pouvait être étendue à des espaces de Banach arbitraires. Parmi ceux-ci, les espaces L^p (l'ensemble des fonctions[3] mesurables de puissance p intégrable) offrent un cadre naturel permettant le traitement de données fonctionnelles. De plus, l'avantage majeur des espaces L^p sur des espaces de Banach arbitraires est qu'ils permettent une caractérisation aisée des formes linéaires continues. En effet, pour $1 \leq p < \infty$, le dual topologique de $L^p(\mu)$, où μ est une mesure σ-finie définie sur un espace X, peut être identifié avec $L^q(\mu)$ (où p et q sont des exposants conjugués, i.e. $\frac{1}{p} + \frac{1}{q} = 1$).

Plus précisément, on a le théorème de représentation de Riesz (voir [68]) :

Théorème 1. *Soit μ une mesure σ-finie sur un espace mesurable X. Soit p un réel tel que $1 \leq p < \infty$, et soit q son exposant conjugué ($\frac{1}{p} + \frac{1}{q} = 1$), alors pour tout élément $\omega \in (L^p(\mu))^*$, il existe un unique élément $f \in L^q(\mu)$ tel que pour chaque $g \in L^p(\mu)$, on ait :*

$$\omega(g) = \int f g d\mu$$

L'identification des éléments de $(L^p(\mu))^*$ grâce au théorème de représentation, permet une caractérisation plus simple des neurones fonctionnels définis sur $L^p(\mu)$: soit la fonction d'activation T (de $\mathbb{R}$ dans $\mathbb{R}$), et soit $b \in \mathbb{R}$, l'expression

[3] plus précisément des classes de fonctions.

du neurone fonctionnel se simplifie alors de la manière suivante :

$$N(g) = T\left(b + \int fgd\mu\right)$$

Par identification du dual de $L^p(\mu)$ avec $L^q(\mu)$, on appelle encore la fonction f "*la fonction de poids*".

Plus généralement, si le produit fg est μ-intégrable pour chaque élément g appartenant à E, alors le neurone fonctionnel $N(g) = T\left(b + \int fgd\mu\right)$ peut encore être défini. Si l'on considére le cas particulier où $E = C_0(\mathbb{R}^n, \mathbb{R})$ ($C_0(\mathbb{R}^n, \mathbb{R})$ désignant l'ensemble des fonctions continues de $\mathbb{R}^n$ dans $\mathbb{R}$ à support compact), la fonction de poids f peut être choisie dans l'ensemble des fonctions μ-essentiellement bornées : $f \in L^\infty(\mu)$. Il faut noter qu'en utilisant une représentation plus simple du neurone fonctionnel (i.e., une forme intégrale), on introduit une restriction sur l'ensemble des fonctions de poids, qui est à présent strictement inclus dans E^* (pour l'exemple présent, le dual topologique de $C_0(\mathbb{R}^n, \mathbb{R})$ n'est pas en effet identifiable à $L^\infty(\mu)$).

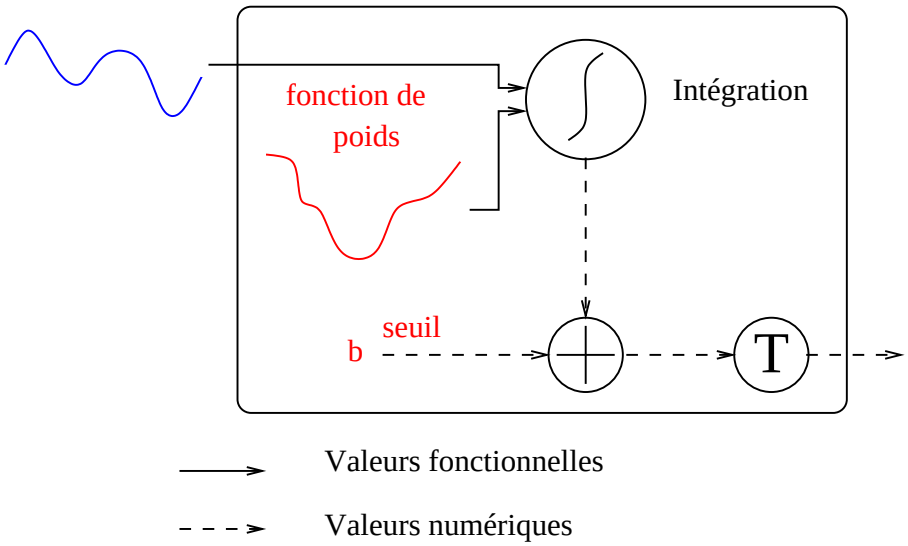


FIG. 3.1 – Le neurone fonctionnel

Définition 2. Soient (X, μ) un espace mesurable, où μ est une mesure σ-finie. On considère E et V deux ensembles de fonctions mesurables de X dans $\mathbb{R}$ (muni de la tribu borélienne) tels que pour chaque $g \in E$ et chaque $f \in V$, le produit fg soit μ-intégrable.

Un neurone fonctionnel défini sur E est caractérisé par une fonction d'activation T (de $\mathbb{R}$ dans $\mathbb{R}$), une fonction de poids $f \in V$ et un biais $b \in \mathbb{R}$. Le

neurone calcule la fonction suivante :

$$N(g) = T\left(b + \int f g d\mu\right) \tag{3.2}$$

La figure 3.1 représente graphiquement le fonctionnement d'un neurone fonctionnel.

3.3 Perceptron multi-couches fonctionnel

On souhaite à présent étendre le perceptron multi-couches numérique aux espaces fonctionnels $L^p(\mu)$, en s'appuyant sur la définition du neurone fonctionnel donnée dans la section précédente.

La sortie du neurone fonctionnel, à l'instar de celle du neurone numérique, est une valeur réelle. La construction du perceptron multi-couches fonctionnel peut donc être réalisée en combinant les neurones fonctionnels et les neurones numériques selon une organisation par couches à la manière d'un perceptron multi-couches classique : la première couche du réseau est composée exclusivement de neurones fonctionnels tandis que les couches suivantes ne sont constituées que de neurones numériques.

Dans le cas d'un perceptron fonctionnel à une couche cachée et à valeurs réelles (voir 3.2), le réseau calcule la fonction suivante :

$$H(g) = \sum_{k=1}^{K} a_k T\left(b_k + \int f_k g d\mu\right)$$

où $g \in L^p(\mu)$ et où les f_k sont les fonctions de poids, éléments de $L^q(\mu)$.

3.4 Entrées multiples

Jusqu'à présent, on a supposé dans les définitions précédentes que chaque individu était décrit par une unique fonction de $\mathbb{R}^n$ dans $\mathbb{R}$. L'extension de ce modèle à des individus décrits par plusieurs fonctions s'effectue aisément : on considère pour cela l'espace produit des espaces d'entrée des fonctions, ainsi que les formes linéaires continues définies sur cet espace.

La définition suivante étend le neurone fonctionnel au cas où chaque individu est décrit par p fonctions réelles.

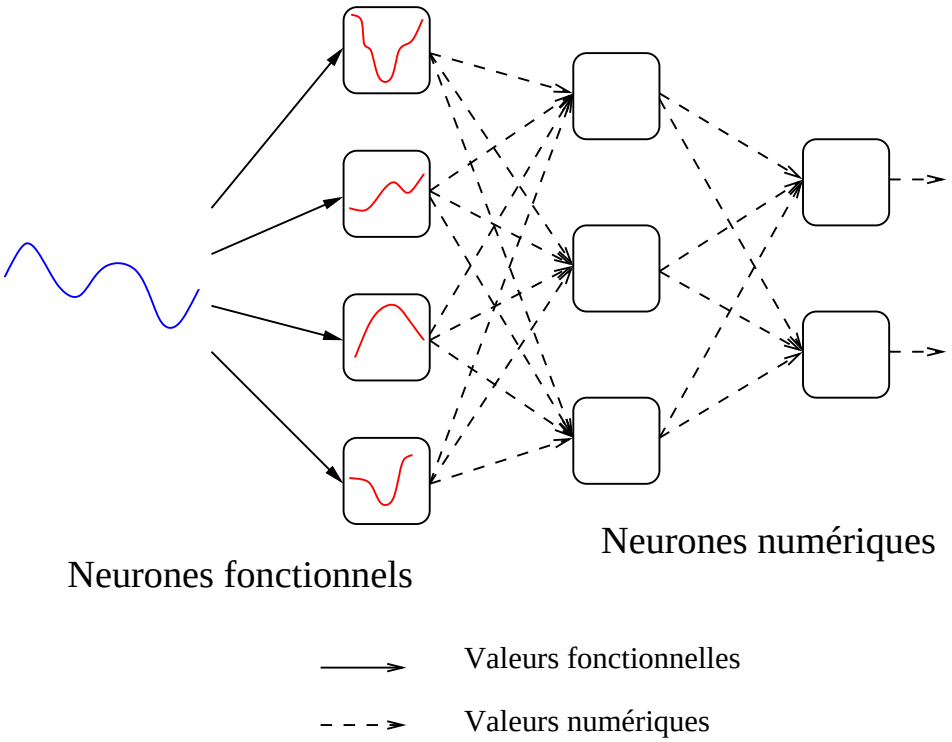


FIG. 3.2 – Le perceptron multi-couches fonctionnel

Définition 3. Soient $(X_1, \mu_1), \ldots, (X_p, \mu_p)$ p espaces mesurables, où chaque μ_i est une mesure σ-finie. On considère E_i et V_i, deux ensembles de fonctions mesurables de X_i vers $\mathbb{R}$ tels que pour chaque $g \in E_i$, et pour chaque $f \in V_i$, le produit fg soit μ_i-intégrable.

Un neurone fonctionnel défini sur l'espace produit $E_1 \times \ldots \times E_p$ et dont les fonctions de poids appartiennent à l'ensemble $V_1 \times \ldots \times V_p$ est une fonction de $E_1 \times \ldots \times E_p$ dans $\mathbb{R}$, caractérisée par une fonction d'activation T (de $\mathbb{R}$ dans $\mathbb{R}$), par p fonctions de poids $f_i \in V_i$ et par un biais $b \in \mathbb{R}$. Le neurone fonctionnel calcule la fonction suivante :

$$N(g_1, \ldots, g_p) = T\left(b + \sum_{i=1}^{p} \int f_i g_i d\mu_i\right) \tag{3.3}$$

Dans le cas particulier où chaque individu est décrit par une (ou plusieurs) fonction à valeurs vectorielles, la définition ci-dessus peut encore être appliquée en considérant les fonctions coordonnées de la fonction associée à chaque individu. La figure 3.3 représente un neurone fonctionnel avec entrées multiples.

L'extension du perceptron multi-couches fonctionnel à des individus décrits

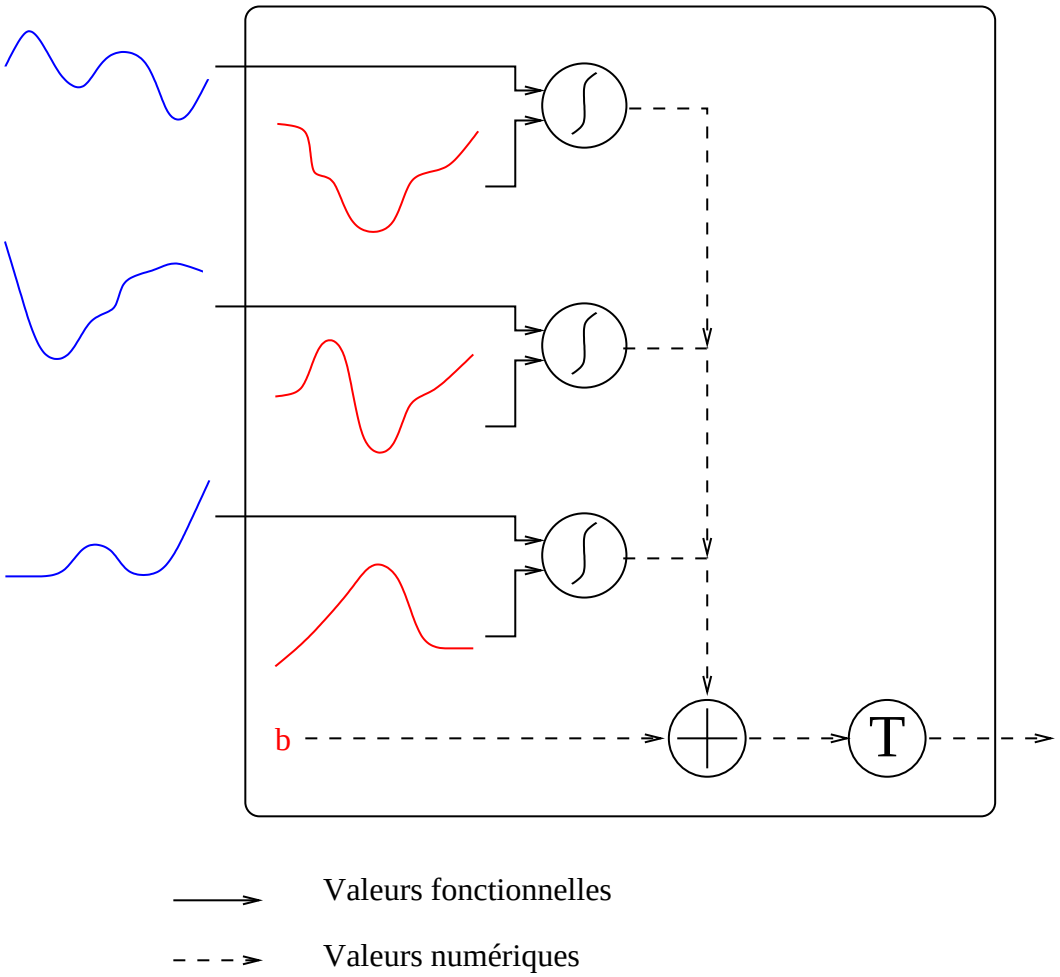


FIG. 3.3 – Le neurone fonctionnel avec entrées multiples

par plusieurs fonctions s'effectue naturellement en considérant cette nouvelle définition du neurone fonctionnel.

3.5 Approche paramétrique

Bien que la définition du neurone fonctionnel ait été simplifiée grâce à l'utilisation du théorème de représentation, sa mise en œuvre pratique reste cependant peu aisée. En effet, manipuler un neurone fonctionnel nécessite la manipulation de sa fonction de poids, qui est un élément arbitraire de $L^q(\mu)$. Cette difficulté est un problème classique de l'Analyse de Données Fonctionnelles (voir section

2.3.1). Afin de résoudre ce problème, on représente les fonctions de poids par des régresseurs paramétriques.

Plus précisément, on a la définition suivante :

Définition 4. Soit (X, μ) un espace mesurable, où μ est une mesure σ-finie. On considère E un ensemble de fonctions mesurables de X dans $\mathbb{R}$ et soient W un ensemble et F une fonction de $W \times X$ dans $\mathbb{R}$ tels que pour chaque $w \in W$ et chaque $g \in E$, le produit $F(w, .)g$ soit μ-intégrable.

Un neurone fonctionnel défini sur E est la fonction de E dans $\mathbb{R}$ caractérisée par une fonction d'activation T (de $\mathbb{R}$ dans $\mathbb{R}$), un paramètre $w \in W$ et un biais $b \in \mathbb{R}$. Le neurone calcule la fonction suivante :

$$N(g) = T\left(b + \int F(w, .)g\right) \tag{3.4}$$

La figure 3.4 représente un neurone fonctionnel paramétrique dont la fonction de poids est calculée par un perceptron multi-couches numérique.

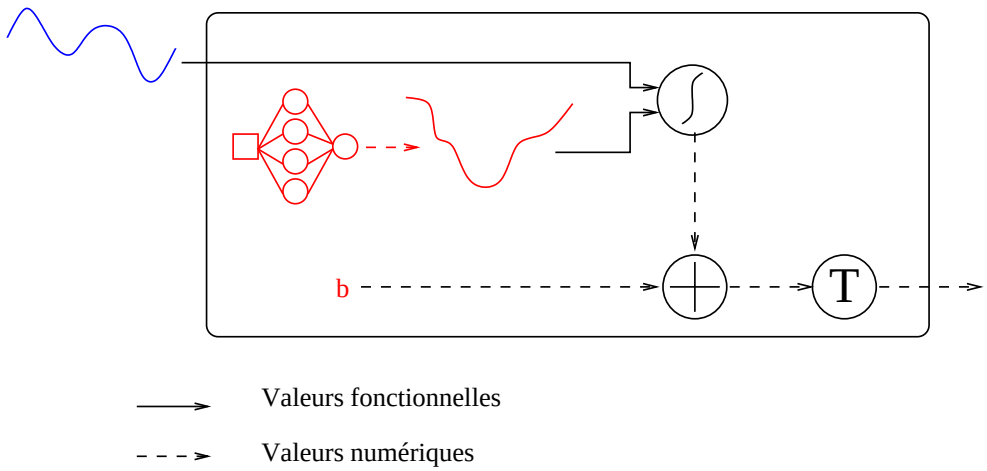


FIG. 3.4 – Le neurone fonctionnel paramétrique

La construction d'un perceptron fonctionnel paramétrique s'effectue alors naturellement en considérant cette définition du neurone paramétrique. Dans la pratique, il est bien sûr possible d'utiliser des régresseurs paramétriques de natures différentes pour chaque neurone fonctionnel du perceptron.

Dans l'exemple du perceptron fonctionnel à une couche cachée, le réseau calcule la fonction suivante :

$$H(g) = \sum_{k=1}^{K} a_k T\left(b_k + \int F_k(w_k, .)g d\mu\right), \tag{3.5}$$

où $(a_1, b_1, \ldots, a_K, b_K) \in \mathbb{R}^{2K}$, $w_k \in W_k$ et $W = W_1 \times \ldots \times W_K$.

Comme on le verra dans le chapitre 5, dans la pratique, $X = \mathbb{R}^n$, μ est une mesure de probabilité, W est un sous-ensemble compact de $\mathbb{R}^t$ et $E = L^p(\mu)$. Chaque F_k peut alors être n'importe quel régresseur paramétrique (perceptron multi-couches, B-spline, séries de Fourier, etc).

3.6 Apprentissage

Si l'on considère un perceptron fonctionnel dont les fonctions de poids sont représentées par des régresseurs paramétriques, le modèle fonctionnel obtenu est entièrement paramétré par un nombre fini de valeurs réelles. Le problème de l'apprentissage d'un tel réseau est donc similaire à celui d'un perceptron multi-couches numérique : il nécessite de pouvoir effectuer des calculs de gradient afin d'appliquer les algorithmes classiques d'optimisation (voir par exemple [60], [35] et [61]). Dans la section suivante, on montre comment la dérivée du neurone en fonction de ses différents paramètres peut être calculée.

3.6.1 Calcul du gradient

Si l'on considère le cas particulier du neurone fonctionnel où chaque individu est décrit par une unique fonction. Le neurone calcule alors la fonction suivante :

$$N(g) = T\left(b + \int F(w, .)g\right)$$

Le gradient de N en fonction du vecteur paramètre w peut alors être calculé comme indiqué dans la proposition suivante :

Proposition 1. *On suppose que $\frac{\partial F}{\partial w}$ existe μ-presque partout, est mesurable et est dominée (i.e. il existe une fonction mesurable positive f tel que $\left|\frac{\partial F}{\partial w}(w, x))\right| \leq f(x)$ pour μ-presque tout x et $\int f g d\mu < \infty$). Alors $w \mapsto \int F(w, x)g(x)d\mu(x)$ est dérivable en fonction de w et le gradient est donné par :*

$$\int \frac{\partial F}{\partial w}(w, x)g(w)d\mu(x) \tag{3.6}$$

Démonstration. Cette proposition est une conséquence directe du théorème de convergence dominée. □

Si de plus la fonction T est elle-même dérivable, on a :

$$\frac{\partial N}{\partial w}(w, b, g) = T'\left(b + \int F(w, x)g(x)d\mu(x)\right) \int \frac{\partial F}{\partial w}(w, x)g(x)d\mu(x) \tag{3.7}$$

3.6.2 Rétro-propagation

L'architecture du perceptron multi-couches fonctionnel étant très proche de celle du perceptron numérique classique (seule la première couche diffère), l'algorithme de rétro-propagation du gradient peut facilement être adapté au modèle fonctionnel.

Si de plus, la fonction de poids F est calculée grâce à un perceptron multi-couches numérique, l'algorithme de rétro-propagation généralisée, présenté dans [37], peut être utilisé afin de calculer efficacement l'expression $\frac{\partial F}{\partial w} g$. On montre en effet, dans le cas où F est à valeurs vectorielles, que l'algorithme de rétro-propagation standard dissocie le calcul de $\frac{\partial F}{\partial w}$, et l'évaluation du produit scalaire avec g. Dans l'algorithme de rétro-propagation généralisée en revanche, l'expression $\frac{\partial F}{\partial w} g$ est évaluée directement, ce qui réduit le coût du calcul (cf [37]).

3.7 Conclusion

Dans ce chapitre, on a pu voir que l'extension du perceptron multi-couches aux espaces fonctionnels s'effectuait naturellement en considérant une définition adaptée du neurone numérique classique. De plus, on a montré que grâce à la représentation des fonctions de poids par des régresseurs paramétriques, le perceptron multi-couches fonctionnel se trouvait paramétré par un nombre fini de paramétres numériques. Ceci permet l'utilisation des algorithmes standards d'optimisation, ainsi que l'adaptation de l'algorithme de rétro-propagation du gradient.

Dans le chapitre suivant, on énonce un résultat d'approximation universelle qui est la justification théorique de l'utilisation de ce modèle dans les problèmes de régression ou de discrimination. Puis dans le chapitre 5, on s'intéressera à la mise en œuvre pratique de ce modèle et au problème de l'estimation consistante de ses paramètres.

Chapitre 4

Approximation universelle

4.1 Introduction

L'approximation de fonctions joue un rôle fondamental dans les problèmes de modélisation statistique tels que les problèmes de régression ou de discrimination. Comme expliqué dans le chapitre précédent, la résolution de tels problèmes nécessite l'utilisation de modéles suffisamment généraux : on dit de tels modèles qu'ils sont des approximateurs universels.

De nombreuses familles de fonctions paramétrées possèdent la propriété d'approximation universelle. Parmi les modèles linéaires généralisés, on peut citer par exemple les polynômes, les séries de Fourier, ou les B-spline. Dans le cas des modèles non-linéaires, on s'intéresse tout particulièrement au perceptron multi-couches numérique, dont l'étude théorique (récente en regard des autres modèles) a permis d'énoncer divers résultats d'approximation universel. On peut citer par exemple Hornik et al. [44], Cybenko [21], Hornik [42] et Hornik [43] (la liste n'est bien sûr pas exhaustive). Ces différents travaux poursuivent en majorité deux buts distincts :

- obtention du résultat d'approximation universelle pour diverses classes de fonctions. Dans la pratique, cependant, on s'intéresse essentiellement à l'ensemble des fonctions continues ou à l'ensemble des fonctions mesurables de puissance p intégrable.
- l'utilisation d'hypothèses moins restrictives sur la nature de la fonction d'activation du perceptron multi-couches numérique.

Il semble important de noter que ces divers résultats d'approximation universel sont des résultats d'existence qui ne fournissent pas de méthodes pour choisir une topologie de réseaux (choix optimal du nombre de couches[1], choix du nombre

[1]Une seule couche cachée est suffisante pour obtenir la propriété d'approximation univer-

de neurones, choix des paramètres numériques (voir [10])). Ce choix nécessite donc de la part du praticien la mise en place d'une méthodologie rigoureuse afin de déterminer la topologie du réseau réalisant efficacement l'approximation.

Lors du chapitre précédent, on a montré que l'extension du perceptron multi-couches à des espaces fonctionnels s'effectuait naturellement en considérant une définition adaptée du neurone numérique classique. On s'intéresse à présent à la justification théorique de l'utilisation d'un tel modèle : le perceptron multi-couches fonctionnel a-t-il la capacité d'approcher arbitrairement près toute fonction donnée suffisamment régulière ?

Dans un cadre très général, Stinchcombe [75] apporte des éléments de réponse à cette question. En effet, dans ce travail, l'auteur propose une extension du perceptron multi-couches à des espaces topologiques arbitraires (voir définition 1), et montre sous certaines conditions générales, que ce modèle est un approximateur universel. L'adaptation de ce résultat au perceptron multi-couches fonctionnel n'est pas immédiate, car les hypothèses nécessaires à son application sont quelques peu techniques. C'est la raison pour laquelle on énonce dans ce chapitre deux corollaires au travail de [75], qui montrent que le perceptron multi-couches fonctionnel est un approximateur universel.

4.2 Définitions

On introduit les notations et les définitions suivantes (issues de Stinchcombe [75]) :

4.2.1 Espaces fonctionnels et distances associées

On note $C(X, Y)$ l'ensemble des fonctions continues de X dans Y, où X et Y sont deux espaces topologiques. En particulier, on note C^n l'ensemble des fonctions continues de $\mathbb{R}^n$ dans $\mathbb{R}$. On note enfin M^n l'ensemble des fonctions (Borel) mesurables de $\mathbb{R}^n$ dans $\mathbb{R}$.

Soit d_C la distance sur M^n qui donne la convergence uniforme sur les sous-ensembles compacts :

$$d_C(f, g) = \sum_{n \in \mathbb{N}^*} \frac{1}{2^n} \min\{\sup_{|x| \leq n} |f(x) - g(x)|, 1\}$$

Pour K un sous-ensemble compact de X, on considère ρ_K, la distance sur

selle.

l'ensemble des fonctions bornées de K dans $\mathbb{R}$, définie par :

$$\rho_K(f,g) = \sup_{x \in K} |f(x) - g(x)|$$

Dans les énoncés suivants, on rappelle les définitions de densité dans X et dans X^* (voir [13]) :

Définition 5. Soit X un espace métrique et d sa distance associée. Soit S et C deux sous-ensembles de X.
- S est d-extérieurement dense dans C, si la d-fermeture de S contient l'ensemble C.
- S est d-intérieurement dense dans C, si la d-fermeture de $S \cap C$ contient l'ensemble C.

Définition 6. Soit X un espace topologique, et soit L un sous-ensemble de X^*. L est dense pour la topologie $*$-faible si pour tout élément $l^* \in X^*$, il existe une suite l_n d'éléments de L qui converge vers l^* au sens de la topologie $*$-faible, i.e. $l_n(x)$ converge vers $l^*(x)$ pour tout $x \in X$.

4.2.2 Perceptron à une couche cachée

Définition 7. Soit T une fonction de $\mathbb{R}$ dans $\mathbb{R}$, et n un entier positif. S_T^n est l'ensemble de fonctions exactement calculées par un perceptron à une couche cachée défini sur $\mathbb{R}^n$, à valeurs réelles, et utilisant T comme fonction d'activation, i.e. l'ensemble des fonctions de $\mathbb{R}^n$ dans $\mathbb{R}$ de la forme :

$$H(x) = \sum_{k=1}^{K} \beta_k T(w_k.x + b_k)$$

où $K \in \mathbb{N}^*$, $\beta_k \in \mathbb{R}$ et $(w_k, b_k) \in \mathbb{R}^{n+1}$.

On a de même la définition suivante pour un perceptron multi-couches défini sur un espace vectoriel topologique :

Définition 8. Soit X un un espace vectoriel topologique, $\mathcal{A}$ un sous-ensemble de X^*, et T une fonction de $\mathbb{R}$ dans $\mathbb{R}$. $S_T^X(\mathcal{A})$ est l'ensemble des fonctions exactement calculées par un perceptron généralisé à une couche cachée défini sur X, à valeurs réelles, et utilisant T comme fonction d'activation, i.e. l'ensemble des fonctions de X dans $\mathbb{R}$ de la forme :

$$H(x) = \sum_{k=1}^{K} \beta_k T(\omega_k(x) + b_k)$$

où $K \in \mathbb{N}^*$, $\beta_k \in \mathbb{R}$, $b_k \in \mathbb{R}$ et $\omega_k \in \mathcal{A}$.

Dans le cas plus général où $\mathcal{A}$ est un ensemble arbitraire de fonctions de X dans $\mathbb{R}$, le biais b_k peut être supprimé de la définition.

4.3 Résultats existants

Comme expliqué dans l'introduction, l'étude théorique du perceptron multicouches numérique a permis d'énoncer divers résultats d'approximation universel. Parmi ces résultats deux travaux précurseurs méritent d'être cités : Hornik et al. [44] et Cybenko [21].

Parmi les résultats les plus récents et les plus généraux, on peut rappeler les deux résultats suivants qui montrent que S_T^n est un approximateur universel pour C^n et pour $L^p(\mu)$.

Théorème 2 (Hornik 93). *Si T est une fonction mesurable de $\mathbb{R}$ dans $\mathbb{R}$ non polynomiale et Riemann intégrable sur un intervalle compact (non réduit à un point) de $\mathbb{R}$, alors S_T^n est d_C-extérieurement dense pour C^n.*

Théorème 3 (Hornik 91). *Si T est une fonction mesurable de $\mathbb{R}$ dans $\mathbb{R}$, bornée et non constante, et si μ est une mesure finie sur $\mathbb{R}^n$, alors S_T^n est dense dans $L^p(\mu)$ pour $1 \leq p < \infty$.*

Afin d'énoncer le résultat d'approximation pour les espaces de dimension infinie, on a besoin de la définition suivante :

Définition 9. Soit X un ensemble arbitraire, et $\mathcal{A}$ un ensemble de fonctions de X dans $\mathbb{R}$. $\mathcal{A}$ sépare les points dans X, si pour tout $x, y \in X$ tel que $x \neq y$, il existe un élément de $a \in \mathcal{A}$ tel que $a(x) \neq a(y)$.

Dans le cadre des espaces de dimension infinie, on a le résultat suivant :

Théorème 4 (Stinchcombe 99). *Soit X un espace topologique, et $\mathcal{K}$ un sous-ensemble compact de X. Soit $\mathcal{A}$ un espace vectoriel de fonctions mesurables de X dans $\mathbb{R}$.*

1. *Si S_T^1 est d_C-intérieurement dense dans C^1, et si la $\rho_{\mathcal{K}}$-fermeture de $C(\mathcal{K}, \mathbb{R}) \cap \mathcal{A}$ contient les fonctions constantes et sépare les points dans $\mathcal{K}$, alors $S_T^X(\mathcal{A})$ est $\rho_{\mathcal{K}}$-intérieurement dense pour $C(\mathcal{K}, \mathbb{R})$.*
2. *Si S_T^1 est d_C-extérieurement dense dans C^1, et si l'intersection de $C(\mathcal{K}, \mathbb{R})$ et de la $\rho_{\mathcal{K}}$-fermeture de $\mathcal{A}$ contient les fonctions constantes et sépare les points dans $\mathcal{K}$, alors $S_T^X(\mathcal{A})$ est $\rho_{\mathcal{K}}$-extérieurement dense pour $C(\mathcal{K}, \mathbb{R})$.*

Dans le cas, où l'on restreint l'ensemble $\mathcal{A}$ à un sous-ensemble dense pour la topologie $*$-faible dans X^*, on a le corollaire suivant :

Corollaire 1 (Stinchcombe 99). *Soit X un espace topologique, et $\mathcal{K}$ un sous-ensemble compact de X. Soit $\mathcal{A}$ un sous-ensemble dense pour la topologie $*$-faible dans X^*.*

1. *Si S_T^1 est d_C-intérieurement dense dans C^1, alors $S_T^X(\mathcal{A})$ est $\rho_{\mathcal{K}}$-intérieurement dense pour $C(\mathcal{K}, \mathbb{R})$.*
2. *Si S_T^1 est d_C-extérieurement dense dans C^1, alors $S_T^X(\mathcal{A})$ est $\rho_{\mathcal{K}}$-extérieurement dense pour $C(\mathcal{K}, \mathbb{R})$.*

4.4 Corollaires pour le perceptron multi-couches fonctionnel

On a vu dans la section précédente que $S_T^X(\mathcal{A})$ était intérieurement ou extérieurement dense dans différents espaces fonctionnels. En effet, le théorème 4 et le corollaire 1 (ainsi que ceux de Sandberg [69]) montrent que les résultats d'approximation de $S_T^X(\mathcal{A})$ sont valables pour presque tout espace arbitraire X. Dans la pratique cependant, ces résultats sont assez complexes à appliquer, car les hypothèses à vérifier sur l'ensemble $\mathcal{A}$ ne sont pas triviales (densité et séparation).

Dans cette section, on montre que ces hypothèses sont satisfaites par les perceptrons fonctionnels à une couche cachée.

Corollaire 2. *Soit μ une mesure de Borel finie sur $\mathbb{R}^n$. Soit $1 < p \leq \infty$ un réel et q sont exposant conjugué. Soit V un sous-ensemble dense de $L^q(\mu)$. Soit $\mathcal{A}_V$ l'ensemble des formes linéaires continues sur $L^p(\mu)$ de la forme $l(f) = \int fgd\mu$, où $g \in V$. Soit T une fonction mesurable de $\mathbb{R}$ dans $\mathbb{R}$ telle que S_T^1 est d_C-intérieurement (resp. d_C-extérieurement) dense dans C^1. Alors $S_T^{L^p(\mu)}(\mathcal{A}_V)$ est $\rho_{\mathcal{K}}$-intérieurement (resp. $\rho_{\mathcal{K}}$-extérieurement) dense dans $C(\mathcal{K}, \mathbb{R})$, où $\mathcal{K}$ est sous-ensemble compact de $L^p(\mu)$*

Démonstration. Si $1 < p < \infty$, on sait que $L^q(\mu)$ (avec $q < \infty$) peut être identifié avec $(L^p(\mu))^*$ (voir par exemple [68]). Plus précisément, pour chaque $l \in (L^p(\mu))^*$, il existe une unique fonction $f \in L^q(\mu)$ telle que $l(g) = \int fgd\mu$. Par hypothèse, V est dense dans $L^q(\mu)$. Ceci implique que $\mathcal{A}_V$ est dense dans $(L^p(\mu))^*$ pour la topologie $*$-faible. La conclusion est alors obtenue par le corollaire 1.

Si $p = \infty$, on ne peut pas appliquer directement le corollaire 1 car le dual de $L^\infty(\mu)$ ne s'identifie pas avec $L^1(\mu)$. Considérons malgré tout $\mathcal{A}$, l'ensemble des fonctions affines de $L^\infty(\mu)$ définies par $l(f) = \alpha + \int fgd\mu$, où α est un réel arbitraire et g une fonction arbitraire de V (un sous-ensemble de $L^1(\mu)$). $\mathcal{A}$ est un espace vectoriel qui contient les fonctions constantes de $C(\mathcal{K}, \mathbb{R})$. On cherche donc à montrer que $\mathcal{A}$ sépare les points dans $\mathcal{K}$. Soient u et v deux fonctions distinctes de $\mathcal{K}$. La fonction $f = u - v$ est une fonction non nulle appartenant à $L^\infty(\mu)$. On peut supposer que l'ensemble mesurable $H = \{x \in \mathbb{R}^n | f(x) > 0\}$ a une mesure (finie) non nulle (dans le cas contraire on remplace f par $-f$). Alors $\int f\chi_H d\mu > 0$, i.e. $\int u\chi_H d\mu \neq \int v\chi_H d\mu$. Comme μ est une mesure finie, χ_H appartient à $L^1(\mu)$. Comme V est dense dans $L^1(\mu)$, il existe une suite h_k de fonctions dans V qui converge vers χ_H. On a bien sûr

$$\left| \int f(h_k - \chi_H) d\mu) \right| \leq |f|_\infty \left| \int h_k - \chi_H d\mu \right|$$

Il existe donc un indice k tel que $\int fh_k d\mu > 0$, i.e. il existe une fonction $h_k \in V$ telle que $\int uh_h d\mu \neq \int vh_k d\mu$. $\mathcal{A}$ sépare donc les points dans $\mathcal{K}$. La conclusion est obtenue en appliquant le théorème 4. □

Corollaire 3. *Soit μ une mesure de Borel finie à support compact sur $\mathbb{R}^n$. Soit T une fonction mesurable de $\mathbb{R}$ dans $\mathbb{R}$, telle que S_T^1 est d_C-intérieurement (resp d_C-extérieurement) dense dans C^1. Soit V un sous-ensemble de $L^\infty(\mu)$ d_C-extérieurement dense pour C^n. Alors $S_T^{L^1(\mu)}(\mathcal{A}_V)$ est $\rho_\mathcal{K}$-intérieurement (resp $\rho_\mathcal{K}$-extérieurement) dense dans $C(\mathcal{K}, \mathbb{R})$, où $\mathcal{K}$ est un sous-ensemble compact de $L^1(\mu)$.*

Démonstration. Grâce au théorème de Lusin (voir [68]), on sait que pour toute fonction f dans $L^\infty(\mu)$, il existe une suite de fonctions continues g_k à support compact, qui converge ponctuellement vers f et telle que $|g|_\infty \leq |f|_\infty$. Une simple application du théorème de convergence dominée montre que pour chaque fonction h dans $L^1(\mu)$, $\int g_k h d\mu \to_{k\to\infty} \int fhd\mu$. Comme μ est à support compact, il existe un compact K tel que $\int g_k h d\mu = \int_K g_k h d\mu$. Grâce aux hypothèses, chaque g_k peut être approchée par une fonction ϕ_k dans V telle que $\sup_{x\in K} |g_k(x) - \phi_k(x)| < \frac{1}{k}$. Dans ce cas $|\int_K g_k h d\mu - \int_K \phi_k h d\mu| < \frac{1}{k} \parallel h \parallel_1$. Comme μ est à support compact, on conclut que $\int \phi_k h d\mu \to_{k\to\infty} \int fhd\mu$. L'ensemble des formes linéaires continues $\mathcal{A}_V$ est donc dense pour la topologie $*$-faible dans $(L^1(\mu))^*$. La conclusion du corollaire est enfin obtenue en appliquant le corollaire 1. □

4.5 Discussion

Le corollaire 2 montre que tant que l'on peut approcher les fonctions de $L^p(\mu)$ et de C^1, alors un perceptron fonctionnel à une couche cachée peut être utilisé pour approcher les fonctions de $C(\mathcal{K}, \mathbb{R})$, où $\mathcal{K}$ est un sous-ensemble compact de $L^p(\mu)$. Les résultats antérieurs imposent de très faibles hypothèses sur T afin que S_T^1 soit d_C-intérieurement ou d_C-extérieurement dense dans C^1 (voir par exemple le théorème 2 ou [51]). Pour résumer, T doit être une fonction non polynomiale et Riemann intégrable sur un intervalle compact non dégénéré de $\mathbb{R}$. Cette condition est bien sûr remplie par les fonctions d'activation usuelles telles que la tangente hyperbolique ou la fonction logistique.

Dans le corollaire 2, le perceptron multi-couches fonctionnel utilise des fonctions de poids appartenant à un sous-ensemble V dense dans $L^q(\mu)$. Dans la pratique, les fonctions de poids sont représentées grâce à des fonctions paramétriques. Cette contrainte n'introduit aucune difficulté tant que l'on choisit une classe de régresseurs dense dans $L^q(\mu)$. Grâce au théorème 3, on peut par exemple utiliser un perceptron à une couche cachée avec comme fonction d'activation U (i.e., $V = S_U^n$) tant que U est une fonction mesurable bornée et non constante (comme $p > 1$ et $q < \infty$, le théorème 3 s'applique). D'autres modèles peuvent être utilisés (B-splines, ondelettes, séries de Fourier, etc.) mais ils impliquent en général des restrictions supplémentaires sur l'espace fonctionnel considéré.

La démonstration du corollaire 2 peut être étendue au cas où $p = 1$, le corollaire 3 peut donc sembler redondant. Comme le souligne dans son introduction Stinchcombe [75], S_T^n n'est pas dense dans $L^\infty(\mu)$. L'hypothèse principale du corollaire 2 (V doit être dense dans $L^q(\mu)$) ne peut être satisfaite par une approximation utilisant des perceptrons multi-couches. Le corollaire 2 présente donc un intérêt limité dans le cas où $p = 1$. C'est la raison pour laquelle le corollaire 3 est nécessaire : comme le montre le théorème 2, S_U^n peut être utilisé pour approcher les fonctions continues sur un ensemble compact. Le cas $p = 1$ est donc similaire aux cas $p > 1$ hormis le fait que la mesure doit être à support compact.

Les corollaires 2 et 3 montrent que si l'on considère un sous-ensemble compact $\mathcal{K}$ de $L^p(\mu)$, toute fonction de $C(\mathcal{K}, \mathbb{R})$ peut être approchée à une précision donnée par un perceptron fonctionnel à une couche cachée utilisant un nombre fini de paramètres. Malgré le changement important dans la nature de l'espace d'entrée (de $\mathbb{R}^n$ à un sous-ensemble compact d'un espace fonctionnel), l'approximation des fonctions continues reste donc toujours valable.

En analyse de données fonctionnelles, il est habituel de supposer que les fonctions étudiées sont régulières, i.e. au moins continues. Si l'on considère que ces

fonctions sont définies sur un sous-ensemble compact, le corollaire 2 peut alors être appliqué. En effet, les fonctions continues définies sur un sous-ensemble compact W de $\mathbb{R}^n$ sont clairement des éléments de $L^\infty(\lambda)$, où λ est la restriction de la mesure de Lebesgue à W (en fait, toute mesure de Borel définie sur W peut être utilisée). De plus, un sous-ensemble compact $\mathcal{K}$ d'un espace de fonctions régulières (muni de la norme uniforme) est un sous-ensemble compact de $L^\infty(\lambda)$. On voit donc que toute fonction continue de $\mathcal{K}$ vers $\mathbb{R}$ peut être approchée par un perceptron multi-couches fonctionnel, à condition de pouvoir approcher les éléments de $L^1(\lambda)$ (comme le montre le théorème 3, S_U^n réalise cette approximation).

4.6 Entrées multiples

Les corollaires de la section précédente sont basés sur le théorème 4 et sur le corollaire 1. Ces deux résultats font peu d'hypothèses sur l'espace d'entrée des perceptrons fonctionnels. Dans le cas particulier où les individus sont décrits par plusieurs fonctions, on voit que ces hypothèses sont satisfaites par des produits d'espaces L^p tels que $X = L^{p_1}(\mu_1) \times \ldots \times L^{p_r}(\mu_r)$. Afin d'appliquer le corollaire 1, il est nécessaire de pouvoir approcher les éléments de X^*. Comme un élément de X^* est une combinaison linéaire d'éléments de $(L^{p_i}(\mu_i))^*$, l'approximation est bien sûr valable pour $1 < p_i < \infty$. Dans ce cas, il est facile d'appliquer le corollaire 1 et d'étendre donc le corollaire 2 à un perceptron multi-couches fonctionnel à r entrées.

Le cas de $p_i = \infty$ est traité de manière similaire, mais s'appuie sur le théorème 4. Pour $p_i = 1$, il est nécessaire d'ajouter l'hypothèse que la mesure correspondante μ_i soit à support compact. Dans tous les cas, les conditions des corollaires 2 et 3 sont bien remplies, on peut donc appliquer le théorème 4 pour conclure que l'approximation universelle reste valable.

4.7 Conclusion

On a vu dans ce chapitre que malgré le changement important qui s'est opéré sur l'espace d'entrée du perceptron multi-couches (d'un espace de dimension finie à un espace fonctionnel), les résultats d'approximation universelle, démontrés dans le cadre classique, se transposent au cadre fonctionnel. On s'intéresse à présent à la mise en œuvre pratique d'un tel modèle. Jusqu'à présent, on a supposé que les fonctions d'entrée étaient connues de manière parfaite et

que l'intégrale du neurone fonctionnel pouvait être calculée de manière exacte. Dans la pratique, une telle hypothèse n'est pas valide, car chaque fonction d'entrée est donnée sous la forme d'un échantillonnage (connaissance discrétisée de la fonction) : l'évaluation exacte des intégrales n'est donc pas possible. Comme on va le voir dans le chapitre suivant, une première solution consiste à remplacer chaque intégrale par une moyenne empirique.

Chapitre 5

Cadre Probabiliste

5.1 Introduction

Le chapitre précédent a permis d'énoncer un résultat théorique important prouvant que le perceptron multi-couches fonctionnel est un approximateur universel. Dans la pratique, ce résultat d'approximation ne permet pas de garantir que l'on peut approcher la fonction à modéliser, car cette fonction n'est connue qu'en un nombre fini de points. On a donc besoin d'un résultat de consistance.

Pour le Perceptron multi-couches numérique, ce problème de consistance a été traité par White (voir [76]) : l'estimation des paramètres du modèle est consistante, i.e., les paramètres du modèle estimés sur un nombre fini d'exemples convergent vers les paramètres optimaux théoriques quand le nombre d'exemples croît vers l'infini.

On souhaite obtenir un résultat de consistance similaire adapté aux perceptrons multi-couches fonctionnels. Malheureusement, le résultat de White n'est valable qu'en dimension finie, et ne peut donc pas être appliqué au cadre fonctionnel. C'est la raison pour laquelle on énonce dans la première partie de ce chapitre, un premier résultat (basé sur les travaux d'Andrews [2]), qui étend la loi forte des grands nombres uniforme à des espaces de dimension infinie. Ce résultat général sera par la suite utilisé afin de prouver la consistance des différents modèles fonctionnels.

Le problème de consistance en modélisation fonctionnelle peut souvent être scindé en deux étapes distinctes. Ces deux étapes sont une conséquence directe du fait que l'on possède une connaissance doublement limitée de la fonctionnelle non-linéaire à modéliser. En effet, comme toujours en analyse de données, la fonctionnelle non-linéaire n'est connue que grâce à un nombre fini de couples entrée/sortie (la fonction d'entrée et sa valeur de sortie associée). De plus, et ceci est spécifique à l'analyse de données fonctionnelles, chaque fonction d'entrée

n'est elle même connue qu'en un nombre fini de points d'évaluation.

On énonce donc dans la section 5.3 un premier résultat de consistance. Ce résultat s'appuie sur une connaissance exacte des fonctions d'entrée, et ne prend donc pas en compte les problèmes liés à l'échantillonnage des fonctions. Puis dans la section 5.4, on énonce un second résultat de consistance tenant compte de la discrétisation des fonctions observées.

5.2 Loi forte des grands nombres uniforme

Dans cette section, on présente une extension de la loi forte des grands nombres uniforme aux espaces de dimension infinie. Ce résultat, nécessaire pour prouver la consistance des différents modèles fonctionnels, est en fait une conséquence d'un énoncé plus général formulé par Andrews [2].

Corollaire 4. *Soit X un espace métrique muni de sa tribu borélienne. Soit $(\Omega, \mathcal{A}, P)$ un espace probabilisé sur lequel est définie une suite de variables aléatoires Z_t. On suppose que les Z_t sont indépendantes identiquement distribuées et à valeurs dans X. Soit W un espace métrique compact, et soit l une fonction de $W \times X$ dans $\mathbb{R}$. On suppose que les hypothèses suivantes sont vérifiées :*

1. *Pour chaque $w \in W$, $l(w, .)$ est une fonction mesurable de X dans $\mathbb{R}$.*
2. *Pour chaque $x \in X$, $l(., x)$ est une fonction continue de W dans $\mathbb{R}$.*
3. *Il existe une fonction mesurable d (de X dans $\mathbb{R}$) telle que pour tout $x \in X$ et pour tout $w \in W$, $|l(w, x)| \leq d(x)$.*
4. *$E(d(Z_t)) < \infty$.*

On a alors :

$$\sup_{w \in W} \left| \frac{1}{n} \sum_{t=1}^{N} l(w, Z_t) - E(l(w, Z)) \right| \rightarrow^{p.s.}_{N \rightarrow \infty} 0$$

Afin de démontrer ce corollaire, on a tout d'abord besoin du résultat suivant :

Lemme 1. *Soit l une fonction de $W \times X$ dans $\mathbb{R}$, où W est un espace métrique séparable et X est un espace métrique muni de sa tribu borélienne. Si l est continue sur W pour chaque $x \in X$ et mesurable sur X pour chaque $w \in W$, alors la fonction $f(x) = \sup_{w \in W} l(w, x)$ est mesurable.*

Démonstration. Comme W est séparable, il existe un ensemble dénombrable $W' = \{w_i | i \in \mathbb{N}^*\}$ dense dans W. On va chercher à montrer que $f(x) = \sup_{w \in W'} l(w, x)$. On considère un $x \in X$ fixé. Soit ε un nombre réel positif

arbitraire. Par définition de f, il existe $w \in W$ tel que $l(w,x) \geq f(x) - \frac{\varepsilon}{2}$. Comme $l(.,x)$ est une fonction continue en w, il existe η tel que $|w' - w| < \eta$ implique $|l(w',x) - l(w,x)| < \frac{\varepsilon}{2}$, ce qui implique que $l(w',x) \geq f(x) - \varepsilon$. Comme W' est dense dans W, il existe $w' \in W'$ tel que $|w' - w| < \eta$. Ceci implique que $f(x) \geq \sup_{w \in W'} l(w,x) \geq f(x) - \varepsilon$. Ceci étant vrai pour tout ε, on a $f(x) = \sup_{w \in W'} l(w,x)$. Donc $f(x) = \sup_{i \in \mathbb{N}} l(w_i,x)$. Comme chaque fonction $l(w_i,x)$ est mesurable, le sup est aussi mesurable. □

On montre à présent le corollaire :

Démonstration. Le corollaire est une conséquence du théorème d'Andrews [2]. La démonstration revient à vérifier 3 hypothèses.

1. L'hypothèse A1 est vérifiée car W est un ensemble compact (W correspond à Θ dans l'article d'Andrews).
2. L'hypothèse A2 se décompose en deux sous-hypothèses :
   (a) L'hypothèse A2 (a) se traduit avec nos notations en l'hypothèse suivante : pour chaque $w_0 \in W$ (et pour tout $t \in \mathbb{N}$), $l(w_0, Z_t)$, $\sup_{w \in W(w_0,\eta)} l(w, Z_t)$ et $\inf_{w \in W(w_0,\eta)} l(w, Z_t)$ sont des variables aléatoires (avec $W(w_0,\eta) = B(w_0,\varepsilon) \cap W$, et $B(w_0,\varepsilon)$ est la boule fermée de centre w_0 et de rayon ε).

   $l(w_0, Z)$ est une variable aléatoire grâce à l'hypothèse 1 du corollaire 4. Grâce aux hypothèses 1 et 2 du corollaire 4 et au fait qu'un compact est séparable, le lemme 1 peut être appliqué à l et à $W(w_0,\eta)$, ce qui permet de conclure que $\sup_{w \in W(w_0,\eta)} l(w, Z)$ est une variable aléatoire. Le cas de $\inf_{w \in W(w_0,Z)} l(w, Z)$ est traité de manière identique en appliquant le lemme à la fonction $-l$.

   L'hypothèse A2 (a) est donc vérifiée.

   (b) L'hypothèse A2 (b) se traduit dans notre cas en l'hypothèse suivante : on peut appliquer la loi forte des grands nombres aux variables aléatoires suivante : $sup_{w \in W(w_0,\eta)} l(w, Z)$, $\inf_{w \in W(w_0,\eta)} l(w, Z)$. On cherche donc à montrer que :

$$\lim_{n \to \infty} \frac{1}{n} \sum_{t=1}^{n} \sup_{w \in W(w_0,\eta)} l(w, Z_t) = E\left(\sup_{w \in W(w_0,\eta)} l(w, Z_t) \right) \; P - p.s.$$

   Comme montré au point précédent, $\left(\sup_{w \in W(w_0,\eta)} l(w, Z_t)\right)_{t \in \mathbb{N}^*}$ et de même $\left(\inf_{w \in W(w_0,\eta)} l(w, Z_t)\right)_{t \in \mathbb{N}^*}$ sont des suites de variables aléatoires indépendantes et identiquement distribuées. De plus, grâce aux

hypothèses 3 et 4 du corollaire 4, ces variables aléatoires sont intégrables. On peut donc leur appliquer la loi forte des grands nombres. L'hypothèse A2 (b) est donc vérifiée.

3. L'hypothèse A3 se traduit dans notre cas en l'hypothèse suivante :

$$\lim_{\eta\to 0}\sup_{n\geq 1}\left|\frac{1}{n}\sum_{t=1}^{n}\left(E\left(\sup_{w\in W(w_0,\eta)} l(w,Z_t)\right)-E(l(w,Z_t))\right)\right|=0$$

On doit montrer la propriété équivalente pour $E\left(\inf_{w\in W(w_0,\eta)} l(w,Z_t)\right)$
Comme l est continue en w pour un x fixé, on a la convergence simple suivante :

$$\lim_{\eta\to 0}\sup_{w\in W(w_0,\eta)} l(w,.)=l(w_0,.)$$

Grâce aux hypothèses 3 et 4 du corollaire 4, on peut appliquer la convergence dominée, qui implique :

$$\lim_{\eta\to 0} E\left(\sup_{w\in W(w_0,\eta)} l(w,Z_t)\right)=E(l(w_0,Z_t))$$

Finalement, comme les Z_t sont indépendantes et identiquement distribuées, l'hypothèse A3 peut être simplifiée en :

$$\lim_{\eta\to 0}\left|E\left(\sup_{w\in W(w_0,\eta)} l(w,Z_t)\right)-E(l(w,Z_t))\right|=0$$

Ce qui est justement le résultat prouvé ci-dessus. On procède de manière similaire pour $E\left(\inf_{w\in W(w_0,\eta)} l(w,Z_t)\right)$.
L'hypothèse A3 est donc vérifiée.

Comme les trois hypothèses sont vérifiées, on peut appliquer le théorème d'Andrews qui donne exactement la conclusion du corollaire 4. □

5.3 Connaissance parfaite des fonctions

5.3.1 Cadre probabiliste

Comme expliqué dans l'introduction, la fonction que l'on cherche à approcher n'est connue que grâce à n couples entrée/sortie (g^i, t^i). Les g^i sont les fonctions d'entrée, et chaque t^i est l'élément associé à la fonction g^i.

Dans la pratique, on cherche à faire apprendre au perceptron multi-couches fonctionnel H (de vecteur poids w) la relation existant entre les fonctions d'entrée g^i et les valeurs à prédire t^i. On considère pour cela une fonction d'erreur c (par exemple une distance) et on cherche à minimiser la quantité $c(H(w, g^i), t^i)$ en moyenne.

Dans le théorème qui va suivre chaque fonction g^i va être modélisée par la variable aléatoire fonctionnelle G^i et de même chaque élément t^i va être modélisé par la variable aléatoire T^i. On suppose les couples (G^i, T^i) indépendants identiquement distribués ($G = G^1$ et $T = T^1$). On veut alors trouver le vecteur poids optimal qui minimise l'erreur théorique $E(c(H(w, G), T))$. Cette minimisation étant dans la pratique irréalisable, on remplace donc l'erreur théorique par une moyenne empirique :

$$E = \frac{1}{n} \sum_{i=1}^{n} c(H(w, g^i), t^i)$$

5.3.2 Consistance

On a le théorème suivant :

Théorème 5. *Soient K un entier, et $F_1, \ldots, F_K$, K régesseurs paramétriques tels que pour chaque k :*

1. *F_k est une fonction de $W_h^k \times \mathcal{X}$ dans $\mathbb{R}$.*
2. *W_h^k est un ensemble compact.*
3. *pour chaque $x \in \mathcal{X}$, $F_k(., x)$ est une fonction continue de W^k vers $\mathbb{R}$.*
4. *pour chaque $w \in W_h^k$, $F_k(w, .)$ est une fonction mesurable de $\mathcal{X}$ vers $\mathbb{R}$.*
5. *il existe une fonction mesurable d_k de $\mathcal{X}$ dans $\mathbb{R}$ qui appartient à $L^q(\mu)$ et telle que pour chaque $w \in W_h^k$ et pour tout $x \in \mathcal{X}$, $|F_k(w, x)| \leq d_k(x)$.*

On note $W_h = W_h^1 \times \ldots \times W_h^K$.

Soit G^i une suite de variables aléatoires fonctionnelles définies sur $(\Omega, \mathcal{A}, P)$ et à valeurs dans $L^p(\mu)$ (où μ est une mesure σ-finie). Soit $\mathcal{T}$ un espace métrique muni de sa tribu borélienne, et soit T^i une suite de variables aléatoires définis sur $(\Omega, \mathcal{A}, P)$ et à valeurs dans $\mathcal{T}$. On suppose que les couples de variables aléatoires (G^i, T^i) sont indépendants et identiquement distribués. On note $G = G^1$ et $T = T^1$.

Soit l une fonction de $\mathbb{R}^K \times \mathcal{T} \times W_o$ dans $\mathbb{R}$, où W_o est un ensemble compact. On suppose que :

1. *pour chaque $t \in \mathcal{T}$, $l(., t, .)$ est une fonction continue de $\mathbb{R}^K \times W_o$ vers $\mathbb{R}$.*

2. *pour chaque* $w_o \in W_o$, $l(.,.,w_o)$ *est une fonction mesurable de* $\mathbb{R}^K \times \mathcal{T}$ *vers* $\mathbb{R}$.
3. *il existe une fonction mesurable* d' *de* $\mathcal{T}$ *vers* $\mathbb{R}$ *telle que* $|l(z,t,w_o)| \leq d'(t)$ *pour tout* z *et* w_o
4. $E(d'(T)) < \infty$.

Pour chaque $\omega \in \Omega$, *on définit :*

$$\lambda^n(w_h, w_o)(\omega) = \frac{1}{n}\sum_{i=1}^{n} l\Bigg(\int F_1\left(w_h^1, x\right) G^i(\omega)(x)d\mu(x), \ldots, \int F_K\left(w_h^K, x\right) G^i(\omega)(x)d\mu(x), T^i(\omega), w_o\Bigg)$$

et

$$\lambda(w_h, w_o) = E\left(l\left(\int F_1\left(w_h^1, x\right) G(x)d\mu(x), \ldots, \int F_K\left(w_h^K, x\right) G(x)d\mu(x), T, w_o\right)\right)$$

Alors pour chaque $\omega \in \Omega$ *et pour chaque* n, *il existe une solution* $w^n(\omega)$ *au problème*

$$\min_{w \in W_h \times W_o} \lambda^n(w_h, w_o)(\omega)$$

Si W^* *est l'ensemble des minimiseurs de* $\lambda(w_h, w_o)$, *alors pour presque tout* $\omega \in \Omega$

$$\lim_{n \to \infty} d(w^n(\omega), W^*) = 0$$

Démonstration. On applique la loi forte des grands nombres uniforme (corollaire 4) à la fonction :

$$h(w_h, w_o, g, t) = l\left(\int F_1(w_h^1, x)g(x)d\mu(x), \ldots, \int F_K(w_h^K, x)g(x)d\mu(x), t, w_o\right)$$

Ceci est possible pour les raisons suivantes :

1. la fonction $h'((w_h, w_o), (g,t)) = h(w_h, w_o, g, t)$ est continue en $w = (w_h, w_o)$ pour chaque $x = (g,t)$, grâce aux hypothèses sur l et sur $F_1, \ldots, F_K$, et sachant que g appartient à $L^p(\mu)$. En effet, la fonction $w_h^k \mapsto \int F_k(w_h^k, x)g(x)d\mu(x)$ est continue pour chaque g : comme F_k est continue en w pour chaque x, la fonction $F_k(w', .)g(.)$ converge simplement vers $F_k(w, .)g(.)$ quand w' converge vers w. De plus, $|F_k(w, .)g(.)|$ est dominée sur W_h^k par $d_k(.)|g(.)|$, laquelle est intégrable (par hypothèse). Grâce au théorème de convergence dominée, ceci implique la continuité de la fonction $w_h^k \mapsto \int F_k(w_h^k, x)g(x)d\mu(x)$.

2. h' est mesurable en (g,t) pour chaque (w_h, w_o). C'est une conséquence directe des hypothèses sur l et du fait que $g \mapsto \int F_k(w_h^k, x)g(x)d\mu(x)$ est continue pour chaque w_h^k.
3. grâce au lemme 1 appliqué à $|h'|$, la fonction
$$c(g,t) = \sup_{(w_h,w_o)\in W_h\times W_o} |h'((w_h,w_o),(g,t))|$$
est mesurable.
4. $E(c(G,T)) < \infty$ par hypothèse sur l.

Grâce au corollaire 4, on a donc

$$\sup_{(w_h,w_o)\in W_h\times W_o} \left| \frac{1}{n}\sum_{i=1}^{n} h(w_h,w_o,G^i,T^i) - E(h(w_h,w_o,G,T)) \right| \to^{p.s.}_{n\to\infty} 0 \quad (5.1)$$

La conclusion finale est obtenue de manière similaire à White [76] :

On pose $w = (w_h, w_o)$, $W = W_h \times W_o$, et $h'(w,g,t) = h(w_h,w_o,g,t)$. On procède alors selon étapes suivantes :

1. $\lambda(w)$ est continue. Par continuité de h', $h'(w,g,t)$ converge simplement vers $h(w,g,t)$ quand w' converge vers w. De plus, h' est dominée par d'. Donc, le théorème de convergence dominée implique que $E(h'(w',G,T))$ converge vers $E(h'(w,G,T))$ quand w' converge vers w.
2. chaque λ^n est continue sur l'ensemble compact W. Il existe donc un minimiseur w^n.
3. on considère un $\omega \in \Omega$ pour lequel la convergence uniforme de λ^n vers λ a lieu. Comme W est un ensemble compact, la suite w^n a au moins un point d'accumulation w_0, et une sous-suite $w^{n'}$ qui converge vers lui. Soit ϵ un réel strictement positif. λ est uniformément continue sur W et donc il existe η tel que $|w' - w| < \eta$ implique $|\lambda(w) - \lambda(w')| < \epsilon$. Par convergence uniforme, pour n' suffisamment grand, $\|\lambda^{n'} - \lambda\|_\infty < \epsilon$. Pour n' suffisamment grand, on a aussi $|w^{n'} - w_0| < \eta$. Ceci implique $|\lambda^{n'}(w^{n'}) - \lambda(w_0)| < 2\epsilon$. Ceci implique que pour tout w, $\lambda(w_0) - \lambda(w) \leq 3\epsilon$, car l'optimalité de $w^{n'}$ implique $\lambda^{n'}(w^{n'}) - \lambda^{n'}(w) \leq 0$, $\lambda^{n'}(w) - \lambda(w) \leq \epsilon$ par convergence uniforme et on vient juste de prouver que $\lambda(w_0) - \lambda^{n'}(w^{n'}) < 2\epsilon$. Comme ceci est vrai pout tout ϵ, on a pour tout w, $\lambda(w_0) \leq \lambda(w)$, ce qui montre que $w_0 \in W^*$.
4. finalement, on suppose que $d(w^n, W^*)$ ne converge pas vers 0. Alors il existe un réel positif ϵ et une sous-suite, $w^{n'}$ tel que $d(w^{n'}, W^*) > \epsilon$ pour chaque n'. Mais $w^{n'}$ est encore une suite de minimiseurs dans un ensemble compact et a donc un point d'accumulation dans W^* ce qui est impossible car $d(w^{n'}, W^*) > \epsilon$.

□

5.3.3 Discussion

Ce théorème apporte une réponse directe au problème de consistance. Il montre en effet que l'estimation des paramètres optimaux est statistiquement valide. La formulation du théorème est quelque peu technique, car la fonction l modélise à la fois le perceptron fonctionnel (excepté les intégrales des neurones fonctionnels) et la fonction d'erreur. Si on note c la fonction d'erreur, on peut définir dans le cas d'un perceptron fonctionnel à une couche cachée et à valeurs réelles, la fonction l comme suit :

$$l(z, t, w_o) = c\left(\sum_{k=1}^{K} a_k T(b_k + z_k), t\right)$$

où $w_o = (a_1, b_1, \ldots, a_K, b_K) \in \mathbb{R}^{2K}$. Grâce à cette définition, on a donc :

$$c(H(w, g), t) = l\left(\int F_1(w_h^1, x)g(x)d\mu(x), \ldots, \int F_K(w_h^K, x)g(x)d\mu(x), t, w_o\right)$$

$\lambda^n(w_h, w_o)$ est donc l'erreur empirique du perceptron multi-couches fonctionnel sur les données restreintes, et $\lambda(w_h, w_o)$ est l'erreur théorique que l'on cherche à minimiser. La signification de ce théorème est que si le nombre de fonctions croît vers l'infini, les paramètres estimés convergent presque sûrement vers les vrais paramètres optimaux.

Si on étudie à présent les différentes hypothèses utilisées dans le théorème 5, on voit que son application au perceptron multi-couches fonctionnel ne pose pas de problème.

- Dans le cas où chaque F_k est un perceptron multi-couches numérique, la continuité de la fonction d'activation implique la continuité en w et la mesurabilité en x. Dans le cas de modèles linéaires généralisés, les fonctions de base doivent être mesurable en x (ce qui est toujours vérifié par définition).
- Dans le cas des modèles linéaires généralisés, la majoration $|F_k(w, x)| \leq d_k(x)$ est vérifiée si on suppose que les fonctions de base appartiennent à $L^q(\mu)$ (ce qui est toujours vérifié par définition). Dans le cas du perceptron multi-couches, on peut supposer que la mesure μ est finie (hypothèse nécessaire au théorème 3 d'approximation universelle), et que de plus la fonction d'activation est continue et bornée (condition vérifiée par les fonctions d'activation habituelles comme la tangente hyperbolique). Sous ces hypothèses, il existe une constante M telle que pour tout w et pour tout x, $|F_k(w, x)| \leq M$. La fonction constante M appartient à $L^q(\mu)$.

– Si l'on suppose que la fonction d'activation du perceptron multi-couches fonctionnel est continue, la fonction l est alors continue en (z, w_o) et mesurable en (z, t).
– La majoration de la fonction l, i.e. $|l(z, t, w_o)| \leq d'(t)$ est obtenue en supposant que la fonction d'activation du perceptron multi-couches fonctionnel est continue et bornée (la sortie du réseau est alors bornée), et en imposant une condition plus forte sur la variable aléatoire T. Si l'on suppose que la fonction d'erreur est quadratique, on peut par exemple imposer à T d'avoir un moment d'ordre 2.

5.4 Connaisance limitée des fonctions

5.4.1 Cadre probabiliste

Comme rappelé dans l'introduction, dans presque toute application d'analyse de données fonctionnelles, les fonctions ne sont connues qu'en un nombre fini de points d'évaluation, i.e. pour une fonction g donnée, on a à notre disposition un nombre fini de couples entrée/sortie (x_j, y_j), où $y_j = g(x_j)$. Une manière naturelle de modéliser cette connaissance consiste à supposer que la suite de ces points d'évaluation $(x_j)_{j\in\mathbb{N}^*}$ est la réalisation d'une suite de variables aléatoires $(X_j)_{j\in\mathbb{N}^*}$. Les X_j sont indépendantes identiquement distribuées, définies sur $(\Omega, \mathcal{A}, P)$ et à valeurs dans $\mathcal{X}$. On note P_X la mesure de probabilité induite par $X = X_1$. Si l'évaluation de la fonction g au point x_j est entâchée d'un bruit de mesure, on aura $y_j = g(x_j) + \varepsilon_j$. Les ε_j sont les réalisations des variables aléatoires $\mathcal{E}_j$. Les $\mathcal{E}_j$ sont indépendantes identiquement distribuées avec $E(\mathcal{E}_j)$=0. On suppose que les $(X_j)_{j\in\mathbb{N}}$ et les $(\mathcal{E}_j)_{j\in\mathbb{N}}$ sont des variables aléatoires indépendantes.

Sous ces hypothèses on a :

$$\int fgdP_X = E(f(X)g(X))$$

Quand $E(f(X)g(X)) < \infty$ et $E(f(X)\mathcal{E}) < \infty$, la loi forte des grands nombres implique que :

$$\frac{1}{m}\sum_{j=1}^{m} f(X_j)(g(X_j) + \mathcal{E}_j) \rightarrow_{m\rightarrow\infty} E(f(X)g(X))\ P - p.s.$$

Dans ce cas, la sortie exacte du neurone fonctionnel $N(g)$ peut être remplacée

par une réalisation $N(g)_m$ de la variable aléatoire suivante :

$$\widehat{N}(g)_m = T\left(b + \frac{1}{m}\sum_{j=1}^{m} f(X_j)(g(X_j) + \mathcal{E}_j)\right)$$

Si T est continue, $\widehat{N}(g)_m$ converge (presque sûrement) vers la sortie exacte $N(g)$ du neurone fonctionnel. Cette approximation reste bien sûr valable pour des perceptrons multi-couches arbitraires (entrées multiples et/ou neurones fonctionnels paramétriques).

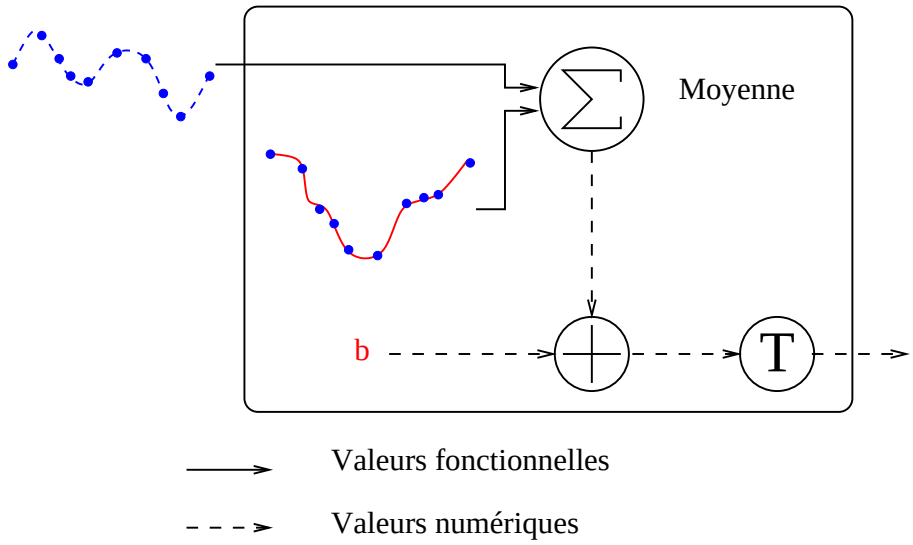


FIG. 5.1 – Approximation du neurone fonctionnel

On a par exemple dans le cas d'un perceptron fonctionnel à une couche cachée et une sortie réelle, la variable aléatoire suivante :

$$\widehat{H}(g)_m = \sum_{k=1}^{K} a_k T\left(b_k + \frac{1}{m}\sum_{j=1}^{m} f_k(X_j)(g(X_j) + \mathcal{E}_j)\right) \tag{5.2}$$

5.4.2 Conséquences pour l'approximation universelle

Comme on a pu le voir dans le chapitre 4 (corollaires 2 et 3), toute fonction continue définie sur $\mathcal{K}$, un sous-ensemble compact de $L^p(\mu)$, et à valeurs réelles peut être approchée à une précision donnée par un perceptron fonctionnel H à une couche cachée utilisant un nombre fini de paramètres. Cette approximation

a lieu au sens de la norme infinie, i.e. si $F \in C(\mathcal{K}, \mathbb{R})$ et $\varepsilon > 0$ sont fixés, il existe H tel que pour tout $g \in \mathcal{K}$, $|F(g) - H(g)| < \varepsilon$.

Dans la section précédente, on a vu que dans la pratique la sortie exacte du perceptron multi-couches fonctionnel $H(g)$ est remplacée par une réalisation de la variable aléatoire $\widehat{H}(g)_m$. Si toutes les fonctions d'activation du réseau sont continues, $\widehat{H}(g)_m$ converge presque sûrement vers $H(g)$ pour un g fixé.

En combinant ces deux résultats, il existe donc pour une fonction g donnée, un entier M (*qui dépend de* g) tel que pour tout $m \geq M$, on ait $|F(g) - \widehat{H}(g)_m| < 2\varepsilon$ presque sûrement. Ce résultat montre donc qu'en tout point (en toute fonction g), la fonction F peut être approchée arbitrairement près par un perceptron multi-couches fonctionnel basé sur une connaissance empirique des fonctions d'entrée : il suffit pour cela que chaque fonction d'entrée g soit connue avec suffisamment de précision. On peut noter que ce résultat de convergence, contrairement à celui de l'approximation universelle, n'est pas uniforme : la vitesse de convergence dépend de la fonction g, i.e. du point d'évaluation de la fonction F (convergence simple).

5.4.3 Consistance

Dans le cas où les fonctions ne sont pas connues de manière parfaite, le premier résultat de consistance (théorème 5) ne peut plus être appliqué. Afin de prouver que l'estimation des paramètres optimaux reste cependant statistiquement valide, on énonce ici un second résultat de consistance plus spécifique. Ce résultat, adapté à cette connaissance empirique à deux niveaux (nombre fini de fonctions, nombre fini de points d'évaluation), impose cependant des hypothèses plus fortes sur la nature des fonctions d'entrée (fonctions continues définies sur un ensemble compact).

Afin d'assurer que l'échantillonnage puisse être distinct d'une fonction d'entrée à une autre, i.e. que la position des points d'évaluation et leur nombre puissent varier d'une fonction à une autre, on considère une suite de suites de variables aléatoires $(X_j^i)_{i,j\in\mathbb{N}^*}$ à valeurs dans $\mathcal{X}$ (un espace métrique compact muni de sa tribu borélienne). Ces variables aléatoires sont indépendantes identiquement distribuées (on note $X = X_1^1$ et P_X la mesure induite sur $\mathcal{X}$). Pour un i fixé, la suite $(X_j^i)_{j\in\mathbb{N}^*}$ est associée à la fonction aléatoire G^i.

Selon le même principe, on considère la suite de suites de variables aléatoires $(\mathcal{E}_j^i)_{i,j\in\mathbb{N}^*}$. Ces variables aléatoires sont indépendantes identiquement distribuées (on note $\mathcal{E} = \mathcal{E}_1^1$). On suppose que $E(\mathcal{E}) = 0$ et que $E(|\mathcal{E}|^p) < \infty$. On considère de plus que les $\mathcal{E}_j^i$ et les X_j^i sont indépendantes.

On a alors le théorème suivant :

Théorème 6. *Soit $\mathcal{X}$ un espace métrique compact muni de sa tribu borélienne. Soit $(\Omega, \mathcal{A}, P)$ un espace probabilisé sur lequel est définie une suite de variables aléatoires X_j^i indépendantes identiquement distribuées et à valeurs dans $\mathcal{X}$. On note P_X la mesure induite sur $\mathcal{X}$ et $X = X_1^1$. Soient p et q des exposants conjugués avec $p \geq 1$. Soit $\mathcal{E}_j^i$ une suite de variables aléatoires indépendantes identiquement distribuées à valeurs dans $\mathbb{R}$. On suppose que $E\left(\mathcal{E}_j^i\right) = 0$ et $E\left(\left|\mathcal{E}_j^i\right|^p\right) < \infty$ (on note $\mathcal{E} = \mathcal{E}_1^1$). On suppose que les X_j^i et les $\mathcal{E}_j^i$ sont indépendantes.*

Soient K un entier, et $F_1, \ldots, F_K$, K régesseurs paramétriques tels que pour chaque k :

1. *F_k est une fonction de $W^k \times \mathcal{X}$ vers $\mathbb{R}$.*
2. *W^k est un ensemble compact.*
3. *pour chaque $x \in \mathcal{X}$, $F_k(., x)$ est une fonction continue de W^k vers $\mathbb{R}$.*
4. *pour chaque $w_h^k \in W_h^k$, $F_k(w_h^k, .)$ est une fonction mesurable de $\mathcal{X}$ vers $\mathbb{R}$.*
5. *il existe une fonction mesurable d_k de $\mathcal{X}$ vers $\mathbb{R}$ qui appartient à $L^q(P_X)$ et telle que pour chaque $w \in W_h^k$ et pour tout $x \in \mathcal{X}$, $\left|F^k(w, x)\right| \leq d_k(x)$.*

On note $W_h = W_h^1 \times \ldots \times W_h^K$.

Soit $\mathcal{K}$ un sous-ensemble compact de $C(\mathcal{X}, \mathbb{R})$. Soit G^i une suite de variables aléatoires fonctionnelles définies sur $(\Omega, \mathcal{A}, P)$ et à valeurs dans $\mathcal{K}$. Soit $\mathcal{T}$ un espace métrique muni de sa tribu borélienne, et soit T^i une suite de variables aléatoires définies sur $(\Omega, \mathcal{A}, P)$ et à valeurs dans $\mathcal{T}$. On suppose que les couples de variables aléatoires (G^i, T^i) sont indépendants et identiquement distribués. On note $G = G^1$ et $T = T^1$.

Soit l une fonction de $\mathbb{R}^K \times \mathcal{T} \times W_o$ dans $\mathbb{R}$, où W_o est un ensemble compact. On suppose que :

1. *pour chaque $t \in \mathcal{T}$, $l(., t, .)$ est une fonction uniformément continue de $\mathbb{R}^K \times W_o$ vers $\mathbb{R}$.*
2. *pour chaque $w_o \in W_o$, $l(., ., w_o)$ est une fonction mesurable de $\mathbb{R}^K \times \mathcal{T}$ vers $\mathbb{R}$.*
3. *il existe une fonction mesurable d' de $\mathcal{T}$ vers $\mathbb{R}$ telle que $|l(z, t, w_o)| \leq d'(y)$ pour tout z et w_o*
4. *$E(d'(T)) < \infty$*

Pour chaque $\omega \in \Omega$, on définit :

$$\lambda_m^n(w_h, w_o)(\omega) = \frac{1}{n}\sum_{i=1}^{n} l\left(\frac{1}{m_i}\sum_{j=1}^{m_i} F_1\left(w_h^1, X_j^i(\omega)\right)\left(G^i(\omega)\left(X_j^i(\omega)\right) + \mathcal{E}_j^i(\omega)\right),\right.$$
$$\left. \ldots, \frac{1}{m_i}\sum_{j=1}^{m_i} F_K\left(w_h^K, X_j^i(\omega)\right)\left(G^i(\omega)\left(X_j^i(\omega)\right) + \mathcal{E}_j^i(\omega)\right), T^i(\omega), w_o\right)$$

avec $m = \inf_{1\le i\le n} m_i$. On définit

$$\lambda(w_h, w_o) = E\left(l\left(\int F_1\left(w_h^1, x\right) G(x) dP_X(x), \ldots, \int F_K\left(w_h^K, x\right) G(x) dP_X(x), T, w_o\right)\right)$$

Alors pour chaque n et m et ω, il existe une solution $w_m^n(\omega)$ au problème

$$\min_{w\in W_h\times W_o} \lambda_m^n(w_h, w_o)(\omega)$$

Si W^* est l'ensemble des minimiseurs de $\lambda(w_h, w_o)$, alors pour presque tout $\omega \in \Omega$:

$$\lim_{n\to\infty}\lim_{m\to\infty} d(w_m^n(\omega), W^*) = 0$$

Démonstration. Premièrement, on considère la fonction h définie sur $(W_h^k \times \mathcal{K}) \times \mathcal{X}$ par :

$$h((w, g), x) = F_k(w, x)g(x)$$

On applique à h la loi forte des grands nombres uniforme (corollaire 4). On vérifie donc les hypothèses suivantes :

1. $h((w, g), .)$ est mesurable pour tout (w, g), grâce aux hypothèses sur F_k et sur $\mathcal{K}$.
2. $h(., x)$ est continue pour tout x, grâce aux hypothèses sur F_k et sur $\mathcal{K}$. En effet, pour tout x, la fonction $g \to g(x)$ est continue sur $\mathcal{K}$.
3. la fonction $g \to \sup_{x\in\mathcal{X}} |g(x)|$ est continue de $\mathcal{K}$ to $\mathbb{R}^+$, et donc il existe $M \in \mathbb{R}^+$ tel que pour tout $g \in \mathcal{K}$, $\sup_{x\in\mathcal{X}} |g(x)| \le M$. On a donc $|F_k(w, x)g(x)| \le Md_k(x)$. Par hypothèse, Md_k est mesurable.
4. $E(d_k(X)M) \le (\int d_k^q dP_X)^{\frac{1}{q}} (\int M^p dP_X)^{\frac{1}{p}} \le M(\int d_k^q)^{\frac{1}{q}} < \infty$ (car $P_X(\mathcal{X}) = 1$ et $d_k \in L^q(P_X)$).

Grâce au corollaire 4, on a donc

$$\sup_{(w,g)\in W_h^k\times\mathcal{K}} \left|\frac{1}{m_i}\sum_{j=1}^{m_i} F_k(w, X_j^i)g(X_j^i) - E(F_k(w, X)g(X))\right| \to^{p.s.}_{m_i\to\infty} 0 \qquad (5.3)$$

Pour prendre en compte le bruit, on applique une seconde fois la loi forte des grands nombres uniforme à la fonction ϕ de $W_h^k \times (\mathcal{X} \times \mathbb{R})$ définie par :

$$\phi(w,(x,t)) = F_k(w,x)t$$

Le corollaire 4 s'applique car :

1. $\phi(w,.)$ est mesurable pour tout w grâce aux hypothèses sur F_k.
2. $\phi(.,(x,t))$ est continue pour tout (x,t) grâce aux hypothèses sur F_k.
3. $|F_k(w,x)t| \leq d_k(x)|t|$ pour chaque w (et tout (x,t)) et $d_k(x)|t|$ est mesurable.
4. $E\left(d_k(X)|\mathcal{E}|\right) \leq E(d_k(X)^q)^{\frac{1}{q}} E(|\mathcal{E}|^p)^{\frac{1}{p}} < \infty$

On a donc :

$$\sup_{(w,g)\in W_h^l \times \mathcal{K}} \left| \frac{1}{m_i} \sum_{j=1}^{m_i} F_k(w, X_j^i)\mathcal{E}_j^i - E(F_k(w,X)\mathcal{E}) \right| \rightarrow^{p.s.}_{m_i \rightarrow \infty} 0 \qquad (5.4)$$

De plus, par indépendance, $E(F_k(w,X)\mathcal{E}) = E(F_k(w,X))E(\mathcal{E}) = 0$. Donc en combinant l'équation 5.3 et l'équation 5.4, on a :

$$\sup_{(w,g)\in W_h^l \times \mathcal{K}} \left| \frac{1}{m_i} \sum_{j=1}^{m_i} F_k(w, X_j^i) \left(g(X_j^i) + \mathcal{E}_j^i\right) - E(F_k(w,X)g(X)) \right| \rightarrow^{p.s.}_{m_i \rightarrow \infty} 0 \qquad (5.5)$$

Dans un deuxième temps, on applique la loi forte des grands nombres uniforme à la fonction :

$$h(w_h, w_o, g, t) = l\left(\int F_1(w_h^1, x)g(x)dP_X(x), \ldots, \int F_K(w_h^K, x)g(x)dP_X(x), t, w_o \right)$$

On utilise pour cela des arguments similaires au théorème 5. On voit que la fonction h est continue en (w_h, w_o), et est majorée par la fonction mesurable c, qui est d'espérance finie. Pour prouver la mesurabilité de h, on montre que la fonction qui à g élément de $C(\mathcal{X}, \mathbb{R})$ (muni de la norme infinie) associe $\int F_k(w_h^k, x)g(x)dP_X$ est continue. On a en effet $|\int F_k(w_h^k, x)g(x)dP_X - \int F_k(w_h^k, x)g'(x)dP_X| \leq \| F_k(w_h^k, .) \|_q \| g - g' \|_p \leq \| F_k(w_h^k, .) \|_q \| g - g' \|_\infty$.

On a donc :

$$\sup_{(w_h,w_o)\in W_h \times W_o} \left| \frac{1}{n} \sum_{i=1}^{n} h(w_h, w_o, G^i, T^i) - E(h(w_h, w_o, G, T)) \right| \rightarrow^{p.s.}_{n \rightarrow \infty} 0 \qquad (5.6)$$

A présent, on considère un ω pour lequel la convergence uniforme des équations 5.6 et 5.5 a lieu. Un tel ω existe presque sûrement. On pose $g^i = G^i(\omega)$ et $t^i = T^i(\omega)$. Soit ε un réel strictement positif. Selon l'équation 5.6, il existe N tel que pour chaque $n \geq N$,

$$\sup_{(w_h,w_o)\in W_h\times W_o} \left| \frac{1}{n}\sum_{i=1}^{n} h(w_h, w_o, g^i, t^i) - E(h(w_h, w_o, G, T)) \right| < \frac{\varepsilon}{2} \tag{5.7}$$

On fixe n supérieur à N. Comme l est uniformément continue en z et w, pour chaque t^i il existe $\eta^i > 0$ tel que pour chaque w, $|l(z, w, t^i) - l(z', w, t^i)| < \frac{\varepsilon}{2}$ tant que $\|z - z'\| < \eta^i$. On appelle $x^i_j = X^i_j(\omega)$ et $\varepsilon^i_j = \mathcal{E}^i_j(\omega)$. Selon l'équation 5.5, il existe M^i tel que, $m_i \geq M^i$ implique

$$\sup_{w\in W_h} \left| \frac{1}{m_i}\sum_{j=1}^{m_i} F_k(w, x^i_j)(g^i(x^i_j) + \varepsilon^i_j) - \int F_k(w, x) g^i(x) dP_X(x) \right| < \eta_i,$$

pour chaque k. On pose $M^n = sup_{i\leq n} M_i$. Pour $m \geq M^n$, on a pour chaque $i \leq n$ et pour tout (w_h, w_o) :

$$\left| l\left(\frac{1}{m}\sum_{j=1}^{m} F_1(w^1_h, x^i_j)(g^i(x^i_j) + \varepsilon^i_j), \ldots, \frac{1}{m}\sum_{j=1}^{m} F_K(w^K_h, x^i_j)(g^i(x^i_j) + \varepsilon^i_j) \right) - h(w_h, w_o, g^i, t^i) \right| < \frac{\varepsilon}{2}$$

Ce qui implique pour tout (w_h, w_o) :

$$\left| \lambda^n_m(w_h, w_o)(\omega) - \frac{1}{n}\sum_{i=1}^{n} h(w_h, w_o, g^i, t^i) \right| < \frac{\varepsilon}{2}$$

En combinant cette inégalité avec l'équation 5.7, on obtient la conclusion suivante : Pour presque tout $\omega \in \Omega$, et pour chaque $\varepsilon > 0$, il existe N tel que pour chaque $n \geq N$, il existe M^n tel que pour chaque $m \geq M^n$

$$\sup_{(w_h,w_o)\in W_h\times W_o} |\lambda^n_m(w_h, w_o)(\omega) - \lambda(w_h, w_o)| < \varepsilon$$

Pour presque tout ω, on a donc

$$\lim_{n\to\infty} \lim_{m\to\infty} \sup_{(w_h,w_o)\in W_h\times W_o} |\lambda^n_m(w_h, w_o)(\omega) - \lambda(w_h, w_o)| = 0 \tag{5.8}$$

La conclusion finale est obtenue grâce aux étapes suivantes :

On pose $h'((w_h, w_o), (g, t)) = h(w_h, w_o, g, t)$.

1. $\lambda(w_h, w_o)$ est continue. On a en effet montré que h' est continue sur (w_h, w_o) pour chaque (g, t). Pour toute suite $(w_h, w_o)_i$ qui converge vers (w_h, w_o), $h'((w_h, w_o)_i, (g, t))$ converge vers $h'((w_h, w_o), (g, t))$. De plus, h' est dominée par c. Finalement , grâce au théorème de convergence dominée, $E\left(h'((w_h, w_o)_i, (G, T))\right)$ converge vers $E\left(h'((w_h, w_o), (G, T))\right)$. On a donc la continuité de λ.
2. on considère $\lambda_m^n(w_h, w_o)(\omega)$. Cette fonction est continue sur l'ensemble compact $W = W_h \times W_o$, et donc atteind son minimum en $w_m^n(\omega)$. On peut donc définir de cette façon une fonction de Ω vers W.
3. On considère des réalisations g^i, t^i et x_j^i pour lesquelles la convergence uniforme a lieu (equation 5.8). Pour chaque n, $\left(w_m^n(\omega)\right)_{m\in\mathbb{N}^*}$ a au moins un point d'accumulation (car W est un ensemble compact), $w_0^n(\omega)$, et il existe une sous-suite $w^n_{\phi_n(m)}$ qui converge vers ce point (i.e., $\lim_{m\to\infty} w^n_{\phi_n(m)}(\omega) = w_0^n(\omega)$). La suite $\left(w_0^n(\omega)\right)_{n\in\mathbb{N}^*}$ a au moins un point d'accumulation, $w_0^0(\omega)$ et il existe une sous-suite $w_0^{\psi(n)}(\omega)$ qui converge vers ce point. On a :

   $$\lim_{n\to\infty}\lim_{m\to\infty} w^{\psi(n)}_{\phi_{\psi(n)}(m)}(\omega) = w_0^0(\omega)$$

   Soit ϵ un nombre réel positif arbitraire. λ est uniformément continue sur W et donc il existe η tel que $|w' - w| < \eta$ implique $|\lambda(w) - \lambda(w')| < \epsilon$.
   Par convergence uniforme, il existe N tel que pour chaque $n > N$, il existe M^n tel que $m > M^n$ implique pour tout $w \in W$, on a $|\lambda_m^n(\omega)(w) - \lambda(w)| < \epsilon$. De plus, on peut choisir N et M^n tel que $n > N$ et $m > M^n$ implique $|w^{\psi(n)}_{\phi_{\psi(n)}(m)}(\omega) - w_0^0| < \eta$. Donc, $n > N$ et $m > M^n$ implique $\left|\lambda^{\psi(n)}_{\phi_{\psi(n)}(m)}(\omega)\left(w^{\psi(n)}_{\phi_{\psi(n)}(m)}(\omega)\right) - \lambda(w_0)\right| < 2\epsilon$.
   Comme $w^{\psi(n)}_{\phi_{\psi(n)}(m)}$ est un minimiseur de $\lambda^{\psi(n)}_{\phi_{\psi(n)}(m)}$, on a alors l'inégalité suivante $\lambda^{\psi(n)}_{\phi_{\psi(n)}(m)}(\omega)(w^{\psi(n)}_{\phi_{\psi(n)}(m)}(\omega)) - \lambda^{\psi(n)}_{\phi_{\psi(n)}(m)}(\omega)(w) \leq 0$. Par convergence uniforme, on a $\lambda^{\psi(n)}_{\phi_{\psi(n)}(m)}(\omega)(w) - \lambda(w) < \epsilon$. Donc, $\lambda(w_0^0) - \lambda(w) \leq 3\epsilon$. Comme ceci est vrai pour tout ϵ, on a pour tout w, $\lambda(w_0^0) \leq \lambda(w)$, ce qui montre que $w_0^0 \in W^*$.
4. supposons que $d(w_m^n(\omega), W^*)$ ne converge pas vers 0. Alors il existe un nombre réel positif ϵ et une sous-suite, $w^{\psi'(n)}_{\phi'_{\psi'(n)}(m)}(\omega)$ tel que pour chaque m et n, on ait $d(w^{\psi'(n)}_{\phi'_{\psi'(n)}(m)}(\omega), W^*) > \epsilon$. Comme $w^{\psi'(n)}_{\phi'_{\psi'(n)}(m)}(\omega)$ est encore une suite de minimiseurs dans un ensemble compact, il existe donc un point d'accumulation dans W^*. Ce qui est impossible car $d(w^{\psi'(n)}_{\phi'_{\psi'(n)}(m)}(\omega), W^*) > \epsilon$.

□

5.4.4 Discussion

Par rapport au premier résultat de consistance (théorème 5), trois modifications majeures ont été apportées :

1. Premièrement, les fonctions d'entrée sont à présent des fonctions continues à support compact, de plus ces fonctions appartiennent à un compact de $C(\mathcal{X}, \mathbb{R})$. Dans le premier théorème au contraire, la seule hypothèse faite sur la nature des fonctions était leur appartenance à l'espace fonctionnel $L^p(P_X)$.

2. La seconde modification apportée par rapport au premier théorème est la modification de l'erreur empirique $\lambda_m^n(w_h, w_o)$ afin qu'elle prenne en compte les deux niveaux d'approximation. Si l'on reprend les notations de la section 5.3.3, où c est la fonction d'erreur, on a donc à présent :

$$c(H(w, g^i)_m, t^i) = \frac{1}{n}\sum_{i=1}^{n} l\left(\frac{1}{m}\sum_{j=1}^{m} F_1(w_h^1, x_j^i)(g^i(x_j^i) + \varepsilon_j^i), \ldots, \right.$$
$$\left. \frac{1}{m}\sum_{j=1}^{m} F_K(w_h^K, x_j^i)(g^i(x_j^i) + \varepsilon_j^i), t^i, w_o\right)$$

3. On peut remarquer que par rapport au théorème 5, la fonction l doit à présent vérifier une hypothèse plus forte : elle doit être uniformément continue en z et en w_o. Cette hypothèse ne pose pas de problème pratique dans le cas du perceptron multi-couches fonctionnel. En effet, la fonction d'activation du réseau peut être choisie uniformément continue et bornée sur $\mathbb{R}$ tout entier. Les hypothèses restantes sont identiques à celles du théorème 5 (voir la discussion de la section 5.3.3).

La signification du théorème 6 peut se résumer ainsi : si le nombre de fonctions et le nombre de points d'évaluation pour chaque fonction croissent vers l'infini, les paramètres estimés convergent alors presque sûrement vers les vrais paramètres optimaux. La seule réelle limitation de ce résultat est que la convergence a lieu séquentiellement en m puis en n. En effet, si l'on cherche à atteindre un niveau de précision donné, alors le nombre minimum M de points d'évaluation nécessaire pour atteindre cette précision dépend du nombre n de fonctions.

5.5 Mise en oeuvre pratique

5.5.1 Introduction

Dans cette section, on s'intéresse au coût algorithmique qu'implique l'évaluation du perceptron multi-couches fonctionnel, ainsi qu'à la complexité de ce modèle (nombre de paramètres ajustables). On cherche notamment à effectuer une comparaison avec l'approche classique dans le cas où le perceptron multi-couches numérique peut être utilisé pour traiter les fonctions d'entrée (cas d'une discrétisation identique pour chaque fonction).

Cette étude montre qu'une distinction importante doit être faite selon la nature des régresseurs paramétriques utilisés pour représenter les fonctions de poids. En effet, comme on pourra le voir par la suite, l'utilisation de modèles pseudo-linéaires permet une réduction considérable du temps de calcul nécessaire à l'évaluation du réseau. De plus, on verra que dans ce cas, le perceptron multi-couches fonctionnel est en tout point semblable à un perceptron multi-couches numérique.

5.5.2 Cas général

Coût algorithmique

Comme on a pu le voir dans le chapitre 3, à l'exception de la première couche du réseau (la couche fonctionnelle), le perceptron multi-couches fonctionnel est en tout point semblable à un perceptron multi-couches numérique. On peut donc restreindre notre étude à la comparaison du neurone fonctionnel et du neurone numérique.

Dans le cas de l'approche standard, le neurone numérique nécessite l'évaluation d'un simple produit scalaire dans $\mathbb{R}^n$ (l'espace des individus d'entrée) :

$$N(x) = T(b + w.x)$$

Dans le cas de l'approche fonctionnelle, on remplace l'évaluation d'une intégrale par le calcul d'une moyenne empirique :

$$N(g) = T\left(b + \frac{1}{m}\sum_{j=1}^{m} F(w, x_j)y_j\right)$$

où y_j est l'évaluation (avec ou sans bruit) de la fonction g au point x_j.

On voit donc que le coût de ce calcul est proportionnel au nombre, m, de points d'évaluation nécessaire pour décrire la fonction d'entrée g, ainsi qu'au coût d'évaluation du régresseur paramétrique F.

Dans le cas où chaque fonction d'entrée est décrite par un nombre important de points d'évaluation, on voit que la phase d'estimation des paramètres optimaux du réseau peut nécessiter d'importantes ressources. En effet, les algorithmes d'optimisation non-linéaire imposent l'évaluation de la fonction à minimiser (et de son gradient) en un très grand nombre de points.

Dans la section suivante, on montre que contrairement à l'approche numérique classique, dans l'approche fonctionnelle, le nombre de paramètres ajustables du modèle ne dépend pas du nombre de points d'évaluation, m.

Complexité

Le modèle proposé peut être comparé avec un perceptron numérique traditionnel, si l'on fait une hypothèse restrictive sur le mode d'échantillonnage des fonctions d'entrée. On suppose en effet que toutes les fonctions d'entrée sont échantillonnées en des points d'évaluation identiques x_j : chaque fonction g est alors décrite par un vecteur $(g_1, \ldots, g_M)$ de dimension M.

Sous ces hypothèses, la sortie du perceptron fonctionnel H peut être réécrite sous la forme suivante :

$$H(a, b, w, g)_M = \sum_{k=1}^{K} a_k T \left(b_k + \frac{1}{M} \sum_{j=1}^{M} w_k(x_j) g_j \right)$$

Si l'on considère à présent un perceptron U à une couche cachée avec la même fonction d'activation. Etant donné un vecteur d'entrée $g \in \mathbb{R}^M$, la sortie du réseau est donnée par la formule suivante :

$$U(\alpha, \beta, p, g) = \sum_{k=1}^{K} \alpha_k T \left(\beta_k + \sum_{j=1}^{M} p_{kj} g_j \right)$$

Les deux modèles sont en fait très proches. La principale différence est que dans l'approche traditionnelle, les poids (p_{kj}) sont choisis librement, alors que dans l'approche fonctionnelle, ils sont remplacés par la sortie d'un régresseur paramètrique $(\frac{1}{M} w_k(.))$. Cette différence a d'importantes conséquences sur la complexité de ces deux modèles.

Dans l'approche traditionnelle, le seul meta-paramètre qui peut être modifié est le nombre de neurones cachés K. Le nombre total de paramètres numériques correspond à $K(2 + M)$. Ce nombre est directement proportionnel au nombre d'entrées. Dans l'approche fonctionnelle, on peut choisir librement le nombre de neurones cachées ainsi que la complexité des régresseurs paramétriques internes. Si l'on suppose que chaque régresseur paramétrique est en fait un perceptron à

une couche cachée, le nombre total de paramètres numériques est $K(2+3m)$, où m est le nombre de neurone cachée dans les perceptrons internes. Le principal avantage dans cette approche est que le nombre de paramètres n'est pas lié à la dimension du vecteur d'entrée.

On peut interpréter le modèle fonctionnel comme une sorte de technique de régularisation. Plutôt que de permettre aux poids de prendre des valeurs arbitraires, on les contraint à appartenir à un ensemble de valeurs calculables par une fonction.

Cette contrainte sur les poids du réseau peut d'un certain point de vue être comparée à la technique de "partage faible des poids" ("soft weight sharing" voir [58]). Cette méthode a pour but d'améliorer les capacités de généralisation du perceptron multi-couches en imposant à l'ensemble des paramètres ajustables du réseau de prendre des valeurs voisines. Plus précisément, l'auteur considère une loi de probabilité dont la densité est donnée par un mélange de gaussiennes. Les paramètres de ce modèle (centre et variance des gaussiennes, ainsi que les coefficients du mélange) sont variables, et sont optimisées lors de la phase d'apprentissage du modèle. L'auteur impose alors aux paramètres du réseau d'être vraisemblables pour cette loi de probabilité en ajoutant un terme de pénalité à la fonction de coût. On voit donc que dans les deux approches (fonctionnelles/partage faible des poids), la contrainte sur les paramètres du réseau est donnée par un modèle paramétrique. Cependant, alors que dans l'approche fonctionnelle, les différents poids du réseau sont liés par une contrainte fonctionnelle, dans la technique de partage faible des poids, la contrainte est de type probabiliste. De plus, dans l'approche fonctionnelle, la contrainte s'excerce sur l'ensemble des poids d'un même neurone, alors que dans la technique de partage faible des poids, la contrainte est globale sur l'ensemble des poids du réseau.

5.5.3 Régresseurs paramétriques pseudo-linéaires

Simplification

On a vu dans la section précédente que le coût d'évaluation du perceptron multi-couches fonctionnel était proportionnel au nombre, m, de points d'évaluation utilisés pour décrire les fonctions d'entrée. Dans cette section, on montre que l'utilisation de modèles pseudo-linéaires pour représenter les fonctions de poids permet de s'affranchir de cette dépendance : l'évaluation du perceptron multi-couches fonctionnel peut alors être réalisée efficacement.

Le neurone paramétrique calcule la fonction suivante :

$$N(g) = T\left(b + \int F(w,.)gd\mu\right)$$

dans le cas où le régresseur F s'exprime sous la forme d'une combinaison linéaire de fonctions $F(w,.) = \sum_{q=1}^{Q} w_q\psi_q$, le calcul de l'intégrale peut être réécrit sous la forme suivante :

$$\int F(w,.)gd\mu = \sum_{q=1}^{Q} w_q \int g\psi_q d\mu = w^T\beta$$

avec le vecteur $\beta = (\int g\psi_q d\mu)_{1\leq q\leq Q}$ et $w \in \mathbb{R}^Q$.

Le point important à noter est qu'à présent chaque intégrale $\int g\psi_q d\mu$ peut être calculée une fois pour toute et indépendamment du perceptron multi-couches fonctionnel. Le pré-calcul du vecteur β, associé à la fonction g, permet alors de simplifier le calcul réalisé par le neurone fonctionnel : l'évaluation du neurone fonctionnel nécessite seulement le calcul d'un produit scalaire dans $\mathbb{R}^Q$.

Dans le cas pratique où l'on remplace l'intégrale par une moyenne empirique, le pré-calcul de β s'effectue de manière similaire :

$$\beta = \left(\frac{1}{m}\sum_{j=1}^{m} y_j\psi_q(x_j)\right)_{1\leq q\leq Q}$$

où y_j est l'évaluation (avec ou sans bruit) de la fonction g au point x_j.

Coût algorithmique

Grâce au pré-calcul du vecteur β, le temps de calcul qu'implique l'évaluation du neurone fonctionnel est à présent proportionnel au nombre, Q, de fonctions de base du régresseur paramétrique F, et est complètement dissocié du nombre de points d'évaluation, m.

Dans la pratique, l'ordre de grandeur de m est généralement nettement supérieur à celui de Q. En effet, il arrive fréquemment que l'échantillonnage des fonctions d'entrée soit réalisé à haute résolution (m grand), alors que le problème à résoudre ne nécessite l'extraction de caractéristisques qu'à basse résolution (Q petit). Dans ce cas, le coût d'évaluation d'un neurone fonctionnel utilisant un modèle pseudo-linéaire est nettement plus faible que celui d'un neurone fonctionnel utilisant un modèle non-linéaire (la non-linéarité de la fonction de poids interdisant tout pré-calcul, le coût algorithmique reste proportionnel à m).

Lien avec le perceptron multi-couches numérique

Comme on a pu le voir dans la section précédente, on associe à chaque fonction d'entrée le vecteur β de $\mathbb{R}^Q$. Le neurone fonctionnel calcule alors la fonction suivante :

$$N(g) = T(b + w.\beta)$$

où w est le vecteur paramètre du modèle linéaire F, et où b est un nombre réel.

Grâce à l'étape de pré-calcul, le neurone fonctionnel calcule de manière similaire au neurone numérique classique, le produit scalaire entre son vecteur d'entrée β et son vecteur de poids w. On voit donc que le perceptron multi-couches fonctionnel est en tout point identique à un perceptron multi-couches classique.

Dans le cas où les fonctions de poids sont représentées par des modèles pseudo-linéaires, le perceptron multi-couches fonctionnel ne nécessite donc pas le développement de logiciels spécifiques : les bibliothèques de réseaux de neurones existantes peuvent être utilisées pour traiter des données fonctionnelles.

Remarque sur les théorèmes de consistance

Comme on vient de le voir, le pré-calcul du vecteur β a des conséquences importantes sur la manière dont la sortie du réseau fonctionnel peut être calculée (utilisation d'un simple perceptron multi-couches numérique). On peut remarquer que ce pré-calcul a de même des conséquences théoriques importantes dans la démonstration des théorèmes de consistance (5 et 6).

En effet, on a vu dans le début de ce chapitre la nécessité d'utiliser une loi des grands nombres uniforme adaptée à la nature fonctionnelle des variables aléatoires traitées. Le corollaire 4 apporte une réponse à ce problème en adaptant la loi forte des grands nombres uniforme aux espaces de dimension infinie.

Dans le cas où les fonctions de poids sont représentées par des modèles pseudo-linéaires, ce résultat n'est plus nécessaire : la démonstration des théorèmes de consistance (5 et 6) peut utiliser le résultat standard de White [76]. En effet, on peut associer à la variable aléatoire fonctionnelle G, la variable aléatoire vectorielle $\widehat{\beta}$ de la manière suivante :

$$\widehat{\beta} = \left(\int G \psi_q d\mu \right)_{1 \leq q \leq Q}$$

L'erreur théorique devient alors :

$$\lambda(w_h, w_o) = E\left(l(w_h^1.\widehat{\beta}, \ldots, w_h^K.\widehat{\beta}, T, w_o) \right)$$

où les w_h^k sont des vecteurs de $\mathbb{R}^Q$.

5.5.4 Liens avec des travaux antérieurs

Dans leur article, Chen et Chen [20] propose un modèle qui est très similaire à l'approche par perceptron numérique standard. En effet, toutes les fonctions sont évaluées en des points identiques choisis grâce au théorème 4 de [20]. Plus précisément, le théorème 4 est un résulat d'approximation universel, qui prouve juste l'existence de ces points d'évaluation, et n'indique rien quant à leur valeur. De plus, ce modèle, à l'instar de l'approche par perceptron numérique, ne permet pas de fixer la complexité du réseau de manière indépendante du nombre de points d'évaluation. On voit donc que le modèle fonctionnel proposé ici est beaucoup plus général.

5.6 Conclusion

Dans ce chapitre, on a pu voir comment la définition du perceptron multi-couches fonctionnel pouvait être étendue afin de prendre en compte la nature échantillonnée des fonctions d'entrée. Dans un deuxième temps, on s'est intéressé au problème de l'estimation des paramètres optimaux du modèle. Deux résultats théoriques ont été démontrés : le premier prouve la consistance du modèle dans le cas d'une connaissance parfaite des fonctions d'entrée, le second montre que cette consistance reste valide dans le cas d'une connaissance discrète de ces fonctions (échantillonnage). Dans la fin de ce chapitre, on a montré que le coût d'évaluation de ce modèle fonctionnel pouvait être pénalisant dans le cas où les fonctions de poids sont représentées par des modèles non-linéaires. Dans le chapitre suivant, on propose une nouvelle approche basée sur une pré-projection des fonctions d'entrée (technique habituelle en Analyse de Données Fonctionnelles). Comme on pourra le voir, cette méthode présente l'avantage de réduire de manière importante le temps nécessaire à l'apprentissage du perceptron multi-couches fonctionnel dans le cas où les fonctions de poids sont représentées par des modèles non-linéaires.

Chapitre 6

Approche par projection

6.1 Introduction

Dans le chapitre précédent, on a vu que la définition du perceptron multi-couches fonctionnel pouvait aisément être adaptée afin de prendre en compte le problème de l'échantillonnage des fonctions d'entrée. Pour cela, le calcul de l'intégrale du neurone a été remplacé par l'évaluation d'une moyenne empirique. Dans ce chapitre, on présente une seconde méthode basée sur la projection des fonctions d'entrée sur une base de fonctions choisie au préalable (B-spline, séries de Fourier, etc). La représentation régularisée[1] ainsi obtenue est alors présentée au perceptron multi-couches fonctionnel afin qu'il calcule la sortie correspondante (la fonction d'entrée initiale n'intervenant plus lors de ce calcul).

La projection des fonctions d'entrée est une technique habituellement utilisée en Analyse de Données Fonctionnelles. Son intérêt par rapport à un traitement direct (voir chapitre précédent) est de permettre un filtrage des entrées du perceptron multi-couches fonctionnel. Ce filtrage permet généralement d'améliorer la qualité d'estimation des paramètres optimaux du réseau. En effet, lors de la phase d'apprentissage, le modèle fonctionnel cherche à approcher la relation déterministe existante entre les fonctions d'entrée et les variables à prédire. Dans le cas d'une connaissance bruitée des fonctions d'entrée, cette phase d'estimation se révele plus difficile : le modèle doit s'affranchir du caractère bruité des fonctions d'entrée, afin d'obtenir une modélisation correcte du phénomène observé. Dans le cas de l'approche par projection, l'étape de régularisation permet de présenter au modèle fonctionnel une entrée partiellement débruitée.

Un parallèle peut être fait entre cette étape de régularisation et les tech-

[1] le terme "régularisé" est ici considéré au sens large, car certaines bases de projection n'ont pas la propriété d'être régulière (ondelettes).

niques de pré-traitement communément utilisées en analyse de données. Comme expliqué dans Bishop [10] (et comme on a pu le voir dans le chapitre 2), le pré-traitement des données et l'extraction de caractéristiques poursuivent trois buts distincts : la réduction de la dimension de l'espace des données, l'amélioration des performances grâce à l'introduction d'une connaissance *a priori*, et enfin le traitement de données corrompues (données manquantes).

On retrouve dans l'étape de projection ces trois caractéristiques :

- l'opérateur de projection effectue une réduction de la dimension de l'espace d'entrée : les éléments de l'espace fonctionnel, espace de dimension infinie, sont projetés afin d'obtenir une représentation régularisée appartenant à un espace de dimension finie ;
- le choix de la base de projection permet d'introduire une connaissance *a priori* sur les fonctions d'entrée : si les fonctions d'entrée présentent une structure périodique, le choix des séries de Fourier permet de modéliser efficacement cette caractéristique. Dans le cas où toutes les fonctions d'entrée partagent une même forme[2] (par exemple des variations importantes localisées identiquement pour toutes les fonctions d'entrée), le choix d'une base adaptée de B-spline permet de modéliser plus finement ce comportement (on augmente le nombre de fonctions de base aux endroits de forte variation) ;
- dans le cadre de l'Analyse de Données Fonctionnelles, la connaissance des fonctions d'entrée sous la forme d'un échantillonnage peut être interprétée comme une forme particulière de données manquantes : chaque fonction n'est connue qu'en un nombre fini de points d'évaluation, en tout autre point la valeur de la fonction n'est pas accessible. Comme on pourra le voir dans la section traitant de la projection empirique (section 6.4.2), l'étape de projection apporte une solution à ce problème, en estimant la valeur de chaque fonction en tout point.

Dans la première partie de ce chapitre, on présente l'étape de projection, ainsi que le perceptron multi-couches fonctionnel basé sur cette approche. On montre dans la section 6.3 que le résultat d'approximation universelle, énoncé dans le chapitre 4, reste valide malgré l'étape de projection. La dernière partie de ce chapitre est consacrée aux problèmes de consistance.

[2] le terme "forme" est considéré ici au sens large, et peut désigner la fonction f ainsi que ses f',f''...

6.2 Approche par projection

6.2.1 Etape de projection

Le but de l'étape de projection est d'obtenir une représentation régularisée des fonctions d'entrée en projetant chacune d'elles sur l'espace vectoriel engendré par un ensemble de fonctions choisi au préalable.

On introduit la définition suivante :

Définition 10. Soit X un espace Hilbertien séparable, une base topologique de X est une famille dénombrable totale et libre d'éléments de X.

On suppose que les fonctions d'entrée appartiennent à l'espace fonctionnel[3] $L^2(\mu)$, où μ est une mesure σ-finie définie sur $\mathbb{R}^n$, et on munit cet espace d'une base topologique $\Phi = (\phi_p)_{p\in\mathbb{N}^*}$.

On considère alors Π_P l'opérateur de projection sur l'espace vectoriel engendré par les P premiers éléments de la base topologique $(vect(\phi_1, \ldots, \phi_P))$. Pour toute fonction $g \in L^2(\mu)$, la projection de g sur $vect(\phi_1, \ldots, \phi_P)$ est donnée par la relation suivante :

$$\Pi_P(g) = \sum_{p=1}^{P} \left(\int g\phi_p d\mu \right) \phi_p$$

On rappelle que l'opérateur de projection Π_P est Lipschitzien de rapport 1, et donc continu de $L^2(\mu)$ dans $L^2(\mu)$ (voir [68]).

6.2.2 Perceptron Multi-couches Fonctionnel basé sur une étape de projection

Comme expliqué dans la section précédente, dans l'approche par projection, chaque fonction d'entrée est tout d'abord projetée sur un espace de dimension finie, afin d'en obtenir une représentation régularisée. Cette représentation est alors soumise au perceptron multi-couches fonctionnel, afin qu'il calcule la sortie correspondante. Plus précisément, si l'on considère H un perceptron multi-couches fonctionnel défini sur $L^2(\mu)$, on évalue $H(\Pi_P(g))$, au lieu de $H(g)$.

Dans le cas d'un perceptron fonctionnel à une couche cachée et à valeurs réelles, la sortie du modèle fonctionnel est calculée de la manière suivante :

$$Ho\Pi_P(g) = \sum_{k=1}^{K} a_k T \left(b_k + \int f_k \Pi_P(g) d\mu \right)$$

[3] $L^2(\mu)$ est un espace Hilbertien séparable.

où $f_k \in L^2(\mu)$, et où a_k et b_k sont des nombres réels.

On voit donc que contrairement à l'approche directe (voir chapitre 5), l'approche par projection est composée de deux étapes distinctes : premièrement, l'étape de projection des fonctions d'entrée, qui est effectuée préalablement et de manière indépendante de l'évaluation du modèle fonctionnel, puis le calcul de la sortie du réseau, qui ne dépend plus des fonctions d'entrée initiales.

6.2.3 Approche paramétrique

L'utilisation de régresseurs paramétriques pour représenter les fonctions de poids permet d'obtenir un perceptron multi-couches fonctionnel paramétré par un nombre fini de paramètres numériques. Dans cette section, on va voir que de manière identique à l'approche directe (voir chapitre précédent), une distinction importante doit être faite selon la nature de ces régresseurs paramétriques (modèles pseudo-linéaires/modèles non-linéaires).

Cas général

Dans le cas où l'on ne fait pas d'hypothèses particulières sur la nature des régresseurs paramétriques, le neurone fonctionnel calcule la fonction suivante :

$$N(g) = T\left(b + \int F(w,.)\Pi_P(g)d\mu\right)$$

où F est un régresseur paramétrique de vecteur poids w.

Grâce à l'étape préalable de projection, la fonction $\Pi_P(g)$ est connue sous une forme analytique : cette fonction est donc évaluable en tout point. On voit donc que contrairement à l'approche directe, il n'est plus nécessaire d'approcher l'intégrale interne au neurone par une moyenne empirique (la précision de cette approximation dépendait du nombre de points d'évaluation de la fonction g, et n'était donc pas sous le contrôle de l'utilisateur). Dans l'approche par projection, l'intégrale $\int F(w,.)\Pi_P(g)d\mu$ peut être calculée de manière approchée[4] à une précision fixée préalablement. Ce calcul peut être réalisé par les techniques classiques de quadrature, ou par une approche de type Monte-Carlo. Dans le cas d'une méthode par quadrature, on réalise le calcul suivant :

$$\int F(w,.)\Pi_P(g)d\mu \simeq \sum_{j=1}^{M} \gamma_j F(w,x_j)\Pi_P(g)(x_j)$$

[4]et dans certains cas de manière exacte

où M est le nombre de points de discrétisation (notés x_j) nécessaire au calcul de l'intégrale (γ_j sont des coefficients qui dépendent du mode de quadrature).

Il est important de noter que le nombre M peut être choisi indépendamment du nombre, m, de points d'évaluation de la fonction g : la précision d'évaluation de l'intégrale est à présent un paramètre ajustable du modèle[5].

Régresseurs paramétriques pseudo-linéaires

Comme dans l'approche directe (voir chapitre précédent), la représentation des fonctions de poids par des modèles pseudo-linéaires permet une simplification du calcul réalisé par le perceptron multi-couches fonctionnel.

On considère une seconde base topologique $\Psi = (\psi_q)_{q\in\mathbb{N}^*}$ de $L^2(\mu)$, et on impose aux fonctions de poids d'appartenir à l'espace vectoriel engendré par les Q premiers éléments de cette base ($vect(\psi_1, \ldots, \psi_Q)$). Si l'on considère une fonction de poids de la forme $F(w,.) = \sum_{q=1}^{Q} w_q\psi_q$, chaque intégrale s'exprime alors de la manière suivante :

$$\int F(w,.)\Pi_P(g)d\mu = \sum_{q=1}^{Q}\sum_{p=1}^{P} w_q \int \phi_p\psi_q d\mu \int g\phi_p d\mu = w^T\Lambda\beta = w^T\widetilde{\beta}$$

où $\Lambda = (\int \phi_p\psi_q d\mu)_{q,p}$, $\beta = (\int g\phi_p d\mu)_p$ et $\widetilde{\beta} = \Lambda\beta$.

Dans cette expression, chaque intégrale $\int g\phi_p d\mu$ est calculée pendant l'étape de projection (plus précisément, on calcule une valeur approchée de $\int g\phi_p d\mu$ comme expliqué dans la section 6.4.2). Les intégrales $\int \phi_p\psi_q d\mu$ sont indépendantes du vecteur de poids w, ainsi que des fonctions d'entrée, on peut donc les calculer préalablement à toute évaluation du perceptron multi-couches fonctionnel. Selon les bases utilisées pour représenter les fonctions de poids et les fonctions d'entrée, le calcul de $\int \phi_p\psi_q d\mu$ peut être effectué soit de manière exacte, soit de manière approchée[6] en utilisant une méthode de quadrature ou une méthode de type Monte Carlo.

Finalement, comme Λ et β sont des constantes, le résultat du produit matriciel $\widetilde{\beta} = \Lambda\beta$ est lui aussi une constante, et peut donc être évalué préalablement à toute évaluation du perceptron multi-couches fonctionnel. Grâce aux pré-calcul de $\widetilde{\beta}$, l'évaluation de chaque intégrale est donc réduite à l'évaluation d'un simple produit scalaire dans $\mathbb{R}^Q$: $w^T\widetilde{\beta}$.

[5] M peut être choisi petit afin de réduire le coût d'évaluation du modèle. Ceci s'effectue bien sûr au détriment de la précision.

[6] avec une précision fixée au préalable

6.2.4 Liens avec le perceptron multi-couches numériques

Comme on a pu le voir dans la section précédente, la représentation linéaire des fonctions de poids permet d'associer à chaque fonction d'entrée g le vecteur $\widetilde{\beta}$ de $\mathbb{R}^Q$. Le neurone fonctionnel calcule alors la fonction suivante :

$$N(g) = T(b + w.\widetilde{\beta})$$

Ce calcul correspond à celui réalisé par un neurone numérique classique. On voit donc que de manière identique à l'approche directe, le perceptron multi-couches fonctionnel basé sur une étape de projection est en fait un perceptron multi-couches numérique.

Si l'on examine le calcul réalisé par le neurone fonctionnel, on peut s'interroger sur la nécessité de calculer le vecteur $\widetilde{\beta}$. La phase de projection ayant extrait un nombre réduit de caractéristiques pour chaque fonction d'entrée (le vecteur β), on cherche à savoir en quoi la présentation directe du vecteur β à un perceptron multi-couches numérique diffère de l'approche proposée ci-dessus (présentation du vecteur $\widetilde{\beta}$).

Il est tout d'abord intéressant de noter que l'approche fonctionnelle (présentation de $\widetilde{\beta}$), permet de dissocier la complexité du modèle (le nombre de paramètres ajustables) de la dimension de l'espace vectoriel sur lequel on projette les fonctions d'entrée. En effet, le nombre Q (la complexité des modèles pseudo-linéaires) peut être choisi indépendamment de P (la dimension de l'espace de projection), ce qui permet de réduire le nombre de paramètres ajustables du modèle : le modèle est plus souple.

Si l'on réécrit à présent le calcul réalisé par le neurone fonctionnel, on a $w^T\widetilde{\beta} = w^T\Lambda\beta = \widetilde{w}^T\beta$, où $\widetilde{w} = \Lambda^T w$. On voit donc que soumettre le vecteur $\widetilde{\beta}$ à un perceptron multi-couches numérique revient à soumettre le vecteur β en imposant une contrainte sur le vecteur de poids $\widetilde{w}$ du neurone ($\widetilde{w} = \Lambda^T w$ i.e. le vecteur de poids doit appartenir à l'image de l'application Λ^T). Cette contrainte est une conséquence de la nature fonctionnelle des individus d'entrée et des fonctions de poids. Si l'on travaille directement sur β sans restreindre les poids du perceptron multi-couches numérique, le modèle obtenu est plus général qu'un perceptron multi-couches fonctionnel, et peut réaliser un plus grand nombre de fonctions. On voit donc que l'estimation des paramètres optimaux associés aux β donne une solution différente de l'estimation réalisée grâce aux $\widetilde{\beta}$. Si l'on fait l'hypothèse supplémentaire que Λ^T est surjective (i.e. Λ injective), alors les deux approches sont totalement équivalentes, car pour un vecteur $\widetilde{w}$ donné, on peut retrouver un vecteur w tel que le neurone numérique et le neurone fonctionnel calculent la même fonction.

6.2.5 Coût algorithmique

On a vu dans le chapitre précédent que la représentation des fonctions de poids par des modèles non-linéaires impliquait un temps de calcul important dans le cas de l'approche directe. Dans cette section, on montre que ce coût peut être réduit dans le cas de l'approche par projection.

Dans cette partie, on suppose que l'on cherche à évaluer la sortie du perceptron multi-couches fonctionnel pour un nombre, N, de fonctions d'entrée (avec le même vecteur poids), ce qui permet par exemple de calculer l'erreur commise par le réseau, ou encore le gradient de cette erreur. Ce calcul doit être le plus rapide possible car il apparait un très grand nombre de fois lors de la phase d'apprentissage. L'optimisation qui va être proposée permettra donc une réduction importante du temps d'apprentissage (le temps d'évaluation du modèle pour une seule fonction restant quant à lui inchangé).

On suppose pour simplifier l'explication que le nombre de points d'évaluation est identique pour toutes les fonctions d'entrée g^i. On note m ce nombre. Dans le cas de l'approche directe, on est donc amené à calculer l'expression suivante pour $1 \leq i \leq N$:

$$N(g^i) = T\left(b + \frac{1}{m}\sum_{j=1}^{m} F(w, x_j^i) y_j^i\right)$$

où y_j^i est l'évaluation (avec ou sans bruit) de la fonction g^i au point d'évaluation x_j^i.

Le coût algorithmique de ce calcul est proportionnel au nombre de fonctions d'entrée, N, au nombre de points d'évaluation, m, ainsi qu'au coût d'évaluation du régresseur paramétrique non-linéaire F. Ce coût est donc donné par[7] $C_{direct} = N * m * (C(F) + 2)$

Dans le cas de l'approche par projection, l'intégrale du neurone fonctionnel peut être réécrite sous la forme suivante :

$$\int F(w, .)\Pi_P(g^i)d\mu = \sum_{p=1}^{P} \beta_p^i \int F(w, .)\phi_p d\mu$$

où le vecteur β^i correspond aux composantes de $\Pi_P(g^i)$ sur la base tronquée $\Phi = (\phi_p)_{1 \leq p \leq P}$.

Dans ce calcul, les intégrales $\int F(w, .)\phi_p d\mu$ peuvent être calculées préalablement à l'évaluation du réseau pour les N fonctions d'entrée. Si l'on suppose que

[7] on suppose que l'addition ainsi que la multiplication de deux nombres correspond à une unité de temps.

chacune de ces intégrales est calculée au moyen d'une méthode de quadrature, on a alors :

$$\int F(w,.)\phi_p d\mu \simeq \sum_{j=1}^{M} \gamma_j F(w,x_j)\phi_p(x_j)$$

où M est le nombre de points de discrétisation (notés x_j) nécessaire au calcul de l'intégrale (γ_j sont des coefficients qui dépendent du mode de quadrature).

Le coût algorithmique de ce pré-calcul est proportionnel à P, le nombre de fonctions de base, à M, la précision de la méthode de quadrature, ainsi qu'au coût d'évaluation de F et de ϕ_p. Le calcul final nécessite de plus le calcul d'un produit scalaire dans $\mathbb{R}^P$. Le coût algorithmique de l'approche par projection est donc donné par $C_{projection} = P * M * (C(F) + C(\phi) + 3) + N * P * 2$

Comme l'intégrale est calculée au moyen d'une méthode de quadrature, les points de discrétisation sont fixés. Le calcul de $\phi_p(x_j)$ peut donc être réalisé une fois pour toute (pour tout p). On s'affranchit donc du coût $C(\phi)$. De plus, le coût d'évaluation du modèle F est généralement nettement supérieur à 3 opérations (addition/multiplication), on peut donc s'affranchir des constantes. Si l'on suppose l'ordre de grandeur de M égale à celui de m, on a alors $C_{direct} = N * m * C(F)$ et $C_{projection} = (2N + m * C(F)) * P$.

On voit donc que dans le cas de l'approche par projection, le coût total n'est plus proportionnel au produit de N et de $m * C(F)$, mais à leur somme. Comme l'ordre de grandeur de P est généralement nettement inférieur à celui de $m * C(F)$, si le nombre, N, de fonctions d'entrée est suffisamment grand, l'approche par projection est plus efficace que l'approche directe.

Dans le cas de l'expérience des cercles décrite dans le chapitre 8, on a les valeurs suivantes pour les différentes variables : $N = 100$, $m = 200$ et $P = 10$. On a donc $C_{direct} = 20000\ C(F)$ et $C_{projection} \simeq 2000\ C(F)$ (on rappelle que $C(F)$ est grand devant 1). On gagne donc dans ce cas un facteur 10.

6.2.6 Régression des fonctions d'entrée

On peut remarquer que rien n'empêche d'utiliser des modèles non-linéaires pour représenter les fonctions d'entrée. Comme précédemment, la phase de régression des fonctions d'entrée peut être effectuée préalablement à l'utilisation du perceptron multi-couches fonctionnel. Si on note R^g, la régression de la fonction d'entrée[8] g, le neurone fonctionnel calcule la fonction suivante :

$$N(g) = T\left(b + \int F(w,.)R^g d\mu\right)$$

[8]par exemple en utilisant un perceptron multi-couches numérique

Dans le où les fonctions de poids sont représentées par des modèles pseudo-linéaires, le pré-calcul des intégrales $\int \phi_q R^g d\mu$ permet de se ramener à un perceptron multi-couches numérique. En revanche, dans le cas où F est non-linéaire, le caractère non-linéaire de la régression interdit l'optimisation décrite dans la section précédente.

6.3 Approximation universelle

6.3.1 Définitions

Définition 11. Soit μ une mesure σ-finie, et soit $\mathcal{A}$ un sous-ensemble de $L^2(\mu)$. On considère $\Phi = (\phi_p)_{p\in\mathbb{N}^*}$ une base topologique de $L^2(\mu)$, et Π_P l'opérateur de projection sur le sous-espace $vect(\phi_1, \ldots, \phi_P)$. Si T est une fonction de $\mathbb{R}$ dans $\mathbb{R}$, on note $S_T^{\Pi_\Phi}(\mathcal{A})$ l'ensemble des fonctions de la forme $g \mapsto Ho\Pi_P(g)$, où $H \in S_T^{L^2(\mu)}(\mathcal{A})$, et $P \in \mathbb{N}^*$, i.e. l'ensemble des fonctions de la forme :

$$Ho\Pi_P(g) = \sum_{k=1}^{K} a_k T\left(b_k + \int f_k \Pi_P(g)\, d\mu\right)$$

où $K \in \mathbb{N}^*$, $g \in L^2(\mu)$, $f_k \in \mathcal{A}$, $P \in \mathbb{N}^*$ et a_k et b_k sont des nombres réels.

Remarque. Dans la définition ci-dessus, on a volontairement identifié $\mathcal{A}$ sous-ensemble de $L^2(\mu)^*$, avec le sous-ensemble correspondant dans $L^2(\mu)$ (identification du dual dans un espace Hilbertien).

6.3.2 Approximation universelle

Le but de cette section est d'adapter le résultat d'approximation du perceptron multi-couches fonctionnel à l'approche régularisée. On montre dans le théorème suivant que malgré l'étape de projection, le perceptron multi-couches fonctionnel est toujours capable d'approcher les fonctions de $C(\mathcal{K}, \mathbb{R})$, où $\mathcal{K}$ est un sous-ensemble compact de $L^2(\mu)$.

Théorème 7. *Soient μ une mesure Borélienne sur $\mathbb{R}^r$, et Φ une base topologique de $L^2(\mu)$. Soit $\mathcal{A}$ un sous-ensemble dense de $L^2(\mu)$. Soit T une fonction continue non polynomiale de $\mathbb{R}$ vers $\mathbb{R}$. Alors $S_T^{\Pi_\Phi}(\mathcal{A})$ est $\rho_\mathcal{K}$-intérieurement dense dans $C(\mathcal{K}, \mathbb{R})$, où $\mathcal{K}$ est un sous-ensemble compact de $L^2(\mu)$.*

Afin de démontrer ce théorème, on a besoin du résultat d'approximation uniforme suivant :

Lemme 2. *Soit ε un nombre réel strictement positif, alors il existe un entier P tel que pour tout $p \geq P$, on a $\| \Pi_p(g) - g \|_2 < \varepsilon$ pour tout $g \in \mathcal{K}$.*

Démonstration. On considère $g_0 \in \mathcal{K}$, et $K(g_0, r) = B(g_0, r) \cap \mathcal{K}$ un voisinage compact de g_0. Comme $(\phi_p)_{p \in \mathbb{N}^*}$ est une base topologique, il existe P_0 tel que pour chaque $p_0 > P_0$, on a $\| \Pi_{p_0}(g_0) - g_0 \|_2 < \varepsilon/2$. Pour chaque $g \in K(g_0, r_0)$, on a l'inégalité $\| \Pi_{p_0}(g) - g \|_2 \leq \| \Pi_{p_0}(g) - \Pi_{p_0}(g_0) \|_2 + \| \Pi_{p_0}(g_0) - g_0 \|_2 + \| g_0 - g \|_2$. Le terme du milieu est majoré par $\varepsilon/2$. Comme l'opérateur Π_{p_0} est Lipschitzien de rapport 1, on a $\| \Pi_{p_0}(g) - \Pi_{p_0}(g_0) \|_2 \leq \| g - g_0 \|_2$. On choisit r_0 afin d'avoir $\| g - g_0 \|_2 \leq \varepsilon/4$. On a donc $\|\Pi_{p_0}(g) - g\|_2 \leq \varepsilon$ pout tout $g \in B(g_0, r_0) \cap \mathcal{K}$. Comme $\mathcal{K}$ est un sous-ensemble compact, il est recouvert par un nombre fini de $K(g_i, r_i)$. On considère $P = \max P_i$, ce qui permet de conclure. □

On démontre à présent le théorème d'approximation universel :

Démonstration. On applique tout d'abord le corollaire 2 pour trouver un perceptron fonctionnel H, avec T comme fonction d'activation, qui approche uniformément F sur $\mathcal{K}$ avec la précision de $\varepsilon/2$. Comme H est une fonction continue sur $L^2(\mu)$, pour chaque $g \in K$, il existe η (dépendant de g) tel que pour chaque $g' \in B(g, \eta)$, on a $|H(g) - H(g')| \leq \varepsilon/4$. $\mathcal{K}$ est recouvert par un nombre fini de $B(g_i, \eta_i/2)$. On pose $\eta = \min \eta_i$. En utilisant le lemme 2, il existe P tel que pour chaque $g \in \mathcal{K}$, on a $\| \Pi_P(g) - g \|_2 < \eta/2$. On a donc pour chaque $g \in B(g_i, \eta_i/2)$, $\| \Pi_P(g) - g_i \|_2 \leq \| \Pi_P(g) - g \|_2 + \| g - g_i \|_2 \leq \eta_i$. On conclut que pour chaque $g \in \mathcal{K}$, il existe g_i tel que $|H(\Pi_P(g)) - H(g_i)| \leq \varepsilon/4$ et $|H(g_i) - H(g)| \leq \varepsilon/4$. Finalement pour chaque $g \in \mathcal{K}$, on a $|F(g) - H(\Pi_P(g))| \leq |F(g) - H(g)| + |H(g) - H(g_i)| + |H(g_i) - H(\Pi_P(g))| \leq \varepsilon$ □

Dans le cas de fonctions de poids pseudo-linéaires, on considère $\Psi = (\psi_q)_{q \in \mathbb{N}}$ une base topologique de $L^2(\mu)$, et $\mathcal{A}$ l'ensemble des fonctions de poids dont la forme est donnée par $F(w, .) = \sum_{q=1}^{Q} w_q \psi_q$ pour tout $Q \in \mathbb{N}^*$. Dans ce cas, le théorème montre que toute fonction de $C(\mathcal{K}, \mathbb{R})$ peut être approchée arbitrairement près par un perceptron multi-couches fonctionnel basé sur une étape de projection et utilisant un nombre fini de paramètres numériques.

6.4 Cadre probabiliste

Le but de cette section est d'adapter les deux résultats de consistance, énoncés dans le chapitre 5, à l'approche par projection. Le premier théorème montre que l'estimation des paramètres est statistiquement valide dans le cas où l'on possède une connaissance parfaite des fonctions. Puis dans le cas où les fonctions

ne sont connues que de manière limitée (échantillonnage), on introduit la notion de projection empirique, et on démontre un second résultat de consistance qui prend en compte cette connaissance discrète.

6.4.1 Connaissance parfaite des fonctions

Consistance

Théorème 8. *Soit $(\mathcal{X}, \mathcal{B}, \mu)$ un espace métrique muni de sa tribu borélienne. Soit $\Psi = (\psi_q)_{q\in\mathbb{N}^*}$ une base topologique de $L^2(\mu)$, et F un régresseur linéaire de la forme $F(w,.) = \sum_{q=1}^{Q} w_q\psi_q$ défini sur $W \times \mathcal{X}$, où W est un ensemble compact. On note $W_h = W^K$.*

Soit G^i une suite de variables aléatoires fonctionnelles définies sur $(\Omega, \mathcal{A}, P)$ et à valeurs dans $L^2(\mu)$. Soit $\mathcal{T}$ un espace métrique muni de sa tribu borélienne, et soit T^i une suite de variables aléatoires définis sur $(\Omega, \mathcal{A}, P)$ et à valeurs dans $\mathcal{T}$.

On suppose que les couples de variables aléatoires (G^i, T^i) sont indépendants et identiquement distribués. On note $G = G^1$ et $T = T^1$.

Soit l une fonction de $\mathbb{R}^K \times \mathcal{T} \times W_o$ dans $\mathbb{R}$, où W_o est un ensemble compact. On suppose que :

1. *pour chaque $t \in \mathcal{T}$, $l(.,t,.)$ est continue sur $\mathbb{R}^K \times W_o$*
2. *pour chaque $w_o \in W_o$, $l(.,.,w_o)$ est mesurable sur $\mathbb{R}^K \times \mathcal{T}$*
3. *il existe une fonction mesurable d' de $\mathcal{T}$ dans $\mathbb{R}$ telle que $|l(z,t,w_o)| < d'(t)$ pour tout z et w_o*
4. *$E(d'(T)) < \infty$*

Pour chaque $\omega \in \Omega$, on définit :

$$\lambda^n(w_h, w_o)(\omega) = \frac{1}{n}\sum_{i=1}^{n} l\Bigg(\int F\left(w_h^1, x\right)\Pi_P(G^i(\omega))(x)d\mu(x), \ldots, \int F\left(w_h^K, x\right)\Pi_P(G^i(\omega))(x)d\mu, T^i(\omega), w_o\Bigg)$$

et

$$\lambda(w_h, w_o) = E\Bigg(l\Bigg(\int F\left(w_h^1, x\right)\Pi_P(G)(x)d\mu(x), \ldots, \int F\left(w_h^K, x\right)\Pi_P(G)(x)d\mu(x), T, w_o\Bigg)\Bigg)$$

Alors pour chaque $\omega \in \Omega$ et pour chaque n, il existe une solution $w^n(\omega)$ au problème

$$\min_{w \in W_h \times W_o} \lambda^n(w_h, w_o)(\omega)$$

Si W^* est l'ensemble des minimiseurs de $\lambda(w_h, w_o)$, alors

$$\lim_{n \to \infty} d(w^n(\omega), W^*) = 0 \; P - p.s.$$

Démonstration. On montre dans un premier temps que $\sup_{w \in W} |F(w, .)|$ appartient à $L^2(\mu)$. On voit tout d'abord que cette fonction est mesurable en appliquant le lemme 1. Par majoration, on a l'inégalité $\sup_{w \in W} |F(w, x)| \leq \sup_{w \in W} \sum_{q=1}^{Q} |w_q||\psi_q(x)|$. Comme W est compact, il existe $w^* \in W$ tel que pour chaque x, $\sup_{w \in W} \sum_{q=1}^{Q} |w_q||\psi_q(x)| = \sum_{q=1}^{Q} w_q^*|\psi_q(x)|$ qui appartient à $L^2(\mu)$.

On applique la loi forte des grands nombres uniforme (corollaire 4) à la fonction :

$$h(w_h, w_o, g, t) = l\left(\int F(w_h^1, x)\Pi_P(g)(x)d\mu(x), \ldots, \int F(w_h^K, x)\Pi_P(g)(x)d\mu(x), t, w_o\right)$$

Ceci est possible pour les raisons suivantes :

1. la fonction $h'((w_h, w_o), (g, t)) = h(w_h, w_o, g, t)$ est continue en $w = (w_h, w_o)$ pour chaque $x = (g, t)$, grâce aux hypothèses sur l et sur F, et sachant que $\Pi_P(g)$ appartient à $L^2(\mu)$. En effet, la fonction $w \mapsto \int F(w, x)\Pi_P(g)(x)d\mu(x)$ est continue pour chaque g : comme F est continue en w pour chaque x, la fonction $F(w', .)\Pi_P(g)(.)$ converge simplement vers $F(w, .)\Pi_P(g)(.)$ quand w' converge vers w. De plus, $|F(w, .)\Pi_P(g)(.)|$ est dominée sur W par $\sup_{w \in W} |F(w, .)||\Pi_P(g)(.)|$, laquelle est intégrable comme le produit de deux fonctions de $L^2(\mu)$. Grâce au théorème de convergence dominée, ceci implique la continuité de $w \mapsto \int F(w, x)\Pi_P(g)(x)d\mu(x)$.
2. h' est mesurable en (g, t) pour chaque (w_h, w_o). c'est une conséquence directe des hypothèses sur l et du fait que $g \mapsto \int F(w, x)\Pi_P(g)(x)d\mu(x)$ est continue pour chaque $w \in W$ (Π_P est un opérateur continu).
3. grâce au lemme 1 appliqué à $|h'|$, la fonction

   $$c(g, t) = \sup_{(w_h, w_o) \in W_h \times W_o} |h'((w_h, w_o), (g, t))|$$

   est mesurable.
4. $E(c(G, T)) < \infty$ par hypothèse sur l.

Grâce au corollaire 4, on a donc

$$\sup_{(w_h,w_o)\in W_h\times W_o}\left|\frac{1}{n}\sum_{i=1}^{n}h(w_h,w_o,G^i,T^i)-E(h(w_h,w_o,G,T))\right|\rightarrow^{p.s.}_{n\rightarrow\infty}0 \quad (6.1)$$

La conclusion finale est obtenue de manière similaire au théorème 5. □

Remarque. Dans ce résultat de consistance, on s'est volontairement restreint au cas où les fonctions de poids sont représentées par des modèles pseudo-linéaires. Les hypothèses du théorème ont ainsi pu être adaptées à ce type de modèles. Cependant, on voit que la démonstration peut facilement être étendue au cas non-linéaire en adaptant au cadre $L^2(\mu)$ les hypothèses du théorème 5 sur les fonctions de poids.

6.4.2 Connaissance limitée des fonctions

La projection empirique

Comme expliqué dans la section 6.2.2, dans l'approche par projection, le calcul de $H(g)$ est remplacé par celui de $H(\Pi_P(g))$. Dans la pratique, la connaissance limitée des fonctions d'entrée (échantillonnage) interdit malheureusement le calcul exact de la projection $\Pi_P(g)$. C'est la raison pour laquelle on remplace la fonction $\Pi_P(g)$ par une fonction approchée : le projeté empirique.

Comme dans la section 5.4, on considère deux suites de variables aléatoires indépendantes identiquement distribuées $(X_j)_{j\in\mathbb{N}^*}$ et $(\mathcal{E}_j)_{j\in\mathbb{N}^*}$, définies sur $(\Omega,\mathcal{A},P)$. On note $X=X_1$ et $\mathcal{E}=\mathcal{E}_1$. On suppose que X et $\mathcal{E}$ sont indépendantes, et que $E(\mathcal{E})=0$ et $E((\mathcal{E})^2)=\sigma^2$.

Pour $\omega\in\Omega$, on pose $x_j=X_j(\omega)$, $\varepsilon_j=\mathcal{E}_j(\omega)$ et $y_j=g(x_j)+\varepsilon_j$. On définit alors pour chaque fonction $g\in L^2(P_X)$, le projeté empirique $\Pi_P(g)^\omega_m$ correspondant : $\Pi_P(g)^\omega_m$ est l'unique élément de $L^2(P_X)$, solution du problème suivant (au sens de Moore-Penrose [64], voir théorème 9) :

$$\Pi_P(g)^\omega_m=\arg\min_{\pi\in vect(\phi_1,\ldots,\phi_P)}\sum_{j=1}^{m}(y_j-\pi(x_j))^2$$

On a le théorème de consistance suivant :

Théorème 9. *Soit $\Phi=(\phi_p)_{p\in\mathbb{N}^*}$ une base topologique de $L^2(P_X)$. On note $\beta(g)$ et $\beta(g)^\omega_m$ les coordonnées respectives de $\Pi_P(g)$ et de $\Pi_P(g)^\omega_m$ sur l'espace vectoriel $vect(\phi_1,\ldots,\phi_P)$. On a alors pour tout $g\in L^2(P_X)$:*

$$\beta(g)^\omega_m\rightarrow_{m\rightarrow\infty}\beta(g)\ P-p.s.$$

Démonstration. Soient l et k deux fonctions, on définit $< l, k >= E(l(X)k(X))$ et $< l, k >_m^\omega = \frac{1}{m}\sum_{j=1}^m l(X_j(\omega))k(X_j(\omega))$ avec $\omega \in \Omega$. De même, on définit $< \mathcal{E}, k >= E(\mathcal{E}\, k(X))$ et $< \mathcal{E}, k >_m^\omega = \frac{1}{m}\sum_{j=1}^m \mathcal{E}_j(\omega)k(X_j(\omega))$. La matrice $(< \phi_l, \phi_k >)_{l,k}$ étant inversible, on a $\beta(g) = (< \phi_l, \phi_k >)_{l,k}^{-1}(< g, \phi_k >)_k$ et de même $\beta(g)_m^\omega = (< \phi_l, \phi_k >_m^\omega)_{l,k}^\dagger(< g + \mathcal{E}, \phi_k >_m^\omega)_k$ (où $M^\dagger$ est l'inverse de Moore-Penrose de la matrice M). Par la loi forte des grands nombres et par intersection finie, on a pour presque tout $\omega \in \Omega$, $< \phi_l, \phi_k >_m^\omega$, $< g, \phi_k >_m^\omega$ et $< \mathcal{E}, \phi_k >_m^\omega$ qui convergent respectivement vers $< \phi_l, \phi_k >$, $< g, \phi_k >$ et $< \mathcal{E}, \phi_k >$ pour tout couple (l, k). Par indépendance entre X et $\mathcal{E}$, on a $< g+\mathcal{E}, \phi_k >=< g, \phi_k > + < \mathcal{E}, \phi_k >=< g, \phi_k >$. On en déduit que $< g + \mathcal{E}, \phi_k >_m^\omega$ converge vers $< g, \phi_k >$.

Finalement, pour presque tout $\omega \in \Omega$, il existe m_0 tel que pour tout $m > m_0$, la matrice $(< \phi_l, \phi_k >_m^\omega)_{l,k}$ est non singulière. En effet, le déterminant est une fonction continue de ses paramètres, et $(< \phi_l, \phi_k >_m^\omega)_{l,k}$ converge vers $(< \phi_l, \phi_k >)_{l,k}$ qui est inversible $(\det((< \phi_l, \phi_k >)_{l,k}) \neq 0)$. On a donc pour $m > m_0$, $(< \phi_l, \phi_k >_m^\omega)_{l,k}^\dagger = (< \phi_l, \phi_k >_m^\omega)_{l,k}^{-1}$. Et par continuité de l'inverse, on a pour presque tout $\omega \in \Omega$, $(< \phi_l, \phi_k >_m^\omega)_{l,k}^\dagger$ qui converge vers $(< \phi_l, \phi_k >)_{l,k}^\dagger = (< \phi_l, \phi_k >)_{l,k}^{-1}$. Ce qui permet de conclure. □

Si l'on considère une fonction g fixée, élément de $L^2(P_X)$, ce théorème montre que pour presque tout $\omega \in \Omega$, la projection empirique $\Pi_P(g)_m^\omega$ converge en norme L^2 vers la projection réelle $\Pi_P(g)$ quand m croît vers l'infini.

Pour un perceptron multi-couches fonctionnel H, le calcul de $H(\Pi_P(g))$ est remplacé par celui de $H(\Pi_P(g)_m^\omega)$. Si la fonction d'activation de H est continue, on voit que la sortie réelle du réseau $H(\Pi_P(g))$ est approchée par la sortie calculée sur les données empiriques :

$$H(\Pi_P(g)_m^\omega) \rightarrow_{m\rightarrow\infty} H(\Pi_P(g))\ P - ps$$

Consistance

On énonce ici le second résultat de consistance, qui prend en compte la connaissance limitée des fonctions d'entrée.

Théorème 10. *Soit $\mathcal{X}$ un espace métrique compact muni de sa tribu borélienne. Soit $(\Omega, \mathcal{A}, P)$ un espace probabilisé sur lequel est défini une suite de variables aléatoires X_j^i indépendantes identiquement distribuées et à valeurs dans $\mathcal{X}$. On note P_X la mesure induite sur $\mathcal{X}$ et $X = X_1^1$. Soit $\mathcal{E}_j^i$ une suite de variables aléatoires indépendantes identiquement distribuées à valeurs dans $\mathbb{R}$. On note $\mathcal{E} = \mathcal{E}_1^1$. On suppose que $E(\mathcal{E}) = 0$ et $E\left(|\mathcal{E}|^2\right) < \infty$. On suppose que les X_j^i et les $\mathcal{E}_j^i$ sont indépendantes.*

Soit $\Psi = (\psi_q)_{q\in\mathbb{N}^*}$ *une base topologique de* $L^2(P_X)$*, et* F *un régresseur linéaire de la forme* $F(w,.) = \sum_{q=1}^{Q} w_q\psi_q$ *défini sur* $W\times\mathcal{X}$*, où* W *est un ensemble compact. On note* $W_h = W^K$*.*

Soit $\mathcal{K}$ *un sous-ensemble compact de* $C(\mathcal{X},\mathbb{R})$*. Soit* G^i *une suite de variables aléatoires fonctionnelles définies sur* $(\Omega,\mathcal{A},P)$ *et à valeurs dans* $\mathcal{K}$*. Soit* $\mathcal{T}$ *un espace métrique muni de sa tribu borélienne, et soit* T^i *une suite de variables aléatoires définis sur* $(\Omega,\mathcal{A},P)$ *et à valeurs dans* $\mathcal{T}$*.*

On suppose que les couples de variables aléatoires (G^i,T^i) *sont indépendants et identiquement distribués. On note* $G = G^1$ *et* $T = T^1$*.*

Soit l *une fonction de* $\mathbb{R}^K\times\mathcal{T}\times W_o$ *dans* $\mathbb{R}$*, où* W_o *est un ensemble compact. On suppose que :*

1. *pour chaque* $t\in\mathcal{T}$*,* $l(.,t,.)$ *est uniformément continue sur* $\mathbb{R}^K\times W_o$
2. *pour chaque* $w_o\in W_o$*,* $l(.,.,w_o)$ *est mesurable sur* $\mathbb{R}^K\times\mathcal{T}$
3. *il existe une fonction mesurable* d' *de* $\mathcal{T}$ *dans* $\mathbb{R}$ *telle que* $|l(z,t,w_o)| < d'(t)$ *pour tout* z *et* w_o
4. $E(d'(T)) < \infty$

Pour chaque $\omega\in\Omega$*, on définit :*

$$\lambda_m^n(w_h,w_o)(\omega) = \frac{1}{n}\sum_{i=1}^{n} l\Bigg(\int F\left(w_h^1,x\right)\Pi_P(G^i(\omega))_m^\omega(x)dP_X,\ldots,\int F\left(w_h^k,x\right)\Pi_P(G^i(\omega))_m^\omega(x)dP_X,T^i(\omega),w_o\Bigg)$$

et

$$\lambda(w_h,w_o) = E\left(l\left(\int F\left(w_h^1,x\right)\Pi_P(G)(x)dP_X,\ldots,\int F\left(w_h^k,x\right)\Pi_P(G)(x)dP_X,T,w_o\right)\right)$$

Alors pour chaque $\omega\in\Omega$ *et pour chaque* n *et* m*, il existe une solution* $w_m^n(\omega)$ *au problème*

$$\min_{w\in W_h\times W_o}\lambda_m^n(w_h,w_o)(\omega)$$

Si W^* *est l'ensemble des minimiseurs de* $\lambda(w_h,w_o)$*, alors*

$$\lim_{n\to\infty}\lim_{m\to\infty} d(w_m^n(\omega),W^*) = 0\ P-p.s.$$

Afin de prouver ce théorème, on a besoin du résultat de convergence uniforme suivant :

Théorème 11. *Sous les hypothèses du théorème 10, pour presque tout $\omega \in \Omega$, on a pour tout $g \in \mathcal{K}$:*

$$\|\Pi_P(g)_m^\omega - \Pi_P(g)\|_2 \to_{m\to\infty} 0$$

Démonstration. On utilise les notations du théorème 9. Pour presque tout $\omega \in \Omega$, $(< \phi_l, \phi_k >_m^\omega)_{l,k}^\dagger$ converge vers $(< \phi_l, \phi_k >)_{l,k}^{-1}$ (toujours selon le théorème 9).

On considère les fonctions h_k définie sur $\mathcal{K} \times (\mathcal{X} \times \mathbb{R})$ par :

$$h_k(g, (x,t)) = (g(x) + t)\phi_k(x)$$

On applique à h_k la loi forte des grands nombres uniforme (corollaire 4). On vérifie donc les hypothèses suivantes :

1. $h_k(g, .)$ est mesurable pour tout g, grâce aux hypothèses sur ϕ_k et sur g.
2. $h_k(., (x,t))$ est continue pour tout (x,t), car la fonction $g \to g(x)$ est continue.
3. $\sup_{g\in\mathcal{K}} |h_k(g, (x,t))|$ est mesurable, grâce au lemme 1.
4. la fonction $g \to \sup_{x\in\mathcal{X}} |g(x)|$ est continue sur le compact $\mathcal{K}$. Il existe donc $M \in \mathbb{R}^+$, tel que pour tout $g \in \mathcal{K}$, $\sup_{x\in\mathcal{X}} |g(x)| \leq M$. Donc $\sup_{g\in\mathcal{K}} |h_k(g, (x,t))| \leq \sup_{g\in\mathcal{K}} |g(x)||\phi_k(x)| + |t||\phi_k(x)| \leq M|\phi_k(x)| + |t||\phi_k(x)|$. Comme $E(M|\phi_k(X)|) \leq (E(M)^2)^{\frac{1}{2}}(E(\phi(X))^2)^{\frac{1}{2}} < \infty$, et que $E(|\mathcal{E}||\phi_k(X)|) \leq (E(\mathcal{E})^2)^{\frac{1}{2}}(E(\phi_k(X))^2)^{\frac{1}{2}} < \infty$. On conclut que $E(\sup_{g\in\mathcal{K}} |h_k(g, (X, \mathcal{E}))|) < \infty$.

Grâce au corollaire 4, on a donc :

$$\sup_{g\in\mathcal{K}} \left| \frac{1}{m} \sum_{j=1}^{m} h_k(g, (X_j, \mathcal{E}_j)) - E(h_k(g, (X, \mathcal{E}))) \right| \to_{m\to\infty}^{p.s.} 0$$

Par indépendance entre X et $\mathcal{E}$, on a $E(h_k(g, (X, \mathcal{E}))) = E(g(X)\phi_k(X))$. Par intersection finie, pour presque tout $\omega \in \Omega$, on a $(< \phi_l, \phi_k >_m^\omega)_{l,k}^\dagger$ qui converge vers $(< \phi_l, \phi_k >)_{l,k}^{-1}$, et pour tout $g \in \mathcal{K}$, $(< g + \mathcal{E}, \phi_k >_m^\omega)_k$ qui converge vers $(< g, \phi_k >)_k$. On conclut que pour presque tout $\omega \in \Omega$, on a pour toute fonction $g \in \mathcal{K}$, $\beta(g)_m^\omega \to_{m\to\infty} \beta(g)$. □

Remarque. Dans le cas unidimensionnel, on peut énoncer un théorème de convergence similaire (voir Abraham et al. [1]), où l'hypothèse de compacité de l'ensemble des fonctions est remplacée par l'hypothèse plus faible suivante :

Théorème 12. *Soit $\mathcal{G}$ l'ensemble des fonctions continues à variations bornées définies sur l'intervalle $[a, b]$ et à valeurs dans $\mathbb{R}$. Pour presque tout $\omega \in \Omega$, on a $\sup_{g\in\mathcal{G}} \|\Pi_P(g)_m^\omega - \Pi_P(g)\|_2 \to_{m\to\infty} 0$.*

Le point important à noter dans ces deux résultats est que l'on cherche à avoir pour presque tout $\omega \in \Omega$ une propriété de convergence simple pour tout g. Dans les deux théorèmes, on a un résultat plus fort, car presque sûrement la convergence a lieu uniformément en g. Les points d'évaluation sont cependant identiques pour toutes les fonctions g.

On démontre à présent le théorème :

Démonstration. Premièrement, on démontre que pout presque tout $\omega \in \Omega$, on a pour chaque $g \in \mathcal{K}$, l'intégrale $\int F(w,x)\Pi_P(g)^\omega_m(x)dP_X$ qui converge uniformément sur W vers $\int F(w,x)\Pi_P(g)(x)dP_X$. C'est une simple conséquence du théorème 11, et du fait que $\sup_{w\in W} \parallel F(w,.) \parallel_2$ est fini. En effet, on a l'inégalité suivante : $\sup_{w\in W} \parallel F(w,.) \parallel_2 \leq \parallel \sup_{w\in W} |F(w,.)| \parallel_2$. Or $\sup_{w\in W} |F(w,.)|$ appartient à l'espace $L^2(P_X)$ (voir théorème 8). On conclut que pour presque tout $\omega \in \Omega$, on a pour chaque $g \in \mathcal{K}$:

$$\sup_{w\in W} \left| \int F(w,x)\Pi_P(g)^\omega_m dP_X - \int F(w,x)\Pi_P(g)dP_X \right| \rightarrow_{m\rightarrow\infty} 0 \qquad (6.2)$$

Dans un second temps, on applique la loi forte des grands nombres uniforme à la fonction :

$$h(w_h, w_o, g, t) = l\left(\int F(w_h^1, x)\Pi_P(g)(x)dP_X(x), \ldots, \int F(w_h^K, x)\Pi_P(g)(x)dP_X(x), t, w_o \right)$$

On utilise pour cela des arguments similaires au théorème 8. On constate que la fonction h est continue en (w_h, w_o), et est majorée par la fonction mesurable c, qui est d'espérance finie. Pour prouver la mesurabilité de h, on montre que la fonction qui à g élément de $C(\mathcal{X}, \mathbb{R})$ (muni de la norme infinie) associe $\int F(w_h^k, x)\Pi_P(g)(x)dP_X$ est continue. On a en effet $| \int F(w_h^k, x)\Pi_P(g)(x)dP_X - \int F(w_h^k, x)\Pi_P(g')(x)dP_X | \leq \parallel F(w_h^k, .) \parallel_2 \parallel \Pi_P(g) - \Pi_P(g') \parallel_2 \leq \parallel F(w_h^k, .) \parallel_2 \parallel g - g' \parallel_\infty$.

On a donc

$$\sup_{(w_h,w_o)\in W_h\times W_o} \left| \frac{1}{n}\sum_{i=1}^{n} h(w_h, w_o, G^i, T^i) - E(h(w_h, w_o, G, T)) \right| \rightarrow^{p.s.}_{n\rightarrow\infty} 0 \qquad (6.3)$$

A présent, on considère un ω pour lequel la convergence uniforme des équations 6.3 et 6.2 a lieu. Un tel ω existe presque sûrement. On appelle $g^i = G^i(\omega)$ et $t^i = T^i(\omega)$. Soit ε un réel strictement positif. Selon l'équation 6.3, il existe N tel que pour chaque $n \geq N$,

$$\sup_{(w_h,w_o)\in W_h\times W_o} \left| \frac{1}{n}\sum_{i=1}^{n} h(w_h, w_o, g^i, t^i) - E(h(w_h, w_o, G, T)) \right| < \frac{\varepsilon}{2} \qquad (6.4)$$

On fixe n supérieur à N. Comme l est uniformément continue en z et w, pour chaque t^i il existe $\eta^i > 0$ tel que pour chaque w, $|l(z,w,t^i) - l(z',w,t^i)| < \frac{\varepsilon}{2}$ tant que $\|z - z'\| < \eta^i$. Selon l'équation 6.2, il existe M^i tel que, $m_i \geq M^i$ implique

$$\sup_{w \in W_h} \left| \int F(w,x) \Pi_P(g^i)^{\omega}_{m_i}(x) dP_X(x) - \int F(w,x) \Pi_P(g^i)(x) dP_X(x) \right| < \eta_i,$$

pour chaque k. On appelle $M^n = sup_{i \leq n} M_i$. Pour $m \geq M^n$, on a pour chaque $i \leq n$ et pour tout (w_h, w_o) :

$$\left| l\left(\int F(w_h^1, x) \Pi_P(g^i)^{\omega}_{m}(x) dP_X(x), \ldots, \int F(w_h^K, x) \Pi_P(g^i)^{\omega}_{m}(x) dP_X(x), t^i, w_o \right) - h(w_h, w_o, g^i, t^i) \right| < \frac{\varepsilon}{2}$$

Ce qui implique pour tout (w_h, w_o) :

$$\left| \lambda^n_m(w_h, w_o)(\omega) - \frac{1}{n} \sum_{i=1}^{n} h(w_h, w_o, g^i, t^i) \right| < \frac{\varepsilon}{2}$$

En combinant cette inégalité avec l'équation 6.4, on obtient la conclusion suivante : Pour presque tout $\omega \in \Omega$, et pour chaque $\varepsilon > 0$, il existe N tel que pour chaque $n \geq N$, il existe M^n tel que pour chaque $m \geq M^n$

$$\sup_{(w_h, w_o) \in W_h \times W_o} |\lambda^n_m(w_h, w_o)(\omega) - \lambda(w_h, w_o)| < \varepsilon$$

Pour presque tout ω, on a donc

$$\lim_{n \to \infty} \lim_{m \to \infty} \sup_{(w_h, w_o) \in W_h \times W_o} |\lambda^n_m(w_h, w_o)(\omega) - \lambda(w_h, w_o)| = 0 \tag{6.5}$$

La conclusion finale est obtenue de manière similaire au théorème 6. □

Remarque. De manière identique au théorème 8, on s'est volontairement limité dans ce résultat au cas où les fonctions de poids sont représentées par des modèles pseudo-linéaires. L'extension de ce théorème à des modèles non-linéaires s'effectue aisément en adaptant au cadre $L^2(P_X)$ les hypothèses du théorème 6 sur les fonctions de poids.

6.5 Conclusion

La projection préalable des fonctions d'entrée est une technique communément utilisée en Analyse de Données Fonctionnelles. Comme on a pu le voir au cours de ce chapitre, elle présente deux avantages importants par rapport à un traitement direct des fonctions d'entrée (voir chapitre prédécent) : premièrement, elle permet de s'affranchir en partie du caractère bruité des fonctions d'entrée, ce qui facilite la phase d'estimation et autorise une meilleure modélisation du phénomène observé. De plus, on a vu que son coût algorithmique[9] était moindre dans le cas d'une représentation non-linéaire des fonctions de poids. Ceci permet donc d'accelérer la phase d'apprentissage.

Dans la seconde partie de ce chapitre, on a montré que les deux propriétés théoriques importantes, démontrées dans le cas direct, se transposaient à l'approche par projection. On a en effet vu dans la section 6.3, que le perceptron multi-couches fonctionnel, basé sur une étape de projection était un approximateur universel. De plus, on a montré dans les sections 6.4.1 et 6.4.2 que l'estimation de ses paramètres était consistante dans le cas d'une connaissance parfaite des fonctions d'entrée, comme dans le cas d'une connaissance empirique. On voit donc que d'un point de vue théorique, l'approche directe et l'approche par projection sont identiques quand les fonctions d'entrée appartiennent à l'espace $L^2(\mu)$.

[9] pour une évaluation multiple du modèle

Chapitre 7

Perceptron multi-couches fonctionnel à valeurs fonctionnelles

7.1 Introduction

On a pu voir lors des chapitres précédents que le perceptron multi-couches fonctionnel était un outil souple et performant, bien adapté aux problèmes de régression fonctionnelle. En effet, dans le cas où la variable explicative X est à valeurs dans un espace de fonctions, et la variable à prédire Y est à valeurs vectorielles, ce modèle permet d'approcher arbitrairement près la fonction de régression $E(Y|X)$ (propriété de l'approximation universelle).

On s'intéresse à présent au cas où la variable à prédire Y est elle aussi à valeurs dans un espace de fonctions (par exemple $L^2(\mu)$). Ramsay et Silverman utilisent une telle modélisation dans leur livre ([63]), afin de prédire les courbes de précipitation annuelles de différentes villes canadiennes en fonction de leur courbe de température annuelle. Ils utilisent à cet effet le modèle intégral (voir chapitre 2.3.3). Une autre application naturelle d'une telle modélisation est la prévision d'un processus $(X_t)_{t\in\mathbb{R}}$ à temps continu sur un intervalle de temps δ. Comme expliqué dans Besse et Cardot [8], on définit à partir de (X_t) un processus à temps discret $(Y_n)_{n\in\mathbb{Z}}$ à valeurs dans un espace fonctionnel (par exemple $L^2(\mu)$), en découpant chaque trajectoire sur des tronçons de longueur δ. La prévision à un pas de temps nécessite alors l'estimation de l'espérance conditionnelle $E(Y_{i+1}|Y_i)$.

L'utilité de modèles régressifs à réponse fonctionnelle (et en particulier de modèles auto-régressifs fonctionnels) suggère naturellement l'adaptation du per-

ceptron multi-couches fonctionnel. Comme on pourra le voir dans la première partie de ce chapitre, l'adaptation présentée dans ce travail présente certaines similitudes avec un modèle proposé par Bishop et Legleye [9] dans un cadre complètement différent.

La première partie de ce chapitre est consacrée à la présentation de ce nouveau modèle. Par la suite, on s'intéresse à l'étude de ses propriétés théoriques : la propriété d'approximation universelle est démontrée dans la section 7.3. Dans la section 7.4, l'estimation consistante des paramètres est prouvée sous l'hypothèse d'indépendance des fonctions d'entrée[1].

7.2 Perceptron fonctionnel à valeurs fonctionnelles

7.2.1 Présentation du modèle

La construction d'un modèle à réponse fonctionnelle peut aisément être réalisée en utilisant tout modèle classique à valeurs vectorielles. Dans ce travail, nous nous limiterons volontairement à des modèles dont la sortie fonctionnelle est à valeurs dans un espace vectoriel de dimension Q finie. On sait alors que tout vecteur de $\mathbb{R}^Q$ s'identifie de manière unique avec un élément de cet espace (on considère pour cela l'isomorphisme défini grâce à une base choisie au préalable). Afin d'obtenir une réponse fonctionnelle, il suffit donc d'identifier le vecteur de sortie du modèle avec l'élément correspondant dans l'espace vectoriel. De manière plus prosaïque, on voit que l'on utilise le vecteur de sortie du modèle comme vecteur paramètre d'un modèle linéaire généralisé.

On considère μ et ν deux mesures σ-finies, et $\Psi = (\psi_q)_{q\in\mathbb{N}^*}$ une base topologique de $L^2(\nu)$. Le perceptron fonctionnel à valeurs fonctionnelles est construit en utilisant la sortie vectorielle d'un perceptron multi-couches fonctionnel afin de calculer les coefficients d'un modèle linéaire généralisé. Ce modèle utilise les Q premiers éléments de la base Ψ comme fonctions de base. On voit donc que le perceptron fonctionnel à réponse fonctionnelle est à valeurs dans l'espace vectoriel $vect(\psi_1, \ldots, \psi_Q)$.

Si on prend le cas particulier d'un perceptron fonctionnel à valeurs fonctionnels avec une couche cachée, on a :

[1] La consistance du modèle dans le cas de processus fonctionnels auto-régressifs nécessite une preuve adaptée.

$$H_{\Pi}(g) = \sum_{q=1}^{Q} \left(\sum_{k=1}^{K} a_{qk} T \left(b_k + \int f_k g d\mu \right) \right) \psi_q$$

où T est une fonction d'activation, a_{qk} et b_k sont des nombres réels, $g \in L^p(\mu)$ et $f_k \in L^q(\mu)$.

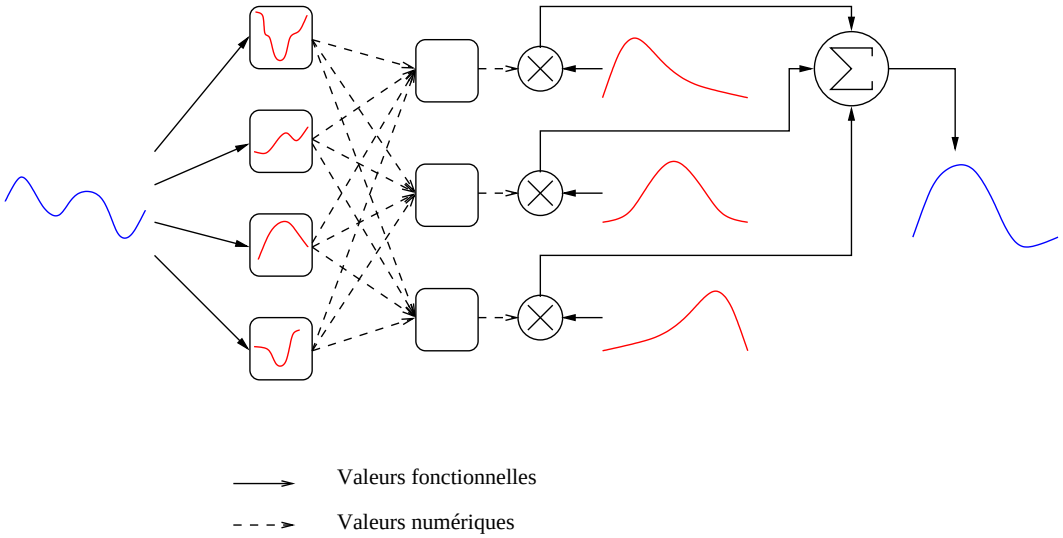


FIG. 7.1 – Perceptron multi-couches fonctionnel à réponse fonctionnelle

Comme on peut le voir dans le paragraphe suivant, il est tout à fait possible de remplacer dans la pratique le modèle linéaire généralisé par un modèle non-linéaire (par un perceptron multi-couches numérique par exemple). De manière identique, les paramètres de ce modèle sont donnés par la sortie d'un perceptron multi-couches fonctionnel.

7.2.2 Liens avec un modèle existant

Dans Bishop et Legleye [9], les auteurs s'intéressent au problème de la modélisation du lien existant entre deux variables aléatoires réelles X et Y. Si on se place dans le cadre usuel de la régression, on va supposer que ces deux variables sont liées par une dépendance fonctionnelle entâchée d'un bruit gaussien. On a donc un modèle de la forme : $Y = f(X) + \mathcal{E}$, et on s'intéresse naturellement à l'estimation de la fonction f (i.e. à l'espérance conditionnelle $E(Y|X)$).

Dans le cas le plus général, et sans connaissance *a priori* sur le phénomène à modéliser, l'hypothèse d'une dépendance fonctionnelle peut être restrictive. On peut par exemple supposer que pour une réalisation de X donnée, plusieurs

valeurs distinctes de Y sont vraisemblables. On voit donc que dans ce cas, la fonction $E(Y|X)$ donne une représentation très pauvre de la réelle structure des données.

Afin de modéliser correctement ce type de problèmes, Bishop et Legleye [9] tentent de modéliser la densité conditionnelle $p(y|x)$ pour tout x en utilisant simultanément deux modèles paramétriques. Le premier modèle (par exemple un mélange de gaussiennes) sert à modéliser la densité conditionnelle $p(y|x)$:

$$p(y|x) = \sum_{j=1}^{M} \alpha_j(x)\phi_j(y|x)$$

où ϕ_j est une gaussienne de centre $\mu_j(x)$ et d'écart-type $\sigma_j(x)$, et où les $\alpha_j(x)$ sont les coefficients du mélange.

Chaque paramètre ajustable de ce modèle (i.e. $\alpha_j(x)$, $\mu_j(x)$, $\sigma_j(x)$) peut alors être donné par l'une des sorties d'un perceptron multi-couches numérique[2]. Ce réseau calcule donc pour une entrée x donnée l'ensemble des paramètres du mélange de lois. En choisissant suffisamment de fonctions noyaux dans le modèle de mélange, et suffisamment de neurones cachés pour le perceptron multi-couches, les auteurs montrent que ce modèle peut approcher arbitrairement près la densité conditionnelle $p(y|x)$ pour tout x.

7.3 Approximation universelle

7.3.1 Définitions

Dans cette section, on introduit diverses notations.

Définition 12. Soit ν une mesure borelienne sur $\mathbb{R}^m$, et $\Psi = (\psi_q)_{q\in\mathbb{N}^*}$ une base topologique de $L^2(\nu)$. On note Iso_Q l'isomorphisme de $vect(\psi_1, \ldots, \psi_Q)$ vers $\mathbb{R}^Q$, défini grâce aux Q premiers éléments de la base Ψ. On a la relation suivante :

$$Iso_Q(g) = \left(\int g\psi_1 d\mu, \ldots, \int g\psi_Q d\mu \right)$$

avec $g \in vect(\psi_1, \ldots, \psi_Q)$.

La fonction inverse Iso_Q^{-1} est alors définie de $\mathbb{R}^Q$ vers $vect(\psi_1, \ldots, \psi_Q)$ par :

$$Iso_Q^{-1}(\beta) = \sum_{q=1}^{Q} \beta_q\psi_q$$

[2]par des changements de variables adéquates, la sortie du réseau reste dans le domaine de définition des paramètres.

avec $\beta \in \mathbb{R}^Q$.

Les deux définitions suivantes introduisent le perceptron fonctionnel à valeurs dans $\mathbb{R}^Q$, et à valeurs dans $L^2(\nu)$.

Définition 13. Soit μ une mesure de Borel sur $\mathbb{R}^n$. Soit p un nombre réel tel que $1 \leq p \leq \infty$. Soit T une fonction de $\mathbb{R}$ dans $\mathbb{R}$, et soit Q un entier. $S_T^{L^p(\mu)\to\mathbb{R}^Q}(\mathcal{A})$ est l'ensemble des perceptrons fonctionnels à une couche caché définis sur $L^p(\mu)$ et à valeurs dans $\mathbb{R}^Q$.

Définition 14. Soit μ une mesure de Borel sur $\mathbb{R}^n$. Soit p un nombre réel tel que $1 \leq p \leq \infty$. Soit T une fonction de $\mathbb{R}$ dans $\mathbb{R}$. Soit ν une mesure de Borel sur $\mathbb{R}^m$, et Ψ une base topologique de $L^2(\nu)$. On note Iso_Q l'isomorphisme de $vect(\psi_1, \ldots, \psi_Q)$ vers $\mathbb{R}^Q$, défini grâce à la base Ψ. $S_T^{L^p(\mu)\to\Pi_\Psi}(\mathcal{A})$ est l'ensemble des fonctions de la forme $g \mapsto Iso_Q^{-1} o H(g)$, où $H \in S_T^{L^p(\mu)\to\mathbb{R}^Q}(\mathcal{A})$ avec Q un entier arbitraire.

Les deux définitions suivantes introduisent le perceptron fonctionnel basé sur une étape de projection à valeurs dans $\mathbb{R}^Q$, et à valeurs dans $L^2(\nu)$.

Définition 15. Soit μ une mesure de Borel sur $\mathbb{R}^n$. Soit Φ une base topologique de $L^2(\mu)$. Soit T une fonction de $\mathbb{R}$ dans $\mathbb{R}$, et soient P et Q deux entiers. $S_T^{\Pi_\Phi\to\mathbb{R}^Q}(\mathcal{A})$ est l'ensemble des fonctions de la forme $g \mapsto Ho\Pi_P(g)$, où $H \in S_T^{L^2(\mu)\to\mathbb{R}^Q}(\mathcal{A})$.

Définition 16. Soit μ une mesure de Borel sur $\mathbb{R}^n$. Soit Φ une base topologique de $L^2(\mu)$. Soit T une fonction de $\mathbb{R}$ dans $\mathbb{R}$. Soit ν une mesure de Borel sur $\mathbb{R}^m$, et Ψ une base topologique de $L^2(\nu)$. On note Iso_Q l'isomorphisme de $vect(\psi_1, \ldots, \psi_Q)$ vers $\mathbb{R}^Q$ vers $\mathbb{R}^Q$, défini grâce à la base Ψ. $S_T^{\Pi_\Phi\to\Pi_\Psi}(\mathcal{A})$ est l'ensemble des fonctions de la forme $g \mapsto Iso_Q^{-1} o H o \Pi_P(g)$, où $H \in S_T^{L^2(\mu)\to\mathbb{R}^Q}(A)$, avec P et Q des entiers arbitraires.

7.3.2 Approximation universelle

Perceptron multi-couches fonctionnel à valeurs vectorielles

Dans les deux corollaires suivants, on montre que les perceptrons multi-couches fonctionnels à valeurs vectorielles sont des approximateurs universels.

Corollaire 5. *Si $S_T^{L^p(\mu)}(\mathcal{A})$ est $\rho_\mathcal{K}$-intérieurement (resp. $\rho_\mathcal{K}$-extérieurement) dense dans $C(\mathcal{K}, \mathbb{R})$, alors $S_T^{L^p(\mu)\to\mathbb{R}^Q}(\mathcal{A})$ est $\rho_\mathcal{K}$-intérieurement (resp. $\rho_\mathcal{K}$-extérieurement) dense dans $C(\mathcal{K}, \mathbb{R}^Q)$.*

Démonstration. Soit F un élément de $C(\mathcal{K}, \mathbb{R}^Q)$, et soient F_i ses fonctions composantes. Soit ε un réel strictement positif, alors il existe H_i élément de $S_T^{L^P(\mu)}(\mathcal{A})$ (resp. $S_T^{L^P(\mu)}(A) \cap C(\mathcal{K}, \mathbb{R})$) tel que $\| F_i - H_i \|_\infty \leq \varepsilon/n$. Soit H le perceptron fonctionnel à une couche cachée à valeurs dans $\mathbb{R}^Q$, tel que chacune de ses fonctions composantes soit donnée par H_i (voir figure 7.2). On a $\| F - H \|_\infty \leq \varepsilon$. $S^{L^p(\mu)\to\mathbb{R}^Q}(\mathcal{A})$ est donc $\rho_\mathcal{K}$-extérieurement dense dans $C(\mathcal{K}, \mathbb{R}^Q)$. Si de plus, chaque H_i appartient à $C(\mathcal{K}, \mathbb{R})$, alors H appartient à $C(\mathcal{K}, \mathbb{R}^Q)$, et $S_T^{L^p(\mu)\to\mathbb{R}^Q}(\mathcal{A})$ est $\rho_\mathcal{K}$-intérieurement dense dans $C(\mathcal{K}, \mathbb{R}^Q)$. □

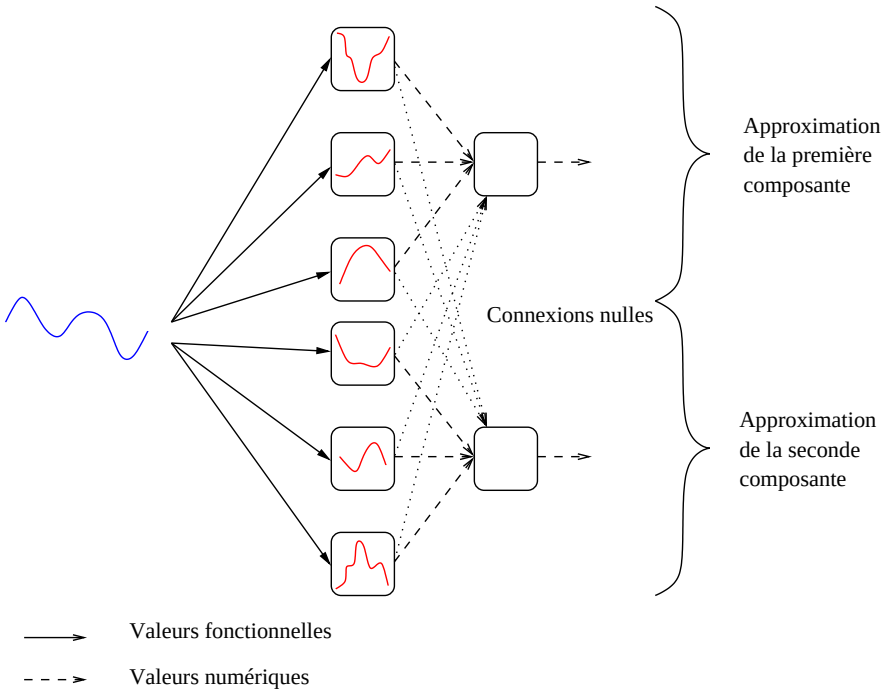


FIG. 7.2 – Perceptron multi-couches fonctionnel

De la même façon, on a le résultat suivant :

Corollaire 6. *Si $S_T^{\Pi_\Phi}(\mathcal{A})$ est $\rho_\mathcal{K}$-intérieurement dense dans $C(\mathcal{K}, \mathbb{R})$, alors $S_T^{\Pi_\Phi\to\mathbb{R}^Q}(\mathcal{A})$ est $\rho_\mathcal{K}$-intérieurement dense dans $C(\mathcal{K}, \mathbb{R}^Q)$.*

Perceptron multi-couches fonctionnel à valeurs fonctionelles

Dans les deux corollaires suivants, on montre que les perceptrons multi-couches fonctionnels à valeurs fonctionnelles sont des approximateurs universels.

Théorème 13. *Si $S_T^{L^p(\mu)}(\mathcal{A})$ est $\rho_{\mathcal{K}}$-intérieurement (resp. $\rho_{\mathcal{K}}$-extérieurement) dense dans $C(\mathcal{K},\mathbb{R})$, alors $S_T^{L^p(\mu)\rightarrow\Pi_\Psi}(\mathcal{A})$ est $\rho_{\mathcal{K}}$-intérieurement (resp. $\rho_{\mathcal{K}}$-extérieurement) dense dans $C(\mathcal{K},L^2(\nu))$.*

Démonstration. Soit F un élément de $C(\mathcal{K},L^2(\nu))$. La continuité de la fonction F implique que $F(\mathcal{K})$ est un sous-ensemble compact de $L^2(\nu)$. Il existe donc un entier Q tel que $\| F-\Pi_Q oF \|_\infty \leq \varepsilon/2$ (voir lemme 2). On considère à présent la fonction $Iso_Q o\Pi_Q oF$. C'est une fonction continue, définie sur le compact $\mathcal{K}$ et à valeurs dans $\mathbb{R}^Q$. Comme $S_T^{L^p(\mu)\rightarrow\mathbb{R}^Q}(\mathcal{A})$ est $\rho_{\mathcal{K}}$-intérieurement (resp. $\rho_{\mathcal{K}}$-extérieurement) dense dans $C(\mathcal{K},\mathbb{R}^Q)$, il existe $H\in S_T^{L^p(\mu)\rightarrow\mathbb{R}^Q}(\mathcal{A})$ tel que $\| Iso_Q o\Pi_Q oF-H \|_\infty\leq\varepsilon/2$. Par l'isomorphisme Iso_Q, on a $\| Iso_Q o\Pi_Q oF-H \|_\infty=\| \Pi_Q oF-Iso_Q^{-1}oH \|_\infty$. On conclut grâce à l'inégalité suivante : $\| F-Iso_Q^{-1}oH \|_\infty\leq\| F-\Pi_Q oF \|_\infty+\| \Pi_Q oF-Iso_Q^{-1}oH \|_\infty\leq\varepsilon$. □

De la même façon, on a le résultat suivant :

Théorème 14. *Si $S_T^{\Pi_\Phi}(\mathcal{A})$ est $\rho_{\mathcal{K}}$-intérieurement dense dans $C(\mathcal{K},\mathbb{R})$, alors $S_T^{\Pi_\Phi\rightarrow\Pi_\Psi}(\mathcal{A})$ est $\rho_{\mathcal{K}}$-intérieurement dense dans $C(\mathcal{K},L^2(\nu))$.*

7.4 Cadre probabiliste

7.4.1 Connaissance parfaite des fonctions

Consistance

Afin d'obtenir un résultat de consistance unifié pour l'approche directe et l'approche par projection, on introduit dans le théorème suivant la fonction $I(w,g)$. Cette fonction correspond dans le cas des deux approches au calcul des intégrales des neurones fonctionnels.

– dans le cas de l'approche directe, on pose :

$$I(w_h,g)=\left(\int F_1(w_h^1,.)gd\mu,\ldots,\int F_K(w_h^K,.)gd\mu\right)$$

– dans le cas de l'approche par projection, on pose :

$$I(w_h,g)=\left(\int F_1(w_h^1,.)\Pi_P(g)d\mu,\ldots,\int F_K(w_h^K,.)\Pi_P(g)d\mu\right)$$

On démontre à présent la propriété de consistance dans le cas où les fonctions d'entrée et les fonctions à prédire sont connues parfaitement :

Théorème 15. *Soit W_h un ensemble compact. Soit $I(w,g)$ une fonction définie sur $W_h \times L^p(\mu)$ et à valeurs dans $\mathbb{R}^K$. On suppose que :*

1. *pour chaque $w \in W_h$, $I(w,.)$ est une fonction mesurable de $L^p(\mu)$ vers $\mathbb{R}^K$.*
2. *pour chaque $g \in L^p(\mu)$, $I(.,g)$ est une fonction continue de W_h vers $\mathbb{R}^K$.*

Soit G^i une suite de variables aléatoires fonctionnelles définies sur $(\Omega, \mathcal{A}, P)$ et à valeurs dans $L^p(\mu)$. Soit T^i une suite de variables aléatoires définis sur $(\Omega, \mathcal{A}, P)$ et à valeurs dans $L^2(\nu)$.

On suppose que les couples de variables aléatoires (G^i, T^i) sont indépendants et identiquement distribués. On note $G = G^1$ et $T = T^1$.

Soit Ψ une base topologique de $L^2(\nu)$, et Π_Q l'opérateur de projection sur l'espace vectoriel $vect(\psi_1, ..., \psi_Q)$.

Soit l une fonction de $\mathbb{R}^K \times L^2(\nu) \times W_o$ vers $\mathbb{R}$, où W_o est un ensemble compact. On suppose que :

1. *pour chaque $t \in L^2(\nu)$, $l(.,t,.)$ est une fonction continue de $\mathbb{R}^K \times W_o$ vers $\mathbb{R}$.*
2. *pour chaque $w_o \in W_o$, $l(.,.,w_o)$ est une fonction mesurable de $\mathbb{R}^K \times L^2(\nu)$ vers $\mathbb{R}$.*
3. *il existe une fonction mesurable d' de $L^2(\nu)$ vers $\mathbb{R}$ telle que $|l(z,t,w_o)| \leq d'(t)$ pour tout z et w_o.*
4. *$E(d'(\Pi_Q(T))) < \infty$*

Pour chaque $\omega \in \Omega$, on définit :

$$\lambda^n(w_h, w_o)(\omega) = \frac{1}{n} \sum_{i=1}^{n} l\left(I(w_h, G^i(\omega)), \Pi_Q(T^i(\omega)), w_o\right),$$

On définit de plus

$$\lambda(w_h, w_o) = E\left(l\left(I(w_h, G), \Pi_Q(T), w_o\right)\right)$$

Alors pour chaque $\omega \in \Omega$ et pour chaque n, il existe une solution $w^n(\omega)$ au problème

$$\min_{w \in W_h \times W_o} \lambda^n(w_h, w_o)(\omega)$$

Si W^* est l'ensemble des minimiseurs de $\lambda(w_h, w_o)$, alors

$$\lim_{n \to \infty} d(w^n(\omega), W^*) = 0 \; P - p.s.$$

Démonstration. On applique la loi forte des grands nombres uniforme (corollaire 4) à la fonction :

$$h(w_h, w_o, g, t) = l\left(I(w_h, g), \Pi_Q(t), w_o\right)$$

Ceci est possible pour les raisons suivantes :

1. la fonction $h'((w_h, w_o), (g, t)) = h(w_h, w_o, g, t)$ est continue en $w = (w_h, w_o)$ pour chaque $x = (g, t)$, grâce aux hypothèses sur l et sur I.
2. h' est mesurable en (g, t) pour chaque (w_h, w_o). c'est une conséquence directe des hypothèses sur l et sur I, et du fait que l'opérateur Π_Q est continu de $L^2(\nu)$ dans $L^2(\nu)$.
3. grâce au lemme 1 appliqué à $|h'|$, la fonction

$$c(g, t) = \sup_{(w_h, w_o) \in W_h \times W_o} |h'((w_h, w_o), (g, t))|$$

   est mesurable.
4. $E(c(G, T)) < \infty$ par hypothèse sur l.

Grâce au corollaire 4, on a donc

$$\sup_{(w_h, w_o) \in W_h \times W_o} \left| \frac{1}{n} \sum_{i=1}^{n} h(w_h, w_o, G^i, T^i) - E(h(w_h, w_o, G, T)) \right| \rightarrow^{p.s.}_{n \rightarrow \infty} 0 \quad (7.1)$$

La conclusion finale est obtenue de manière similaire au théorème 5. □

Discussion

Le théorème de consistance ci-dessus s'applique grâce à la fonction I aux deux approches présentées dans les chapitres précédents : l'approche directe (chapitre 5), et l'approche par projection (chapitre 6).

En effet, dans le cas de l'approche directe, on rappelle que :

$$I(w_h, g) = \left(\int F_1(w_h^1, x) g(x) d\mu(x), \ldots, \int F_K(w_h^K, x) g(x) d\mu(x) \right)$$

Les hypothèses sur la fonctions I dans le théorème 15 se transposent en des hypothèses sur les régresseurs paramétriques F_k (voir les hypothèses du théorème 5).

Dans le cas de l'approche par projection, et on rappelle que :

$$I(w_h, g) = \left(\int F_1(w_h^1, x) \Pi_P(g)(x) d\mu(x), \ldots, \int F_K(w_h^K, x) \Pi_P(g)(x) d\mu(x) \right)$$

Comme ci-dessus, les hypothèses sur la fonctions I se traduisent en des hypothèses sur les régresseurs paramétriques F_k (voir les hypothèses du théorème 8 dans le cas pseudo-linéaire, et théorème 5 dans le cas non-linéaire).

De manière similaire au chapitre 5 section 5.3.3, la fonction l modélise à la fois le perceptron multi-couches fonctionnel à valeurs fonctionnelles (excepté les intégrales internes aux neurones fonctionnels) ainsi que la fonction d'erreur. Dans le cas d'un perceptron à une couche cachée et à valeurs dans $\mathbb{R}^Q$, on a :

$$l(z,t,w_o) = \left\| t - \sum_{q=1}^{Q} \left(\sum_{k=1}^{K} a_{qk} T(b_k + z_k) \right) \psi_q \right\|_2^2$$

avec $w_o = ((a_{qk})_{1\leq q\leq Q, 1\leq k\leq K}, (b_k)_{1\leq k\leq K}) \in \mathbb{R}^{(Q+1)K}$.

7.4.2 Connaissance limitée des fonctions

Consistance

De manière identique au théorème précédent, on introduit la fonction $I(w,g)^{\omega}_{i,m}$ afin d'obtenir un résultat unifié pour l'approche directe et l'approche par projection. Cette fonction correspond pour les deux approches au calcul des intégrales des neurones fonctionnels dans le cas d'une connaissance discrète des fonction d'entrée :

– pour l'approche directe, on pose :

$$I(w_h, g^i)^{\omega}_{i,m} = \left(\sum_{j=1}^{m} F_1(w_h^1, X_j^i(\omega))(g^i(X_j^i(\omega)) + \mathcal{E}_j^i(\omega)), \ldots, \right.$$
$$\left. \sum_{j=1}^{m} F_K(w_h^K, X_j^i(\omega))(g^i(X_j^i(\omega)) + \mathcal{E}_j^i(\omega)) \right)$$

– pour l'approche par projection, on pose :

$$I(w_h, g^i)^{\omega}_{i,m} = \left(\int F_1(w_h^1, .)\Pi_P(g^i)^{\omega}_m dP_X, \ldots, \int F_K(w_h^K, .)\Pi_P(g^i)^{\omega}_m dP_X \right)$$

On démontre à présent le résultat de consistance dans le cas où les fonctions d'entrée et les fonctions à prédire sont connues de manière discrète.

Théorème 16. *Soient $\mathcal{X}$ et $\widetilde{\mathcal{X}}$ deux espaces métriques compacts munis de leur tribu borélienne. Soit $\mathcal{K}$ un sous-ensemble compact de $C(\mathcal{X}, \mathbb{R})$, et soit $\widetilde{\mathcal{K}}$ un sous-ensemble compact de $C(\widetilde{\mathcal{X}}, \mathbb{R})$.*

Soit W_h un ensemble compact. Soit $I(w,g)_{i,m}^{\omega}$ une suite de fonctions définies sur $W_h \times \mathcal{K}$ et à valeurs dans $\mathbb{R}^K$. Soit $I(w,g)$ une fonction définie sur $W_h \times \mathcal{K}$ et à valeurs dans $\mathbb{R}^K$. On suppose que :

1. *pour chaque $w \in W_h$, $I(w,.)$ est une fonction mesurable de $\mathcal{K}$ vers $\mathbb{R}^K$.*
2. *pour chaque $g \in \mathcal{K}$, $I(.,g)$ est une fonction continue de W_h vers $\mathbb{R}^K$.*
3. *pour presque tout $\omega \in \Omega$, on a pour tout $g \in \mathcal{K}$ et pour tout i, $I(w,g)_{i,m}^{\omega}$ qui converge uniformément sur W_h vers $I(w,g)$ quand m croît vers l'infini.*

Soit une suite de variables aléatoires $\widetilde{X}_j^i$ indépendantes identiquement distribuées et à valeurs dans $\widetilde{\mathcal{X}}$. On note $P_{\widetilde{X}}$ la mesure induite sur $\widetilde{\mathcal{X}}$ et $\widetilde{X} = \widetilde{X}_1^1$. Soit $\widetilde{\mathcal{E}}_j^i$ une suite de variables aléatoires indépendantes identiquement distribuées à valeurs dans $\mathbb{R}$. On note $\widetilde{\mathcal{E}} = \widetilde{\mathcal{E}}_1^1$. On suppose que $E(\widetilde{\mathcal{E}}) = 0$ et $E(|\widetilde{\mathcal{E}}|^2) < \infty$. On suppose que les $\widetilde{X}_j^i$ et les $\widetilde{\mathcal{E}}_j^i$ sont indépendantes.

Soit G^i une suite de variables aléatoires fonctionnelles définies sur $(\Omega, \mathcal{A}, P)$ et à valeurs dans $\mathcal{K}$. Soit T^i une suite de variables aléatoires définis sur $(\Omega, \mathcal{A}, P)$ et à valeurs dans $\widetilde{\mathcal{K}}$.

On suppose que les couples de variables aléatoires (G^i, T^i) sont indépendants et identiquement distribués. On note $G = G^1$ et $T = T^1$.

Soit Ψ une base topologique de $L^2(P_{\widetilde{X}})$, et Π_Q l'opérateur de projection sur l'espace vectoriel $vect(\psi_1, ..., \psi_Q)$. On note $\mathcal{V}$ un voisinage compact de $\Pi_Q(\widetilde{\mathcal{K}})$ dans $vect(\psi_1, ..., \psi_Q)$.

Soit l une fonction de $\mathbb{R}^K \times L^2(P_{\widetilde{X}}) \times W_o$ vers $\mathbb{R}$, où W_o est un ensemble compact. On suppose que :

1. *$l(.,.,.)$ est une fonction uniformément continue de $\mathbb{R}^K \times \mathcal{V} \times W_o$ vers $\mathbb{R}$.*
2. *il existe une fonction mesurable d' de $L^2(P_{\widetilde{X}})$ vers $\mathbb{R}$ telle que $|l(z,t,w_o)| \leq d'(t)$ pour tout z et w_o.*
3. *$E(d'(\Pi_Q(T))) < \infty$*

Pour chaque $\omega \in \Omega$, on définit :

$$\lambda_{m,m'}^n(w_h, w_o)(\omega) = \frac{1}{n} \sum_{i=1}^{n} l\left(I(w_h, G^i(\omega))_{i,m}^{\omega}, \Pi_Q(T^i(\omega))_{i,m'}^{\omega}, w_o\right),$$

On définit de plus

$$\lambda(w_h, w_o) = E\left(l\left(I(w_h, G), \Pi_Q(T), w_o\right)\right)$$

Alors pour chaque $\omega \in \Omega$ et pour chaque n, m et m', il existe une solution $w_{m,m'}^n(\omega)$ au problème

$$\min_{w \in W_h \times W_o} \lambda_{m,m'}^n(w_h, w_o)(\omega)$$

Si W^* est l'ensemble des minimiseurs de $\lambda(w_h, w_o)$, alors

$$\lim_{n\to\infty} \lim_{(m,m')\to\infty} d(w^n_{m,m'}(\omega), W^*) = 0 \; P - p.s.$$

Démonstration. On applique la loi forte des grands nombres uniforme à la fonction :

$$h(w_h, w_o, g, t) = l\left(I(w_h, g), \Pi_Q(t), w_o\right)$$

On utilise pour cela des arguments similaires au théorème 15. On voit que la fonction h est continue en (w_h, w_o), et est majorée par la fonction mesurable c, qui est d'espérance finie. Pour prouver la mesurabilité de h en (g, t), on voit que la fonction qui à g élément de $\mathcal{K}$ associe $I(w_h, g)$ est mesurable.

On a donc

$$\sup_{(w_h,w_o)\in W_h\times W_o} \left| \frac{1}{n}\sum_{i=1}^{n} h(w_h, w_o, G^i, T^i) - E(h(w_h, w_o, G, T)) \right| \to^{p.s.}_{n\to\infty} 0 \qquad (7.2)$$

A présent, on considère un ω pour lequel la convergence uniforme ci-dessus a lieu, et tel que pour chaque $g \in \mathcal{K}$, $I(w_h, g)^{\omega}_{i,m}$ converge uniformément vers $I(w_h, g)$, et de plus pour chaque $t \in \widetilde{\mathcal{K}}$, $\Pi_Q(t)^{\omega}_{m'}$ converge vers $\Pi_Q(t)$ (voir théorème 11). Par intersection finie, un tel ω existe presque sûrement.

On appelle $g^i = G^i(\omega)$ et $t^i = T^i(\omega)$. Soit ε un réel strictement positif. D'après 7.2, il existe N tel que pour chaque $n \geq N$,

$$\sup_{(w_h,w_o)\in W_h\times W_o} \left| \frac{1}{n}\sum_{i=1}^{n} h(w_h, w_o, g^i, t^i) - E(h(w_h, w_o, G, T)) \right| < \frac{\varepsilon}{2} \qquad (7.3)$$

On fixe n supérieur à N. Comme l est uniformément continue sur $\mathbb{R}^K \times \mathcal{V} \times W_o$, il existe $\eta^i > 0$ tel que pour chaque $w_o \in W_o$, $|l(z, \pi, w_o) - l(z', \pi', w_o)| < \frac{\varepsilon}{2}$ tant que $\|z - z'\| < \eta^i$ et $\|\pi - \pi'\|_2 < \eta_i$, avec z et z' éléments de $\mathbb{R}^K$, et π et π' éléments de $\mathcal{V}$. Selon les hypothèses sur I, il existe M^i tel que, $m_i \geq M^i$ implique $sup_{w_h\in W_h}\|I(w_h, g^i)^{\omega}_{i,m_i} - I(w_h, g^i)\| < \eta_i$ De plus, pour chaque t^i, il existe M'_i tel que $m'_i \geq M'_i$ implique $\|\Pi_Q(t^i)^{\omega}_{m'_i} - \Pi_Q(t^i)\|_2 < \eta_i$ avec $\Pi_Q(t^i)^{\omega}_{m'_i}$ élément de $\mathcal{V}$.

On appelle $M^n = sup_{i\leq n} M_i$ et $M'^n = \sup_{i\leq n} M'_i$. Pour $m \geq M^n$ et pour $m' \geq M'^n$, on a pour chaque $i \leq n$ et pour tout (w_h, w_o) :

$$\left| l(I(w_h, g^i)^{\omega}_{i,m}, \Pi_Q(t^i)^{\omega}_{m'}, w_o) - h(w_h, w_o, g^i, t^i) \right| < \frac{\varepsilon}{2}$$

Ce qui implique pour tout (w_h, w_o) :

$$\left| \lambda^n_{m,m'}(w_h, w_o)(\omega) - \frac{1}{n}\sum_{i=1}^{n} h(w_h, w_o, g^i, t^i) \right| < \frac{\varepsilon}{2}$$

En combinant cette inégalité avec l'équation 7.3, on obtient la conclusion suivante : Pour presque tout $\omega \in \Omega$, et pour chaque $\varepsilon > 0$, il existe N tel que pour chaque $n \geq N$, il existe M^n et M'^n tel que pour chaque $m \geq M^n$, et pour chaque $m' \geq M'^n$, $\sup_{(w_h,w_o)\in W_h\times W_o} |\lambda^n_{m,m'}(w_h, w_o)(\omega) - \lambda(w_h, w_o)| < \varepsilon$. Pour presque tout ω, on a donc

$$\lim_{n\to\infty} \lim_{(m,m')\to\infty} \sup_{(w_h,w_o)\in W_h\times W_o} |\lambda^n_{m,m'}(w_h, w_o)(\omega) - \lambda(w_h, w_o)| = 0 \qquad (7.4)$$

La conclusion finale est obtenue de manière similaire au théorème 6. □

Discussion

De manière similaire au théorème 15, les fonctions $I(.,.)$ et $I(.,.)^\omega_m$ permettent de traiter simultanément l'approche directe, présentée au chapitre 5, et l'approche par projection, présentée au chapitre 6.

Pour l'approche directe, on rappelle que :

$$I(w_h, g^i)^\omega_{i,m} = \left(\sum_{j=1}^{m} F_1(w^1_h, X^i_j(\omega))(g^i(X^i_j(\omega)) + \mathcal{E}^i_j(\omega)), \ldots, \right.$$
$$\left. \sum_{j=1}^{m} F_K(w^K_h, X^i_j(\omega))(g^i(X^i_j(\omega)) + \mathcal{E}^i_j(\omega)) \right)$$

On considère les variables aléatoires X^i_j et $\mathcal{E}^i_j$, et les régresseurs paramétriques $F_k(w^k_h, x)$. Alors les hypothèses sur les fonctions $I(.,.)$ et $I(.,.)^\omega_m$ se transposent en des hypothèses sur les fonctions F_k (voir théorème 6).

Pour l'approche par projection, on rappelle que :

$$I(w_h, g^i)^\omega_{i,m} = \left(\int F_1(w^1_h, x)\Pi_P(g^i)^\omega_m dP_X(x), \ldots, \int F_K(w^K_h, x)\Pi_P(g^i)^\omega_m dP_X(x) \right)$$

On considére les variables aléatoires X^i_j et $\mathcal{E}^i_j$, et les régresseurs paramétriques $F_k(w^k_h, x)$. Alors comme ci-dessus les hypothèses sur les fonctions $I(.,.)$ et $I(.,.)^\omega_m$ se traduisent en des hypothèses sur les fonctions F_k (voir théorème 10 dans le cas pseudo-linéaire, et théorème 6 dans le cas non-linéaire).

L'existence du voisinage compact $\mathcal{V}$ de $\Pi_Q(\widetilde{\mathcal{K}})$ dans $vect(\psi_1, ..., \psi_Q)$ ne pose pas de problème pratique car $\Pi_Q(\widetilde{\mathcal{K}})$ est un ensemble compact dans un espace de dimension finie. L'ensemble $\mathcal{V}$ doit être un voisinage compact pour cet espace.

L'uniforme continuité de la fonction l peut aisément être vérifiée si l'on impose à la fonction d'activation du perceptron multi-couches fonctionnel d'être uniformément continue et bornée (hypothèse vérifiée pour les fonctions d'activation habituellement utilisées).

7.5 Conclusion

Dans ce chapitre, on a montré que le perceptron multi-couches fonctionnel pouvait aisément être adapté afin d'obtenir une réponse fonctionnelle. Ce nouveau modèle offre un potentiel d'applications important, car il permet de modéliser entre autre des processus fonctionnels auto-régressifs.

La seconde partie de ce chapitre a été consacrée à l'étude des propriétés théoriques de ce modèle : la propriété d'approximation universelle a été démontrée dans la section 7.3. Dans la section 7.4, l'estimation consistante des paramètres a été prouvées sous l'hypothèse d'indépendance des fonctions d'entrée[3].

[3]La consistance du modèle dans le cas de processus fonctionnels auto-régressifs nécessite une preuve adaptée.

Chapitre 8

Simulations

8.1 Introduction

Le but de ce chapitre est de comparer sur des données simulées l'approche classique (le perceptron numérique) et les deux approches fonctionnelles (l'approche directe et l'approche par projection). Deux expériences différentes sont présentées :

- Dans la première expérience, on utilise les différents modèles (numériques et fonctionnels) afin de discriminer deux classes distinctes de fonctions. Ces fonctions, engendrées à partir de formes sinusoidales, sont définies de $\mathbb{R}$ dans $\mathbb{R}$. A travers cette première expérience, on montre les limites de l'approche classique naïve face à l'approche fonctionnelle dans le cas où la résolution d'échantillonnage devient trop faible pour décrire correctement les fonctions d'entrée (peu de points d'évaluation).
- Dans la seconde expérience, on considère des fonctions définies de $\mathbb{R}^2$ dans $\mathbb{R}$, et on cherche à prédire un vecteur binaire associé à chaque fonction d'entrée. On montre alors que la représentation non-linéaire des fonctions de poids permet d'obtenir de meilleurs résultats qu'une approche pseudo-linéaire. En effet, dans cette expérience, la représentation des fonctions de poids par des perceptrons multi-couches numériques se révèle plus efficace que celle réalisée grâce à des RBF (les fonctions de base sont fixes).

Dans la première partie de ce chapitre, on montre comment la technique d'initialisation géométrique, proposée dans [67], peut être adaptée à l'approche fonctionnelle. Dans les sections 8.3 et 8.4, la comparaison des différents modèles (numériques et fonctionnels) est réalisée grâce aux deux expériences présentées ci-dessus.

8.2 Initialisation

La phase d'apprentissage des perceptrons multi-couches (numériques ou fonctionnels) est réalisée grâce à des techniques classiques d'optimisation (par exemple, les algorithmes de descente de gradient). Dans la pratique, cette phase d'optimisation est une tâche difficile, car la fonction à minimiser présente généralement de nombreux minima locaux. La présence de tels minima peut être expliquée par le fait que certains neurones produisent une sortie identique sur l'ensemble d'apprentissage : de tels neurones sont donc redondants.

Le but des méthodes d'initialisation est de fournir un point de départ "acceptable" à l'algorithme d'optimisation, afin d'assurer une convergence rapide de l'algorithme, ainsi qu'une utilisation optimale des neurones du réseau. Plusieurs techniques d'initialisation ont déja été proposées pour le perceptron multi-couches numérique (initialisation par prototypes [25], initialisation par arbres de décision [45]). On présente ici une technique d'initialisation géométrique (voir Rossi et Gegout [67]), qui présente l'avantage d'être facilement adaptable au perceptron multi-couches fonctionnel.

8.2.1 Initialisation géométrique

Un perceptron à une couche cachée et à valeurs réelles calcule la fonction suivante :

$$f(x) = \sum_{k=1}^{K} a_k T(w_k.x + b_k) \tag{8.1}$$

où a_k et b_k sont des réels, et où w_k et x sont des vecteurs de $\mathbb{R}^n$.

Dans l'expression 8.1, chaque équation $w_k.x + b_k = 0$ définit un hyperplan de $\mathbb{R}^n$. Comme expliqué dans [54], la position relative de ces hyperplans dans l'espace des données joue un rôle fondamental dans le bon déroulement de la phase d'apprentissage : une configuration mal adaptée peut ralentir l'algorithme d'optimisation, et même mener à des résultats sous-optimaux. Afin d'identifier et d'éviter ce type de configurations, il est intéressant d'analyser le fonctionnement du neurone numérique d'un point de vue géométrique.

Dans la pratique, le choix de la fonction d'activation T s'effectue dans une classe restreinte de fonctions (les "squashing functions", voir [44]). Toutes ces fonctions partagent la propriété d'être quasi-constante (saturée) en dehors d'un intervalle localisé en l'origine. Cette propriété a d'importantes répercussions sur la fonction réalisée par le neurone. En effet, si l'hyperplan $w_k.x + b_k = 0$ est très éloigné de l'ensemble des données, la sortie du neurone sera constante sur l'ensemble des individus d'entrée. On voit donc qu'un tel neurone ne participe pas utilement au calcul de la fonction réalisée par le perceptron multi-couches.

Lors de la phase d'apprentissage, on souhaite que ces neurones inutiles prennent une part active dans le calcul de la sortie du réseau. Pour cela, l'algorithme d'optimisation va progressivement déplacer chaque hyperplan vers l'ensemble des individus composant la base d'apprentissage. La progression de ces hyperplans s'effectue d'autant plus rapidement que la dérivée du neurone en fonction de ses paramètres est importante. Malheureusement, dans le cas d'un hyperplan très éloigné, cette dérivée est quasiment nulle, car la sortie du neurone est entièrement déterminée par la zone saturée (constante) de la fonction d'activation. La progression des hyperplans nécessite donc un nombre très important d'itérations (la modification de la position d'un hyperplan peut même dans certains cas se révéler impossible).

L'initialisation aléatoire, habituellement utilisée pour les perceptrons multicouches numériques, n'apporte pas une réponse satisfaisante à ce problème. En effet, le choix aléatoire des paramètres w_k et b_k dans l'expression 8.1 ne permet pas de contrôler précisément la position de l'hyperplan dans l'espace des données : cette position dépend entièrement du rapport $\frac{-b_k}{\|w_k\|}$, qui peut prendre des valeurs arbitraires.

Le but de l'initialisation géométrique, proposée dans [67], est de contrôler efficacement la position de chaque hyperplan afin que chaque neurone soit "sensible" à une partie des données. Pour ce faire, on modifie l'équation 8.1, afin d'en obtenir une version plus géométrique et plus intuitive :

$$f(x) = \sum_{k=1}^{K} a_k T(d_k.(x - t_k)) \tag{8.2}$$

où les a_k sont des réels, et où d_k et t_k sont des vecteurs de $\mathbb{R}^n$.

Le passage de la forme géométrique à la forme standard s'effectue aisément, en développant le produit scalaire $d_k.(x - t_k)$. Les paramètres de l'équation 8.1 sont alors donnés par $w_k = d_k$ et $b_k = -d_k.t_k$.

Comme précédemment, l'équation $d_k.(x - t_k) = 0$ définit un hyperplan dans l'espace des données. Grâce à cette nouvelle expression, l'initialisation de chaque neurone peut être réalisée aisément. En effet, la position de l'hyperplan dépend uniquement du paramètre de translation, t_k, de même sa direction est uniquement fonction du paramètre de direction $d_k/\|d_k\|$. Enfin, la selectivité de la fonction d'activation dépend uniquement de la norme $\|d_k\|$.

Une manière simple de choisir ces trois paramètres est réalisée en imposant à chacun d'eux d'appartenir aux trois ensembles distincts suivants :

1. t_k est choisi aléatoirement dans "l'ensemble de translation", par exemple le sous-ensemble sur lequel est définie la fonction à modéliser.

2. $d_k/\|d_k\|$ est choisi aléatoirement dans "l'ensemble de direction" (par exemple, la boule unité).
3. finalement, $\|d_k\|$ est choisi aléatoirement dans "l'ensemble de sélectivité". Dans la pratique, cet ensemble est fonction de la nature de la fonction d'activation utilisée, ainsi que de l'ensemble de données. Si l'on considère par exemple la fonction d'activation $\tanh(\frac{x}{2})$, son domaine de saturation est caractérisé approximativement par l'ensemble des x vérifiant $|x| > 5$. De plus, dans le cas où $|x| < 1$, sa sortie est quasi-linéaire, et peut être approchée par $\frac{x}{2}$. Le choix de l'ensemble de sélectivité doit donc éviter la partie saturée de la fonction d'activation, tout en garantissant une bonne répartition des données sur la zone restante. On réalise pratiquement ce choix, en calculant l'expression $M = \max_{x \in App} |d_k.(x - t_k)|$. On renormalise alors le paramètre d_k trouvé à l'étape précédente, en choisissant aléatoirement le coefficient multiplicateur dans l'intervalle $[\frac{1}{M}, \frac{5}{M}]$.

L'initialisation des paramètres a_k de la couche de sortie du perceptron multi-couches ne pose pas de problème. En effet, comme la fonction d'erreur est quadratique en ces paramètres, les algorithmes d'optimisation classiques (gradient conjugué, BFGS) trouvent la solution optimale en un nombre limité d'itérations.

8.2.2 Adaptation aux modèles fonctionnels

Dans le cas où les fonctions de poids sont représentées par des modèles pseudo-linéaires, on a vu que le perceptron multi-couches fonctionnel était en fait un perceptron multi-couches numérique. L'initialisation géométrique (ainsi que tout autre méthode d'initialisation des perceptrons multi-couches numériques) peut donc lui être appliquée sans modification.

Dans le cas d'une représentation non-linéaire, l'initialisation géométrique nécessite une adaptation au cadre fonctionnel.

Le perceptron fonctionnel à une couche cachée calcule la fonction suivante :

$$H(g) = \sum_{k=1}^{K} a_k T(b_k + \int f_k g d\mu)$$

où $g \in L^p(\mu)$, $f_k \in L^q(\mu)$.

Comme dans la section précédente, on reformule l'expression du perceptron multi-couches fonctionnel d'un point de vue plus géométrique :

$$H(g) = \sum_{k=1}^{K} a_k T\left(\int d_k(g - t_k) d\mu\right)$$

où $g \in L^p(\mu)$, $d_k \in L^q(\mu)$ et $t_k \in L^p(\mu)$.

Le passage de la forme géométrique à la forme standard, s'effectue naturellement, en posant $f_k = d_k$ et $b_k = -\int d_k t_k d\mu$.

Comme précédemment, l'expression $\int d_k(g - t_k)d\mu = 0$ définit un hyperplan dans l'espace des fonctions d'entrée. Le neurone fonctionnel dépend donc des trois paramètres suivants : la fonction t_k, la fonction $d_k/\|d_k\|$ et le scalaire $\|d_k\|$.

Le choix de ces trois paramètres peut dans la pratique être réalisé ainsi :

1. on choisit aléatoirement une fonction d'entrée g_o dans l'ensemble d'apprentissage, et on impose à l'hyperplan de passer par ce point. On pose donc $t_k = g_o$.
2. on choisit alors aléatoirement la fonction de direction d_k, afin d'orienter l'hyperplan dans une direction aléatoire. Dans la pratique, la fonction d_k est représentée à l'aide d'un régresseur paramétrique. Choisir aléatoirement d_k revient donc à choisir aléatoirement les paramètres de ce régresseur paramétrique. Dans le cas particulier où d_k est représentée à l'aide d'un perceptron multi-couches numérique, le choix des paramètres numériques doit être réalisé avec soin, afin d'éviter un nouveau problème d'initialisation. On peut par exemple réappliquer à d_k la technique de l'initialisation géométrique en considérant l'ensemble sur lequel sont définies les fonctions d'entrées.
3. finalement, on calcule l'expression $\int d_k(g - t_k)d\mu = \int d_k g d\mu - \int d_k g_o d\mu$ pour chaque fonction d'entrée g appartenant à l'ensemble d'apprentissage. On choisit ainsi une valeur de $\|d_k\|$ qui évite une saturation excessive de la fonction d'activation (on applique pour cela le même procédé que dans la section précédente). La modification de la norme de d_k s'effectue aisément en normalisant les poids de la sortie linéaire du régresseur paramétrique.

Dans le cas de l'approche directe, le calcul des intégrales $\int d_k g d\mu$ et $\int d_k g_o d\mu$ est remplacé par l'évaluation d'une moyenne empirique (car les fonctions g et g_o ne sont connues qu'en un nombre fini de points d'évaluation). Le principe de l'initialisation géométrique n'est pas modifié par ce calcul approché. De même dans le cas de l'approche par projection, la fonction g est remplacée par sa projection $\Pi_P(g)$, ce qui ne modifie pas non plus la méthode proposée ici.

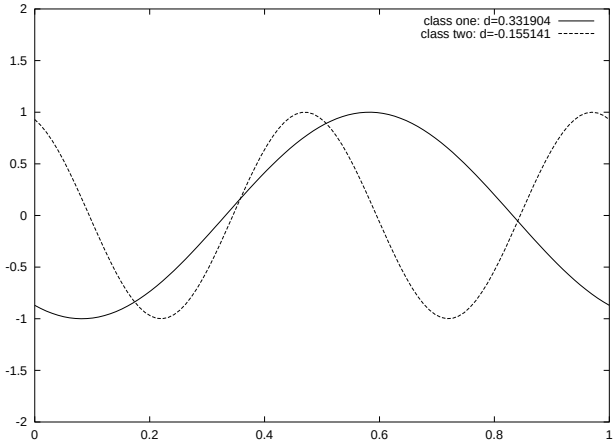
8.3 Les fonctions sinus

8.3.1 Les fonctions d'entrée

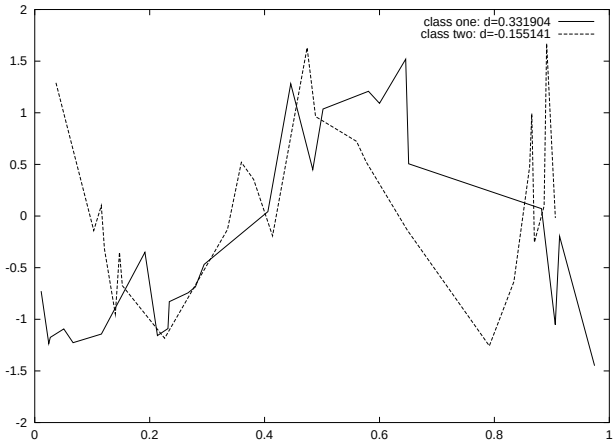
Le but de cette première expérience est de discriminer deux classes de fonctions. Ces deux classes sont engendrées selon le procédé suivant.

Pour la première classe, on considère la fonction génératrice $f_d(x) = \sin(2\pi(x-d))$, paramétrée par la variable d (le paramètre de translation). Afin de générer les fonctions composant la classe, on choisit aléatoirement selon une loi uniforme la valeur de d dans l'intervalle $[0, 1]$. Le schéma d'échantillonnage est alors identique pour toutes les fonctions : on choisit aléatoirement selon une loi uniforme 25 points d'évaluation répartis sur l'intervalle $[0, 1]$. Pour chacun de ces points d'évaluation, on calcule la valeur correspondante de la fonction. Enfin, on ajoute un bruit gaussien d'écart-type 0.7 à chacune de ces mesures.

On engendre la seconde classe de manière identique, en considérant la fonction génératrice $g_d(x) = \sin(4\pi(x-d))$.



(a) Fonctions non bruitées



(b) Fonctions bruitées

FIG. 8.1 – Fonctions d'entrée

Dans la figure 8.1(a), un exemple de fonction de chacune des deux classes

est représenté, tandis que dans la figure 8.1(b), ces deux même fonctions sont représentées après échantillonnage, ajout du bruit gaussien et interpolation linéaire.

On a généré selon ce procédé 500 fonctions différentes : 250 pour chaque classe. Lors de la phase d'apprentissage, ces 500 fonctions sont réparties en trois sous-ensembles distincts : 100 fonctions sont affectées à l'ensemble d'apprentissage, 100 autres fonctions sont affectées à l'ensemble de validation. Finalement, l'ensemble de test est constitué de 300 fonctions. Chaque optimisation a été effectuée grâce à un algorithme de gradient conjugué en utilisant la technique de l'arrêt prématuré (*early stopping*).

8.3.2 Les différents modèles

Afin de différencier l'approche directe de l'approche par projection, on introduit la notation suivante : on note $PMCF^{\Pi}$, le perceptron multi-couches fonctionnel basé sur une étape préalable de projection, et $PMCF$, le perceptron multi-couches fonctionnel de l'approche directe.

Afin de faciliter la comparaison des deux modèles fonctionnels, le $PMCF$ et le $PMCF^{\Pi}$ utilisent des architectures similaires. Les perceptrons sont tous deux constitués d'une unique couche cachée fonctionnelle, et utilisent tous deux 19 paramètres numériques. Pour l'approche directe, on a donc le perceptron fonctionnel suivant :

$$H(g) = c + \sum_{k=1}^{3} a_k T(b_k + \int f_k g d\mu)$$

Chaque fonction de poids f_k est représentée au moyen d'une B-spline, composée de 4 fonctions de base cubiques. Dans le cas de l'approche par projection, chaque fonction d'entrée est de même représentée au moyen d'une B-spline, composée de 6 fonctions de base cubiques. La figure 8.2 montre les deux fonctions d'entrée de la figure 8.1(a) après l'étape de projection.

Le but de ces expériences est de comparer les deux modèles fonctionnels ($PMCF^{\Pi}$ et $PMCF$) au perceptron numérique classique. Il n'est bien sûr pas possible de soumettre dans la pratique une fonction d'entrée à un perceptron multi-couches numérique. En effet, bien que le nombre de points d'évaluation soit identique pour toutes les fonctions d'entrée, leurs positions varient d'une fonction à une autre. Les description ne sont donc pas directement comparables.

La manière la plus simple de transformer les fonctions d'entrée en des vecteurs est d'utiliser une technique de moyennage. Pour ces expériences, on a divisé l'intervalle $[0, 1]$ en un nombre fini de sous-intervalles. On calcule alors

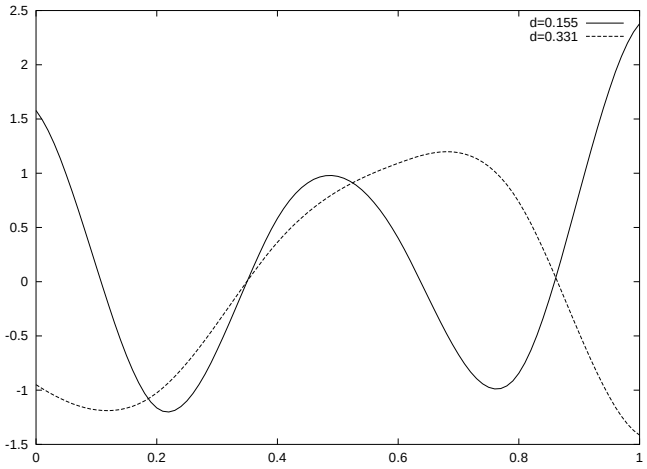


FIG. 8.2 – Projection des fonctions d'entrée

la moyenne de la fonction sur chacun de ces sous-intervalles afin d'obtenir un nouveau vecteur d'entrée.

Comme pour les approches fonctionnelles, le perceptron multi-couches numérique utilise une seule couche cachée et une sortie réelle. La phase d'optimisation est réalisée grâce un algorithme de gradient conjugué en utilisant la technique de l'arrêt prématuré (*early stopping*).

On peut remarquer que cette approche multivariée est très similaire à un cas particulier de l'approche par projection. En effet, si l'on choisit de représenter les fonctions d'entrée et les fonctions de poids grâce au modèle linéaire suivant : $\sum_{p=1}^{P} \beta_p \chi_{I_p}(x)$ où $\chi_{I_p}(x)$ est la fonction caractéristique du sous-intervalle I_p, l'approche multivariée et l'approche par projection sont totalement équivalentes ([26] utilise une approche similaire).

8.3.3 Résultats

Les résultats de cette première expérience sont résumés dans le tableau suivant :

Modèles	projection/ moyennage	neurone caché numérique/ fonctionnel	nombre de poids	erreur quadratique	taux de réussite
$PMCF$		3 (4 B-splines)	19	0.046	94.4 %
$PMCF^{\text{II}}$	6 B-splines	3 (4 B-splines)	19	0.029	97.0 %
PMC	4 intervalles	3	19	0.051	94.7 %

Les résultats de cette première expérience ne sont pas facilement interprétable, car le $PMCF$ et le perceptron numérique ont des résultats similaires, quant au $PMCF^{\Pi}$, ses performances sont à peine meilleures. La principale explication des bonnes performances du perceptron numérique s'explique par le fait que la stratégie de moyennage utilisée cette première expérience est relativement bien adaptée à la structure des données. Les modèles fonctionnels ne sont donc pas plus performants dans ce cas précis. Comme on pourra le voir dans la prochaine expérience, ces résultats sont modifiés si le nombre de points d'évaluation est trop faible pour donner une représentation correcte des fonctions d'entrée.

La figure 8.3 représente les fonctions de poids du $PMCF$ dans le cas où des B-splines à 4 fonctions de base cubiques sont utilisées.

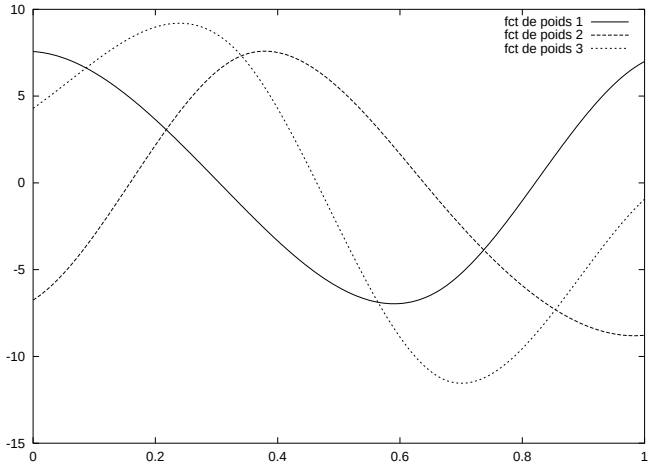


FIG. 8.3 – Les fonctions de poids du $PMCF$ (B-spline)

Bien que cette première expérience ne soit pas totalement concluante, il est intéressant néanmoins d'étudier le comportement des deux modèles fonctionnels quand les fonctions de poids sont représentées par différentes familles de régresseurs paramétriques. Deux résultats supplémentaires méritent une attention particulière : celui utilisant des séries de Fourier et celui utilisant des perceptrons multi-couches numériques.

Comme on pouvait s'y attendre, les modèles fonctionnels utilisant les séries de Fourier donnent de meilleurs résultats que ceux basés sur les B-splines : dans le cas du $PMCF^{\Pi}$, on obtient par exemple un taux de succès de 97.7% avec seulement 16 paramètres. Ce résultat montre que le modèle fonctionnel utilisant les séries de Fourier est pleinement avantagé par la structure sinusoidale des fonctions d'entrée. Le second résultat intéressant est celui du $PMCF$

utilisant des perceptrons multi-couches numériques comme fonctions de poids. Chacun de ces perceptrons utilise un seul neurone caché, et une sortie réelle. Ce modèle obtient le plus mauvais résultat, avec 87.4% de taux de réussite pour 19 paramètres. Cette contre-performance peut être expliquée par le fait que les modèles linéaires généralisés sont plus efficaces que les modèles non linéaires de type perceptron multi-couches pour représenter des fonctions définies sur des espaces de petite dimension (voir [3]) : dans le cas unidimensionnel, les perceptrons multi-couches numériques ont besoin de plus de paramètres que les B-splines pour représenter correctement les fonctions de poids (il faudrait donc rajouter des paramètres).

Si on s'intéresse au coût algorithmique nécessaire à l'apprentissage des différents modèles fonctionnels, on voit que durant la phase d'optimisation, chaque itération prend environ 20 fois plus de temps dans le cas d'une représentation non-linéaire des fonctions de poids ($PMCF$ et perceptron multi-couches numérique) que dans le cas pseudo-linéaire (B-spline, série de Fourier). Cette différence est bien sûr étroitement liée au nombre de points d'évaluation utilisé pour décrire chaque fonction d'entrée (25 points d'évaluation).

Afin de montrer les différences qui existent entre les modèles fonctionnels et l'approche standard, on présente les résultats d'une expérience similaire. Dans cette nouvelle expérience, on accroît la vitesse de variation des fonctions d'entrée, tandis que le nombre de points d'évaluation reste identique (25 points d'évaluation). On considère cette fois les deux fonctions génératrices suivantes : $f_d(x) = \sin(4\pi(x-d))$ et $g_d(x) = \sin(6\pi(x-d))$:

Modèles	projection/ moyennage	neurone caché numérique/ fonctionnel	nombre de poids	erreur quadratique	taux de réussite
$PMCF$		3 (5 B-splines)	22	0.139	84.0 %
$PMCF^{\text{II}}$	8 B-splines	3 (5 B-splines)	22	0.178	78.4 %
PMC	4 intervalles	3	19	0.173	76.0 %
PMC	6 intervalles	3	25	0.214	74.0 %

Cette fois les résultats sont plus faciles à interpréter. Il est clair que le $PMCF$ est plus performant que les autres modèles dans cette expérience : le nombre peu élevé de points d'évaluation ne le pénalise pas autant que les autres approches. Bien que l'approche par projection soit légèrement plus performante que l'approche standard (78.4% contre 76%), le résultat n'est cependant pas comparable à celui du $PMCF$. Cette différence peut être expliquée par le fait que l'étape de projection est sans doute trop sensible au bruit gaussien appliqué aux fonctions d'entrée. Une voie possible pour accroître les performances du

$PMCF^{\Pi}$ serait d'utiliser une méthode plus robuste de projection (B-spline avec validation croisée). Cela permettrait sans doute de représenter chaque fonction d'entrée de manière plus précise. Ce procédé est bien sûr plus coûteux en temps de calcul.

La figure 8.4 représente les fonctions de poids du $PMCF$ dans le cas où des B-splines à 5 fonctions de base cubiques sont utilisées.

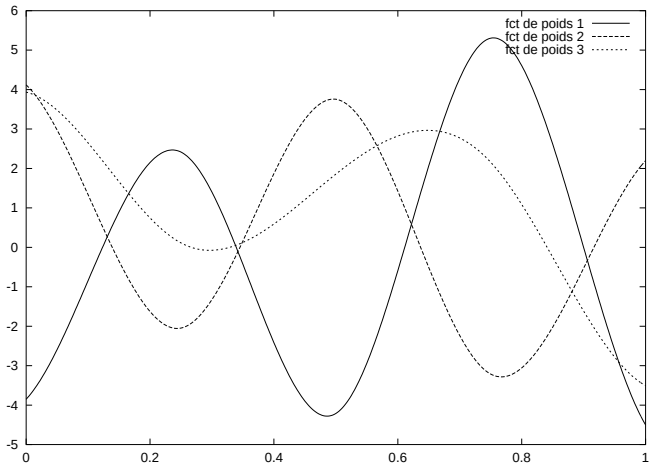


FIG. 8.4 – Les fonctions de poids du $PMCF$ (B-spline)

8.4 Les cercles

Le but de cette seconde expérience est d'étudier le comportement des modèles fonctionnels dans le cas où la dimension de l'espace d'entrée des fonctions augmente : les fonctions d'entrée sont à présent définies de $\mathbb{R}^2$ dans $\mathbb{R}$.

8.4.1 Les fonctions d'entrée

Dans cette expérience, l'ensemble des fonctions d'entrée est engendré selon le procédé suivant :

On considère cinq cercles du plan de rayon 0.1. Les centres sont uniformément espacés sur un cercle de rayon plus important (de rayon 0.3). Les cinq cercles sont représentés sur la figure 8.5. On génère alors chaque fonction d'entrée de la manière suivante :

1. On choisit aléatoirement selon une loi uniforme un entier compris entre 0 et 31 inclus. La représentation binaire de ce nombre, $b_0b_1b_2b_3b_4$, est alors

utilisée comme vecteur à prédire pour la fonction d'entrée. Chaque modèle (fonctionnel ou numérique) aura donc 5 sorties, chacune correspondant à une coordonnée de ce vecteur binaire. De plus, chaque chiffre dans la représentation binaire correspond à un cercle.

2. On choisit aléatoirement selon une loi uniforme 200 points d'évaluation dans le carré $[0,1] \times [0,1]$. La valeur de la fonction à un point d'évaluation donné vaut 0 si le point est extérieur aux cinq cercles. Si le point est intérieur au cercle i, la valeur de la fonction vaut b_i. Par exemple, la figure 8.5(a) correspond à la fonction d'entrée $11111 = 31$, et la figure 8.5(b) correspond à $11010 = 26$.

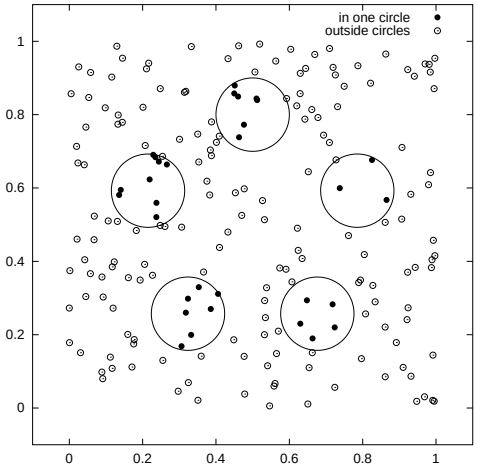
On a généré selon ce procédé 500 fonctions différentes. Lors de la phase d'apprentissage, ces 500 fonctions sont réparties en trois sous-ensembles distincts : 100 fonctions sont affectées à l'ensemble d'apprentissage, 100 autres fonctions sont affectées à l'ensemble de validation. Finalement, l'ensemble de test est constitué de 300 fonctions. Toutes les optimisations ont été effectuées grâce à un algorithme de gradient conjugué en utilisant la technique de l'arrêt prématuré (*early-stopping*).

8.4.2 Modèles

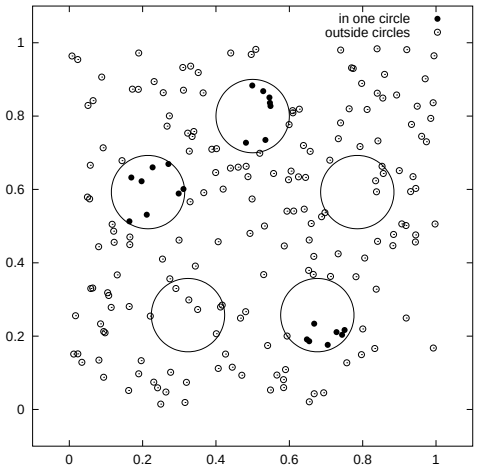
Dans cette expérience, le $PMCF$ utilise des perceptrons numériques pour représenter ses fonctions de poids. Ceux-ci sont donc composés de deux entrées, d'une couche cachée et d'une sortie réelle. De façon similaire, le $PMCF^{\Pi}$ utilise des fonctions de base radiale (RBF) pour représenter ses fonctions de poids. Les gaussiennes du RBF sont organisées uniformément sur le carré unité. Finalement, chaque RBF calcule la fonction suivante :

$$R(x) = \alpha_0 + \sum_{q=1}^{Q} \alpha_q \psi_q(x)$$

Comme dans l'expérience précédente, on cherche à comparer les modèles fonctionnels à l'approche standard (perceptron multi-couches numérique). Afin de transformer chaque fonction d'entrée sous la forme d'un vecteur, on utilise une seconde fois une technique de moyennage. Le carré unité est divisé régulièrement en $r \times r$ sous-carrés. Pour chaque fonction, on calcule la moyenne de la fonction sur chaque sous-carré. On obtient finalement un vecteur de r^2 valeurs réelles. Dans le cas de la fonction d'entrée représentée dans la figure 8.5(b), on obtient le vecteur suivant (pour $r = 4$) :



(a) 11111



(b) 11010

FIG. 8.5 – Les cercles

	$x \in [0, \frac{1}{4}[$	$x \in [\frac{1}{4}, \frac{1}{2}[$	$x \in [\frac{1}{2}, \frac{3}{4}[$	$x \in [\frac{3}{4}, 1]$
$y \in [0, \frac{1}{4}[$	0	0	0.37	0.14
$y \in [\frac{1}{4}, \frac{1}{2}[$	0	0	0	0
$y \in [\frac{1}{2}, \frac{3}{4}[$	0.45	0.24	0.056	0
$y \in [\frac{3}{4}, 1]$	0	0.077	0.36	0

On peut faire le rapprochement entre cette technique de moyennage et une approche naïve de projection des fonctions d'entrée sur une base de fonctions constantes par morceaux (voir la remarque dans la section 8.3.2).

Le perceptron numérique a r^2 entrées, une couche cachée et 5 sorties. La phase d'apprentissage est réalisée de manière identique aux approches fonctionnelles (gradient conjugué et arrêt prématuré).

8.4.3 Résultats

Tous les résultats sont résumés dans le tableau suivant :

Modèles	projection/ moyennage	neurone caché numérique/ fonctionnel	nombre de poids	erreur quadratique	taux de réussite
$PMCF$		5 (PMC avec 1 neurone caché)	60	0.071	92.0 %
$PMCF$		6 (PMC avec 1 neurone caché)	71	0.059	97.0 %
$PMCF$		5 (PMC avec 3 neurone caché)	100	0.016	100.0 %
$PMCF^{\Pi}$	16 RBF +1	5 (9 RBF +1)	85	0.040	97.0 %
PMC	9 carrés	5	80	0.053	94.0 %
PMC	16 carrés	5	115	0.013	99.0 %

Ces expériences montrent clairement que le perceptron multi-couches numérique, ainsi que le $PMCF^{\Pi}$ ont besoin de plus de paramètres que le $PMCF$ afin d'atteindre des performances comparables.

Si l'on compare les deux approches fonctionnelles, la différence de performance peut être expliquée par la nature des régresseurs paramétriques utilisés pour représenter les fonctions de poids. Dans cette expérience, il est clair que l'utilisation de modèles non linéaires tels que les perceptrons multi-couches numériques est plus efficace que les approches basées sur des modèles linéaires. En effet, comme on peut le voir dans [3], les perceptrons multi-couches sont plus parcimonieux (i.e. ils nécessitent moins de paramètres) que les modèles linéaires généralisés pour représenter des fonctions définies sur des espaces de dimension importante.

Dans le cas du $PMCF$ défini par 5 neurones fonctionnels (où chaque neurone fonctionnel utilise un perceptron numérique composé d'un seul neurone caché), la figure 8.6 montre comment le $PMCF$ effectue pratiquement son calcul. L'unique neurone caché de chaque perceptron numérique définit une droite dans le plan, qui indique la zone de transition entre les deux parties saturées de sa fonction d'activation. On voit qu'après la phase d'apprentissage, l'ensemble de

ces droites forme une partition en régions du carré unité, qui respecte la disposition des cinq cercles : en effet, chaque région de la partition contient entièrement un unique cercle. On voit donc que le perceptron multi-couches fonctionnel s'est adapté à la structure du problème lors de la phase d'apprentissage.

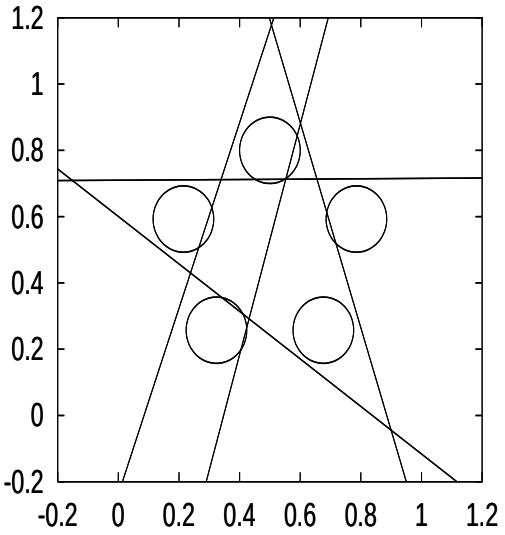


FIG. 8.6 – Les droites définies par les neurones fonctionnels du $PMCF$

Si l'on compare à présent le $PMCF$ et le perceptron multi-couches numérique, on peut expliquer les différents résultats par le fait que le $PMCF$ peut adapter sa méthode de moyennage aux données (stratégie dynamique de moyennage comme on peut le voir sur la figure 8.6), tandis que la méthode de moyennage dans le cas du perceptron multi-couches numérique est fixée avant l'apprentissage. Cette contrainte est l'une des raisons qui explique les performances moyennes de l'approche standard comparée à l'approche fonctionnelle. Bien sûr, il eut été possible d'utiliser une stratégie de moyennage plus adaptée aux données, afin d'obtenir une amélioration des performances du perceptron multi-couches numérique. Cette approche nécessite cependant une très bonne compréhension des données utilisées (dans notre cas, la position des cercles). Pour le $PMCF$ en revanche aucune connaissance *a priori* n'a été nécessaire.

Enfin, comme dans l'expérience précédente, on cherche à comparer le temps nécessaire à l'apprentissage des différents modèles fonctionnels : chaque itération de l'algorithme d'optimisation prend environ 100 fois plus de temps dans le cas d'une représentation non-linéaire des fonctions de poids ($PMCF$ et perceptron multi-couches numérique) que dans le cas pseudo-linéaire (RBF). Cette différence est bien sûr étroitement liée au nombre de points d'évaluation utilisé afin de décrire les fonctions d'entrée (200 points d'évaluation).

8.5 Conclusion

Dans la première partie de ce chapitre, on a pu voir que la technique de l'initialisation géométrique pouvait aisément être transposée au cadre fonctionnel. Cette technique permet une réduction du nombre d'itérations nécessaires à la phase d'apprentissage, ainsi qu'une meilleure utilisation des ressources du perceptron multi-couches fonctionnel.

Dans la seconde partie, on a montré que l'approche fonctionnelle était supérieure à l'approche classique dans le cas où la résolution d'échantillonnage n'était pas suffisante pour décrire correctement les fonctions d'entrée. Dans la seconde expérience, on a pu voir que la représentation non-linéaire des fonctions de poids se révélait plus efficace que celle réalisée par une approche pseudo-linéaire dans le cas où l'espace de définition des fonctions d'entrée était de dimension élevée.

Chapitre 9

Conclusions et Perspectives

Le but de cette thèse est d'effectuer la jonction entre le domaine de l'Analyse de Données Fonctionnelles et celui des techniques neuronales classiques. Comme on a pu le voir au cours de ce travail, l'extension du perceptron multi-couches numérique au cadre fonctionnel s'est révélée prolifique tant du point de vue pratique que du point de vue théorique :

- du point de vue pratique, on a tout d'abord montré que le perceptron multi-couches fonctionnel pouvait aisément être adapté afin de prendre en compte la discrétisation des fonctions d'entrée. Deux approches distinctes ont pour cela été proposées : l'approche directe et l'approche par projection. Dans les deux cas, la nature des fonctions de poids (non-linéaires/pseudo-linéaires) joue un rôle fondamental dans le coût d'évaluation du réseau : en effet, la représentation par des modèles pseudo-linéaires est nettement moins coûteuse que celle réalisée par des modèles non-linéaires.

  Toujours d'un point de vue pratique, on a vu que la représentation des fonctions de poids par des modèles paramétriques permettait au perceptron multi-couches fonctionnel d'être paramétré par un nombre fini de paramètres ajustables. On voit donc que la phase d'apprentissage de ce modèle est en tout point semblable à celle du perceptron multi-couches numérique, et nécessite l'utilisation d'algorithmes classiques d'optimisation (algorithmes de descente de gradient).

  Dans le chapitre 8, on a finalement montré que l'étape d'apprentissage du perceptron multi-couches fonctionnel pouvait bénéficier d'une technique d'initialisation issue de l'approche classique : l'initialisation géométrique.
- d'un point de vue théorique, on a montré que l'extension du perceptron multi-couches numérique au cadre fonctionnel était réalisée en conservant les propriétés théoriques importantes du modèle. En effet, on a vu que

dans les deux approches proposées (directe et par projection), le modèle était un approximateur universel, et que de plus, dans chacun des deux cas, l'estimation consistante des paramètres était possible : une première fois dans le cas d'une connaissance parfaite des fonctions d'entrée, puis une deuxième fois dans le cas d'une discrétisation de ces mêmes fonctions.

Dans le chapitre 7, on a montré que le perceptron multi-couches fonctionnel pouvait encore être adapté afin d'obtenir une réponse fonctionnelle. Cette extension est très intéressante, car elle permet d'obtenir par exemple un modèle auto-régressif fonctionnel adapté à la prévision de processus fonctionnel à temps discret. La propriété d'approximation universelle ainsi que les propriétés de consistance ont été prouvées (la consistance n'a cependant pas été étendue au cas de dépendance).

Les perspectives à l'issue de ce travail sont nombreuses et diverses. Par exemple, dans le chapitre 8, on s'est intéressé à deux expériences sur données simulées qui ont permis la comparaison des différents modèles fonctionnels. On souhaite compléter ces simulations par des expériences sur données réelles (ou simulées) afin d'obtenir une meilleure compréhension des différentes techniques. D'autre part, les techniques classiques de l'Analyse de Données Fonctionnelles proposent une solution alternative aux problèmes qui nous intéressent. On peut citer par exemple les modèles non paramétriques, le modèle linéaire fonctionnel, ou le modèle de régression inverse par tranche (**?**]). Il nous faudrait donc comparer sur des données réelles ces différents modèles et la solution proposée ici.

Un autre point important qui nécessite de plus amples développements est le problème de la modélisation du *design* des fonctions d'entrée. Bien que la modélisation proposée ici se soit révélée fructueuse, d'autres solutions peuvent être envisagées. On peut considérer par exemple un *design* asymptotiquement identique. Dans ce cas les points d'évaluation ne sont pas identiquement distribués, mais on impose en revanche à la loi limite d'être la même pour toutes les fonctions observées. Dans le même genre d'idée, on peut s'intéresser à un *design* déterministe, où l'on impose donc des contraintes sur l'espacement entre les points d'évaluation. Enfin, le *design* conditionné à la fonction d'entrée permettrait d'obtenir une souplesse accrue, car on pourrait par exemple tenir compte d'une adaptation de la discrétisation à la complexité de la fonction.

On peut noter que tous les résultats de consistance énoncés dans ce travail font l'hypothèse d'indépendance sur les individus d'entrée. Il est donc légitime de s'intéresser au cas de dépendance. L'une des applications possibles est bien sûr la prévision de processus fonctionnels à temps discret (voir par exemple Besse et Cardot [8]).

Enfin plus généralement, on peut s'intéresser à l'extension des diverses techniques neuronales au cadre fonctionnel. On a vu par exemple dans le chapitre 8 que la technique d'initialisation géométrique, issue des approches classiques avait été transposé avec succès au cas du perceptron multi-couches fonctionnel. D'autres techniques d'initialisation peuvent être adaptées de façon identique afin de pouvoir être appliquées au perceptron multi-couches fonctionnel. Enfin, toujours dans l'optique d'effectuer une jonction entre l'ADF et les méthodes neuronales, d'autres modèles neuronaux semblent se préter au cadre fonctionnel, et pourraient donc faire partie de l'arsenal fonctionnel.

Bibliographie

[1] Christophe Abraham, Pierre-André Cornillon, Eric Matzner-Lober, et Nicolas Molinari. Unsupervised curve clustering using b-splines. Technical Report XX–XX, ENSAM–INRA–UM II–Montpellier, June 2002.

[2] Donald W. K. Andrews. Consistency in nonlinear econometric models : A generic uniform law of large numbers. *Econometrica*, 55(6) :1465–1471, November 1987.

[3] Andrew R. Barron. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Trans. Information Theory*, 39(3) :930–945, May 1993.

[4] Philippe Besse. Approximation spline de l'analyse en composantes principales d'une variable aléatoire hilbertienne. *Annales de la Faculté des sciences de Toulouse*, 12(3) :329–349, 1991. Série 5.

[5] Philippe Besse, Hervé Cardot, et Frédéric Ferraty. Simultaneous nonparametric regressions of unbalanced longitudinal data. *Computational Statistics and Data Analysis*, 24 :255–270, 1997.

[6] Philippe Besse, Hervé Cardot, et David Stephenson. Autoregressive forecasting of some functional climatic variations. *Scandinavian Journal of Statistics*, 4 :673–688, 2000.

[7] Philippe Besse et Jim Ramsay. Principal component analysis of sampled curves. *Psychometrica*, 51 :285–311, 1986.

[8] Philippe C. Besse et Hervé Cardot. Approximation spline de la prévision d'un processus fonctionnel autorégressif d'ordre 1. *Canadian Journal of Statistics*, 24 :467–487, 1996.

[9] C. Bishop et C. Legleye. Estimating conditional probability densities for periodic variables, 1995.

[10] Christopher Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

[11] Denis Bosq. Modelization, non-parametric estimation and prediction for continuous time processes. In G. Roussas, editor, *Nonparametric Functional Estimation and Related Topics*, Asi, pages 509–529. Nato, 1991.

[12] Babette A. Brumback et John A. Rice. Smoothing spline models for the analysis of nested and crossed samples of curves. *J. Amer. Statist. Assoc.*, 93 :961–994, 1998.

[13] Hïam Brézis. *Analyse fonctionnelle : Théorie et applications*. Dunod, 1994.

[14] Hervé Cardot. Convergence du lissage spline de la prévision des processus autorégressifs fonctionnels. *C. R. Acad. Sci. Paris*, 326 :755–758, 1998. Série I.

[15] Hervé Cardot. Nonparametric estimation of smoothed principal components analysis of sampled noisy functions. *Journal of Nonparametric Statistics*, 12 :503–538, 2000.

[16] Hervé Cardot, Frédéric Ferraty, André Mas, et Pascal Sarda. Testing hypotheses in the functional linear model. *to appear in Scandinavian Journal of Statistics*, 2001.

[17] Hervé Cardot, Frédéric Ferraty, et Pascal Sarda. Functional linear model. *Statist. & Prob. Letters*, 45 :11–22, 1999.

[18] Hervé Cardot, Frédéric Ferraty, et Pascal Sarda. Etude asymptotique d'un estimateur spline hybride pour le modèle linéaire fonctionnel. *C. R. Acad. Sci. Paris*, 330 :501–504, 2000. Série I.

[19] Tianping Chen. A unified approach for neural network-like approximation of non-linear functionals. *Neural Networks*, 11 :981–983, May 1998.

[20] Tianping Chen et Hong Chen. Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems. *IEEE Transactions on Neural Networks*, 6 (4) :911–917, July 1995.

[21] G. Cybenko. Approximation by superpositions of a sigmoidal function, mathematics of control. *Signals and Systems*, 2 :303–314, 1989.

[22] J. Dauxois, Louis Ferré, et Anne-Françoise Yao. Un modèle semi-paramétrique pour variables hilbertiennes. *C. R. Acad. Sci. Paris*.

[23] J. Dauxois et A. Pousse. *Les analyses factorielles en calcul des probabilités et en statistiques : essai d'étude synthétique*. PhD thesis, Université Paul Sabatier, 1976.

[24] J. Dauxois, A. Pousse, et Y. Romain. Asymptotic theory for the principal component analysis of a vector of random function : some applications to statistical inference. *Journal of Multivariate Analysis*, 1982.

[25] Thierry Denoeux et Régis Lengellé. Initializing back propagation networks with prototypes. *Neural Networks*, 6 :351–363, 1993.

[26] J.C. Deville. Méthodes statistiques et numériques de l'analyse harmonique. *Annales de l'INSEE*, 1974.

[27] Jianqing Fan et Sheng-Kuei Lin. Test of significance when data are curves. *Journal of American Statistical Association*, 93 :1007–1021, 1998.

[28] Jianqing Fan et Jin-Ting Zhang. Two-step estimation of functional linear models with applications to longitudinal data. *Journal of the Royal Statistical Society B*, 62 :303–322, 2000.

[29] Louis Ferré et Anne-Françoise Yao. Functional sliced inverse regression analysis. *Statistics*, 2002.

[30] F. Ferraty et P. Vieu. Statistique fonctionnelle : Modèles de régression pour variables aléatoires uni, multi et infiniment dimensionnées. Technical Report LSP-2001-03, Laboratoire de Statistique et Probabilités, Université Paul Sabatier, Toulouse, France, 2001.

[31] Frédéric Ferraty. Estimation non-paramétrique et discrimination de courbes. In *Actes des huitièmes journées de la SFC*, 2001.

[32] Frédéric Ferraty, Aldo Goia, et Philippe Vieu. Régression non-paramétrique pour des variables aléatoires fonctionnelles mélangeantes. *C. R. Acad. Sci. Paris*, 334 :217–220, 2002. Série I.

[33] Frédéric Ferraty et Philippe Vieu. Dimension fractale et estimation de la régression dans des espaces vectoriels semi-normés. *C. R. Acad. Sci. Paris*, 330 :139–142, 2000. Série I.

[34] Frédéric Ferraty et Philippe Vieu. Statistique fonctionnelle : modèles de régression pour variables aléatoires uni, multi et infiniment dimensionnées. Technical Report LSP-2001-03, Laboratoire de statistique et probabilités, Université Paul Sabatier, Toulouse, 2001.

[35] R. Fletcher. *Practical Methods of Optimization*. New York : John Wiley, 1987.

[36] I.E. Frank et J. H. Friedman. A statistical view of some chemometrics regression tools. *Technometrics*, 35 :109–148, 1993.

[37] Cedric Gegout, Bernard Girau, et Fabrice Rossi. Generic back-propagation in arbitrary feedforward neural networks. In D. W. Pearson, N. C. Steele, et R. F. Albrecht, editors, *Int. Conf. on Artificial Neural Nets and Genetic Algorithms*, pages 168–171, Alès, April 1995. Springer Verlag.

[38] T. Hastie, A. Buja, et R. Tibshirani. Penalized discriminant analysis. *Annals of Statistics*, 23 :73–102, 1995.

[39] T. Hastie et C. Mallows. A discussion of "a statistical view of some chemometrics regression tools" by i.e. frank and j.h. friedman. *Technometrics*, 35 :140–143, 1993.

[40] Trevor Hastie et Robert Tibshirani. Varying-coefficient models. *Journal of the Royal Statistical Society B*, 55 :757–796, 1993.

[41] Donald R. Hoover, John A. Rice, Colin O. Wu, et Li-Ping Yang. Nonparametric smoothing estimates of time-varying coefficient models with longitudinal data. *Biometrika*, 85(4) :809–822, 1998.

[42] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4 :251–257, 1991.

[43] Kurt Hornik. Some new results on neural network approximation. *Neural Networks*, 6(8) :1069–1072, 1993.

[44] Kurt Hornik, Maxwell Stinchcombe, et Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2 :359–366, 1989.

[45] K. Sethi Ishwar. From decision trees to neural networks, October 1990.

[46] Gareth M. James. Generalized linear models with functional predictor variables. *Journal of the Royal Statistical Society Series B*, (64) :411–432, 2002.

[47] Gareth M. James et Trevor J. Hastie. Functional linear discriminant analysis of irregularly sampled curves. *Journal of the Royal Statistical Society Series B*, 63 :533–550, 2001.

[48] Gareth M. James, Trevor J. Hastie, et Catherine A. Sugar. Principal component models for sparse functional data. *Biometrika*, 87(3) :587–602, September 2000.

[49] Gareth M. James et Catherine A. Sugar. Clustering for sparsely sampled functional data. Technical report, Marshall School of Business, University of Southern California, 2002.

[50] Ludovic Lebart, Alain Morineau, et Marie Piron. *Statistique exploratoire multidimensionnelle*. Dunod, 3ème edition, 1995.

[51] Moshe Leshno, Vladimir Ya. Lin, Allan Pinkus, et Schocken Shimon. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *neural networks*, 6(6) :861–867, 1993.

[52] S. Leurgans, R. Moyeed, et B. Silverman. Canonical correlation analysis when the data are curves. *Journal of the Royal Statistical Society B*, 55 (3) :725–740, 1993.

[53] Ker-Chau Li. Sliced inverse regression for dimension reduction. *J. Amer. Statist. Assoc.*, 86 :316–342, 1991.

[54] F. A. Wessels Lodewyk et Etienne Barnard. Avoiding false local minima by proper initialization of connections, November 1992.

[55] B. D. Marx et P. H. Eilers. Generalized linear regression on sampled signals with penalized likelihood. In R. Hatzinger A. Forcina, G. M. Marchetti et G. Galmacci, editors, *Statistical Modelling. Proceedings of the 11th International workshop on Statistical Modelling*, Orvietto, 1996.

[56] C. Michel-Briand et Y. Escouffier. Segmentation d'un ensemble de courbes. *Revue de Statistique Appliquée*, 4 :5–24, 1994.

[57] Jean-Michel Morel et Sergio Solimini. *Variational methods in image segmentation*, volume 14 of *Progress in Nonlinear Differential Equations and Their Applications*. Birkhäuser, Boston, 1995.

[58] Steven J. Nowlan et Geoffrey E. Hinton. Simplifying neural networks by soft weight-sharing. *Neural Computation*, 4(4) :473–493, 1992.

[59] S. Pezzulli et B. Silverman. On smoothed principal components analysis. *Computational Statistics*, 8 :1–16, 1993.

[60] E. Polak. *Computational Methods in Optimization : A Unified Approach*. New York : Academic Press, 1971.

[61] William H. Press, Saul A. Teukolsky, William T. Vetterling, et Brian P. Flannery. *Numerical Recipes : The Art of Scientific Computing*. Cambridge University Press, 2ème edition.

[62] B. Pumo. *Estimation et prévision de processus fonctionnels auto-régressifs : Application aux processus à temps continu*. Thèse de troisième cycle, Université Paris VI, 1992.

[63] Jim Ramsay et Bernard Silverman. *Functional Data Analysis*. Springer Series in Statistics. Springer Verlag, June 1997.

[64] C. R. Rao et S. K. Mitra, editors. *Generalized Inverse of Matrices and Its Applications*. Wiley, 1971.

[65] John A. Rice et B. Silverman. Estimating the mean and covariance structure nonparametrically when the data are curves. *Journal of the Royal Statistical Society B*, 53(1) :233–243, 1991.

[66] John A. Rice et Colin O. Wu. Nonparametric mixed effects models for unequally sampled noisy curves. *Biometrics*, 57(1) :253–259, March 2001.

[67] Fabrice Rossi et Cedric Gegout. Geometrical initialisation, parametrization and control of multilayer perceptrons : Application to function approximation, June 1994.

[68] Walter Rudin. *Real and complex Analysis*. Mc Graw Hill, 1974.

[69] Irwin W. Sandberg. Notes on weighted norms and network approximation of functionals. *IEEE Transactions on Circuits and Systems–I : Fundamental Theory and Applications*, 43(7) :600–601, July 1996.

[70] Irwin W. Sandberg et Lilian Xu. Network approximation of input-output maps and functionals. *Circuits Systems Signal Processing*, 15(6) :711–725, 1996.

[71] B. Silverman. Incorporating parametric effects into functional principal components analysis. *Journal of the Royal Statistical Society B*, 57(4) : 673–689, 1995.

[72] B. Silverman. Smooth functional principal components analysis by choice of norm. *The Annals of Statistics*, 24(1) :1–24, 1996.

[73] STAPH. Statistique fonctionnelle : résumés des exposés 1999-2000. Technical Report LSP-2001-05, Laboratoire de statistique et probabilités, 2001.

[74] STAPH. Statistique fonctionnelle : résumés des exposés 2000-2001. Technical Report LSP-2001-07, Laboratoire de statistique et probabilités, 2001.

[75] Maxwell B. Stinchcombe. Neural network approximation of continuous functionals and continuous functions on compactifications. *Neural Networks*, 12(3) :467–477, 1999.

[76] Halbert White. Learning in artificial neural networks : A statistical perspective. *Neural Computation*, 1(4) :425–464, 1989.

[77] Yan Yu et Diane Lambert. Fitting trees to functional data, with an application to time of day patterns. *J. of Computational and Graphical Statistics*, 8 :749–762, 2000.