

REPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE
SCIENTIFIQUE

UNIVERSITÉ FRÈRES MENTOURI CONSTANTINE
FACULTÉ DES SCIENCES EXACTES
DÉPARTEMENT DE MATHÉMATIQUES



NUMÉRO D'ORDRE :30/D3C/2021
NUMÉRO DE SÉRIE :01/Math/2021

EN VUE DE L'OBTENTION DE GRADE DE DOCTEUR LMD

Option : Statistique Appliquée

RÉSEAUX DE NEURONES DANS LA STATISTIQUE EXPLORATOIRE ET À L'AIDE
DE LA DÉCISION

Présentée par :
Assia Ayache

• Devant le jury :

Z. MOHDEB	Prof, E. Polytechnique	UC3	Président
F.L RAHMANI	Prof, E. Biotechnologie	UC3	Rapporteur
S. KHARFOUCHI	Prof, Faculté de Médecine	UC3	Co-encadreur
S. BELALOUI	Prof, Département de Mathématiques	UMC	Examinatrice
N. ABADA	MCA, ENS Assia Djebbar	UNSC	Examinatrice

Date de soutenance orale le : 08.04.2021

Remerciements

Après avoir rendu grâce à Dieu le Tout Puissant, mes premiers remerciements vont d'abord à mon co-encadrante de thèse madame Soumia KHARFOUCHI qui m'a fait partager ses brillantes intuitions. Je lui suis également reconnaissante pour le temps conséquent qu'elle m'a accordé, sa gentillesse et sa disponibilité.

Je remercie également mon directeur de thèse monsieur Fouad Lazhar RAHMANI, qui a su inspirer le titre de cette thèse, pour toutes nos discussions et ses conseils qui m'ont accompagné tout au long de ce parcours.

Je voudrais remercier les rapporteurs de ce travail messieurs et mesdames Zaher MOHDEB, Soheir BELALUOI, Nadjet ABADA pour avoir accepté de participer à mon jury de thèse.

Mon cher époux pour son soutien quotidien indéfectible, à mes enfants Mohamed, Abd Allah et Hazar.

Je remercie ma chère mère ainsi que mon frère et mes sœurs, pour leur soutien au cours de ces années et sans lesquels je n'en serais pas là aujourd'hui.

Je dédie ce travail à :

L'âme de mon père.

L'âme de monsieur Ahmed CHIBAT, leur soutien et leur encouragement m'ont toujours donné de la force pour persévérer et pour prospérer dans la vie.

Un honorable remerciement à mon très chère amie madame Dalel ZERDAZI pour son soutien et son aide surtout dans les moments les plus difficiles.

Je remercie toutes les personnes avec qui j'ai partagé mes études, notamment pendant ces années de thèse.

À tous ces intervenants, je présente mes remerciements, mon respect et ma gratitude.

Résumé

L'analyse des données (aussi appelée analyse exploratoire des données ou AED) est une famille de méthodes statistiques dont les principales caractéristiques sont d'être multidimensionnelles et descriptives. Ces méthodes peuvent également être considérées comme des méthodes neuronales particulières.

Dans ce travail de thèse, une mise au point sur les aspects statistiques des méthodes neuronales est proposée. Comme contribution innovante dans ce domaine, une technique par région en croissance est utilisée pour réaliser une segmentation d'images en fusionnant certains points de départ ou de petites zones internes si elles sont homogènes selon une mesure d'une propriété de la région locale. Un modèle autorégressif à coefficients aléatoires 2D RCA est ajusté afin d'identifier les différentes textures présentes dans l'image.

Tout d'abord, une procédure d'estimation utilisant la méthode des moments généralisée (GMM) est proposée pour extraire certaines propriétés de région locale. Pour cela, un réseau de neurones à base de gradient (GNN) est utilisé pour estimer les paramètres du modèle 2D RCA à partir d'une texture donnée. La fonction de coût du (GNN) proposée est basée sur une forte correspondance des moments statistiques du modèle 2D-RCA aux moments empiriques issus des données d'image. Les résultats expérimentaux démontrent l'efficacité et la pertinence de la méthode proposée.

Mots clés: Statistique exploratoire, Méthodes factorielles, Segmentation d'image; Modèles 2D RCA; ANNs; GMM.

Abstract

Data analysis (also called exploratory data analysis) is a family of statistical methods whose main characteristics are that they are multidimensional and descriptive. These methods can also be considered as special neural methods.

In this thesis work, a focus on the statistical aspects of neuronal methods is proposed. As an innovative contribution in this field, a region growing technique is used to achieve image segmentation by merging some starting points or internal small areas if they are homogeneous according to a measurement of a local region property. A 2D random coefficients autoregressive model (2D RCA) is fitted in order to identify the different textures present in the image. First, an estimation procedure using a generalized method of moments (GMM) technique is proposed to extract some local region properties. For this, a gradient-based neural network (GNN) is used to estimate the 2D RCA model parameters from a given texture. The cost function of the proposed (GNN) is based on a strong matching of the statistical moments of the corresponding 2D-RCA model and the sample moments of population image data. Experimental results demonstrate the effectiveness and the relevance of the proposed method.

Key words: Exploratory statistics, Factorial methods, Image segmentation; 2D RCA models; ANNs; GMM.

ملخص

تحليل البيانات (يسمى أيضاً تحليل البيانات الاستكشافية) هو مجموعة من الأساليب الإحصائية التي تتمثل خصائصها الرئيسية في أنها متعددة الأبعاد ووصفية. يمكن أيضاً اعتبار هذه الطرق طرقاً عصبية خاصة.

في هذا العمل، تم اقتراح التركيز على الجوانب الإحصائية للطرق العصبية. كمساهمة مبتكرة في هذا المجال، يتم استخدام تقنية نمو المنطقة لتحقيق تجزئة الصورة عن طريق دمج بعض نقاط البداية أو المساحات الداخلية الصغيرة إذا كانت متجانسة وفقاً لقياس خاصية المنطقة المحلية. تم تركيب نموذج الانحدار الذاتي ثنائي الأبعاد ذو المعاملات العشوائية من أجل تحديد القومات المختلفة الموجودة في الصورة.

أولاً، تم اقتراح إجراء تقدير باستخدام طريقة معمة لتقنية اللحظات لاستخراج بعض خصائص المنطقة المحلية. لهذا، يتم استخدام شبكة عصبية قائمة على التدرج لتقدير معالم النموذج المكاني من نسيج معين. تعتمد دالة التكلفة للمقترح على مطابقة قوية للحظات الإحصائية للنموذج المقابل ولحظات العينة لبيانات الصورة السكانية. تظهر النتائج التجريبية فعالية وملاءمة الطريقة المقترحة

الكلمات المفتاحية: الإحصاء الاستكشافي، طرق العوامل، تجزئة الصورة؛

2D RCA ؛ ANNs. GMM.

Table des matières

Introduction général	4
1 Les méthodes factorielles	9
1.1 Analyse en Composantes Principales	9
1.1.1 Les types de tableaux pouvant être traités par l'ACP	10
1.1.2 Avantages et inconvénients de l'ACP	15
1.2 Analyse Factorielle des Correspondances	16
1.2.1 Définitions et notations	17
1.2.2 Métrique du χ^2 pour l'évaluation de la liaison entre X et Y	20
1.2.3 Caractère significatif du χ^2	20
1.3 Analyse des correspondances multiples (ACM)	21
1.3.1 Cadre général de l'analyse des correspondances	21
1.3.2 ACM via une AFC sur la matrice des indicatrices	24
1.3.3 ACM via une AFC sur le tableau de Burt	25
1.3.4 ACM via une ACP sur le tableau des profils lignes	25
2 Les méthodes neuronales	27
2.1 Aperçu historique sur les réseaux de neurones	28
2.2 Le neurone formel	30
2.3 Apprentissage	30
2.3.1 L'apprentissage supervisé	31

2.3.2	L'apprentissage non supervisé	32
2.4	Types de réseaux de neurones	32
2.4.1	Réseaux de neurones à propagation avant	32
2.4.2	Réseaux de neurones récurrents	33
2.4.3	Réseaux de neurones convolutifs	33
2.5	Les tâches des réseaux de neurones	34
2.5.1	La régression	34
2.5.2	Les tâches de classification	35
2.6	Liens des réseaux de neurones avec l'analyse factorielle	36
3	Les méthodes de classification neuronales	37
3.1	Les données	37
3.2	Les objectifs	38
3.3	Classification supervisée par réseaux multicouches	38
3.4	Les principales méthodes neuronales de classification non supervisée	40
3.4.1	L'algorithme des cartes auto-organisatrices	40
3.4.2	Cartes topologiques et nuées dynamiques	44
3.4.3	Algorithme de Kohonen révisé	44
3.4.4	Version nuées dynamiques des cartes topologiques	45
3.4.5	Analogie avec l'algorithme de Kohonen	47
3.4.6	Carte topologique et approximation de la fonction densité	48
3.5	Intérêt et limitations de l'algorithme de Kohonen	49
3.5.1	Avantages	49
3.5.2	Limitations	49
3.6	Expérimentations	50
3.7	Conclusion	53
4	Combinaison d'un modèle 2D-RCA et des RNA pour la segmentation des images texturées	56
4.1	Définition du modèle et estimation des paramètres	58

4.1.1	Notations	58
4.1.2	Le modèle 2D-RCA	58
4.1.3	Estimation des paramètres du modèle 2D-RCA	59
4.2	Le processus d'apprentissage	61
4.3	Architecture ANN	62
4.3.1	Le sous réseau d'entrée	62
4.3.2	Le sous réseau d'analyse	63
4.3.3	Le sous réseau de classification	64
4.4	Segmentation d'images par regions en croissance	64
4.4.1	L'étape initiale	65
4.4.2	L'étape de raffinage	65
4.5	Résultats expérimentaux	66
4.6	Conclusion	70

Introduction général

Dans ce travail de thèse, nous nous intéressons aux différentes méthodes statistiques de prévision et de discrimination, à l'analyse des données, la reconnaissance de formes et les réseaux de neurones. Il s'agira beaucoup plus d'une mise au point sur les aspects statistiques des méthodes neuronales. Cette thèse a donc l'ambition de contribuer à une profonde compréhension des frontières entre les deux communautés des statisticiens et neuromiméticiens.

L'analyse des données consiste à construire des représentations simplifiées des données brutes, pour mettre en évidence les relations, les dominantes, la structure interne du nuage des observations. On peut distinguer deux grands groupes de techniques classiques : les méthodes factorielles et les méthodes de classification.

► Les méthodes factorielles :

Les méthodes factorielles sont essentiellement linéaires ; elles consistent à chercher des sous-espaces vectoriels, des changements de repères, permettant de réduire les dimensions tout en perdant le moins d'information possible. Les plus connues sont :

- Analyse en Composantes Principales (p variables quantitatives)
- Analyse Factorielle Discriminante (p variables quantitatives, 1 variable qualitative),
- Analyse Factorielle des Correspondances Binaire (2 variables qualitatives) et Multiple (p variables qualitatives),
- Analyse Canonique (p et q variables quantitatives),
- "Multidimensional Scaling" (M.D.S.) ou positionnement multidimensionnel ou analyse factorielle d'un tableau de distances.
- Analyse en Facteurs ("Factor Analysis"), ou analyse en facteurs communs et spécifiques

► les méthodes de classification :

Les méthodes de classification sont très nombreuses et diverses. Elles permettent de grouper et de ranger les observations. Les plus utilisées sont :

- Classification ascendante hiérarchique,
- Algorithmes de réallocation dynamique,
- Cartes de Kohonen (réseaux de neurones)

Les méthodes statistiques de prévision et de discrimination ont pour objectifs de prédire des variables à partir de prédicteurs issus d'un échantillon où les groupes d'appartenance ainsi que les variables explicatives sont connus. Les problèmes traités par ces méthodes apparaissent très souvent dans le traitement d'images, celui de la parole, ou le diagnostic. Il y a une panoplie de méthodes de prévisions qu'on peut souvent subdiviser en catégories, à savoir, les courbes de croissance, les moyennes mobiles, la décomposition saisonnière, le lissage exponentiel, la régression multiple, la méthode de Box et Jenkins, et plus récemment les méthodes neuronales.

Les réseaux de neurones artificiels (RNA) sont des approximateurs universels. Ils permettent d'ajuster des modèles non linéaires arbitrairement complexes à des données multidimensionnelles au niveau de précision souhaité Hornik et al (1989). En effet, les auteurs de Masters (1995) ont prouvé que les réseaux à trois couches sont suffisants pour approcher toute fonction discontinue. Ils peuvent être utilisés pour la reconnaissance de formes (en particulier la reconnaissance de la parole et des contorsions), la prédiction, l'analyse des risques, la classification, l'identification et le contrôle, et le regroupement. D'un autre côté, les capacités informatiques offertes par les ordinateurs modernes ont permis aux RNA de développer des applications industrielles à grande échelle. Cependant, les chercheurs en réseaux de neurones ont été stoppés dans leurs élan innovant par des problèmes que les statisticiens ont toujours connus. Ils se sont rendus compte qu'ils ne pouvaient plus se contenter de considérer des pratiques largement heuristiques pour développer leurs méthodes : la base théorique du choix de la taille de l'échantillon doit être fournie, la représentativité des données et l'importance des résultats et des estimations doivent être rigoureusement établies, le choix de l'architecture du réseau doit être menée avec précision. Ainsi, ils ont reconstruit des méthodes qui sont connues dans la littérature statistique depuis des décennies ou des siècles mais, souvent, sans comprendre comment ces méthodes fonctionnent Specht (1991). Ils se sont donc tournés vers la statistique où Vladimir Vapnik a été l'un des premiers, au début des années 80, à introduire le raisonnement et la logique de la statistique dans les réseaux de neurones Vapnik (1982), Vapnik (1995). D'autres statisticiens purs se sont également intéressés aux RNA et ont commencé à produire des bases théoriques pour eux Friedman (1994), Mac Kay (1992a), Mac Kay (1992b), Mac Kay (1992c). Par la suite, les RNA et les statistiques ont connu un rapprochement spectaculaire au point où la littérature est restée confuse quant à savoir si les

RNA ne seraient pas simplement une catégorie particulière de modèles statistiques. Les travaux publiés qui donnent un bon aperçu de la relation entre les statistiques et les réseaux de neurones sont rares. L'une des meilleures références à cette fin, à ce jour, est certainement celle de Ripley (1993).

L'une des principales tâches des RNA est l'apprentissage des réseaux de neurones à partir des données. Cette tâche est effectuée à peu près de la même manière que l'estimation de nombreux modèles statistiques. En fait, en fin de compte, l'apprentissage et l'estimation sont deux approches indépendantes qui convergent. À partir de là, le monde scientifique a vu l'émergence de la théorie de l'apprentissage statistique Hastie et al (2001). Le but de l'apprentissage statistique est de combiner l'apprentissage automatique, comme les RNA, et les méthodes d'estimation statistique afin de modéliser une distribution de probabilité de manière de plus en plus algorithmique.

Notre contribution à ce sujet consiste dans la construction d'un nouveau algorithme pour la segmentation supervisée des images texturées basée sur les techniques des RNA. La segmentation d'images est un problème difficile car il n'existe pas de modèle mathématique général, proposé dans la littérature, susceptible d'être appliqué à tous les types d'images.

Les années 90 ont vu l'essor des méthodes basées sur les RNA comme une approche différente de la segmentation d'image. Leur mise en œuvre facile et leur capacité à être utilisées dans des applications en temps réel leur permettent de jouer un rôle de premier plan dans presque toutes les disciplines avec un degré de réussite différent. Les RNA de Kohonen et Hopfield sont les plus utilisés dans la pratique. On peut consulter Pal et Pal (1993) et Tang et al (2007) pour une brève revue des méthodes RNA utilisées dans la segmentation d'images.

Pal et Pal (1993) ont divisé les techniques de segmentation d'images en quatre approches :

- a) seuillage du niveau de gris ;
- b) segmentation basée sur les pixels ;
- c) segmentation basée sur la région ;
- d) segmentation des images couleurs.

Le modèle de champ aléatoire de Markov (MRF) et le modèle autorégressif 2D (AR) sont deux modèles stochastiques fréquemment utilisés pour la représentation des caractéristiques de texture. Dans de telles méthodes de modèle stochastique, une texture est considérée comme une

réalisation d'un champ aléatoire. Les paramètres du modèle sont ensuite estimés et choisis comme caractéristiques de texture. La méthode de segmentation proposée peut être classée comme une méthode basée sur la région en utilisant un critère de similarité construit sur les paramètres du modèle autorégressif à coefficients aléatoires 2D RCA. Le modèle 2D RCA a été récemment introduit par les auteurs de Boulemdjadjel et al (2016) pour capturer le comportement variant dans l'espace de la volatilité. Ils ont utilisé avec succès l'approche de la méthode des moments généralisés (GMM) pour estimer les paramètres du modèle et les ont considérés comme des caractéristiques de pixels dans la classification des images de texture. Dans ce travail de thèse, nous proposons une procédure d'estimation des paramètres du modèle 2D-RCA à l'aide d'une technique GMM implémentée avec un réseau de neurones à gradient (GNN). Les auteurs de Weilu et Xu (1995) avaient déjà utilisé un réseau de neurones pour estimer les paramètres d'un modèle autorégressif 2D (2D AR), censé représenter une image texturée donnée, en minimisant une erreur totale correspondant à la différence entre l'image et celle estimée. Leur méthode est formulée comme une fonction de coût, basée sur la similitude entre l'image texturée et celle obtenue à partir du modèle AR ajusté, qui doit ensuite être minimisée. Dans l'approche proposée, le problème est formulé de la même manière mais la fonction de coût est basée sur une forte correspondance des moments statistiques du modèle 2D-RCA correspondant et des moments d'échantillonnage des données empiriques d'image. Cela fait de l'approche proposée une méthode d'information partielle, dérivée d'un ensemble minimal de conditions de moment.

Le manuscrit est composé de quatre chapitres précédés d'une introduction générale :

- Le premier chapitre est consacré aux méthodes neuronales ainsi qu'aux différents concepts qui s'y rattachent utilisés tout au long de cette thèse.
- On s'intéresse dans le deuxième chapitre aux méthodes factorielles pour dont on va préciser les principes.
- Le troisième chapitre sera consacré à la notion de classification qui sera placée dans son cadre théorique complet.
- Dans le quatrième chapitre nous présentons les principaux résultats que nous avons obtenus tout au long de nos années de recherche. Nous proposons ainsi une technique de segmentation basée sur la région en croissance et qui utilise les modèles autorégressifs à coefficients

aléatoires bidimensionnels et des techniques neuronales pour sa réalisation.

Chapitre 1

Les méthodes factorielles

L'analyse factorielle est une technique statistique déjà centenaire représentant une famille de méthodes dont l'objectif principal est de résumer l'information apportée par un grand nombre de variables permettant ainsi de réduire la dimension d'un domaine empirique donné. Les méthodes factorielles, qui sont apparues au début des années 30, ont été surtout développées en France dans les années 60, en particulier par Jean-Paul Benzècri qui a beaucoup exploité les aspects géométriques et les représentations graphiques. Dans la mesure où ce sont des méthodes descriptives, elles ne s'appuient pas sur un modèle probabiliste, mais elles dépendent d'un modèle géométrique.

L'objectif de ce chapitre est de présenter de façon simple les principes de base des méthodes d'analyse factorielle, et plus particulièrement de l'analyse en composantes principales et de l'analyse factorielle des correspondances. L'accent sera mis sur les éléments utiles lors de l'interprétation des résultats et les méthodes seront illustrées par des exemples élémentaires, traités de façon détaillée.

1.1 Analyse en Composantes Principales

L'Analyse en composantes principales (ACP) fait partie du groupe des méthodes descriptives multidimensionnelles appelées méthodes factorielles.

L'ACP propose, à partir d'un tableau rectangulaire de données comportant les valeurs de p

variables quantitatives pour n unités (appelées aussi individus), des représentations géométriques de ces unités et de ces variables. Ces données peuvent être issues d'une procédure d'échantillonnage ou bien de l'observation d'une population toute entière. Les représentations des unités permettent de voir s'il existe une structure, non connue a priori, sur cet ensemble d'unités. De façon analogue, les représentations des variables permettent d'étudier les structures de liaisons linéaires sur l'ensemble des variables considérées. Ainsi, on cherchera si l'on peut distinguer des groupes dans l'ensemble des unités en regardant quelles sont les unités qui se ressemblent, celles qui se distinguent des autres, etc. Pour les variables, on cherchera quelles sont celles qui sont très corrélées entre elles, celles qui, au contraire ne sont pas corrélées aux autres, etc.

Nous verrons après l'exposé de la méthode, quelles précautions il faut prendre pour interpréter correctement les représentations obtenues. Dans tous les cas, il ne faut pas oublier d'où sont issues les données utilisées et ce qu'elles représentent et signifient pour le problème que l'on se pose.

Enfin, comme pour toute méthode descriptive, réaliser une ACP n'est pas une fin en soi. L'ACP servira à mieux connaître les données sur les quelles on travaille, à détecter éventuellement des valeurs suspectes, et aidera à formuler des hypothèses qu'il faudra étudier à l'aide de modèles et d'études statistiques différentielles. On pourra aussi, a posteriori, se servir des représentations fournies par l'ACP pour illustrer certains résultats dans un but pédagogique.

1.1.1 Les types de tableaux pouvant être traités par l'ACP

L'ACP s'applique à des tableaux à deux dimensions croisant des individus et des variables quantitatives ou pouvant être considérées comme telles. Selon la nature de ces variables, on distingue trois grandes catégories de tableaux qui peuvent être traités par l'ACP, à savoir :

a. Les tableaux de mesures : les variables sont obtenues à partir de comptages ou de recensements. Ces variables, continues ou entières, sont quantitatives.

Exemple : le RGP (Recensements Général de la Population) en 2008.

b. Les tableaux de notes : Les variables sont obtenues à partir de notations. Les notes sont des (variables qualitatives ordinales) qui peuvent être généralement assimilées à des variables quantitatives.

c. Les tableaux de rangs : Les variables sont des rangs, les n individus sont classés de 1 à n ,

du meilleur au plus mauvais, du plus rapide au plus lent, etc.

On possède donc un tableau rectangulaire de mesure (comme une matrice) dont les colonnes sont des variables (mensurations, taux, températures, ...) et dont les lignes représentent des individus statistiques (unités élémentaires telles que des êtres humains, des pays, des années ...). On note \mathbf{X} ce tableau de données.

$$\mathbf{X} = \begin{bmatrix} X_{11} & \dots & \dots & \dots & X_{1p} \\ \vdots & \dots & X_{ij} & \dots & \vdots \\ X_{n1} & \dots & \dots & \dots & X_{np} \end{bmatrix}$$

\mathbf{X} : Tableau de données

- X_{ij} : Valeur de la i^{eme} observation pour la j^{eme} variable.
- $X_{i.}$: i^{eme} observation du tableau.
- $X_{.j}$: j^{eme} variable du tableau.
- n : Effectif des individus.
- p : Nombre de variable.

Ainsi dans une ACP, il s'agit de synthétiser les données contenus dans le tableau \mathbf{X} . Pour cela, on construit un petit nombre de variables, $\mathbf{C}^1, \mathbf{C}^2, \dots$ appelées composantes principales permettant de saisir l'essentiel du tableau \mathbf{X} . En résumé, les p variables seront représentées dans un sous-espace F_k de \mathbb{R}^p de dimension k (k petit 2, 3...). Les axes qu'elles déterminent seront appelés « axes principaux » et les formes linéaires associées « facteurs principaux ».

Pratiquement, à l'étape 1, on détermine une variable synthétique \mathbf{C}^1 appelée première composante principale, combinaison linéaire des variables X_j .

$$\mathbf{C}^1 = \mathbf{a}_1^1 X_{.1} + \dots + \mathbf{a}_j^1 X_{.j} + \dots + \mathbf{a}_m^1 X_{.m}$$

Ce qui signifie que la valeur de \mathbf{C}^1 pour l'individu i est donnée par :

$$\mathbf{C}_i^1 = \mathbf{a}_1^1 X_{i1} + \dots + \mathbf{a}_j^1 X_{ij} + \dots + \mathbf{a}_m^1 X_{im}$$

Cette première composante principale ne suffit généralement pas pour résumer de façon sa-

tisfaisante les données du tableau \mathbf{X} , aussi, on construit une deuxième composante principale, puis une troisième. . .

De façon générale, à l'étape k , on construit la composante d'ordre k ,

$$\mathbf{C}^k = \mathbf{a}_1^k X_{.1} + \dots + \mathbf{a}_j^k X_{.j} + \dots + \mathbf{a}_m^k X_{.m}$$

Matriciellement, $\mathbf{C}^k = \mathbf{X}\mathbf{a}^k$ ou \mathbf{a}^k est un vecteur colonne à m éléments, l'élément d'ordre j étant égal à \mathbf{a}_j^k , le vecteur \mathbf{a}^k est appelé facteur d'ordre k (ou $k^{\text{ième}}$ facteur).

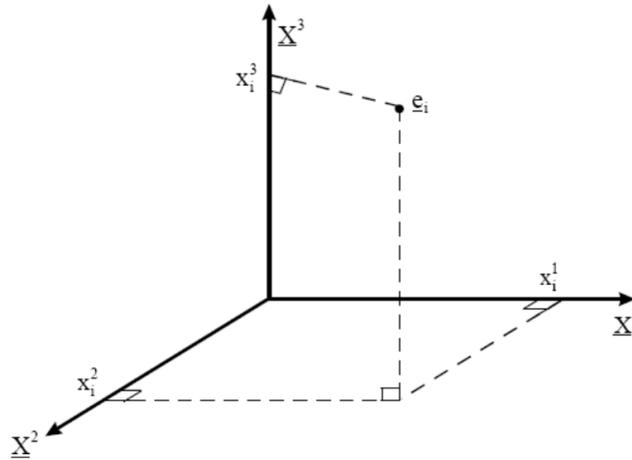
Les facteurs fournissent un système de « poids » pour les variables ; certains « poids » \mathbf{a}_j^k sont négatifs, d'autres positifs. En fait ce qui importe n'est pas la valeur de chacun de ces poids, mais le rapport de ces « poids » les uns par rapport aux autres. Les facteurs sont donc définis à une constante multiplicative près, aussi, une contrainte de normalisation pour chacun des facteurs est imposée :

$$\sum_{j=1}^m (\mathbf{a}_j^k)^2 = 1.$$

Les composantes principales sont des variables de moyenne nulle, puisque les variables d'origine sont centrées, la valeur pour l'individu i de la composante principale k est :

$$\mathbf{C}_i^k = \mathbf{a}_1^k \mathbf{X}_{i1} + \dots + \mathbf{a}_j^k \mathbf{X}_{ij} + \dots + \mathbf{a}_m^k \mathbf{X}_{im}.$$

Le principe de l'ACP dans la construction des composantes principales est de perdre le moins d'information possible. Pour cela le sous espace F_k devra être « ajusté » le mieux possible au nuage des points du tableau : F_k est le sous-espace tel que le nuage de points projeté ait une inertie totale (dispersion) maximale. L'inertie totale exprime la quantité d'information portée par les données. Elle repose dans ce cas sur les distances et les projections orthogonales. il est possible de la définir de deux manières : à travers les distances entre paires d'individus ou entre paires de variables selon que l'on considère le tableau \mathbf{X} comme un ensemble de lignes ou un ensemble de colonnes. C'est pour cela, deux études peuvent éventuellement être considérées.



Etude des individus

L'objectif principal de cette étude est de faire un bilan des ressemblances entre individus. Pour cela, on cherche d'abord à représenter le nuage des individus. A chaque individu noté e_i , on peut associer un point dans $R^p =$ espace des individus.

A chaque variable du tableau \mathbf{X} est associée un axe de \mathbb{R}^n , impossible à visualiser dès que $p > 3$. L'étude des individus équivaut à l'étude de la forme du nuage de points ainsi obtenu. Pour pouvoir étudier la structure du nuage, il est indispensable de réduire et centrer les données notamment si les unités de mesure sont différentes d'une variable à l'autre. On parle alors d'analyse en composante principale normée. D'autre part, la notion de ressemblance entre individus équivaut à l'étude des distances entre ces individus. Dans \mathbb{R}^p , les $n(n-1)$ distances associées aux couples de points représentant les individus i et i' données par,

$$d^2(i, i') = \sum_{j=1}^p \left(\frac{X_{ij} - X_{i'j}}{S_j \sqrt{n}} \right)^2, \quad \text{avec } S_j^2 = \frac{1}{n} \sum_{i=1}^n (X_{ij} - \bar{X}_j)^2, \quad \text{et } \bar{X}_j = \frac{1}{n} \sum_{i=1}^n X_{ij}$$

ont une interprétation directe dans l'étude de la structure du nuage des individus.

Il s'agit ici de la distance euclidienne classique. Donc, si les p coordonnées de deux points sont très proches les deux points sont très voisins. En bref, l'analyse du nuage des points-individus dans \mathbb{R}^p s'est faite grâce à une translation de l'origine au centre de gravité de ce nuage menant à un changement des échelles sur les différents axes. Toutes les variables $R_{ij} = \frac{X_{ij} - X'_{ij}}{S_j \sqrt{n}}$ ainsi transformées sont "comparables" et ont même dispersion. L'inertie est alors la somme pondérée des carrés des distances des individus au centre de gravité. Cette inertie mesure la dispersion totale du nuage de points.

Etude des variables

La proximité entre deux points variables j et j' , prises à partir du tableau \mathbf{X} , découle de l'analyse précédente dans \mathbb{R}^p . Rappelons que la distance euclidienne usuelle entre deux variables j et j' est donnée par :

$$d^2(j, j') = \sum_{i=1}^n (R_{ij} - R_{ij'})^2,$$

D'où la relation liant la distance dans \mathbb{R}^n entre deux point(-variables j et j' et le coefficient de corrélation $c_{jj'}$ entre les variables s'écrit :

$$d^2(j, j') = 2(1 - c_{jj'}) = \cos(j, j'). \quad (1.1)$$

L'interprétation du système de proximités entre points-variables induit par la relation (1.1) se fait en termes de corrélations.

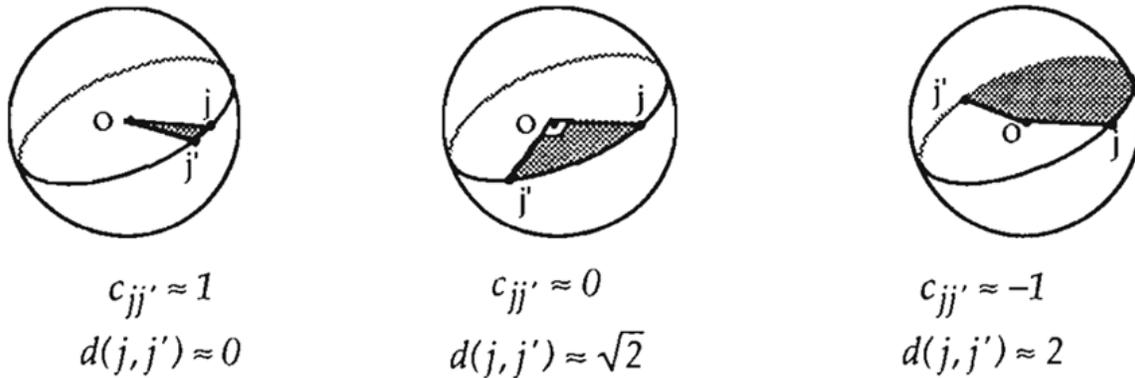


Figure 1.1. Corrélations et distances entre points-variables

Ainsi, l'analyse du tableau transformé \mathbf{X} nous conduit à diagonaliser une matrice de corrélation définie par $\mathbf{C} = \mathbf{X}'\mathbf{X}$ où le terme général c_{ij} n'est autre que le coefficient de corrélation empirique entre les variables j et j' . En conclusion, La recherche d'axes portant le maximum d'inertie équivaut à la construction de nouvelles variables (auxquelles sont associés ces axes) de variance maximale. En d'autres termes, on effectue un changement de repère dans \mathbb{R}^p de façon à se placer dans un nouveau système de représentation où le premier axe apporte le plus possible de l'inertie totale du nuage, le deuxième axe le plus possible de l'inertie non prise en compte par le premier axe, et ainsi de suite. Cette réorganisation s'appuie sur la diagonalisation de la matrice de variances-covariances.

1.1.2 Avantages et inconvénients de l'ACP

Avantages

Simplicité mathématique :

Il s'agit seulement de combinaisons linéaires, les seuls véritables outils mathématiques utilisés dans l'ACP sont le calcul des valeurs/vecteurs propres d'une matrice, et les changements de base. Sur le plan mathématique, l'ACP est donc une méthode simple à mettre en œuvre.

Simplicité des résultats :

Grâce aux graphiques qu'elle fournit, l'Analyse en Composantes Principales permet d'appréhender une grande partie de ses résultats d'un simple coup d'œil.

Puissance :

L'ACP a beau être simple, elle n'en est pas moins puissante. Elle offre, en quelques opérations seulement, un résumé et une vue complète des relations existant entre les variables quantitatives d'une population d'étude, résultats qui n'auraient pas pu être obtenus autrement, ou bien uniquement au prix de manipulations fastidieuses.

Flexibilité :

L'ACP est une méthode très souple, puisqu'elle s'applique sur un ensemble de données de contenu et de taille quelconques, pour peu qu'il s'agisse de données quantitatives organisées sous forme individus/variables. Cette souplesse d'utilisation se traduit surtout par la diversité des applications de l'ACP, qui touche tous les domaines.

Inconvénients

En tant que méthode d'analyse de données, l'ACP n'a pas réellement d'inconvénients en soi. Elle s'applique simplement sur des cas précis et pour générer un type de résultat particulier. Ce la n'aurait donc aucun sens de dire que c'est un inconvénient de l'ACP qu'elle ne s'applique pas en dehors de ce contexte. De même, étant donné qu'il s'agit avant tout d'une technique de résumé de données, la perte d'informations forcément engendrée n'est pas un inconvénient, mais plutôt une condition d'obtention du résultat, même si elle occulte parfois des caractéristiques pourtant représentatives dans certains cas particuliers.

1.2 Analyse Factorielle des Correspondances

Comme l'analyse en composantes principales, l'analyse des correspondances peut être présentée selon divers points de vue. Il est d'ailleurs difficile de faire l'historique précis de cette méthode. Cette méthode n'est pas un cas particulier de l'analyse en composantes principales bien que l'on puisse se ramener à cette technique en faisant des changements de variables appropriés (à condition de traiter chaque espace séparément). On peut montrer qu'il s'agit de la

recherche de la meilleure représentation simultanée de deux ensembles constituant les lignes et les colonnes d'un tableau de données. L'AFC cherche au fait à décrire la liaison entre deux variables qualitatives.

1.2.1 Définitions et notations

L'approche pour réaliser une AFC que nous considérons dans ce travail consiste à définir l'Analyse Factorielle des Correspondances (AFC) comme étant le résultat d'une double Analyse en Composantes Principales :

- l'ACP des profils-lignes,
- l'ACP des profils-colonnes,

relativement à la métrique dite du χ^2 . Cette métrique sera présentée à la section 2 de ce chapitre.

Dans toute la suite, on considère un tableau de deux variables qualitatives X et Y , observées simultanément sur n individus telles que :

- X a I modalités.
- Y a J modalités.
- On suppose $I > J$

Table de contingence (permet d'analyser la correspondance entre X et Y)

X/Y	y_1	...	y_j	...	y_J	Total
x_1	n_{11}		n_{1j}		n_{1J}	$n_{1.}$
⋮						
x_i	n_{i1}		n_{ij}		n_{iJ}	$n_{i.}$
⋮						
x_I	n_{I1}		n_{Ij}		n_{IJ}	$n_{I.}$
Total	$n_{.1}$		$n_{.j}$		$n_{.J}$	n

n_{ij} = Nombre d'observations ayant la modalité x_i de X et y_j de Y .

$n_{i.}$ =effectif marginal : Nombre d'observations ayant la modalité x_i de X

$n_{.j}$ = effectif marginal : Nombre d'observations ayant la modalité y_j de Y.

On déduit de ce tableau un ensemble de fréquences particulières d'où on déduit une matrice de profil-lignes PL et une matrice des profil-colonnes PC .

Matrice des Fréquences :

Fréquence de (i, j) : $f_{ij} = \frac{n_{ij}}{n}$.

Fréquence marginale de la modalité i : $f_{i.} = \frac{n_{i.}}{n}$.

Fréquence marginale de la modalité j : $f_{.j} = \frac{n_{.j}}{n}$.

Fréquence conditionnelle j sachant i : $f_i^j = \frac{n_{ij}}{n_{i.}} = \frac{f_{ij}}{f_{i.}}$.

Fréquence conditionnelle i sachant j : $f_j^i = \frac{n_{ij}}{n_{.j}} = \frac{f_{ij}}{f_{.j}}$.

$$\sum_{i=1}^I \sum_{j=1}^J n_{ij} = \sum_{i=1}^I n_{i.} = \sum_{j=1}^J n_{.j} = n; \quad \sum_{i=1}^I \sum_{j=1}^J f_{ij} = \sum_{i=1}^I f_{i.} = \sum_{j=1}^J f_{.j} = 1,$$

$$\sum_{j=1}^J n_{ij} = n_{i.}, \quad \sum_{i=1}^I n_{ij} = n_{.j}, \quad \sum_{j=1}^J f_{ij} = f_{i.}, \quad \sum_{i=1}^I f_{ij} = f_{.j}$$

Profiles

- On appelle i° profil ligne, le vecteur de dimension J des fréquences de la variable Y conditionnellement à la valeur x_i de X :

$$PL^{(i)} = (f_i^1, \dots, f_i^j, \dots, f_i^J).$$

- *Tableau des profils ligne*

$$PL = \begin{array}{|c|c|c|c|c|} \hline f_1^1 & \dots & f_1^j & \dots & f_1^J \\ \hline \vdots & & \vdots & & \vdots \\ \hline f_i^1 & \dots & f_i^j & \dots & f_i^J \\ \hline \vdots & & \vdots & & \vdots \\ \hline f_I^1 & \dots & f_I^j & \dots & f_I^J \\ \hline \end{array}$$

- PL permet de comparer la répartition des valeurs de Y dans les différentes modalités de X.

- On appelle j° profil colonne, le vecteur de dimension I des fréquences de la variable X

conditionnellement à la valeur j de Y :

$$PC^{(j)} = (f_j^1, \dots, f_j^j, \dots, f_j^I).$$

- *Tableau des profils colonnes*

$$PC = \begin{array}{|c|c|c|c|c|} \hline f_1^1 & \dots & f_1^i & \dots & f_1^I \\ \hline \vdots & & \vdots & & \vdots \\ \hline f_j^1 & \dots & f_j^i & \dots & f_j^I \\ \hline \vdots & & \vdots & & \vdots \\ \hline f_J^1 & \dots & f_J^i & \dots & f_J^I \\ \hline \end{array}$$

- PC permet de comparer la répartition des valeurs de X dans les différentes modalités de Y .

La relation entre X et Y peut être évaluée de deux manières :

- En probabilité, s'il y a indépendance entre X et Y , on a :

$$(1) \forall (i, j) P(X = i \text{ et } Y = j) = P(X = i) P(Y = j)$$

$$(2) \forall (i, j) P(X = i / Y = j) = P(X = i)$$

$$(2) \forall (i, j) P(/Y = j / X = i) = P(Y = j)$$

- En statistique, ces relations équivalent à :

$$(1) \forall (i, j) f_{ij} = f_{i.} \times f_{.j} \iff (f = p, f = (f_{ij}), p = f_{i.} \times f_{.j})$$

$$(2) \forall (i, j) f_j^i = f_{i.} \iff (Y^{(j)} = p_X, p_X = (f_{i.}))$$

$$(2) \forall (i, j) f_i^j = f_{.j} \iff (X^{(i)} = p_Y, p_Y = (f_{.j}))$$

- En conclusion, lorsque X et Y sont indépendants

- Toutes les distributions conditionnelles de X/Y (profils colonnes) sont égales. Elles sont toutes égales à la loi marginale de X (profil moyen)

- Toutes les distributions conditionnelles de Y/X (profils lignes) sont égales. Elle sont toutes égales à la loi marginale de Y (profil moyen)

- La distribution jointe (fréquence) est égale au produit des marginales (fréquences marginales)

Remarques

1. Toute structure d'ordre existant éventuellement sur les modalités de X ou de Y est ignorée par l'AFC
2. Tout individu présente une modalité et une seule de chaque variable.
3. Chaque modalité doit avoir été observée au moins une fois ; sinon, elle est supprimée

1.2.2 Métrique du χ^2 pour l'évaluation de la liaison entre X et Y

La mesure de la liaison entre X et Y se fait en évaluant un des écarts entre distributions "profils colonnes-profile moyen X" ou "profils lignes-profile moyen Y" ou "distribution jointe-produit des marginales". Pour cela, nous devons définir la notion de distance entre distributions.

Définition 1.1. Si l'on considère trois distributions de probabilités o, q et r, on appelle distance du χ^2 entre o et q centrée sur r la quantité :

$$\chi_r^2 = \sum_i \frac{(o_i - q_i)^2}{r_i}.$$

Dans le cas de l'AFC, la mesure de la liaison entre X et Y est donnée par :

$$\chi_{AFC}^2 = n \sum_{i,j} \frac{(f_{ij} - f_{i.}x f_{.j})^2}{f_{i.}x f_{.j}},$$

Alors au plus χ_{AFC}^2 est grand au plus la correspondance (dépendance) est forte entre les deux variables.

1.2.3 Caractère significatif du χ^2

La question qui s'impose à ce stade, à partir de quelle valeur de χ_{AFC}^2 doit-on considérer que les variables X et Y sont indépendantes ? Pour répondre à cette question, la stratégie adoptée est la suivante : on suppose que X et Y sont issues de tirages de deux variables aléatoires indépendantes. On peut alors montrer que χ_{AFC}^2 est une réalisation d'une variable aléatoire qui suit une loi $\chi_{(I-1)(J-1)}^2$. On effectue alors un test du χ^2 : on se fixe un risque d'erreur α (0.01 ou 0.05 en général) et on calcule la valeur d_c^2 telle que $P(\chi_{(I-1)(J-1)}^2 > d_c^2) = \alpha$;

- si $\chi_{AFC}^2 > d_c^2$, on considère que l'événement est trop improbable et que donc l'hypothèse originale d'indépendance doit être rejetée. On trouvera en général ces valeurs dans une table pré calculée.

1.3 Analyse des correspondances multiples (ACM)

L'analyse des correspondances multiples (ACM) est une extension de l'analyse en composantes principales qui s'appliquerait à des tableaux constitués de variables qualitatives. Elle a vu le jour à partir des années 1960 par l'« école française d'analyse des données » qui plaidait pour une pratique statistique plus inductiviste que celle qui domine dans la tradition anglo-saxonne.

Nous disposons de n individus décrits par p variables qualitatives X_1, \dots, X_p à m_1, \dots, m_p modalités.

1.3.1 Cadre général de l'analyse des correspondances

L'analyse des correspondances est une méthode adaptée aux tableaux de contingence et permet d'étudier les éventuelles relations entre deux variables nominales. Il est tout à fait aisé de montrer qu'il s'agit de la recherche de la meilleure représentation simultanée de deux ensembles constituant les lignes et les colonnes d'un tableau de nombres positifs.

Pour une description complète de cette méthode, ses applications et le détail des calculs, on peut se référer à l'ouvrage de Greenacre (1984). Ces méthodes ont été initialement développées en France par Jean-Paul Benzécri au début des années 1960 et dans les années 1970 (par exemple, voir Benzécri, 1973 ; voir également Lebart, Morineau, et Tabard, 1977). Notez que des techniques similaires ont été développées indépendamment dans différents pays, où elles sont plus connues sous les noms de graduation optimale, de moyennes réciproques, de scoring optimal, de méthode de quantification ou d'analyse d'homogénéité.

Dans une analyse des correspondances "typique", les effectifs du tableau sont tout d'abord standardisés, de sorte que les fréquences (relatives) de toutes les cellules totalisent 1,0. Pour schématiser l'objectif d'une analyse type, disons que l'on cherche à représenter les entrées du tableau de fréquences en termes de distances entre les lignes et/ou colonnes individuelles dans

	Catégorie de Fumeur				
Position	(1)	(2)	(3)	(4)	Totaux
Hiérarchique	Non Fumeur	Occasionnel	Moyen	Gros Fumeur	Lignes
(1) Cadre Exp.	4	2	3	2	11
(2) Cadre Déb.	4	3	7	4	18
(3) Employé Exp.	25	10	12	4	51
(4) Employé Déb.	18	24	33	13	88
(5) Secrétaire	10	6	7	2	25
Totaux Colonnes	61	45	62	25	193

un espace à peu de dimensions.

Nuages de points-lignes et points colonnes

Contrairement à l'ACP, deux transformations distinctes sont appliquées au tableau de données, l'une sur les profils-lignes, l'autre sur sur les profils colonnes, à partir desquelles vont être construits les nuages de points dans \mathbb{R}^p et \mathbb{R}^n .

Critères à minimiser et matrice

Pour représenter les proximités entre profils, il faut se placer aux centres de gravité des nuages. Cependant, et c'est là une particularité de l'AC, il est équivalent de procéder à l'analyse par rapport à l'origine ou par rapport aux centres de gravité, à condition d'écarter dans le premier cas l'axe principal qui joint l'origine au centre de gravité.

L'exemple ci-dessous illustrera sans doute mieux ces propos. Il y a un certain parallèle entre l'interprétation d'une analyse des correspondances et d'une Analyse Factorielle, et nous mettrons en évidence certains concepts similaires aux deux techniques.

Exemple 1.1 (exemple tiré du site <https://www.statsoft.fr/>)

Supposons que vous ayez collecté des données sur les habitudes de différents salariés d'une entreprise concernant la cigarette. Les données suivantes sont présentées dans l'ouvrage de Greenacre (1984, p. 55); ces données sont aussi fournies dans le fichier d'exemple Smoking.sta.

Vous pouvez considérer les 4 valeurs des colonnes pour chaque ligne comme des coordonnées

dans un espace à 4 dimensions, et calculer les distances (euclidiennes) entre les 5 points lignes de cet espace à 4 dimensions. Les distances entre les points de cet espace à 4 dimensions synthétisent toute l'information concernant les similitudes entre les lignes du tableau ci-dessus. Maintenant, supposez que vous pouvez trouver un espace avec moins de dimensions, dans lequel placer les points ligne de sorte à conserver toute l'information ou presque, sur les différences entre les lignes. Vous pouvez alors représenter toute l'information sur les similitudes entre les lignes (position hiérarchique dans ce cas) dans un graphique simple à 1, 2, ou 3 dimensions. Si cette procédure peut sembler d'un intérêt limité pour traiter de petits tableaux comme celui ci-dessus, vous pouvez aisément imaginer l'intérêt dans la représentation et l'interprétation des tableaux plus conséquents (par exemple, les préférences différentielles de 10 produits entre 100 groupes de prospects dans une enquête de consommation) grâce à la simplification induite par l'analyse des correspondances (par exemple, représenter les 10 produits dans un espace à deux dimensions).

Masse. Pour poursuivre avec l'exemple élémentaire de tableau à double entrée reporté ci-dessus, le programme va tout d'abord calculer les fréquences relatives de la table de fréquences, de sorte que la somme de toutes les entrées de la table totalise 1,0 (chaque élément sera rapporté au total, c'est-à-dire 193). On pourrait dire que cette table donne maintenant la manière dont les unités de masse sont distribuées entre les cellules. Dans la terminologie de l'analyse des correspondances, les totaux ligne et colonne de la matrice des fréquences relatives sont appelés respectivement masse ligne et masse colonne.

Inertie. Le terme Inertie dans l'analyse des correspondances est utilisé par analogie avec la définition du "moment d'inertie" en mathématiques appliquées, qui représente l'intégrale de la masse multipliée par la distance au carré au centre de gravité (par exemple, Greenacre, 1984, p. 35). L'inertie se définit comme le χ^2 total de Pearson de la table d'ordre deux (par exemple, tel qu'il est également calculé dans les modules Statistiques Élémentaires ou Analyse Log-Linéaire) rapporté à la somme totale (193 dans notre exemple).

Inertie et profils ligne et colonne. Si les lignes et colonnes d'une table sont complètement indépendantes entre elles, les entrées de la table (distribution de la Masse) peuvent être reproduites à partir des totaux ligne et colonne seuls (totaux marginaux), ou profils ligne et colonne dans la terminologie de l'analyse des correspondances. Selon la formule de calcul bien connue

du Chi^2 pour des tables à double entrée, la fréquence théorique (attendue) d'une table, où les colonnes et lignes sont indépendantes entre elles, est égale au total de la colonne respective multiplié par le total ligne, et rapporté au total général. Tout écart aux valeurs théoriques (attendues sous l'hypothèse d'indépendance complète des variables lignes et colonnes) va contribuer à la statistique globale du Chi^2 (voir la rubrique *Détail des Calculs*). Par conséquent, une autre manière d'appréhender l'analyse des correspondances revient à considérer qu'il s'agit d'une manière de décomposer la statistique globale du Chi^2 (ou Inertie= Chi^2/N Total) en identifiant un nombre restreint de dimensions dans lesquelles les écarts aux valeurs théoriques peuvent être représentés. L'objectif de l'Analyse Factorielle, est similaire, puisque la variance totale est décomposée, de façon à permettre une représentation des variables sur quelques dimensions, nous permettant de reconstruire l'essentiel de la matrice de variance/covariance des variables.

Analyser les lignes et colonnes. Nous nous sommes attachés dans cet exemple simple à la présentation des points lignes dans le tableau précédent. Toutefois on aurait pu s'intéresser plutôt aux totaux colonnes, et dans ce cas, tracer les points colonnes dans un espace à quelques dimensions reproduisant de façon satisfaisante les similitudes (et distances) entre les fréquences relatives des colonnes, selon les lignes, dans le tableau ci-dessus. En fait, il est courant de tracer simultanément les points colonne et les points ligne dans un seul graphique afin de synthétiser l'information contenue dans le tableau à double entrée.

L'ACM est la généralisation la plus simple de l'AC au cas où plus de deux ensembles sont mis en correspondance. Elle peut se réaliser de trois manières : Par une AFC sur la matrice des indicatrices, par une AFC sur le tableau de Burt ou par une ACP sur le tableau des profils lignes (ou sur le tableau des profils colonnes aussi d'ailleurs).

1.3.2 ACM via une AFC sur la matrice des indicatrices

Définissons d'abord le tableau des indicatrices : le tableau des indicatrices, appelé aussi tableaux disjonctifs complets, s'obtient en juxtaposant les p tableaux d'indicatrices de chaque variable X_i .

- La somme des éléments de chaque ligne du tableau des indicatrices est égale à p
- La somme des éléments d'une colonne de ce tableau donne l'effectif marginal de la catégorie

Avec l'AFC, on va analyser :

1. Les relations entre les modalités (colonnes)
2. Les proximités entre les individus (lignes)
3. Les association « individus x modalités »

L'analyse à partir du tableau disjonctif complet fournit des représentations des barycentres de classes d'individus en calculant les axes d'inertie du nuage d'individus, puis projeter les barycentres sur ces axes.

1.3.3 ACM via une AFC sur le tableau de Burt

L'hypertable étant problématique pour un grand nombre de variables, pour généraliser le tableau de contingence à deux variables, il est possible de considérer les tableaux de contingence entre variables prises deux à deux. Nous obtenons ainsi une juxtaposition de tableaux de contingence. Un tel tableau est appelé tableau de Burt du nom de son auteur. Dans cette approche, une AFC est appliquée directement sur le nuage des barycentres obtenu par le tableau de Burt.

Avec l'AFC, on va analyser principalement les relations entre les modalités (en colonnes et en lignes du tableau de Burt).

- (1) Mais... à l'instar de l'ACP où on diagonalise la matrice des corrélations, on pourra revenir sur les individus
- (2) Attention, les informations sont dupliquées (des individus sont comptés plusieurs fois), il faudra corriger les résultats de l'AFC.

1.3.4 ACM via une ACP sur le tableau des profils lignes

La troisième piste enfin consiste à appliquer une analyse en composantes principales sur le tableau des profils. Nous pouvons obtenir les résultats de l'ACM via un programme d'ACP en appliquant la pondération suivante (sur les variables indicatrices x_{ij}) :

$$d_{ACP \rightarrow ACM}^2(i, i') = \sum_{k=1}^I u_k \times \left[\frac{1}{\sigma_k^2} (x_{ik} - x_{i'k})^2 \right],$$

où $u_k = \frac{n-n_k}{n \times p}$, $\sigma_k^2 = \frac{n_k(n-n_k)}{n^2}$ (σ_k^2 est la variance de la k ème colonne des indicatrices).

Chapitre 2

Les méthodes neuronales

Les réseaux de neurones artificiels ont connu un développement fulgurant au cours des deux dernières décennies. Cet intérêt a démarré avec l'application réussie de cette technique puissante pour des problématiques très différentes, et dans des domaines aussi divers que la finance, la médecine, la production industrielle, la géologie ou encore la physique.

Le succès croissant des réseaux de neurones sur la plupart des autres techniques statistiques peut s'attribuer à leur puissance, leur polyvalence et à leur simplicité d'utilisation. Les réseaux de neurones sont des techniques extrêmement sophistiquées de modélisation et de prévision, en mesure de modéliser des relations entre des données ou des fonctions particulièrement complexes. Derrière ce développement se cache les travaux effectués pour essayer de comprendre le comportement du cerveau humain. Ces travaux ont menés à représenter le cerveau humain par un ensemble de composants structurels appelés neurones, massivement interconnectés entre eux. Il en contiendrait plusieurs centaines de milliards, et chacun de ceux-ci serait, en moyenne, connecté à dix mille autres. Le cerveau est capable d'organiser ces neurones, selon un assemblage complexe, non-linéaire et extrêmement parallèle, de manière à pouvoir accomplir des tâches très élaborées. Par exemple, n'importe qui est capable de reconnaître des visages, alors que c'est là une tâche quasiment impossible pour un ordinateur classique. C'est la tentative de donner à l'ordinateur les qualités de perception du cerveau humain qui a conduit à une modélisation électrique de celui-ci. C'est cette modélisation que tentent de réaliser les réseaux de neurones artificiels.

2.1 Aperçu historique sur les réseaux de neurones

Un réseau de neurones est un processus distribué de manière massivement parallèle, qui a une propension naturelle à mémoriser des connaissances de façon expérimentale et de les rendre disponibles pour utilisation (Haykin, 94). Il ressemble au cerveau en deux points.

- la connaissance est acquise au travers d'un processus d'apprentissage,
- les poids des connections entre les neurones sont utilisés pour mémoriser la connaissance.

Les premières tentatives de modélisation du cerveau sont anciennes et précèdent même l'ère informatique. C'est en 1943 que Mc Culloch (neurophysiologiste) et Pitts (logicien) ont proposé les premières notions de 'neurone formel'. Ce concept fut ensuite mis en réseau avec une couche d'entrée et une sortie par Rosenblatt en 1959 pour simuler le fonctionnement rétinien et tacher de reconnaître des formes. C'est l'origine du perceptron. Cette approche dite connexionniste a atteint ses limites technologiques, compte tenu de la puissance de calcul de l'époque, mais aussi théoriques au début des années 70.

L'approche connexionniste à connaissance répartie a alors été supplantée par l'approche symbolique ou séquentielle qui promouvait les systèmes experts à connaissance localisée. L'objectif était alors d'automatiser le principe de l'expertise humaine en associant trois concepts :

- Une base de connaissance dans laquelle étaient regroupées "toutes" les connaissances d'experts humains sous forme de propositions logiques élémentaires ou plus élaborées en utilisant des quantificateurs (logique du premier ordre).

- une base de faits contenant les observations du cas à traiter comme, par exemple, des résultats d'examens, d'analyses de sang, de salive pour des applications biomédicales de choix d'un antibiotique .

- un moteur d'inférence chargé d'appliquer les règles expertes sur la base de faits afin d'en déduire de nouveaux faits jusqu'à la réalisation d'un objectif comme l'élaboration du traitement d'une infection bactérienne.

Face aux difficultés rencontrées lors de la modélisation des connaissances d'un expert humain, au volume considérable des bases de connaissance qui en découlaient et au caractère exponentiel de la complexité des algorithmes d'inférence mis en jeu, cette approche s'est éteinte avec les

années 80. En effet, il a été montré que les systèmes basés sur le calcul des prédicats du premier ordre conduisaient à des problèmes NP complets.

L'essor technologique et quelques avancées théoriques :

- ▶ estimation du gradient par rétro-propagation de l'erreur (Hopkins, 1982),
- ▶ analogie de la phase d'apprentissage avec les modèles markoviens de systèmes de particules de la mécanique statistique (verres de spin) par (Hopfield, 1982),

Au début des années 80, l'approche connexionniste a été relancée. Elle connaîtra au début des années 90 un développement considérable si l'on considère le nombre de publications et de congrès qui lui ont été consacrés mais aussi les domaines d'applications très divers où elle apparaît. Le petit ouvrage de Blayo et Verleysen (1996) dans la collection "Que Sais-je?" est une excellente introduction à ce domaine. Outre le fait qu'elles sont partiellement issues d'une inspiration biologique ou cognitive, ces méthodes ont rencontré rapidement un certain succès en particulier à cause de leur caractère de « boîte noire », d'outil à tout faire, ayant de très nombreux domaines d'applications. Une fois dépassés un certain excès d'enthousiasme et des difficultés de mise en oeuvre, les chercheurs et utilisateurs disposent maintenant d'un arsenal de techniques alternatives, non linéaires en général et algorithmiques. On pourra consulter par exemple l'ouvrage de Ripley (1996) qui intègre les techniques neuronales parmi les méthodes statistiques.

Ainsi en analyse des données, les méthodes non supervisées sont très attractives et en particulier l'algorithme de Kohonen est de nos jours largement utilisé dans ce contexte (Kohonen, 1984, 1993, 1995, 1997). Il réussit la double tâche de « projection » et de classification.

Sur de nombreux objectifs, justement ceux propres au data mining, les réseaux neuronaux ne rentrent pas nécessairement en concurrence avec des méthodes statistiques bientôt centenaires mais apportent un point de vue complémentaire qu'il est important de considérer.

Finalement, la motivation initiale de simulation du cortex cérébral a été rapidement abandonnée alors que les méthodes qui en découlaient ont trouvé leur propre intérêt de développement méthodologique et leurs champs d'applications.

Les premiers à proposer un modèle sont deux bio-physiciens de Chicago, McCulloch et Pitts en 1943, qui inventent le premier neurone formel qui portera leurs noms.

2.2 Le neurone formel

De façon très réductrice, un neurone biologique est une cellule qui se caractérise par des synapses, les points de connexion avec les autres neurones, fibres nerveuses ou musculaires, des dendrites, les “entrées” du neurones, l’axone, la “sortie” du neurone vers d’autres neurones ou fibres musculaires, le noyau qui active la sortie en fonction des stimulations en entrée.

Par analogie, le neurone formel est un modèle qui se caractérise par un état interne $s \in S$, des signaux d’entrée x_1, \dots, x_p et une fonction d’activation.

$$s = h(x_1, \dots, x_p) = f\left(w_0 + \sum_{j=1}^p w_j x_j\right) = f(w_0 + w'x).$$

La fonction d’activation opère une transformation d’une combinaison affine des signaux d’entrée, w_0 étant appelé le biais du neurone. Cette combinaison affine est déterminée par un vecteur de poids $[w_0, \dots, w_p]$ associé à chaque neurone et dont les valeurs sont estimées dans la phase d’apprentissage. Ils constituent “la mémoire” ou “connaissance répartie” du réseau.

Les différents types de neurones se distinguent par la nature de leur fonction d’activation f . Les principaux types sont :

- linéaire (f est la fonction identité),
- sigmoïde ($f(x) = \frac{1}{1+e^x}$),
- seuil ($f(x) = 1_{[0,+\infty[}(x)$),
- radiale ($f(x) = \sqrt{\frac{1}{2\pi}} \exp(-x^2/2)$),
- stochastiques $f(x) = 1$ avec la probabilité $\frac{1}{1+e^{-x/H}}$, 0 sinon (H intervient comme une température dans un algorithme de recuit simulé),

2.3 Apprentissage

L’apprentissage est vraisemblablement la propriété la plus intéressante des réseaux neuronaux. Elle ne concerne cependant pas tous les modèles, mais les plus utilisés. L’apprentissage est une phase du développement d’un réseau de neurones durant laquelle le comportement du réseau est modifié jusqu’à l’obtention du comportement désiré. L’apprentissage neuronal fait appel à des

exemples de comportement.

Durant cette phase de fonctionnement, le réseau adapte sa structure (le plus souvent, les poids des connexions) afin de fournir sur ses neurones de sortie les valeurs désirées. Cet apprentissage nécessite des exemples désignés aussi sous l'appellation d'échantillon d'apprentissage ainsi qu'un algorithme d'apprentissage.

Après initialisation des poids du réseau (en général des valeurs aléatoires), il y a présentation des exemples au réseau et calcul des sorties correspondantes. Une valeur d'erreur ou de correction est calculée et une correction des poids est appliquée.

Au niveau des algorithmes d'apprentissage, il a été défini deux grandes classes selon que l'apprentissage est dit supervisé ou non supervisé. Cette distinction repose sur la forme des exemples d'apprentissages. Dans le cas de l'apprentissage supervisé, les exemples sont des couples (Entrée, Sortie associée) alors que l'on ne dispose que des valeurs (Entrée) pour l'apprentissage non supervisé.

Définition 3.1

On appelle « apprentissage » des réseaux de neurones la procédure qui consiste à estimer les paramètres des neurones du réseau, afin que celui-ci remplisse au mieux la tâche qui lui est affectée.

Dans le cadre de cette définition, on peut distinguer deux types d'apprentissages : l'apprentissage « supervisé » et l'apprentissage « non supervisé ».

2.3.1 L'apprentissage supervisé

Nous avons vu, dans le paragraphe précédent, qu'un réseau non bouclé réalise une relation algébrique entre ses entrées et ses sorties. On peut donc affecter à un tel réseau la tâche qui consiste à réaliser une fonction algébrique non linéaire ; celle-ci peut être :

- connue analytiquement : le réseau réalise alors une tâche d'approximation de fonction,
- inconnue analytiquement, mais pour laquelle on dispose de valeurs, en nombre fini, qui sont entachées de bruit si ces valeurs résultent de mesures effectuées sur un processus physique, chimique, économique, biologique, etc. : le réseau réalise alors une tâche de modélisation statique ou une régression. C'est essentiellement pour ce dernier type d'applications que sont mis en

oeuvre les réseaux de neurones à apprentissage supervisé.

2.3.2 L'apprentissage non supervisé

Un réseau de neurones non bouclé peut également être utilisé dans un but de visualisation ou d'analyse de données : on dispose d'un ensemble de données, représentées par des vecteurs de grande dimension, et l'on cherche à les regrouper, selon des critères de ressemblance qui sont inconnus a priori.

L'apprentissage non-supervisé s'appuie généralement sur un processus compétitif permettant d'engendrer un modèle où les poids synaptiques des neurones représentent des prototypes de stimuli. La qualité du modèle résultant doit s'évaluer à l'aide d'une métrique permettant de mesurer la distance entre les stimuli et leurs prototypes. Souvent, cette métrique est basée sur la norme L . C'est le processus de compétition qui permet de sélectionner le prototype associé à chaque stimulus en recherchant le neurone dont le vecteur de poids synaptique est le plus proche (au sens de la métrique choisie) du stimulus en question.

2.4 Types de réseaux de neurones

On distingue différents types de réseaux de neurones. En règle générale, les Neural Networks sont catégorisés en fonction du nombre d'épaisseurs qui séparent l'entrée de données de la production du résultat, en fonction du nombre de noeuds cachés du modèle, ou encore du nombre d'entrées et de sorties de chaque noeud. En fonction du type de réseau, la propagation des informations entre les différents tiers de neurones peut varier.

2.4.1 Réseaux de neurones à propagation avant

Dans la variante la plus simple, celle du réseau de neurones dit à «propagation vers l'avant» , c'est-à-dire les informations passent directement de l'entrée aux noeuds de traitement puis aux sorties. Les réseaux peuvent être monocouches, c'est-à-dire constitués uniquement de couches

d'entrée et de sortie (*perceptron simple*), ou multicouches, c'est-à-dire disposant d'un certain nombre de couches cachées (*perceptron multicouches PMC*).

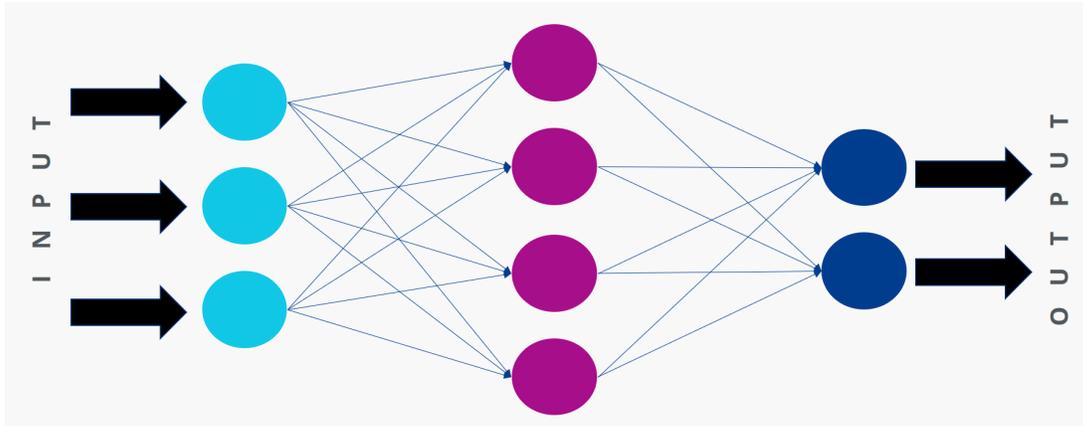


Figure 2.1. RNA à propagation avant

2.4.2 Réseaux de neurones récurrents

Les réseaux de neurones récurrents sauvegardent les résultats produits par les noeuds de traitement et nourrissent le modèle à l'aide de ces résultats. Ce mode d'apprentissage est un peu plus complexe. Ainsi, dans ces réseaux, il est possible de faire passer l'information dans des boucles de rétroaction, et ainsi de la faire revenir vers une couche précédente. Ces rétroactions permettent au système de se constituer une mémoire. Les réseaux de neurones récurrents sont utilisés par exemple en matière de reconnaissance vocale, de traduction et de reconnaissance d'écriture manuscrite.

2.4.3 Réseaux de neurones convolutifs

Un réseau neuronal convolutif est un type de réseau multicouche. Il est composé d'un minimum de cinq couches. Il est procédé sur chacune de ces couches à une reconnaissance de motif. Le résultat obtenu sur chaque couche est transmis à la couche suivante. Ce type de réseau neuronal artificiel est utilisé en matière de reconnaissance d'images, numérisation de texte, traitement naturel du langage...

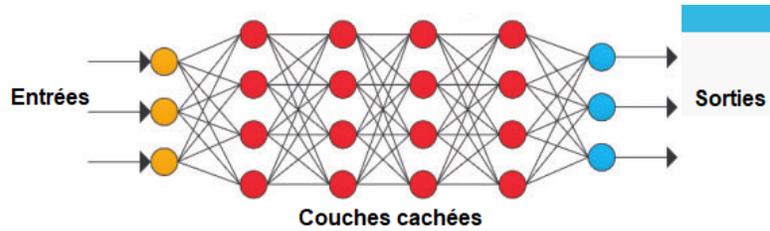


Figure 2.2. RNA convolutifs

2.5 Les tâches des réseaux de neurones

Dans ce qui suit, nous énumérons différentes catégories de tâches que l'on peut vouloir réaliser avec un réseau de neurones.

2.5.1 La régression

Dans la régression paramétrique, l'inférence statistique mène à considérer que nous pouvons modéliser les données en utilisant une forme fermée de fonction pouvant contenir un certain nombre de paramètres (poids) ajustables que nous pouvons estimer, de sorte que le modèle peut nous donner la meilleure explication possible de nos données. Par exemple, considérons une problématique de régression dans laquelle nous cherchons à modéliser ou approcher une variable cible unique, t à l'aide d'une fonction linéaire de la variable d'entrée, x . La fonction mathématique que nous utilisons pour modéliser ces relations est donnée simplement par une transformation linéaire f à deux paramètres, connus sous le nom d'ordonnées à l'origine, a et de pente, b ,

$$t = f(x) = a + bx.$$

Notre tâche consiste à trouver des valeurs de a et de b qui vont permettre de faire le lien entre une entrée x et la variable t . Cette problématique est connue sous le nom de régression linéaire. Ainsi, dans les modèles paramétriques, la relation entre les entrées et les sorties s'exprime par une fonction mathématique de forme fermée.

Par opposition, dans les modèles non-paramétriques, la relation entre les entrées et les sorties est pilotée par un approximateur (comme un réseau de neurones) que nous ne pouvons pas représenter par une fonction mathématique standard. Les modèles non-paramétriques ne font généralement aucune hypothèse quant à la relation entre x et t . En d'autres termes, ces modèles supposent que la véritable fonction sous-jacente qui gouverne la relation entre x et t n'est pas connue a priori, d'où le terme de boîte noire. Ils vont donc chercher une fonction mathématique (qui généralement ne possède pas une forme fermée) capable d'approcher convenablement la représentation de x et de t . Les exemples les plus courants de modèles non-paramétriques sont les fonctions polynomiales avec des paramètres adaptables et les réseaux de neurones.

Dans la mesure où nous ne faisons aucune hypothèse par rapport à la forme fermée de la relation entre x et t , une méthode non-paramétrique doit être suffisamment flexible pour modéliser un spectre relativement large de relations fonctionnelles. Par exemple, plus l'ordre du polynôme sera élevé, plus le modèle sera flexible. De la même manière, plus un réseau de neurones possédera de neurones, plus le modèle sera puissant.

2.5.2 Les tâches de classification

Les réseaux de neurones, comme la plupart des outils statistiques, peuvent également s'utiliser pour résoudre des problématiques de classification. Contrairement aux problématiques de régression, la classification par les réseaux de neurones va affecter l'appartenance des individus aux classes, à une entrée x . Par exemple, si le fichier de données possède trois catégories $\{A, B, C\}$, un réseau de neurones va affecter chacune des entrées à l'une de ces trois classes. L'information relative à l'appartenance des individus aux différentes classes est alors transmise à la variable cible, t . C'est la raison pour laquelle, dans une analyse de classification, la variable cible doit toujours être catégorielle. Une variable est dite catégorielle si (a) elle ne comporte que des valeurs discrètes qui (b) ne peuvent être classées dans un ordre hiérarchique. Par exemple, une variable cible $\{\text{HOMME, FEMME}\}$ est une variable catégorielle à deux modalités. En revanche, une variable cible constituée de valeurs-date n'est pas véritablement catégorielle dans la mesure où nous pouvons la classer dans un ordre chronologique (numérique).

2.6 Liens des réseaux de neurones avec l'analyse factorielle

L'analyse des correspondances des tables de contingence (ou tableaux croisés) peut être présentée comme un cas particulier de décomposition aux valeurs singulières, et aussi comme une analyse discriminante particulière. Dans le cadre des réseaux de neurones, elle peut être décrite comme un cas particulier de trois types de réseaux différents [Leb96].

- Elle peut être considérée comme un perceptron à une couche cachée (les couches d'entrée et de sortie correspondent alors aux lignes et aux colonnes de la table de

 - contingence) dans un contexte supervisé

- Elle peut être également décrite comme un perceptron multicouche auto-associatif (cas non-supervisé). Dans ce cas ce sont les lignes (ou les colonnes) qui constituent à la

 - fois la couche d'entrée et la couche de sortie.

- Enfin, l'analyse des correspondances peut aussi être décrite comme un réseau linéaire adaptatif particulier.

Chapitre 3

Les méthodes de classification neuronales

Comme nous l'avons déjà souligné au chapitre précédent, au cours de ces dernières décennies est apparu un ensemble de disciplines fortement interdépendantes, portant sur le traitement de l'information, la théorie de la décision et les méthodes de reconnaissances des formes en l'occurrence les réseaux de neurones. Leurs champs d'applications s'est beaucoup élargi et s'est étendu à plusieurs domaines, en particulier à la classification.

L'objectif de ce chapitre est de présenter quelques architectures des réseaux de neurones destinés à la classification supervisée et non supervisée mettant le point sur les limites de chaque modèle neuronal.

3.1 Les données

Les données peuvent se présenter sous différentes formes, elles concernent n individus supposés affectés, pour simplifier, du même poids :

- Un tableau de distances (ou dissimilarités, ou mesures de dissemblance), $n \times n$, entre les individus pris deux à deux.
- Les observations de p variables quantitatives sur ces n individus.
- Les observations, toujours sur ces n individus, de variables qualitatives ou d'un mélange

de variables quantitatives et qualitatives. D'une façon ou d'une autre, il s'agit, dans chaque cas, de se ramener au tableau des distances deux à deux entre les individus (c'est-à-dire au premier cas). Le choix d'une matrice de produit scalaire permet de prendre en compte simplement un ensemble de variables quantitatives tandis que le troisième cas nécessite plus de développements ; il n'est pas présenté ici car de peu d'intérêt pour des données d'expression.

3.2 Les objectifs

L'objectif d'une méthode de classification déborde du cadre strictement exploratoire. C'est la recherche d'une typologie, ou segmentation, c'est-à-dire d'une partition, ou répartition des individus en classes, ou catégories. Ceci est fait en optimisant un critère visant à regrouper les individus dans des classes, chacune le plus homogène possible et, entre elles, les plus distinctes possible. Nous sommes dans une situation d'apprentissage non-supervisé, ou en anglais de clustering.

3.3 Classification supervisée par réseaux multicouches

Un Perceptron MultiCouche (PMC) est une architecture stratifiée de neurones formels. C'est un des réseaux de neurones les plus utilisés actuellement, pour la classification supervisée notamment. Ils sont en mesure de minimiser la probabilité totale d'erreurs de classification. La classification par réseaux multicouches repose sur l'idée suivante : Supposons que l'on désire classer des formes en deux catégories, A ou B, en fonction de certaines caractéristiques de ces formes, on peut définir une fonction ϕ qui vaut (+1) pour toutes les formes de la classe A et (-1) pour toutes les formes de la classe B. Les réseaux de neurones sont de bons candidats pour réaliser une approximation de cette fonction ϕ , et on peut démontrer que cette approximation constitue une estimation de la probabilité d'appartenance de la forme inconnue à la classe A. Dans ce réseau, les neurones sont arrangés par couche. Il n'y a pas de connexion entre neurones d'une même couche, et les connexions ne se font qu'avec les neurones de couches avales. Habituellement, chaque neurone d'une couche est connecté à tous les neurones de la couche suivante et celle-ci seulement. Ceci nous permet d'introduire la notion de sens de parcours de l'information

(de l'activation) au sein d'un réseau et donc définir les concepts de neurone d'entrée, neurone de sortie. Dans cette situation particulière (càd classification supervisée) l'apprentissage est supervisée.

Dans l'apprentissage supervise, on présente a l'ordinateur des exemples sous la forme de paires (entrée, sortie désirée). Autrement dit, dans ce type d'apprentissage, on fournit au réseau de neurones la donnée a traiter mais aussi la sortie attendue. Le réseau effectue une évaluation de la donnée, puis compare la valeur obtenue avec la valeur désirée, il va ensuite modifier ses paramètres internes afin de minimiser l'erreur constatée. L'apprentissage 'supervise' consiste donc a calculer les coefficients de telle manière que les sorties du réseau de neurones soient, pour les exemples utilises lors de l'apprentissage, aussi proches que possibles des sorties 'désirées', qui peuvent être : la classe d'appartenance de la forme que l'on veut classer, la valeur de la fonction que l'on veut approcher ou de la sortie du processus que l'on veut modéliser, ou encore la sortie souhaitée du processus a commander. Il est habituel de le présenter en utilisant le paradigme du professeur et de l'élève.

De façon conceptuelle, on admet qu'il existe un professeur qui connaît la relation exacte entre toutes les entrées et leurs sorties, mais le réseau (élève) ne connaît pas cette relation. Si l'élève et le professeur sont exposes a une même entrée, le professeur est capable d'indiquer a l'élève la réponse désirée. Les paramètres (dans le cas des réseaux de neurones, le nombre de neurones et les poids) de l'élève doivent être ajustés pour donner la même réponse que celle du professeur. Cet ajustement, l'apprentissage, est réalise en général de façon itérative, en minimisant une mesure de l'erreur, jusqu'a ce que le réseau élève puisse émuler aussi bien que possible le professeur.

Les réseaux de neurones fournissent donc une information très riche, qui est loin d'être une simple réponse binaire. Cette propriété remarquable, que les réseaux de neurones partagent avec d'autres classificateurs, n'est malheureusement pas mise a profit dans la plupart des applications.

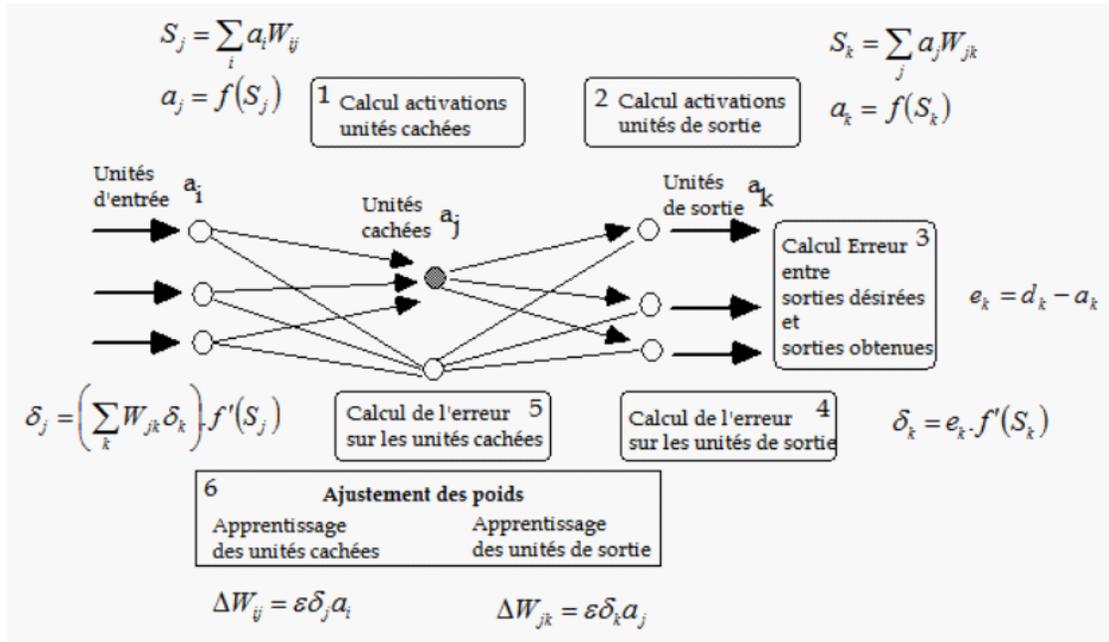


Figure 3.1 : Algorithme d'apprentissage du PMC

3.4 Les principales méthodes neuronales de classification non supervisée

3.4.1 L'algorithme des cartes auto-organisatrices

L'algorithme des cartes auto-organisatrices de Kohonen est l'un des modèles neuronaux les plus anciens, il représente le prototype de l'apprentissage non supervisé dans le domaine. Un grand nombre de résultats théoriques sont maintenant disponibles, qui explicitent les liens entre ce modèle et les méthodes statistiques de reconnaissance des formes : classification automatique, analyse en composantes principales non linéaires, approximation de fonctions densité (Ritter 1992

Voisins de la cellule k

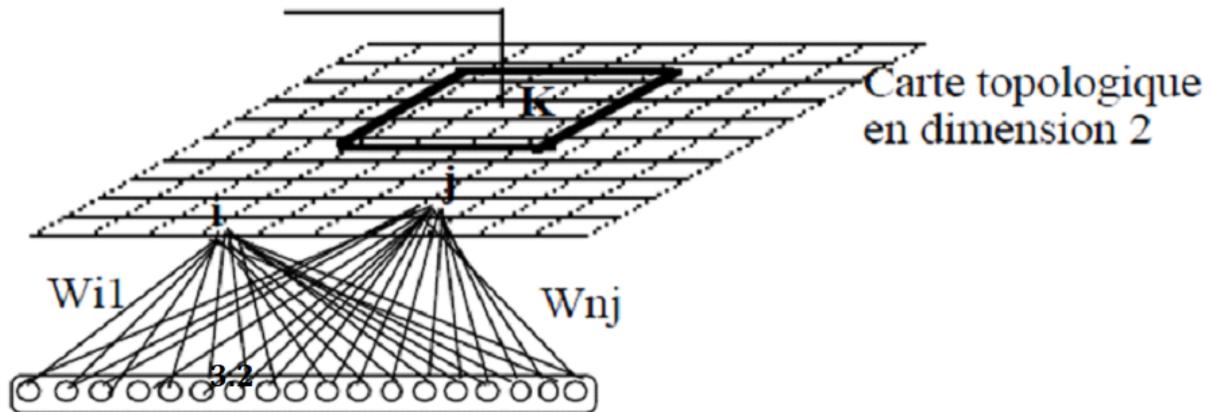


Fig.3.2 . carte topologique. treillis de neurones muni d'un système de voisinages donné et entièrement connecté à la couche d'entrée.

, Kohonen 1994). C'est en fait un procédé d'auto-organisation qui cherche à projeter des données dont la dimension est quelconque et en général grande dans un espace de faible dimension. En fin d'apprentissage, le but du réseau est de reproduire sur la couche de sortie les corrélations présentes dans les données d'entrées.

D'une manière générale, les cartes topologiques vont projeter les données initiales sur un espace discret et régulier de faible dimension (en général 1 ou 2).

Les espaces utilisés sont des treillis réguliers dont chacun des nœuds est occupé par un neurone. La notion de voisinage entre neurones découle alors directement de la structure et définit une topologie de la carte. Grâce au procédé d'auto-organisation la topologie qui lie les données initiales est conservée au niveau des réponses proposées par le réseau. La localisation des neurones actifs reproduit les liens existants au niveau des données initiales.

Les réseaux utilisés sont constitués de deux couches (**figure 3.2**).

- La couche d'entrée sert uniquement à la présentation des formes à classer. Les états de tous ses automates sont forcés aux valeurs des signaux d'entrées.

- La couche d'adaptation est formée du treillis de neurones évoqué précédemment. Le choix de la géométrie du réseau employé est fait à priori. Les neurones utilisés à ce niveau sont de

simples neurones linéaires, chacun d'entre eux étant connecté à tous les éléments de la couche d'entrée. De manière à permettre le processus d'auto-organisation, les poids qui lient les deux couches du réseau sont adaptatifs.

Les neurones de la carte calculent leur état, en parallèle, à partir des mêmes informations fournies par la forme présentée en entrée.

La principale caractéristique du processus d'auto-organisation est de permettre une adaptation des poids uniquement sur la région de la carte la plus active.

L'algorithme le plus simple détermine ce centre d'activité, comme étant le voisinage de la carte associé au neurone dont l'état est le plus grand. C'est l'utilisation de ce voisinage qui introduit les contraintes topologiques dans la représentation finale : ceci modélise de façon simplifiée un couplage latéral entre le neurone sélectionné et ses voisins dans la structure du réseau.

De cette façon, en fin d'apprentissage, les poids de chaque neurone vont converger vers des valeurs telles qu'un neurone ne sera plus actif que pour un sous ensemble précis d'éléments de la base d'apprentissage. Un neurone représente alors un exemple ainsi que la classe des données proches de cet exemple et chaque élément de la base d'apprentissage n'activera qu'un seul neurone.

Rechercher le neurone de plus grande réponse revient à déterminer le vecteur de poids dont le produit scalaire avec l'exemple présenté est le plus grand.

Les vecteurs poids w ayant la même dimension que celle des formes d'entrée z , ils représentent des points à l'intérieur du même espace. Un second critère de sélection du neurone le plus actif peut être de chercher celui dont le vecteur de poids est le plus proche au sens de la distance euclidienne de la forme présentée.

C'est ce critère qui est à l'heure actuelle utilisé dans l'algorithme des cartes topologiques par Kohonen. Son avantage est de permettre une formulation mathématique plus simple du problème. Certaines variantes de l'algorithme cherchent à normaliser les vecteurs de poids à une longueur constante.

L'égalité :

$$\|w - z\|^2 = \|w\|^2 - 2\langle w, z \rangle + \|z\|^2$$

montre que sous cette condition le neurone le plus actif au sens du produit scalaire est le

neurone le plus proche au sens de la norme euclidienne. Les deux critères évoqués plus hauts sont alors identiques et l'on parlera donc par la suite de neurones distances.

Différents procédés sont proposés pour l'adaptation des poids, nous ne présentons que le plus simple d'entre eux et renvoyons à (Kohonen, 1994) pour les variantes.

La description précise de l'algorithme nécessite donc de définir la notion de voisinage sur la carte. Le voisinage V_c de rayon δ du neurone c est composé de tous les neurones du réseau qui se situent à l'intérieur d'un disque de rayon δ .

La valeur choisie pour δ permet de considérer des voisinages de formes différentes et de prendre en compte un nombre variable de neurones. Cette valeur va varier pendant le déroulement de l'algorithme et permettre d'améliorer la convergence.

La modification appliquée dans le voisinage choisi revient à rapprocher les vecteurs poids sélectionnés de l'exemple présenté. En reconnaissance des formes, les méthodes de classification automatique utilisent souvent des techniques similaires.

Si l'on note c le neurone sélectionné, V_c son voisinage et w_r le vecteur de poids d'un neurone r , les modifications qui vont avoir lieu après présentation de la forme z à l'itération t vont être :

$$\begin{cases} w_r^t = w_r^{(t-1)} + \alpha(t)(w_r^{(t-1)} - z) & \text{si } r \in V_c \\ w_r^t = w_r^{(t-1)} & \text{sinon} \end{cases} \quad (3.1)$$

Le procédé d'adaptation nécessite donc l'introduction d'un nouveau paramètre, le pas d'apprentissage, qui varie en fonction du processus itératif : $\alpha(t)$.

Nous présentons la version de l'algorithme pour laquelle le neurone qui réagit le plus fort est celui dont le vecteur de poids est le plus proche au sens de la distance euclidienne de la forme présentée. Il utilise deux paramètres qui sont la taille des voisinages et le pas d'apprentissage. L'initialisation des poids et la présentation des éléments de l'ensemble d'apprentissage sont aléatoires. Le critère d'arrêt peut être un nombre d'itérations fixé a priori. L'algorithme est résumé par l'Algorithme I

Algorithme I

1. A l'itération t , présenter $z = (z_1, z_2, \dots, z_n)$
2. Déterminer le vecteur de poids le plus proche w_c^t de z tel que : $\|z - w_c^t\| = \min \|z - w_r^t\|$ où la minimisation est prise sur l'ensemble des vecteurs de poids associés à chaque neurone du réseau.

3. Déterminer la taille du voisinage de modification V_c et le pas d'apprentissage $\alpha(t)$.
4. Modifier les poids des neurones r selon le procédé décrit par la relation (3.1).
5. Faire $t = t + 1$

3.4.2 Cartes topologiques et nuées dynamiques

Définitions et notations :

Nous introduisons maintenant les notations utilisées pour permettre l'analogie entre le formalisme des nuées dynamiques et les cartes topologiques. Nous notons Dom l'espace de représentation des données $Dom \subset \mathbb{R}^2$ et $App = \{z_i, 1, N\}$ l'ensemble d'apprentissage formé d'éléments de Dom . Nous utilisons une carte C formée de K neurones, et supposons défini sur la carte un système de voisinages. La carte se présente sous forme d'un graphe dont les sommets sont les neurones et les arêtes sont les segments reliant les neurones voisins. La distance $\delta(c, r)$ entre deux neurones c et r sur la carte est alors définie par la longueur sur le graphe du plus court chemin entre c et r . A chaque neurone c est associé un vecteur poids $w_c = (w_c^1, w_c^2, \dots, w_c^n)$ de dimension n que nous appelons référent. Nous notons $w = \{w_c; c = \overline{1, K}\}$ l'ensemble de tous les référents et $L(C, W)$ la carte topologique.

Soit $K()$ une fonction noyau positive, symétrique et d'intégrale égale à l'unité, on utilise $K()$ pour déterminer une zone d'influence autour d'un neurone. En général l'importance relative d'un neurone c par rapport à un neurone r est pondérée par la valeur $K(\delta(c, r))$, le choix de $K()$ permet de gérer le voisinage par l'intermédiaire de la zone d'influence attachée à chaque neurone.

3.4.3 Algorithme de Kohonen révisé

Une version actuelle de l'algorithme de Kohonen utilise ces fonctions noyaux K pour effectuer l'auto-organisation, l'étape 4 de l'algorithme initial est alors remplacée pour toute cellule c de la carte par

$$w_c^{(t+1)} = w_c^t + \alpha(t)K^T(\delta(c, c_i^t))(z_i - w_c^t) \quad (3.2)$$

Où T et α sont deux fonctions décroissantes de \mathbb{R}^+ dans \mathbb{R}^+ . La fonction température $T(t)$

permet de contrôler, sur la carte, la taille du voisinage du neurone c_i^t le plus proche de l'exemple z_i présenté à l'itération t . La fonction $\alpha(t)$ contrôle la modification des référents et la convergence de l'algorithme.

Dans la formule (3.2) l'adaptation se fait par l'intermédiaire de la fonction noyau :

$$K^T(\delta(c, c_i^t)) = \frac{1}{T(t)} K\left(\frac{\delta(c, c_i^t)}{T(t)}\right) \quad (3.3)$$

Une version de cet algorithme en terme de minimisation d'une fonction coût est présentée dans (Luttrell, 1990 ; Kohonen, 1994). Il s'agit d'une minimisation globale qui adapte les référents en fonction de l'ensemble de toutes les données.

Nous présentons maintenant cette version qui peut être interprétée comme une méthode de nuées dynamiques.

3.4.4 Version nuées dynamiques des cartes topologiques

Les nuées dynamiques sont des méthodes générales qui permettent d'obtenir un minimum local d'un critère à optimiser. Elles reposent sur l'utilisation de deux entités : l'ensemble des partitions de l'espace des données et l'ensemble des représentations.

- Pour définir l'ensemble des partitions liée à la carte, nous appelons fonction d'affectation sur C , toute fonction $\Phi : Dom \rightarrow C$. Nous définissons à partir de Φ une partition F_Φ dont certains éléments peuvent être l'ensemble vide, en associant tout élément z de Dom à son image sur la carte. De manière formelle $F_\Phi = \{F_c; c = 1..K\}$ où chaque partie est définie par $F_c = \{z; \Phi(z) = c\}$ et K représente le nombre de neurones.

- L'espace des représentations est la carte $L(C, W)$.

La méthode des nuées dynamiques fonctionne d'une manière itérative. Chaque itération comporte deux phases : une phase d'optimisation suivie d'une phase de réaffectation.

- Pour les cartes, la phase d'optimisation utilise un critère défini à partir d'une distance d entre une donnée z et un neurone de la carte C :

$$d(z, c) = \sum_{r \in C} K(\delta(c, r)) \|z - w_r\|^2 \quad (3.4)$$

Cette distance utilise une fonction noyau K centrée en c , il s'agit de l'une des fonctions K^T définie par la formule (3.3). Pendant toute l'adaptation la valeur $T(t)$ est maintenue constante ($T(t) = T$), l'indice T est omis dans la notation pour simplifier l'écriture des formules. Le critère à minimiser $E(\Phi, W)$ mesure l'adéquation entre une partition F_Φ et une carte topologique $L(C, W)$: cette adéquation se mesure par :

$$E(\Phi, L(C, W)) = \sum_{Z_i \in App} d(Z_i, \Phi(Z_i)) \quad (3.5)$$

Dans toute la suite la topologie de la carte est constante, pour simplifier l'écriture $E(\Phi, L(C, W))$ sera noté $E(\Phi, W)$.

Pour un système de référents donné W , la fonction d'affectation Φ qui minimise $E(\Phi, W)$ est celle qui affecte toute forme z au référent de la carte L le plus proche au sens de la distance définie en (3.4). Nous notons Φ_w cette fonction d'affectation :

$$\Phi_w(Z) = \underset{c}{\operatorname{argmin}} d(z, c) \quad (3.6)$$

Comme toute fonction d'affectation, Φ_w définit la partition F_{Φ_w} . La connaissance d'une partition F_Φ permet de déterminer un système de poids W^* qui améliore l'adéquation $E(\Phi, W)$:

On choisit pour cela le système de poids W^* qui minimise $E(\Phi, W)$. $W^* = \{W_c^*; 1, \dots, K\}$ est la solution unique du système :

$$\frac{\partial E}{\partial W_c} = \sum_{Z_i \in App} K(\delta(c, \Phi(Z_i)))(w_c - Z_i) = 0 \quad (3.7)$$

Son expression est donnée par (3.8) :

$$W_c^* = \frac{\sum_{r \in C} K(\delta(c, r))Z_r}{\sum_{r \in C} K(\delta(c, r))n_r} \quad (3.8)$$

Où $Z_r \in \mathbb{R}^n$ est tel que $Z_r = \sum_{Z_i \in App \cap F_r} Z_i$ et $n_r = \operatorname{card}(App \cap F_r)$.

Il est possible, en procédant de manière itérative par affectation et minimisation successives,

d'obtenir un minimum local de $E(\Phi, W)$. Nous décrivons maintenant cet algorithme d'apprentissage.

3.4.5 Analogie avec l'algorithme de Kohonen

Si l'on remplace la phase d'optimisation de l'algorithme I par une descente du gradient, le nouveau système de poids w_k déterminé à cette étape n'est plus optimal.

$$V_K = E(\Phi_{w^k}, w^k)$$

est conservée ainsi que la convergence de cette suite. La stationnarité de la suite n'est plus assurée, il faut déterminer alors un critère d'arrêt du type

$$\frac{V_{K-1} - V_K}{V_K} \leq \text{seuil}$$

le système de poids w^k obtenu dans ce cas à l'étape k est défini par modification de w^{k-1} de la manière suivante :

$$W_C^K = W_C^{K-1} - \alpha(K) \frac{\partial E(\Phi_{w^{k-1}}, W)}{\partial W_c}$$

Une version de gradient stochastique de cette version adaptative consiste à effectuer la modification après la présentation de chaque forme Z_i .

On reconnaît alors la modification des poids de l'algorithme classique des cartes topologiques proposé par Kohonen pour une taille de voisinage fixée ($T(t) = T$ fixée). La différence entre l'algorithme I et celui de Kohonen réside maintenant uniquement dans la fonction d'affectation. Kohonen choisit comme fonction d'affectation celle qui affecte une forme z au neurone dont le vecteur poids est le plus proche au sens de la distance euclidienne, la version "nuées dynamiques" présentée ici utilise la distance pondérée d définie à partir de la carte (relation 3.4).

D'autre part, l'algorithme de Kohonen fait varier la taille du voisinage, au cours du temps, à travers la fonction température $T(t)$. Une version analogue peut être écrite pour l'algorithme I en prenant une fonction noyau KT qui varie en fonction des itérations t . Généralement, la température décroît de T_{max} et T_{min} de telle façon que les dernières itérations correspondent à un voisinage réduit à un seul neurone. À ce stade l'algorithme I et celui de Kohonen se comportent

comme l'algorithme classique des K-moyennes et permettent d'obtenir, sous la contrainte de la conservation de la topologie induite par la topologie de la carte, un minimum local de la fonction de distorsion :

$$E(\Phi, W) = \sum_{Z_I \in App} \|Z_I - W_{\Phi_{Z_I}}\|^2$$

Les simulations que nous présentons au paragraphe 4 pour montrer l'équivalence de l'algorithme I et celui de Kohonen utilisent cette version (décroissance de T de $Tmax$ à $Tmin$).

3.4.6 Carte topologique et approximation de la fonction densité

Comme nous venons de le montrer l'algorithme de Kohonen et l'algorithme I sont similaires et sont de la même famille que l'algorithme des K-moyennes.

Nous présentons maintenant une autre série d'algorithmes d'apprentissage de cartes topologiques dont le formalisme est différents. Ils permettent d'approximer la fonction de densité de l'espace des données selon un modèle probabiliste complexe.

Formalisme probabiliste des cartes topologiques :

Nous introduisons dans ce paragraphe un formalisme probabiliste du fonctionnement des cartes topologiques introduit dans Luttrel, 1994.

La carte $L(C, W)$ sera modélisée sous la forme d'une architecture à 3 couches :

Une couche d'entrée de n neurones qui reçoit les données z .

La carte C initiale est maintenant dupliquée et donne naissance à deux cartes C_1 et C_2 identiques composées de K neurones et munies de la même topologie que C . Les deux cartes C_1 et C_2 constituent respectivement la première et la seconde couche (figure 3.3).

Estimation des paramètres

Nous utilisons, pour l'estimation des paramètres, une approche par classification (Schroeder 76). C'est une méthode de type nuées dynamiques, il faut donc préciser l'ensemble des partitions à considérer et le critère à optimiser. Comme dans le paragraphe précédent nous utilisons une

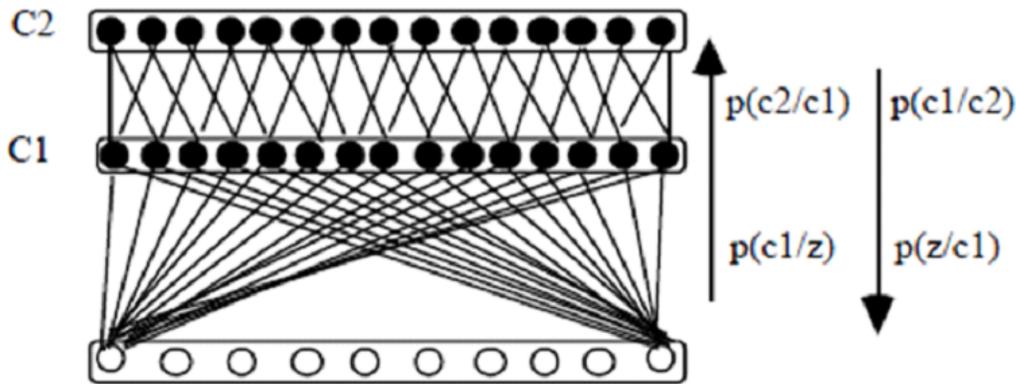


Fig.3.3 . Architecture à trois couches

fonction d'affectation que nous noterons dans la suite χ , elle affecte chaque observation z_i de l'ensemble d'apprentissage à l'un des K phénomènes aléatoires du mélange. La fonction χ définit donc une partition de l'ensemble d'apprentissage App . Si les exemples $\{Z_i, 1..N\}$ sont générés d'une manière indépendante

3.5 Intérêt et limitations de l'algorithme de Kohonen

Les cartes de Kohonen sont d'un grand intérêt et fournissent des résultats spectaculaires dans beaucoup de cas. Cependant, elles ont leurs limitations qu'il faut connaître. Leurs avantages et leurs inconvénients sont résumés ci-après :

3.5.1 Avantages

- formation de cartes préservant la topologie
- quantification vectorielle sous la contrainte de voisinage des codes
- approximation de la densité de probabilité des données pour minimiser la distorsion moyenne

3.5.2 Limitations

- nombre d'unités fixe

- dimension fixe et limitée (1, 2 ou 3 en pratique), ce qui n'est pas toujours compatible avec tous les problèmes
- la forme est rigide et n'est donc pas forcément compatible avec celle de la sous variété des données d'où le risque de prototype situés en-dehors des données
- l'algorithme est très sensible à la densité de probabilité

3.6 Expérimentations

Dans ce paragraphe nous illustrons sur deux exemples simulés le comportement des deux algorithmes *I* et *II* que nous venons de présenter. Les deux algorithmes sont définis pour des vecteurs d'entrée de dimension quelconque, cependant pour visualiser les résultats obtenus nous avons généré des données en dimension deux. Pour permettre une comparaison simple avec l'algorithme classique des cartes de Kohonen, la première base d'apprentissage utilise 800 exemples uniformément réparties dans un carré de taille $[-12, 12] \times [-12, 12]$. La deuxième base d'apprentissage est générée selon un mélange de 9 lois normales dont les matrices de variances-covariances sont diagonales et différentes les unes des autres. Parmi ces lois, 3 ont des variances égales dans les deux directions et 6 ont deux variances différentes selon les deux directions.

Cette base est formée de 900 exemples (100 par chaque distribution normale). Toutes les simulations utilisent des cartes carrées munies de voisinage carrés, pour la distribution uniforme nous présentons les résultats avec une carte de 10×10 neurones et pour le mélange de lois normales une carte de 6×6 neurones. Pour chaque expérience, nous montrerons les résultats obtenus, après organisation de la carte, en visualisant les référents et la topologie induite de la carte. Les référents de deux neurones voisins sur la carte sont reliés par un segment. Dans toutes les simulations les référents ont été initialisés d'une manière régulière de façon à couvrir l'ensemble d'apprentissage utilisé. Dans tous les cas nous avons utilisé la version des algorithmes faisant décroître la température au cours du temps suivant la formule

Les figures 3.4 et 3.5 montrent respectivement la carte obtenue par l'algorithme de Kohonen pour la première et le seconde base d'apprentissage.

Les figures 3.6 et 3.7 montrent les cartes obtenues pour les deux bases d'apprentissage après

	Algo de Kohonen	Algorithme I
Première base	0.605	0.526
Deuxième base	0.315	0.275

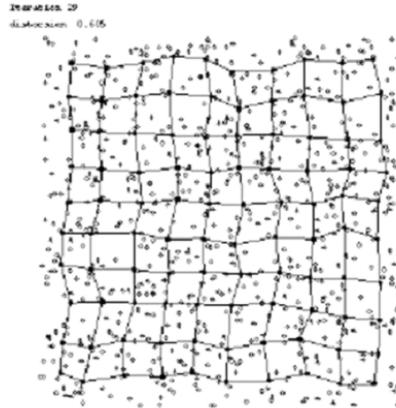


Fig. 3.4 . Carte obtenue après apprentissage pour la répartition uniforme avec l'algorithme de Kohonen

stabilisation de l'algorithme I .

Dans les deux cas les cartes obtenues montrent qu'il y a conservation de la topologie induite, l'organisation des référents étant comparable à celle obtenue par l'algorithme de Kohonen. Le calcul de la distorsion

$$E(\Phi, W) = \sum_{Z_I \in App} \|Z_I - W_{\Phi(Z_I)}\|^2$$

en fin d'apprentissage donne :

Les figures obtenues illustrent le comportement similaire de l'algorithme I et celui de Kohonen. Ce constat est prévisible dans la mesure où la version stochastique de l'algorithme I ne diffère de l'algorithme de Kohonen que par la distance utilisée pour le choix du neurone le plus proche.

Les figures 3.8(a) et 3.9(a) montrent les cartes obtenues pour les deux bases après apprentissage avec l'algorithme II . Sur ces figures, nous observons de même une conservation de la

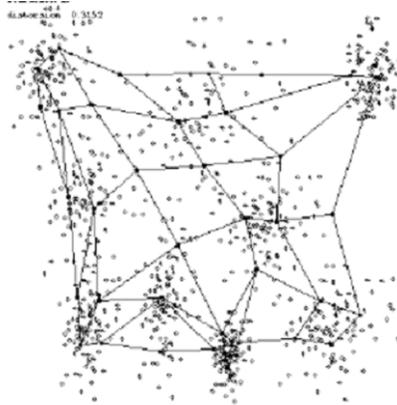


Fig. 3.5. Carte obtenue après apprentissage pour le mélange de lois normales avec l'algorithme de Kohonen

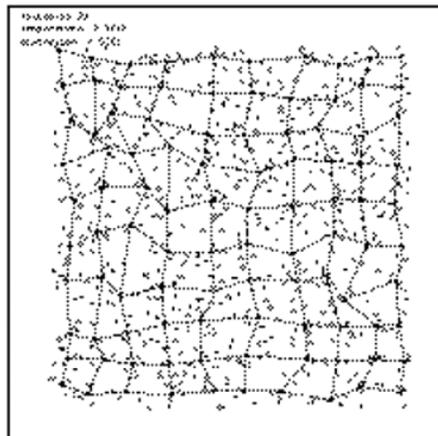


Fig. 3.6. Carte obtenue après apprentissage pour la répartition uniforme avec l'algorithme I

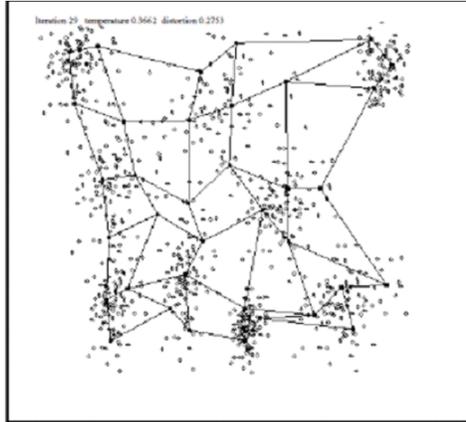


Fig. 3.7 . Carte obtenue après apprentissage pour le mélange de lois normales avec l'algorithme I

topologie de la carte. Les figures 3.8(b) et 3.9(b) visualisent les écarts types calculés en fin d'apprentissage par l'algorithme *II* sous la forme d'un cercle centré sur le référent et de rayon l'écart type.

Les figures 3.8(b) et 3.9(b) nous permettent de mettre en évidence l'influence de l'introduction du paramètre de la variance. Ces variances s'adaptent à la distribution locale des données ; elles sont petites dans les régions de forte densité et grandes dans les régions de faible densité. Nous constatons que nous pouvons regrouper localement plusieurs lois normales associées à des cellules voisines afin de recouvrir les domaines d'influence des lois normales utilisées pour générer les données.

3.7 Conclusion

Dans ce chapitre nous avons montré l'analogie existant entre l'algorithme des cartes topologiques de Kohonen et une méthode du type nuées dynamiques. La seule différence entre les deux réside dans la détermination du neurone le plus actif en réponse à une donnée présentée en entrée de la carte. Nous avons ensuite présenté un formalisme probabiliste des cartes topologiques en terme d'approximation de fonctions densité. La méthode des nuées dynamiques appliquée à la recherche du maximum de vraisemblance donne lieu à un nouveau type de cartes topologiques, dont un cas particulier se trouve être l'algorithme de Kohonen. Les simulations présentées illustrent

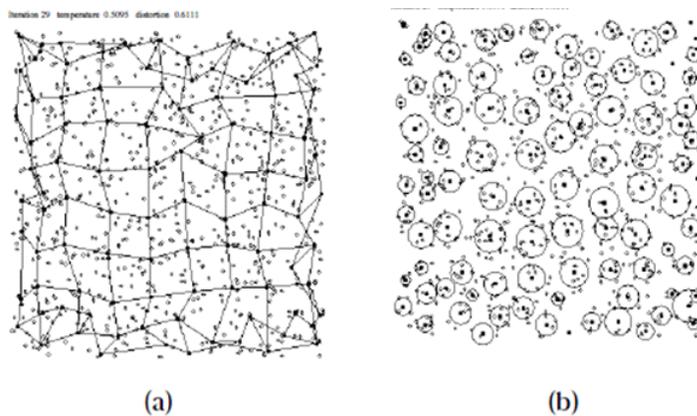


Fig. 3.8 . (a) Représente la carte obtenue après apprentissage avec l'algorithme II pour la première base. (b) Visualise des cercles, chaque cercle est centré sur un référent et de rayon égal à son écart type.

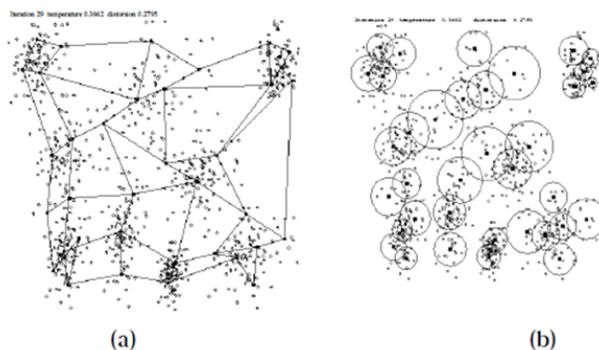


Fig. 3.9 . (a) Représente la carte obtenue après apprentissage avec l'algorithme II pour la deuxième base. (b) Visualise des cercles, chaque cercle est centré sur un référent et de rayon égal à son écart type

le comportement des différents algorithmes.

Chapitre 4

Combinaison d'un modèle 2D-RCA et des RNA pour la segmentation des images texturées

Le modèle de champ aléatoire de Markov (MRF) et le modèle autorégressif 2D (AR) sont deux modèles stochastiques fréquemment utilisés pour la représentation des caractéristiques des textures. Dans de telles méthodes de modèle stochastique, une texture est considérée comme une réalisation d'un champ aléatoire. Les paramètres du modèle sont ensuite estimés et choisis comme caractéristiques de texture. La méthode de segmentation proposée peut être classée comme une méthode basée sur la région en utilisant un critère de similarité construit sur les paramètres du modèle autorégressif à coefficients aléatoires 2D RCA. Le modèle 2D RCA a été récemment introduit par les auteurs de Boulemnadjel et al (2016) pour capturer le comportement variant dans l'espace de la volatilité. Ils ont utilisé avec succès l'approche de la méthode généralisée des moments (GMM) pour estimer les paramètres du modèle et les ont considérés comme des caractéristiques de pixels dans la classification des images de texture. Dans ce travail, nous proposons une procédure d'estimation des paramètres du modèle 2DRCA à l'aide d'une technique GMM implémentée avec un réseau de neurones à gradient (GNN). Les auteurs de Weilu et Xu (1995) avaient déjà utilisé un réseau de neurones pour estimer les paramètres d'un modèle AR 2D censé représenter une image texturée donnée en minimisant une erreur totale correspondant

à la différence entre l'image et celle estimée. Leur méthode est formulée comme une fonction de coût, basée sur la similitude entre l'image texturée et celle obtenue à partir du modèle AR ajusté, qui doit ensuite être minimisée. Dans l'approche proposée, le problème est formulé de la même manière mais la fonction de coût est basée sur une forte correspondance des moments statistiques du modèle 2D-RCA correspondant et des moments d'échantillonnage des données d'image de population. Cela fait de l'approche proposée une méthode d'information partielle, dérivée d'un ensemble minimal de conditions de moment. Pour plus de détails sur les raisons du choix d'une modélisation 2D RCA et la procédure GMM pour l'estimation des paramètres, on peut consulter Boulemnadjel et al (2016).

Dans ce chapitre la section 1 présente et décrit la procédure d'estimation du modèle 2D RCA. La section 2 est consacrée au processus d'apprentissage. Dans la section 3, nous présentons notre architecture ANN. Dans la section 4, une technique de region en croissance est présentée pour réaliser une segmentation d'images. Enfin, les résultats expérimentaux et les remarques finales sont donnés respectivement dans les sections 5 et 6. La procédure d'analyse proposée est illustrée dans la figure suivante.

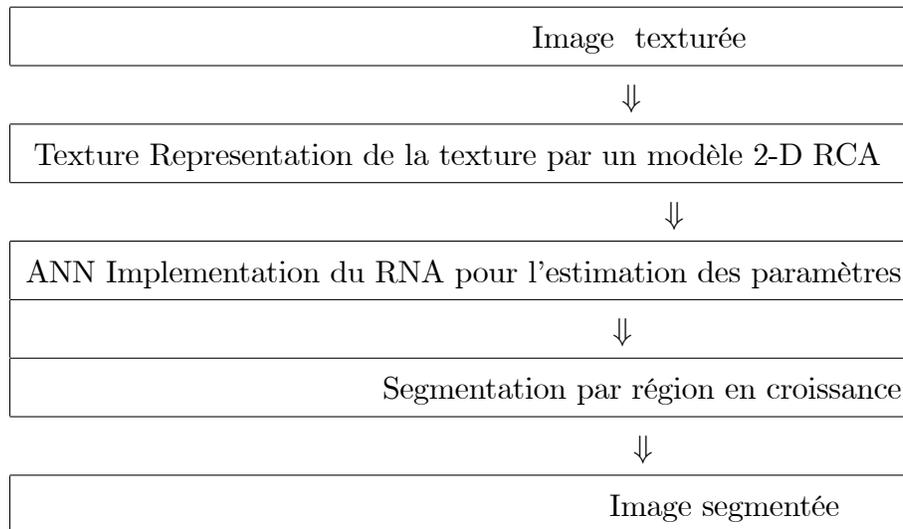


Figure 1 : Étapes de la procédure de segmentation proposée

4.1 Définition du modèle et estimation des paramètres

Les modèles 2D-RCA sont une extension naturelle des modèles 2D-AR introduits dans Boulemadjel et al (2016) afin de modéliser certains ensembles de données spatiales nonGaussiennes tels que les images texturées. Sa construction a été inspirée du célèbre modèle 1D-RCA largement utilisé en économétrie Conslik (1975), Conslik (1976) et en applications engineering Andel (1976).

4.1.1 Notations

Dans la suite, si M est une matrice donné, M^T est sa transposé, $M^{\otimes 2} = M \otimes M$ est le produit de Kronecker de M et M elle même. Si M est une matrice carrée, $\rho(M)$ est le rayon spectral de M . Tous les vecteurs sont soulignés sauf ceux de \mathbb{Z}^2 seront écrits en gras, donc $\mathbf{0} = (0, 0)$ $\mathbf{1} = (1, 1)$, $\mathbf{e}_0 = (0, 1)$, $\mathbf{e}_1 = (1, 0)$ et $\mathbf{n} = (n_1, n_2)$. Pour tout $\mathbf{s} = (s_1, s_2)$, $\mathbf{t} = (t_1, t_2) \in \mathbb{Z}^2$, nous définissons l'ordre lexicographique \preceq tel que $\mathbf{s} \preceq \mathbf{t}$ si et seulement si $[(s_1 < t_1) \vee (s_1 = t_1 \text{ et } s_2 \leq t_2)]$. Cependant, pour $\mathbf{a} = (a_1, a_2)$, $\mathbf{b} = (b_1, b_2) \in \mathbb{Z}^2$ tel que $\mathbf{a} \preceq \mathbf{b}$, considérons le sousensemble d'indexation dans \mathbb{Z}^2 suivant $S[\mathbf{a}, \mathbf{b}] := \{(l, m) \in \mathbb{Z}^2, a_1 \leq l \leq b_1, a_2 \leq m \leq b_2\}$ en ordonnant ses termes par ordre lexicographique, $S[\mathbf{a}, \mathbf{b}] = S[\mathbf{a}, \mathbf{b}] \setminus \{\mathbf{a}\}$ et $S[\mathbf{a}, \mathbf{b}[= S[\mathbf{a}, \mathbf{b}] \setminus \{\mathbf{b}\}$.

Pour tout ensemble A , $\#A$ est le cardinal de A .

4.1.2 Le modèle 2D-RCA

Un processus stochastique 2D $X(\mathbf{t})_{\mathbf{t} \in \mathbb{Z}^2}$ suit un modèle 2D-RCA si,

$$X(\mathbf{t}) = \sum_{\mathbf{s} \in S[\mathbf{0}, \mathbf{p}]} a_{\mathbf{s}}(\mathbf{t}) X(\mathbf{t} - \mathbf{s}) + e(\mathbf{t}), \mathbf{t} \in \mathbb{Z}^2, \quad (4.1)$$

où, $\mathbf{p} = (p_1, p_2)$, $\mathbf{0} = (0, 0)$, $a_{\mathbf{s}}(\mathbf{t}) = \alpha_{\mathbf{s}} + \beta_{\mathbf{s}}(\mathbf{t})$ et $d = \#S[\mathbf{0}, \mathbf{p}]$.

Pour ce modèle, nous avons besoin des hypotheses suivantes :

a) $\{e(\mathbf{t}); \mathbf{t} \in \mathbb{Z}^2\}$ est une suite de variables aléatoires indépendantes stationnaires de moyenne zero et de variance σ_e^2 .

b) Les $\alpha_{\mathbf{s}}, \mathbf{s} \in S[\mathbf{0}, \mathbf{p}]$ sont des constants réelles.

c) Si $\underline{\beta}(\mathbf{t}) = (\beta_{\mathbf{s}}(\mathbf{t}); \mathbf{s} \in S] \mathbf{0}, \mathbf{p}])'$, alors $(\underline{\beta}(\mathbf{t}), \mathbf{t} \in \mathbb{Z}^2)$ est une suite indépendante de vecteurs aléatoires de moyenne zero et $E \{ \underline{\beta}(\mathbf{t}) \underline{\beta}'(\mathbf{t}) \} = C$.

d) $\underline{\beta}(\mathbf{t})$ et $e(\mathbf{t})$ sont indépendants.

Soit la σ -algèbre $\mathcal{F}_{\mathbf{t}} := \sigma(e(\mathbf{t}'), \underline{a}(\mathbf{t}'), \mathbf{t}' \preceq \mathbf{t}, \mathbf{t}' \neq \mathbf{t})$. Boulemnadjel et al (2016) ont obtenu des conditions d'existence de solutions mesurables stationnaires du second ordre de l'équation 4.1. Ils ont proposé, d'abord, de représenter l'équation 4.1 sous forme d'espace d'état définissant le vecteur $d \times 1$ $\underline{X}(\mathbf{t})$ tel que $\underline{X}(\mathbf{t}) = \underline{H}' \underline{X}(\mathbf{t})$, où,

$$\underline{X}(\mathbf{t}) = A_1(\mathbf{t})\underline{X}(\mathbf{t} - \mathbf{e}_0) + A_2(\mathbf{t})\underline{X}(\mathbf{t} - \mathbf{e}_1) + e(\mathbf{t})\underline{H}, \quad \mathbf{t} \in \mathbb{Z}^2, \quad (4.2)$$

avec \underline{H} un vecteur constant approprié et $\{A_l(\mathbf{t}), \mathbf{t} \in \mathbb{Z}^2\}$, $l = 1, 2$ une suite de matrices aléatoires $d \times d$ indépendantes.

Les propriétés de stationnarité des modèles définis par l'équation 4.1, directement liées aux propriétés de stationnarité des modèles définis par l'équation 4.2, sont résumés dans le théorème suivant

Théorème 4.1 Considérons le modèle (4.1) avec une représentation d'espace d'états (4.2). Soit $\mathcal{A}_l = E \{A_l(\mathbf{t})^{\otimes 2}\}$, $l = 1, 2$. Si,

$$\rho(\mathcal{A}_1 + \mathcal{A}_2) < 1, \quad (4.3)$$

Alors l'équation (4.2) a une solution stationnaire du second ordre unique.

4.1.3 Estimation des paramètres du modèle 2D-RCA

Pour le modèle 2-D RCA (4.1), en plus des conditions (a), (b), (c) et (d), nous allons supposer que les paramètres $\alpha_{\mathbf{s}}, \mathbf{s} \in S] \mathbf{0}, \mathbf{p}]$ et C sont tels qu'une solution stationnaire au second ordre unique existe. Notre but est l'estimation des paramètres du modèle $2D - RCA(\mathbf{p})$ basée sur les observations $\{X(\mathbf{t}), \mathbf{t} \in S] \mathbf{0}, \mathbf{N}]\}$, $\mathbf{N} = (N, M)$. Pour cela, nous récrivons le modèle (4.1)

sous forme d'un modèle de regression linéaire, càd,

$$X(\mathbf{t}) = \underline{x}'(\mathbf{t})\underline{\theta} + U(\mathbf{t}), \quad (4.4)$$

où $\underline{\theta} = (\alpha_{\mathbf{s}}; \mathbf{s} \in S] \mathbf{0}, \mathbf{p}])' \in \mathbb{R}^K$, $K = (p_1 + 1)(p_2 + 1) - 1$, $U(\mathbf{t}) = \underline{\beta}(\mathbf{t}) \underline{x}(\mathbf{t}) + e(\mathbf{t})$, $\underline{\beta}(\mathbf{t}) = (\beta_{\mathbf{s}}(\mathbf{t}); \mathbf{s} \in S] \mathbf{0}, \mathbf{p}])'$ et $\underline{x}(\mathbf{t}) = (X(\mathbf{t} - \mathbf{s}); \mathbf{s} \in S] \mathbf{0}, \mathbf{p}])'$. Comme $\underline{\beta}(\mathbf{t})$ et $e(\mathbf{t})$ sont indépendants de

$(\underline{\beta}(\mathbf{t} - \mathbf{k}), e(\mathbf{t} - \mathbf{k}), \mathbf{0} \preceq \mathbf{k}, \mathbf{k} \neq (0, 0))$ et $\underline{x}(\mathbf{t})$ est une fonction mesurable de cet ensemble uniquement ; nous avons :

$$E[(X(\mathbf{t}) - \underline{x}'(\mathbf{t}) \underline{\theta}_0) \setminus \mathcal{F}_{\mathbf{t}}] = 0,$$

de plus,

$$\begin{aligned} E[\underline{x}(\mathbf{t}) (X(\mathbf{t}) - \underline{x}'(\mathbf{t}) \underline{\theta}_0)] &= E[E[\underline{x}(\mathbf{t}) (X(\mathbf{t}) - \underline{x}'(\mathbf{t}) \underline{\theta}_0)] \setminus \mathcal{F}_{\mathbf{t}}] \\ &= E[\underline{x}(\mathbf{t})] E[(X(\mathbf{t}) - \underline{x}'(\mathbf{t}) \underline{\theta}_0) \setminus \mathcal{F}_{\mathbf{t}}], \end{aligned}$$

est une fonction mesurable de cet ensemble uniquement ; nous avons :

$$\begin{cases} h(\underline{\theta}, \mathbf{t}) = \underline{x}(\mathbf{t}) [X(\mathbf{t}) - \underline{x}'(\mathbf{t}) \underline{\theta}] \\ E[h(\underline{\theta}_0, \mathbf{t})] = 0 \end{cases}, \quad (4.5)$$

Ce qui représente un système de d conditions d'orthogonalité. Dans chaque equation du système (4.5) les régresseurs sont exogènes tandis que les régresseurs des autres équations du système peuvent également être utilisés comme instruments. Définissons maintenant la contrepartie empirique de la première équation de (4.5) en utilisant les observations $\{X(\mathbf{t}), \mathbf{t} \in S] \mathbf{0}, \mathbf{N}]\}$,

$$h_{\mathbf{N}}(\underline{\theta}) = \frac{1}{N \times M} \sum_{t_1=1}^N \sum_{t_2=1}^M h(\underline{\theta}, \mathbf{t}). \quad (4.6)$$

L'estimateur GMM (ou minimum distance) $\widehat{\underline{\theta}}_{\mathbf{N}}$ de $\underline{\theta}$ sera choisi de manière à rendre les moments empiriques $\underline{h}_{\mathbf{N}}(\underline{\theta})$ aussi proche que possible de zéro. Comme il y a autant de paramètres que d'équations, par rapport à $\underline{h}_{\mathbf{N}}(\underline{\theta})$, le paramètre $\underline{\theta}$ constitue une surface de distance où chaque ensemble de valeurs de paramètres arbitraires correspond à un point unique dans la surface de distance. Ainsi, à partir de l'équation (4.6), nous pouvons définir une fonction d'erreur à valeur vectorielle $\xi(t) \in \mathbb{R}^K$, $K = (p_1 + 1)(p_2 + 1) - 1$, tel que :

$$\xi(t) = A \cdot \underline{\theta}(t) - B, \quad (4.7)$$

Où $A = \frac{1}{N \times M} \sum_{t_1=1}^N \sum_{t_2=1}^M \underline{x}(\mathbf{t}) \underline{x}'(\mathbf{t})$ et $B = \frac{1}{N \times M} \sum_{t_1=1}^N \sum_{t_2=1}^M \underline{x}(\mathbf{t}) X(\mathbf{t})$.

Pour trouver le jeu de paramètres correspondant à la distance minimale de $h_{\mathbf{N}}(\underline{\theta})$ de zero, nous faisons en sorte que la fonction d'erreur atteigne une trajectoire nulle $\xi(t) = 0$ en suivant la méthode de conception dynamique neuronale basée sur le gradient Zhang et al (2009), Zhang et Chen (2008). Par conséquent, selon la formule de conception GNN $\dot{\underline{\theta}}(t) = -\gamma \left(\frac{\partial \xi(t)}{\partial \underline{\theta}} \right)$, et en adoptant le réseau de neurones généralisé proposé dans Zhang et Ge (2005), l'équation dynamique du réseau de neurones devient la suivante :

$$\dot{\underline{\theta}}(t) = -\gamma A^T f(A\underline{\theta}(t) - B), \quad (4.8)$$

où $f(\cdot)$ est une fonction d'activation sigmoïdale bipolaire non linéaire et γ est le taux d'apprentissage. Le taux d'apprentissage, dans ce cas, détermine la rapidité ou la lenteur avec laquelle le réseau met à jour les paramètres. Habituellement, un taux d'apprentissage élevé est utilisé et diminue progressivement à mesure que la formation progresse. En bref, γ doit être suffisamment bas pour que le réseau converge vers quelque chose d'utile, mais suffisamment haut pour que nous n'ayons pas à passer des années à le former. Les paramètres sont initialisés aléatoirement, ce qui constitue un point de départ dans l'espace des paramètres.

4.2 Le processus d'apprentissage

La formation de l'architecture présentée dans la section suivante afin d'ajuster le réseau proposé pour accomplir sa tâche de classification implique la sélection de l'ensemble de poids qui caractérise la texture avec le plus haut degré de possibilité. L'algorithme suivant décrit les étapes ordonnées pour apprendre chaque texture particulière :

Algorithme 1 :

Input. I_k : k -ième motif de texture parmi les K textures à distinguer

$M \times N$: taille de l'échantillon

- Etape 1. Sélectionnez cinq échantillons représentatifs de taille $M \times N$ pour I_k

Pour chaque échantillon

- (i) Attribuer de petites valeurs aléatoires à l'ensemble de poids $\{\underline{\theta}^{(k)}\}$ comme initialisation
- (ii) Mettre à jour $\{\underline{\theta}^{(k)}\}$ par :

$$\dot{\underline{\theta}}(t) = -\gamma A^T f(A\underline{\theta}(t) - B),$$

- Etape 2. Prendre l'ensemble des poids pour I_k comme la moyenne des ensembles de poids obtenus à partir du groupe d'échantillons

- Etape 3. Répéter les étapes 1 et 2 jusqu'à l'apprentissage de tous les types de textures

- Etape 4. Attribuer l'ensemble des poids obtenus $\{\alpha_s^{(k)}\}$ pour I_k à partir de l'étape 2 au k -ième canal

4.3 Architecture ANN

Trois sous-réseaux constituent le réseau neuronal proposé : le sous-réseau d'entrée (ISN), le sous-réseau d'analyse (ASN) et le sous-réseau de classification (CSN). En pratique, les modèles spatiaux sont pour la plupart de premier ordre, c'est-à-dire que seules les positions voisines immédiates sont utilisées pour modéliser la valeur d'un site (ou pixel) particulier Basu et Reinsel (1993). Ainsi, par la suite, nous considérons toute texture comme une réalisation d'un processus 2D RCA stationnaire du premier ordre.

4.3.1 Le sous réseau d'entré

Dans le ISN, un motif de texture d'entrée $\{y(i, j)\}$, $1 \leq i \leq M$, $1 \leq j \leq N$, où $M \times N$ est la taille de la texture, est accepté. Sa sortie est un motif de texture normalisé de l'entrée d'origine définie comme :

$$y_I(i, j) = \frac{y(i, j)}{\sqrt{\sum_{i=1}^N \sum_{j=1}^M y^2(i, j)}}$$

4.3.2 Le sous réseau d'analyse

L'ASN est constitué de K canaux correspondant à K types de textures que nous devons distinguer. Chaque canal de l'ASN se compose de trois couches, à savoir la couche tampon (BL), la couche d'estimation (EL) et la couche de sommation (SL). First, a normalized input pattern is received by channel k at the (BL).

Pour chaque canal k , $k = 1, \dots, K$, le noeud à la position (i, j) du sous réseau (EL) est lié à l'ensemble des voisins $\{(i, j - 1), (i - 1, j), (i - 1, j - 1)\}$ de la position correspondante (i, j) dans le sous réseau (BL). Donc, le noeud (i, j) dans la couche (EL) présente une valeur de calculée par :

$$y_{EL}^{(k)}(i, j) = f \left(y_I^{(k)}(i, j) - \sum_{(l, m) \in S[\mathbf{0}, \mathbf{1}]} \left[\alpha_{(l, m)}^{(k)} + \beta_{(l, m)}^{(k)}(i - l, j - m) \right] X(i - l, j - m) \right),$$

où $\left\{ \alpha_{(l, m)}^{(k)} \right\}_{(l, m) \in S[\mathbf{0}, \mathbf{1}]}$ sont les poids de connexion d'entrée trouvés dans le processus d'apprentissage pour le k -ième canal, $\beta_{(l, m)}^{(k)}$ sont des suites de variables aléatoires normales avec une moyenne nulle et une variance estimée par la méthode GMM proposée dans Boulemnadjel *at al* (2016) et f est une fonction d'activation défini par :

$$f(x) = \frac{1 - e^{-\lambda x}}{1 + e^{-\lambda x}}, \quad (4.9)$$

où le paramètre λ modifie la forme de la fonction sigmoïdale. L'intégration du paramètre λ dans la fonction d'activation permet d'atténuer les effets du bruit pour les valeurs larges des entrées car il fonctionne proche de la saturation, et peut donc mettre en œuvre un seuillage et une réduction du bruit.

La valeur de la sortie du canal k est une valeur unique $y_{SL}^{(k)}$, qui est représentée comme entrée au

sous réseau de classification, calculée par :

$$y_{SL}^{(k)} = \sum_{i=1}^N \sum_{j=1}^M \left| y_{EL}^{(k)}(i, j) \right|.$$

4.3.3 Le sous réseau de classification

Le sous-réseau de classification se compose de deux couches : les couches inverses et concurrentes ((IL) et (CL) respectivement). Toute les entrées $y_{SL}^{(k)}$, $k = 1, \dots, K$ sont liés à

chaque nœud de la couche inverse. Les poids des liens sont fixés à 1, sauf pour ceux entre deux nœuds avec le même index. Par conséquent, la valeur de sortie dans (IL) est égale à :

$$y_{IL}^{(k)} = \left[\sum_{k=1}^K y_{SL}^{(k)} \right] - 2 \times y_{SL}^{(k)}.$$

L'expression de $y_{IL}^{(k)}$ garantit que le nœud qui produit la plus grande sortie est celui avec la plus petite entrée. Le (CL) est un réseau gagnant-emporte-tout qui identifie le nœud gagnant unique comme celui ayant une sortie positive. Par conséquent, le motif d'entrée est classé selon le type de texture correspondant à ce nœud gagnant.

4.4 Segmentation d'images par regions en croissance

Haralick et Shapiro (1985) ont fourni une belle description de la segmentation d'image. Ils ont dit que la segmentation est un moyen de séparer l'image en régions simples avec un comportement homogène. La technique de croissance de région est l'une des principales procédures pour réaliser la segmentation d'image en fusionnant certains points de départ ou de petites zones internes si elles sont homogènes selon une mesure d'une propriété de région locale. Dans ce travail, l'extraction de la propriété de la région locale est effectuée à l'aide de la fonction de coût définie par l'équation (4.7). Autrement dit, en supposant que chaque fenêtre observée est une réalisation d'un modèle RCA 2D de premier ordre, ses paramètres estimés peuvent être utilisés comme mesure de la propriété de région locale. La procédure de croissance régionale proposée

comprend les deux étapes décrites dans ce qui suit.

4.4.1 L'étape initiale

Après avoir partitionné une image donnée en blocs de taille égale $2^q \times 2^q$, le but de l'étape initiale est de développer des régions connectées uniformes en examinant chaque bloc pour décider s'il appartient à un ensemble de départ de blocs d'amorçage présélectionnés. Un bloc est ajouté à une partie interne d'une région si et seulement si :

- il n'a pas déjà été attribué à une autre région
- c'est un voisin de cette région
- il a le même type de texture, identifié par le réseau de neurones proposé, que ses quatre

blocs voisins dans les directions horizontale et verticale

Dans le cas contraire, il est considéré comme un bloc indéterminé qui devrait être examiné plus avant au stade du raffinage.

4.4.2 L'étape de raffinage

Dans l'étape de raffinage, l'extension de chaque région interne est effectuée en divisant les blocs non dans les régions internes (blocs indéterminés) en sous-blocs de taille égale. Le processus est effectué de manière récursive jusqu'à ce qu'aucune division ou fusion ne puisse être effectuée dans un sous-bloc. Un bloc indéterminé est examiné à l'étape de raffinage de la manière suivante :

- il est divisé en quatre sous-blocs de taille égale
- chaque sous-bloc étudié est appelé un bloc central
- pour chaque centre, un cadre centré sur ce bloc central est constitué
- le type de texture de tous les sous-blocs voisins est déterminé par le réseau
- si tous ces voisins sont identifiés comme ayant la même texture que la région interne

adjacente, le bloc central correspondant est fusionné dans cette région de texture adjacente

- sinon, la procédure d'affinage continue pour tous les blocs centraux indéterminés jusqu'à ce qu'il ne reste plus de blocs indéterminés dans l'image ou que la taille de bloc de 1×1 soit atteinte.

4.5 Résultats expérimentaux

La formation et le test du système proposé sont menés à l'aide de motifs de texture numérisés à partir de l'album Brodatz (Brodatz (1966)). Premièrement, 15 textures sont utilisées pour effectuer un test de classification après la formation. 10 échantillons de test ont été fournis au réseau pour chaque texture. Le tableau 1 montre des performances assez prometteuses du réseau.

À des fins de comparaison, l'approche proposée et la procédure de classification proposée dans Boulemnadjel et al (2016) (qui émane de la même idée de considérer l'image comme une réalisation d'un modèle RCA 2D de premier ordre) ont été mises en œuvre. Notez que dans Boulemnadjel et al (2016), la procédure proposée a été comparée à l'approche des matrices de cooccurrence et au classificateur SVM (en utilisant les estimations GMM des paramètres 2D

AR (1,1) comme vecteur de caractéristiques) et a donné de meilleurs résultats résultats pour différentes textures. Donc dans ce travail, la comparaison avec ces deux dernières méthodes est omise par manque de place. Un échantillon de 5 textures, pour lesquelles 10 tests ont été fournis au réseau, a été utilisé. Pour tous les résultats présentés dans cette section, la taille du bloc au stade initial est choisie égale à $2^5 \times 2^5$ car elle est la plus proche de celle utilisée dans Boulemnadjel et al (2016), et qui était également la taille de la fenêtre qui produit la meilleure segmentation résultats. Le réseau proposé est formé et testé à l'aide d'une fonction d'activation sigmoïdale définie par l'équation (4.9) pour différentes valeurs de γ et λ .

Nous avons retenu les valeurs 10^{-4} et 1 pour γ et λ respectivement car elles ont amené le réseau à atteindre une erreur de convergence égale à 10^{-10} en un nombre minimum d'itérations. Dans l'AS, les séquences de variables aléatoires $\beta_k(i, j)$, $k = 1, 2, 3$, ont été considérées comme normales avec une moyenne nulle et une variance estimée par la méthode GMM proposée dans Boulemnadjel et al (2016). Les résultats (tableau 1) ont montré de meilleures performances de classification pour le réseau proposé par rapport à la méthode proposée dans Boulemnadjel et al (2016).

L'algorithme est également examiné sur un composite de deux textures Brodatz et un composite de quatre textures (Figure 1. (a), (d) et (g) respectivement). Les résultats de la classification par la méthode proposée et la procédure proposée dans Boulemnadjel et al (2016) sont présentés

dans le Tableau 2 en termes de taux de classification (et illustrés à la figure 1). On peut voir dans le Tableau 2 que pour toutes les images testées, l'algorithme proposé a donné de meilleurs résultats que l'autre algorithme en termes de minimisation du nombre de pixels mal classés.

TABLEAU 1.

Taux de classification obtenu par la méthode proposée et la méthode proposée dans Boulelnadjel et al (2016)

	Proposed method	SVM with 2-D RCA
Fig1.a	96.65	95.95
Fig1.d	97.5	90.68
Fig1.g	84.42	72.56

TABLEAU 2.

Comparaison entre les performances de la méthode proposée et la méthode basée sur l'approche proposée dans Boulelnadjel et al (2016)

Assigned texture → Input texture ↓	1	2	3	4	5
1	9(8)		0(1)	1(1)	
2		10(10)			
3	0(1)		8(7)	1(0)	1(2)
4		1(1)		8(9)	1(0)
5	0(1)				10(9)

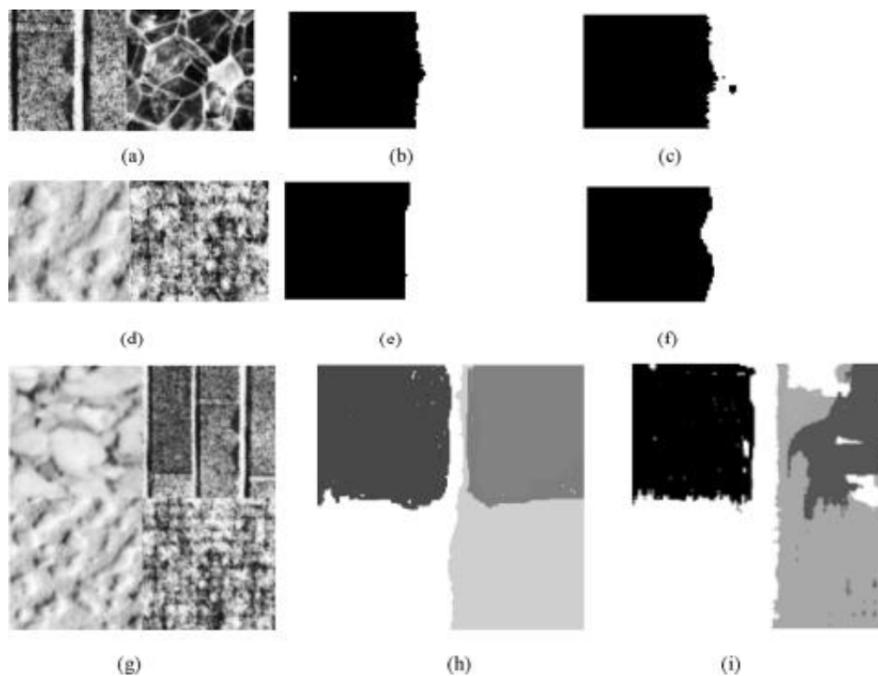


Fig 1. a, d, g : images originales. b, e, h : algorithme proposé. c, f, i : proposed algorithme proposé dans Boulemnadjel et al '2016)

4.6 Conclusion

Dans ce chapitre, nous avons proposé une nouvelle méthode pour segmenter des images texturées à l'aide d'une procédure d'estimation GMM et les ANN pour la modélisation 2D RCA. Un réseau neuronal basé sur le gradient (GNN) a été mis en œuvre pour réaliser l'estimation GMM en minimisant une fonction de coût dérivée d'un ensemble minimal de conditions de moment. Les résultats expérimentaux ont été comparés aux résultats de classification de la procédure proposée dans Boulemnadjel et al (2016) en termes de taux de classification et de performances. La méthode proposée a pris en considération les statistiques réelles du signal. Elle était également efficace pour classer les images synthétiques richement texturées et présentait des résultats compétitifs.

Bibliographie

- [1] Ayache A., Kharfouchi S and Fouad R. (2021) Combination of a 2D-RCA model and ANNs for texture image segmentation. *Int. J. Computing Science and Mathematics*
- [2] Lebart L. (1996) : Correspondence analysis, discrimination, and neural networks. in : *Data Science, Classification and Related Methods, IFCS96, (Fifth International Conference of the International Federation of Classification Societies, Kobe, Japan)*, p 3-6.
- [3] Sylvie, T. Yves, L. Olivier, G. Stéphane, C. (1997) : *Statistique et Méthodes Neuronales*, Dunod, Paris.
- [4] Cottrell, M., Girard, B., Girard, Y., Muller, C. & Rousset, P., (1995) : *Daily Electrical Power Curves : Classification and Forecasting Using a Kohonen Map, From Natural to Artificial Neural Computation*, Proc. IWANN'95, Springer,.
- [5] Cottrell M., Girard B. et Rousset P. (1998) : *Forecasting of curves using a Kohonen classification*, *Journal of Forecasting*, 17, Rousset P. (1999) : *Applications des algorithmes d'auto-organisation à la classification et à la prévision*, Thèse Université Paris 1.
- [6] Cottrell, M., Letrémy, P. (1994) : *Classification et analyse des correspondances au moyen de l'algorithme de Kohonen : application à l'étude de données socioéconomiques*, Proc. Neuro-Nîmes,
- [7] Cottrell M., Gaubert P., Letrémy P., Rousset P., : *Analyzing and representing multidimensional quantitative and qualitative data : Demographic study of the Rhone valley. The domestic consumption of the Canadian families, WSOM'99*, In : Oja E., Kaski S. (Eds), *Kohonen Maps*, Elsevier, Amsterdam,

- [8] Cottrell, M., Letrémy, P. & Roy, E. (1993) : Analyzing a contingency table with Kohonen maps : a Factorial Correspondence Analysis, Proc. IWANN'93, J.Cabestany, J. Mary, A. Prieto (Eds.), Lecture Notes in Computer Science, Springer-Verlag,
- [9] Cottrell, M. & Letrémy, P. (1994) : Classification et analyse des correspondances au moyen de l'algorithme de Kohonen : application à l'étude de données socio-économiques, Proc. Neuro-Nîmes,
- [10] Cottrell M., Letrémy P., (2001, 2003) : Working times in atypical forms of employment : the special case of part-time work, Conf. ACSEG 2001, Rennes, 2001, to appear in Connectionist Approaches in Economics and Management Sciences, Lesage C. et Cottrell M., Eds, Kluwer, Neural networks. A Comprehensive , simon haykin 2 Ed- prentice hall internatational, inc ; 1999.
- [11] Hornik, K., Stinchcombe, M., and White H. (1989). Multilayer feedforward networks are universal approximators, Neural Networks, 2, 359-366.
- [12] Masters, T. (1995) Advanced Algorithms for Neural Networks. A C++ Sourcebook, John Wiley & Sons, New York.
- [13] Specht, D.F. (1991) A Generalized Regression Neural Network. IEEE Transactions on Neural Networks, 2, 568-576.
- [14] Vapnik V. N. (1982) Estimation of dependencies based on emperical data, Springer-Verlag, New York. Vapnik V.N. (1995) The nature of statistical learning theory, Springer-Verlag, New York.
- [15] Friedman J. H. (1994) An overview of predictive learning and function approximation, in : V. Cherkassky, J. H. Friedman, H. Wechsler (eds.), From Statistics to neural networks : theory and pattern recognition application, NATO ASI Series, F136, Springer-Verlag,
- [16] Mac Kay D. J. C. (1992a) Bayesian interpolation, Neural Computation,
- [17] Mac Kay D. J. C. (1992b) A practical bayesian framework for backpropagation networks, Neural Computation,
- [18] Mac Kay D. J. C. (1992c) The evidence framework applied to classification networks, Neural Computation, 4

- [19] Ripley, B. D. (1993) Statistical Aspects of Neural Networks, in Barndorff-Nielsen, O.E., Jensen, J.L. and Kendall, W.S., ed., Networks and Chaos : Statistical and Probabilistic Aspects, London : Chapman & Hall.
- [20] Hastie, T., Tibshirani, R., and Friedman, J. H. (2001) The Elements of Statistical Learning : Data Mining, Inference and Prediction, Springer-Verlag, New York.
- [21] Pal, N. R. and Pal, S. K. (1993) A review on image segmentation techniques, Pattern Recognition, 26.(9).
- [22] Tang, H., Tan, K. C. and Yi, Z. (2007) Competitive neural networks for image segmentation, Studies in computational intelligence, 53.
- [23] Boulemnadjel, A., Hachouf, F., and Kharfouchi, S. (2016) GMM estimation of 2D RCA models with applications to texture image classification. IEEE, Transactions on Image Processing. 25(2), 528-539.
- [24] Weilu, S., and Xu, H. (1995) Textured image segmentation using autoregressive model and artificial neural network. Pattern Recognition,.
- [25] Conlisk, J. (1976) A further note on stability in a random coefficient model, Int. Econ. Rev., 17 (3), 759-764.
- [26] Andel, J. (1976) Autoregressive series with random parameters, Math. Operations for schung statistik, 7 (5), 735-741.
- [27] Zhang, Y., Chen, K., and H. Tan, H. (2009) Performance Analysis of Gradient Neural Network Exploited for Online Time-Varying Matrix Inversion, IEEE Trans. Automa. Contr., 54 (8), 1940-1945.
- [28] Zhang, Y., and Chen, K. (2008) Comparison on Zhang Neural Network and Gradient Neural Network for Time-Varying Linear Matrix Equation $AXB = C$ Solving, IEEE Int. Conf Industr. Tech., Chengdu, China, 1-6.
- [29] Zhang, Y., and Ge, SS. (2005) Design and analysis of a general recurrent neural network model for time-varying matrix inversion. IEEE Trans Neural Netw, 16, 1477-1490.
- [30] Basu, S., and Reinsel, G. C. (1993) Properties of the spatial unilateral first-order ARMA model, Adv. Appl. Probab., 25 (3), 631-648.

- [31] Brodatz, P. (1966) Textures : A Photographic Album for Artists and Designers. New York, NY, USA : Dover.
- [32] Saporta, S (1990). Probabilités Analyse de données et Statistiques. Editions Technip
- [33] Sadegui A, (1997). Self-organization property of Kohonen's map with general type of stimuli distribution Neural Networks.
- [34] Lee K.Y., Cha Y.T. (1990), Park J.H. Artificial neural network methodology for short-term load forecasting NSF Workshop on Artificial Neural Network Methodology in Power System Engineering Clemson University, SC, Apr., pp. 9-10.
- [35] Linde Y., Buzo A., Gray R.M. (January 1980), An algorithm for vector quantizer design IEEE Transactions on Communications, COM-28, 1.
- [36] Kohonen T. (1982), Self-organized formation of topologically correct feature maps Biol. Cybernet, 43, pp. 59-69.
- [37] Kohonen T. (1982), Analysis of a simple self-organizing process Biol. Cybernet, 44, pp. 135-140.
- [38] Kohonen T. (1989), Self-organization and Associative Memory 3rd ed., Springer, New York.
- [39] Saporta G. (1975 a) - Liaisons entre plusieurs ensembles de variables et codages de données qualitatives. Thèse 3^eC., Université Paris VI.
- [40] Saporta G. (1975 b) - Dépendance et codage de deux variables aléatoires. Revue Statist. App/. 23, P 43-63.
- [41] Saporta G. (1977) - Une méthode et un programme d'analyse discriminante sur variables qualitatives. In : Premières Journées Int. Analyse des Données et Informatiques, INRIA, Rocquencourt.
- [42] Saporta G. (1990) - Probabilités, analyse des données et statistiques. Technip, Paris.
- [43] Saporta G., Hatabian G. (1986) - Régions de confiance en analyse factorielle. In : Data Analysis and Informatics, 4, Oiday E. et al. (eds), North-Holland, Amsterdam, p 499508
- [44] ROUSSEL P., MONCET F., BARRIEU B., VIOLA A. (2001) Modélisation d'un processus dynamique à l'aide de réseaux de neurones bouclés. Application à la modélisation de la

- relation pluie-hauteur d'eau dans un réseau d'assainissement et à la détection de défaillances de capteurs, *Innovative technologies in urban drainage*, 1, 919-926, G.R.A.I.E.
- [45] STOPPIGLIA H. [1997], Méthodes statistiques de sélection de modèles neuronaux ; applications financières et bancaires, Thèse de Doctorat de l'Université Pierre et Marie Curie, Paris. Disponible sur le site <http://www.neurones.espci.fr>.
- [46] VAPNIK V. [1995], *The nature of statistical learning theory*, Springer.
- [47] Andel, J. (1976) 'Autoregressive series with random parameters', *Math. Operations for Schung Statistik*, Vol. 7, No. 5, pp.735–741.
- [48] Basu, S. and Reinsel, G.C. (1993) 'Properties of the spatial unilateral first-order ARMA model', *Adv. Appl. Probab.*, Vol. 25, No. 3, pp.631–648.
- [49] Boulemnadjel, A., Hachouf, F. and Kharfouchi, S. (2016) 'GMM estimation of 2D RCA models with applications to texture image classification', *IEEE, Transactions on Image Processing*, Vol. 25, No. 2, pp.528–539.
- [50] Brodatz, P. (1966) *Textures : A Photographic Album for Artists and Designers*, Dover, New York, NY, USA.
- [51] Conlisk, J. (1975) 'Stability in a random coefficient model', *Int. Econ. Rev.*, Vol. 15, No. 2, pp.529–533.
- [52] Conlisk, J. (1976) 'A further note on stability in a random coefficient model', *Int. Econ. Rev.*, Vol. 17, No. 3, pp.759–764.
- [53] Friedman, J.H. (1994) 'An overview of predictive learning and function approximation, in Cherkassky, V., Friedman, J.H. and Wechsler, H. (Eds.) : *From Statistics to Neural Networks : Theory and Pattern Recognition Application*, NATO ASI Series, F136, Springer-Verlag, pp.1–61.
- [54] Haralick, R.M. and Shapiro, L.G. (1985) 'Survey of image segmentation techniques', *Computer Vision Graphics Image Process*, Vol. 29, pp.100–132.
- [55] Hastie, T., Tibshirani, R. and Friedman, J.H. (2001) *The Elements of Statistical Learning : Data Mining, Inference and Prediction*, Springer-Verlag, New York.

- [56] Hornik, K., Stinchcombe, M. and White, H. (1989) ‘Multilayer feedforward networks are universal approximators’, *Neural Networks*, Vol. 2, pp.359–366.
- [57] Mac Kay, D.J.C. (1992a) ‘Bayesian interpolation’, *Neural Computation*, Vol. 4, pp.415–447.
- [58] Mac Kay, D.J.C. (1992b) ‘A practical Bayesian frame work for back propagation networks’, *Neural Computation*, Vol. 4, pp.448–472.
- [59] Mac Kay, D.J.C. (1992c) ‘The evidence frame work applied to classification networks’, *Neural Computation*, Vol. 4, pp.720–736.
- [60] Masters, T. (1995) *Advanced Algorithms for Neural Networks, A C++ Source book*, John Wiley & Sons, New York.
- [61] Pal, N.R. and Pal, S.K. (1993) ‘A review on image segmentation techniques’, *Pattern Recognition*, Vol. 26, No. 9, pp.1277–1294.
- [62] Ripley, B.D. (1993) ‘Statistical aspects of neural networks’, in Barndorff-Nielsen, O.E.,
- [63] Jensen, J.L. and Kendall, W.S. (Ed.) : *Networks and Chaos : Statistical and Probabilistic Aspects*, Chapman & Hall, London.
- [64] Specht, D.F. (1991) ‘A generalized regression neural network’, *IEEE Transactions on Neural Networks*, Vol. 2, pp.568–576.
- [65] Tang, H., Tan, K.C. and Yi, Z. (2007) ‘Competitive neural networks for image segmentation’, *Studies in Computational Intelligence*, Vol. 53, pp.129–144.
- [66] Vapnik, V.N. (1982) *Estimation of Dependencies Based on Empirical Data*, Springer-Verlag, New York.
- [67] Vapnik, V.N. (1995) *The Nature of Statistical Learning Theory*, Springer-Verlag, New York.
- [68] Combination of a 2D-RCA model and ANNs for texture image segmentation 13 Weilu, S. and Xu, H. (1995) ‘Textured image segmentation using autoregressive model and artificial neural network’, *Pattern Recoognition*, Vol. 28, No. 12, pp.1807–1817.
- [69] Zhang, Y. and Chen, K. (2008) ‘Comparison on Zhang neural network and gradient neural network for time-varying linear matrix equation $AXB = C$ solving’, *IEEE Int. Conf Industr. Tech.*, Chengdu, China, pp.1–6.

- [70] Zhang, Y. and Ge, S.S. (2005) ‘Design and analysis of a general recurrent neural network model for time-varying matrix inversion’, *IEEE Trans. Neural Netw.*, Vol. 16, pp.1477–1490.
- [71] Zhang, Y., Chen, K. and Tan, H. (2009) ‘Performance analysis of gradient neural network exploited for online time-varying matrix inversion’, *IEEE Trans. Automa. Contr.*, Vol. 54, No. 8, pp.1940–1945.