Un codage neuro-flou pour la modélisation et l'exploitation d'informations incomplètes

Stéphanie Muller [1,2,3], Patrick Garda [3], Jean-Denis Muller [1], René Crusem [4], Yves Cansi [4]

[1] **Commissariat à l'Energie Atomique** - Direction des Applications Militaires - Direction des Recherches en Ile-de-France
Département de Conception et Réalisation des Expérimentations - B.P. 12 - F-91680 Bruyères-Le-Châtel

[2] **Laboratoire d'Electronique Philips**
22, avenue Descartes - B.P. 15 - F-94453 Limeil-Brévannes Cedex

[3] **Université Pierre et Marie Curie - Laboratoire des Instruments et Systèmes**
Case 252 - 4, place Jussieu - F-75252 Paris Cedex 05

[4] **Commissariat à l'Energie Atomique** - Direction des Applications Militaires - Direction des Recherches en Ile-de-France
Département d'Analyse et Surveillance de l'Environnement - B.P. 12 - F-91680 Bruyères-Le-Châtel

RÉSUMÉ

Cet article présente une méthodologie permettant de gérer des données incomplètes par des systèmes connexionnistes. L'approche proposée consiste à appliquer un codage flou aux données présentées en entrée d'un réseau de neurones. Après avoir rappelé les différents types d'imperfections, nous décrivons la méthodologie proposée pour prendre en compte les données incomplètes dans des systèmes connexionnistes. Nous comparons les résultats obtenus avec ce type de codage et un codage local dans le cadre de la discrimination automatique d'événements sismiques.

ABSTRACT

In this paper, we present a method to model and classify incomplete data in connexionist systems. Our approach consists in a fuzzy coding of the input data of a neural network. After an introduction of the different types of imperfections, we propose a neuro-fuzzy coding in order to take incomplete data into account. Then, we compare the efficiency of this coding to other codings on the problem of the automatic discrimination of seismic events.

1. Introduction

Dans des situations complexes d'analyse de données, la fusion d'informations apporte souvent des solutions satisfaisantes. Ces informations présentent parfois des imperfections rendant leur exploitation difficile. Dans cet article, nous présentons un codage neuro-flou permettant le traitement de données incomplètes et son utilisation sur un problème de discrimination d'événements sismiques.

2. Imperfections des données

Les données sont représentées par un vecteur de caractéristiques et peuvent être entachées d'imperfections. Il en existe trois principaux types [1] :

- les *incertitudes* correspondent à des données pour lesquelles la validité est hypothétique ou subjective,
- les *imprécisions* sont associées à des données pour lesquelles les valeurs de certaines caractéristiques sont difficilement exprimables, voire même ambiguës lorsque deux valeurs sont possibles. Elles sont souvent dues aux limites de mesure des capteurs.
- les *incomplétudes* caractérisent les données pour lesquelles certaines valeurs de caractéristiques sont inconnues.

Plusieurs approches ont été développées pour gérer ces imperfections : la théorie de Dempster-Shafer permet la gestion des incertitudes non probabilistes, la théorie des sous-ensembles flous gère des imprécisions sans toutefois traiter simultanément les incertitudes, enfin la théorie des possibilités associée à la théorie des sous-ensembles flous constitue la « logique floue », seul cadre permettant actuellement la gestion simultanée des imprécisions et des incertitudes.

3. Codage des données

Nous étudions la représentation de données présentant ces différents types d'imperfections en vue d'une classification par un réseau neuronal. Le codage local des données, consistant à associer un unique neurone à chaque caractéristique de la donnée en entrée, est inadapté aux situations d'imperfection. Le codage distribué, où les neurones peuvent participer au codage de plusieurs caractéristiques, peut conduire à des pertes partielles d'information dans certaines situations. Entre le codage local et le codage distribué, on définit le codage semi-distribué [2] : chaque caractéristique est codée par plusieurs neurones, mais un neurone n'est affecté au codage que d'une seule caractéristique. On génère ainsi un motif d'activation des cellules de type « curseur ». Ce codage peut être appliqué aux valeurs d'entrée et/ou de sortie du réseau neuronal. Il permet de représenter explicitement les valeurs des caractéristiques inconnues ou ambiguës comme le montre la figure 1 et donc de traiter les données incomplètes.

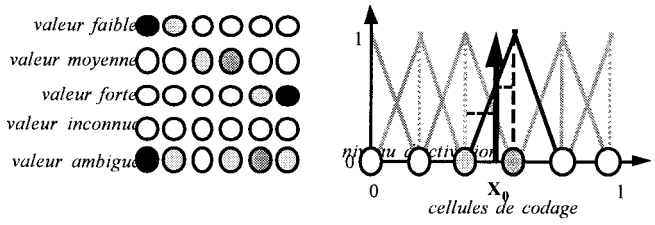

Figure 1 : description du codage semi-distribué. *A gauche : représentation des valeurs possibles d'une caractéristique y compris les imperfections. A droite : principe d'activation des cellules de codage associées à une caractéristique. Une fonction d'activation paire (ici triangulaire) est associée à chaque cellule de codage dont l'activation est représentée par leur niveau de gris. X_0 est la valeur normalisée de la caractéristique à coder.*

Les difficultés posées par ce type de codage sont liées au choix du nombre de cellules de codage, à la dimension de l'espace d'entrée et à la complexité du réseau.

Une première amélioration du codage semi-distribué utilisant les cartes auto-organisatrices de Kohonen est proposée dans [3] dans le cas de données continues. Les cellules ne sont plus espacées d'un pas fixe mais correspondent aux prototypes obtenus par quantification vectorielle de la densité des exemples projetés selon chaque caractéristique. En revanche, la forme de la fonction d'activation de chaque cellule impose, lorsqu'une cellule a une activation maximale, un niveau d'activation des cellules voisines de 0,5. Cette contrainte a pour effet d'atténuer l'impact du placement des cellules de codage obtenu précédemment par quantification vectorielle.

Les méthodes de placement et d'activation des cellules décrites dans [3] n'étant pas optimales, nous proposons deux améliorations du codage semi-distribué pouvant être interprétées dans le cadre de la logique floue :

La première amélioration porte sur le choix de l'algorithme de quantification vectorielle permettant de trouver la position des cellules de codage. Une étude comparative portant sur quatre algorithmes de quantification vectorielle [4] a mis en évidence la supériorité de la qualité de convergence de l'algorithme Neural-Gas [5].

La seconde amélioration porte sur les conditions d'activation des cellules de codage. Nous proposons une activation des cellules dépendant non plus de leur voisinage, mais de leur distance à la valeur à coder. Ainsi, le niveau d'activation de chaque cellule s'interprète comme le degré d'appartenance de la valeur à coder au sous-ensemble flou caractérisé par la forme de la fonction d'activation de la cellule activée. Nous unifions ainsi de manière naturelle deux concepts : le codage flou et le codage neuronal semi-distribué.

4. Validation sur un problème de discrimination d'événements sismiques

La classification automatique d'événements sismiques par des méthodes neuronales a fait l'objet de plusieurs études menées pour la plupart sur des bases de données de petites tailles (quelques dizaines d'événements) composées de signaux sismiques ou de caractéristiques déduites de ces signaux (rapport d'amplitude, cepstre, etc.) [6, 7, 8, 9].

Nous nous intéressons à la classification d'événements sismiques de faible magnitude localisés en France. La décision s'effectue uniquement à partir d'informations de haut niveau obtenues indépendamment de cette étude à partir des signaux sismiques. La base de données, fournie par le Laboratoire de Détection et Géophysique du CEA-DAM, comprend l'ensemble des événements naturels détectés en France entre le 01/01/90 et le 30/04/96, et une partie des événements artificiels (figure 2). Les événements sont répartis en trois classes :

- 5290 séismes naturels,
- 864 tirs (carrières, mines, etc.),
- 1000 coups de terrain (effondrements miniers).

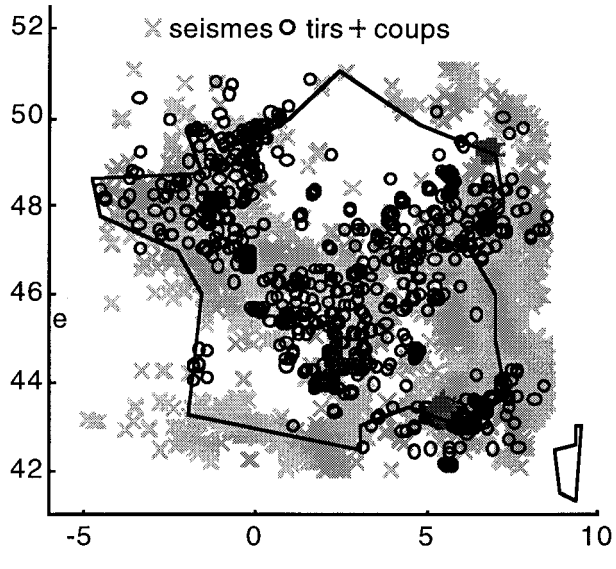

Figure 2 : Répartition événements sismiques de types séisme, tir et coup de terrain dans la base de données.

Chaque événement sismique est défini par un ensemble de caractéristiques de haut niveau parmi lesquelles nous utilisons la date et l'heure de l'événement, la latitude et la longitude de l'épicentre, la magnitude. Le modèle inverse produisant des valeurs imprécises et ne permettant pas systématiquement le calcul de certaines caractéristiques (notamment la magnitude), nous sommes dans l'obligation d'adopter un codage permettant de traiter à la fois des données complètes et incomplètes.

L'architecture neuronale choisie est un perceptron multi-couches. Nous validons les réseaux avec la méthode proposée dans [10] à base de validations croisées de type « bootstrap » [11], qui permet de trouver l'architecture optimale pour un problème donné et d'estimer l'erreur et les performances de généralisation exactes à partir d'expériences réalisées avec un nombre limité d'exemples.

4.1. Classification des données complètes

Dans un premier temps, seuls les événements complets, représentant 95 % de la base de données totale, ont été traités. Trois types de codage ont été comparés : le codage local, le codage semi-distribué et le codage neuro-flou. Dans les deux derniers cas, chaque caractéristique a été codée avec 8 cellules de codage, exceptée la date qui a été codée avec 3 cellules représentant les jours fériés, les samedis et les jours ouvrables. La figure 3 montre le codage neuro-flou de la longitude :

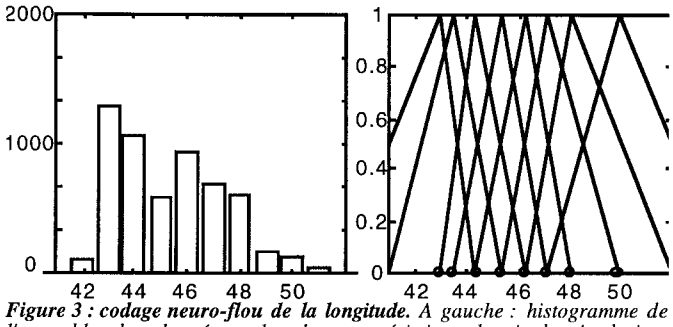

Figure 3 : codage neuro-flou de la longitude. *A gauche : histogramme de l'ensemble des données selon la caractéristique longitude. A droite : position des 8 cellules de codage trouvées par quantification vectorielle avec l'algorithme Neural-Gas et fonctions d'activations associées. Une fonction d'activation recouvre un intervalle de la caractéristique d'autant plus étroit que le nombre de données présent dans cet intervalle est élevé. Le codage neuro-flou s'adapte donc à la répartition des données selon chaque caractéristique.*

Les figures 4 et 5 montrent les performances moyennes obtenues en apprentissage et en test avec des perceptrons monocouches (i.e. sans neurone interne) (figure 4) puis avec des perceptrons comportant 5 neurones internes (figure 5).

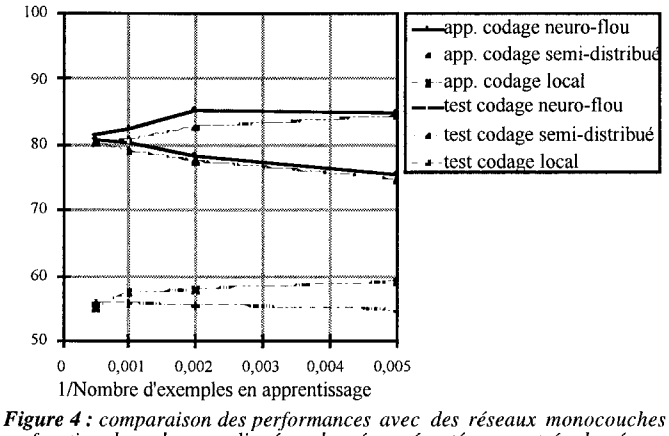

Figure 4 : *comparaison des performances avec des réseaux monocouches en fonction du codage appliqué au données présentées en entrée du réseau de neurones. Le codage neuro-flou, correspondant à un prétraitement non linéaire des données, apporte une amélioration des performances.*

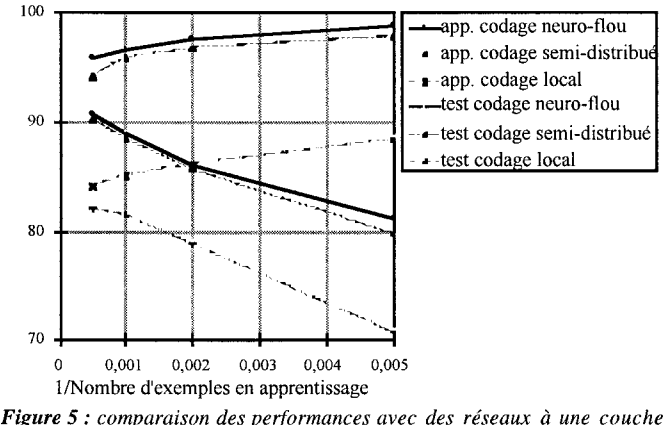

Figure 5 : *comparaison des performances avec des réseaux à une couche interne. On observe toujours une nette supériorité du codage neuro-flou sur le codage local. Néanmoins, l'ajout de neurones internes atténue l'importance du codage des données d'entrées.*

Ces résultats montrent l'importance d'un codage des données adapté en entrée du réseau de neurones : les performances en test, estimées par extrapolation des courbes de la figure 5 jusqu'à l'axe des ordonnées, montrent un gain de 10 points avec le réseau monocouche, passant en moyenne de 83 % de bonnes classifications avec le codage local à 93 % avec le codage neuro-flou. Dans le même temps, l'écart-type est réduit d'un facteur 2,6. La supériorité du codage neuro-flou par rapport au codage semi-distribué s'accroît lorsque le nombre de cellules servant au codage de chaque caractéristique diminue, ce qui accentue l'importance d'un placement des cellules adapté aux données. En revanche, l'ajout de neurones sur la couche interne du réseau réduit l'influence du codage des données. La recherche de la meilleure architecture est toutefois beaucoup plus rapide avec le codage neuro-flou, peu sensible aux variations du nombre de neurones par couche, qu'avec le codage local, qui est en outre structurellement inadapté à l'exploitation de données incomplètes.

4.2. Classification des données incomplètes

Le problème du traitement des données incomplètes a longtemps été ignoré dans la littérature. On recommande même parfois de les éliminer de la base de données lorsqu'elles sont peu nombreuses [12]. Le principal inconvénient de cette approche est la modification de la distribution des données. D'autres solutions consistent à remplacer la caractéristique manquante par une valeur calculée à partir des données complètes : la moyenne de la caractéristique est ainsi couramment choisie comme valeur de substitution. Il est également possible de tester différentes valeurs de remplacement choisies aléatoirement ou de chercher une solution vérifiant le maximum de vraisemblance avec des algorithmes E.M.[13]. Ces solutions peuvent néanmoins conduire à des résultats peu satisfaisants et dépendent fortement de la distribution des données.

Au contraire, nous proposons de coder explicitement l'absence de la valeur d'une caractéristique en affectant un niveau d'activation nul de ses cellules de codage neuro-flou (figure 1). Notons que ce schéma d'activation ne correspond à aucune valeur possible de la caractéristique.

En ajoutant progressivement des données incomplètes dans les bases d'apprentissage, nous avons étudié l'évolution des performances en fonction de la caractéristique manquante et du pourcentage de données incomplètes. L'architecture choisie est un perceptron multi-couches à deux couches internes fournissant les meilleures performances sur les données complètes (figure 6).

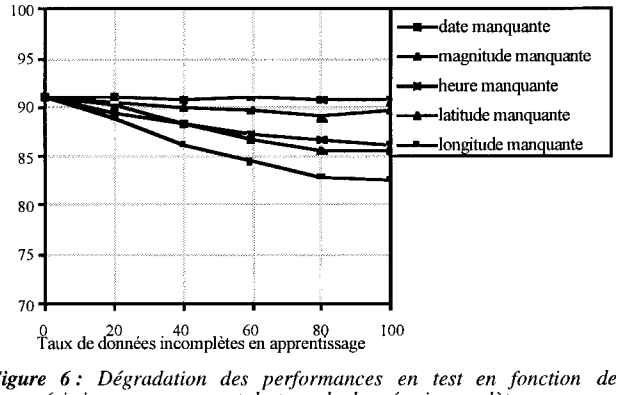

Figure 6 : *Dégradation des performances en test en fonction de la caractéristique manquante et du taux de données incomplètes.*

On ne constate pratiquement aucune dégradation des performances lorsque la caractéristique manquante est la date ou la magnitude : le réseau s'adapte à la classification d'exemples incomplets. A l'opposé, la longitude est la caractéristique la plus sensible : son absence systématique entraîne une dégradation d'au plus 8,5 %. L'analyse des exemples erronés montre que les exemples incomplets sont aussi bien classés que les exemples complets. L'évolution des

performances lorsque les autres caractéristiques sont manquantes a un comportement intermédiaire.

En pratique, le pourcentage de données incomplètes est inconnu. Il est donc souhaitable que les réseaux obtiennent des performances correctes à la fois pour la classification des données incomplètes et des données complètes. Nous avons comparé les performances obtenues en test sur la classification des données complètes avec deux réseaux ayant appris à partir de bases composées d'exemples complets et incomplets en proportions variables, la caractéristique manquante étant successivement la magnitude et la longitude. Nous observons que :

- une incomplétude en apprentissage sur la magnitude n'a pratiquement aucune influence sur la classification des données complètes, quel que soit le pourcentage de données incomplètes en apprentissage,
- une incomplétude en apprentissage sur la longitude conduit à une légère dégradation des performances sur les données complètes atteignant 6,5 % pour une base d'apprentissage comportant 90 % d'exemples incomplets.

Enfin, nous avons mis en évidence l'intérêt d'un apprentissage à partir d'exemples complets et incomplets en présentant des bases de données incomplètes à un réseau n'ayant appris qu'à partir d'exemples complets. On observe alors une forte dégradation des performances atteignant 23 % lorsque la longitude est absente de l'ensemble des exemples de test.

5. Conclusion et perspectives

Nous avons proposé un codage neuro-flou unifiant le codage semi-distribué et des concepts de logique floue. Il permet :

- une amélioration des performances de classifieurs neuronaux sur les données complètes obtenue grâce à une bonne adéquation entre le placement des cellules de codage et leur mécanisme d'activation,
- l'intégration efficace de données incomplètes lors de l'apprentissage sans dégradation notable des performances sur la classification des données complètes.

Nous avons mis en œuvre ce codage pour la classification d'événements sismiques à partir de données pouvant présenter des incomplétudes. Les résultats obtenus sur notre base comportant 7154 événements sismiques sont largement supérieurs à ceux mentionnés dans la littérature.

Les perspectives de cette étude concernent l'extension du codage neuro-flou au cas multidimensionnel, l'extraction de caractéristiques pertinentes à partir du signal sismique pour réduire le taux de confusions tir/séisme et l'application du codage neuro-flou à la gestion de données imprécises. Nous projetons également le passage à une base de données de plus grande dimension.

6. Références

[1] B. Bouchon-Meunier. La logique floue et ses applications. Edition Addison-Wesley, 1995.

[2] P.J.B. Hancock. « *Data representation in neural networks* ». Proceeding of the 1988 Connectionist Models Summer School, Touretzky, Hinton, Sejnowski (Ed.), Morgan Kaufmann, San Mateo, CA, 1988.

[3] V. Lorquet. « *Etude d'un codage semi-distribué adaptatif pour les réseaux multi-couches. Application au diagnostic, à la modélisation et à la commande* ». Thèse de doctorat, Ecole Nationale Supérieure des Télécommunications / CEA, 1992.

[4] S. Muller-Carceles & E. Moser. « *Etude d'algorithmes de quantification vectorielle* ». Rapport de DEA / étude bibliographique. INSTN, CEA-Univ. Paris XI, 1995.

[5] T. M. Martinetz, S. G. Berkovich & K. J. Shulten. « *Neural-Gas network for vector quantization and its application to time-series prediction* ». IEEE Trans. on Neural Networks, Vol. 4, N°4, pp. 558-569, 1993.

[6] F. U. Dowla, S. R. Taylor & R. W. Andersen. « *Seismic discrimination with artificial neural networks: preliminary results with regional spectral data* ». Bulletin of seismological society of America, vol. 80, n°5, pp 1346-1373, 1990.

[7] J. J. Pulli & P. S. Dysart. « *An experiment in the use of trained neural networks for regional seismic event classification* ». Geophysical Research Letters, vol. 17, pp 977-980, 1990.

[8] G. B. Patnaik, T. S. Sereno, R. D. Jenkins. « *Test and evaluation of neural network applications for seismic signal discrimination* ». Phillips Laboratory, Technical Report n° PL-TR-92-2218, 1992.

[9] M. Musil & A. Plesinger. « *Discrimination between local microearthquakes and quarry blasts by multi-layer perceptrons and Kohonen maps* ». Bulletin of seismological society of America, vol. 86, n°4, pp 1077-1090, 1996.

[10] C. Monrocq. « *Approche probabiliste pour l'élaboration et la validation de systèmes de décision. Application aux réseaux de neurones* ». Thèse de doctorat, Université Paris-Dauphine, 1994.

[11] L. Lebart. « *Validation et réechantillonnage* ». Ecole Modulad : statistiques et méthodes neuronales. Montpellier, 1995.

[12] C. Bishop. « *Neural networks for Pattern Recognition* », Oxford University Press, New York, 1996.

[13] Z. Ghahramani & M. I. Jordan. « *Supervised learning from incomplete data via an EM approach* ». In J. D. Cowan, G. T. Tesauro & J. Alspector, Advances in Neural Information Processing Systems, vol. 6, pp 120-127, Morgan Kaufmann (Ed.), San Mateo, CA, 1994.

Ces travaux ont été effectués pour partie dans le cadre d'un Contrat de Thèse CEA - Industrie passé entre le CEA/DAM et les Laboratoires d'Electroniques Philips S.A.S.