Classification à base de clustering : ou comment décrire et prédire simultanément

O. Alaoui Ismaili[1,2] V. Lemaire[1] A. Cornuéjols[2]

[1] Orange Labs, 2 av. Pierre Marzin 22307 Lannion, France
[2] AgroParisTech, 16, rue Claude Bernard 75005 Paris, France
oumaima.alaouiismaili@orange.com

Résumé

Dans certains domaines applicatifs, la compréhension (la description) des résultats issus d'un classifieur est une condition aussi importante que sa performance prédictive. De ce fait, la qualité du classifieur réside donc dans sa capacité à fournir des résultats ayant de bonnes performances en prédiction tout en produisant simultanément des résultats compréhensibles par l'utilisateur. On parle ici du compromis interprétation vs. performance des modèles d'apprentissage automatique. Dans cette thèse, on s'intéresse à traiter cette problématique. L'objectif est donc de proposer un classifieur capable de décrire les instances d'un problème de classification supervisée tout en prédisant leur classe d'appartenance (simultanément).

Mots Clef

Classification, Clustering, Interprétation, Performance, Prédiction, Description

Abstract

In some application areas, the ability to understand (describe) the results given by a classifier is as an important condition as its predictive performance is. In this case, the classifier is considered as important if it can produce comprehensible results with a good predictive performance. This is referred as a trade-off "interpretation vs. performance". In our study, we are interested to deal with this problem. The objectif is then to find out a model which is able to describe the instances from a supervised classification problem and to simultaneously predict their class membership.

Keywords

Classification, Clustering, Interpretation, Performance, Prediction, Description

1 Introduction

De nos jours, les données récoltées par ou pour les entreprises sont devenues un atout important. Les informations présentes, mais à découvrir au sein des grands volumes de données, sont devenues pour ces entreprises un facteur de compétitivité et d'innovation. Par exemple, les grandes entreprises comme *Orange* et *Amazon* peuvent avoir un aperçu des attentes et des besoins de leurs clients à travers la connaissance de leurs comportements. Ces données permettent aussi de découvrir et d'expliquer certains phénomènes existants ou bien d'extrapoler des nouvelles informations à partir des informations présentes.

Pour pouvoir exploiter ces données et en extraire des connaissances, de nombreuses techniques ont été développées. Par exemple, l'analyse multivariée regroupe les méthodes statistiques qui s'attachent à l'observation et au traitement simultané de plusieurs variables statistiques en vue d'en dégager une information synthétique pertinente. Les deux grandes catégories de méthodes d'analyse statistique multivariées sont, d'une part, les méthodes dites descriptives et, d'autre part, les méthodes dites explicatives.

Les méthodes descriptives ont pour objectif d'organiser, de simplifier et d'aider à comprendre les phénomènes existant dans un ensemble important de données. Cet ensemble de données $X = \{x_i\}_1^N$ est composé de N instances sans étiquette (ou classe), chacune décrite par plusieurs variables. On notera $x_i = \{x_i^1, \ldots, x_i^d\}$, l'ensemble de d variables décrivant l'instance $i \in \{1, \ldots, N\}$. Dans d, aucune des variables n'a d'importance particulière par rapport aux autres. Toutes les variables sont donc prises en compte au même niveau. Les méthodes descriptives d'analyse multivariée les plus utilisées sont l'analyse en composantes principales (ACP) [1], l'analyse factorielle des correspondances (AFC) [1], l'analyse des correspondances multiples (ACM) [1] et les méthodes de clustering [2] qui visent à trouver une typologie ou une répartition des instances en K groupes distincts où les instances dans chaque groupe $C_k (k \in \{1, \ldots, K\})$ (ou cluster) doivent être les plus homogènes possibles (*i.e.,* partagent les mêmes caractéristiques).

Les méthodes prédictive ont, quant à elles, pour objectif d'expliquer l'une des variables (dite dépendante) à l'aide d'une ou plusieurs variables explicatives (dites indépendantes). Les principales méthodes explicatives utilisées [3] sont la régression multiple , la régression logistique, l'analyse de variance, l'analyse discriminante, les arbres de décision ([4], [5]), les SVM [6], *etc*.

Si on se place dans le cadre de l'apprentissage automatique

[3], on peut placer les méthodes descriptives dans le domaine de l'apprentissage non supervisé et les méthodes explicatives dans le cas de l'apprentissage supervisé.
Dans cet article, on se place dans le cadre de la classification supervisée où il s'agit d'apprendre un concept cible ($X \rightarrow Y$) dans le but de le prédire ultérieurement pour des nouvelles instances. Dans ce cas, le concept cible Y est une variable *'nominale'*, composée de J classes. Afin d'atteindre cet objectif, de nombreux algorithmes ont vu le jour [7]. Certains entre eux fournissent des résultats difficilement compréhensibles de manière immédiate par l'utilisateur : c'est le cas des modèles appelés "boîtes noires" (*e.g.,* les réseaux de neurones [8] (ANN) et les séparateurs à vaste marge [6] (SVM)). D'autres sont naturellement plus interprétables : c'est le cas des modèles boîtes blanches (*e.g.,* les arbres de décision [4], [5]).
Dans certains domaines appelés "domaines critiques" (*e.g.,* la médecine, les service bancaires, ...), la compréhension (la description) des résultats issus d'un modèle d'apprentissage est une condition aussi importante que sa performance prédictive. De ce fait, la qualité de ces modèles réside donc dans leurs capacités à fournir des résultats ayant de bonnes performances en prédiction tout en produisant simultanément des résultats compréhensibles par l'utilisateur. Dans ce genre de situation, il semble logique de se diriger vers les modèles boîtes blanches. Cependant, certaines études comparatives (*e.g.,* [9]) ont montré que les modèles boîtes noires sont plus performants que les modèles boîtes blanches. De ce fait, il existe un grand intérêt à étudier le compromis *"interprétation - performance"*.
Pour tenter de résoudre cette problématique, deux grandes voies existent (voir Figure 1). La première voie consiste à rendre les algorithmes les plus performants (*e.g.,* les SVM, les ANN, les forêts aléatoires, *etc*) plus interprétables. Cette voie a fait l'objet de nombreuses recherches dans les années passées et cela dans de nombreuses disciplines ([10],[11],[12]). La deuxième voie a, quant à elle, pour but d'améliorer la performance des modèles "naturellement" interprétables". Plusieurs techniques ont été proposées dans ce cadre ([13],[14]). Néanmoins, les améliorations apportées se font souvent en détériorant la qualité d'interprétation du modèle.

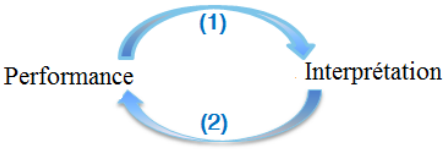

FIGURE 1 – La problématique d'interprétation - performance

Dans cet article, nous nous intéressons à l'étude de la deuxième voie (*i.e.,* de l'interprétation vers la performance). Pour résoudre cette problématique, on part du principe que, d'une part, la performance du modèle dépend de son pouvoir prédictif (*i.e.,* sa capacité à bien prédire la classe des nouvelles instances), et que d'autre part, l'interprétation des résultats du modèle dépend de sa capacité à décrire les données (*i.e.,* sa capacité à fournir des règles compréhensibles par les utilisateurs). De ce fait, notre objectif devient alors : *partir de la description vers la prédiction*.

2 Objectif

L'objectif de cet article est de traiter la problématique "de l'interprétation vers la performance" mais dans un cadre bien précis : *"lorsque chaque classe (ou quelques-unes) dispose d'une structure qui la caractérise"*. La figure 2 présente un exemple illustratif d'un jeu de données caractérisé par la présence de deux classes 'Setosa' et 'Virginica' où chacune dispose d'une structure spécifique : la classe 'Setosa' contient deux sous-groupes distincts ayant chacun des instances homogènes (*i.e.,* partagent les mêmes caractéristique). Tandis que la classe 'Virginica' contient trois sous-groupes distincts.

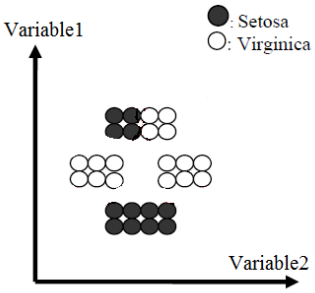

FIGURE 2 – Jeu de données caractérisé par la présence de deux classes ayant chacune une structure qui la caractérise

Dans ce cadre d'étude, on cherche à découvrir les différentes voies qui peuvent mener à une même prédiction. Il s'agit alors de découvrir la structure du concept cible. A titre d'exemple, deux patients x_1 et x_2 ayant comme prédiction un test positif pour l'AVC (*i.e.,* une grande probabilité d'avoir un Accident Vasculaire Cérébral) n'ont pas forcément les mêmes motifs et\ou les mêmes symptômes : il se peut que le patient x_1 soit une personne âgée, qui souffrait de la fibrillation auriculaire et par conséquent, elle a eu des maux de têtes et des difficultés à comprendre. Tandis que le patient x_2, pourrait être une jeune personne qui consommait de l'alcool d'une manière excessive et par conséquent, il a perdu l'équilibre. Généralement, ces différentes raisons que l'on souhaite détecter.
Cette étude a pour but donc, de découvrir, si elle existe, la structure du concept cible à apprendre puis muni de cette structure de pouvoir prédire l'appartenance au concept cible. L'objectif est donc d'essayer de ***décrire et de pré-***

dire d'une manière simultanée. Si le but est atteint, on aura alors à la fois : (1) la prédiction du concept cible et (2) le pourquoi de la prédiction grâce à la découverte de la structure de ce dernier.

Pour atteindre l'objectif désiré, au moins pour l'axe "performance-prédiction", on pense alors immédiatement aux arbres de décisions [4, 5] qui figurent parmi les algorithmes de classification les plus interprétables. Ils ont la capacité de fournir à l'utilisateur des résultats compréhensibles (sous formes de règles) et qui semblent donner une structure du concept cible appris. En effet, l'arbre de décision modélise une hiérarchie de tests sur les valeurs d'un ensemble de variables discriminantes. De ce fait, les instances obtenues suivant un certain chemin (de la racine vers les feuilles terminales) ont normalement la même classe et partagent ainsi les mêmes caractéristiques. Cependant, selon la distribution des données dans l'espace d'entré, l'arbre de décision crée naturellement des polytopes (fermés et ouverts) à l'aide des règles. La présence des polytopes ouverts empêche l'algorithme de découvrir la structure *'complète'*du concept cible Y. La figure 3 présente un exemple illustratif du fonctionnement de l'arbre de décision sur un jeu de données caractérisé par la présence de deux classes ('rouge' et 'noire'), 350 instances et deux variables descriptives $x1$ et $x2$. A partir de ce résultat, on constate que cet algorithme fusionne les deux sous-groupes de classe 'rouge' (situés à la droite de la figure), malgré le fait que les exemples du premier sous-groupe ont des caractéristiques différentes de celles du deuxième sous-groupe. Dans le cas extrême, ces deux sous-groupes peuvent même être très éloignés et donc être de caractéristiques assez différentes.

Pour ce qui est de l'axe "interprétation - description", la découverte de la structure globale d'une base de données (non étiquetées) peut être réalisée à l'aide d'un algorithme de clustering. Les résultats fournit par les algorithmes de clustering sont souvent interprétables. L'utilisateur peut facilement identifier les profils moyens des individus appartenant à un groupe mais aussi les variables qui ont un grand impact sur la formation de chaque groupe. Ceci peut être réalisé à l'aide des outils statistiques simples mais variés (*e.g.,* le tableau des distances, les représentations graphiques et les indices de qualité). Dans notre cas d'étude, l'utilisation d'un algorithme de clustering s'avère insuffisante. En effet, le concept cible à apprendre n'est pas pris en compte et les prédictions ne peuvent pas être réalisées avec une bonne performance : deux instances qui partagent les mêmes caractéristiques n'ont pas forcément la même classe (voir le cluster A de la figure 5 a)).

Partant de ce constat, nous nous sommes fixés comme objectif d'essayer de réaliser au sein du même algorithme d'apprentissage la *description* et la *prédiction* du concept cible à apprendre. Pour cela nous proposons d'adapter un algorithme de clustering aux problèmes de classification supervisée. Autrement dit, l'idée est de modifier un algorithme de clustering afin qu'il soit un bon classifieur (en

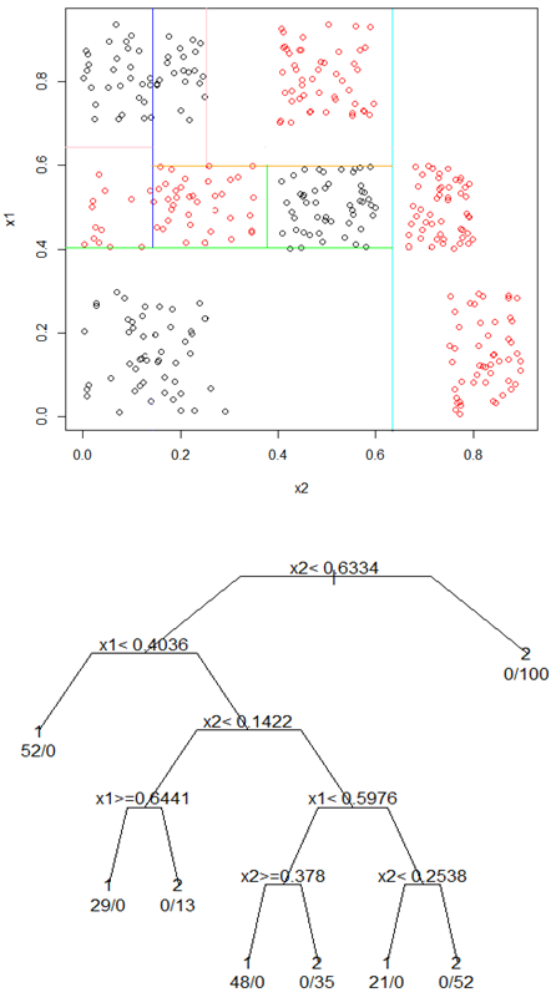

FIGURE 3 – Résolution d'un problème de classification binaire par un arbre de décision. Les feuilles étant pures en termes de classes, l'arbre ne se développe plus.

termes de prédiction) tout en gardant sa faculté à décrire les données et donc le concept cible à apprendre. On parle alors de la ***classification à base de clustering*** (ou décrire et prédire simultanément).
Notre objectif est donc de chercher un modèle qui prend en considération les points suivants :

1. La découverte de la structure interne de la variable cible (la proximité entre les individus).
2. La maximisation de la performance prédictive du modèle.
3. L'interprétation des résultats.
4. La minimisation des connaissances *a priori* requises de la part de l'utilisateur (*i.e.,* pas ou peu de paramètres utilisateur).
5. La minimisation de la taille (complexité) du modèle prédictif - descriptif

3 Classification à base de clustering

Dans notre contexte, la classification à base de clustering est définie comme étant un problème de classification supervisée où un algorithme de clustering standard est soumis à des modifications afin qu'il soit capable à prédire correctement la classe des nouvelles instances. Plus formellement, L'objectif des algorithmes de classification à base de clustering est le même que celui de la classification supervisée, c'est-à-dire prédire la classe des nouvelles instances à partir d'un ensemble de données étiquetées. Dans la phase d'apprentissage (voir Figure 4), ces algorithmes visent à découvrir la structure complète du concept cible à partir la formation des clusters à la fois pures en termes de classe et distincts (*i.e.,* les instances dans chaque cluster doivent être le plus homogènes possibles et différentes de celles appartenant aux autres clusters). A la fin du processus d'apprentissage, chaque groupe appris prend j comme étiquette si la majorité des instances qui le forme sont de la classe j (*i.e.,* l'utilisation du vote majoritaire). Au final, la prédiction d'une nouvelle instance se fait selon son appartenance à un des groupes appris. Autrement dit, l'instance reçoit j comme prédiction si elle est plus proche du centre de gravité du groupe de classe j (*i.e.,* utilisation du 1 plus proche voisin).

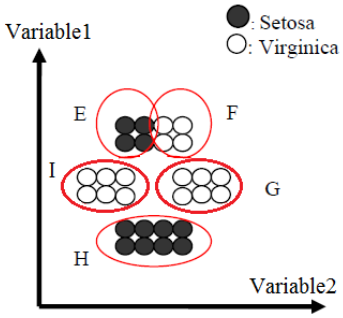

FIGURE 4 – Principe de la classification à base de clustering

Dans la littérature, plusieurs algorithmes de clustering standard ont soumis à des modification afin qu'il soient adapté au problème supervisé ([15],[20],[19]). Ces algorithmes sont connus sous le nom de *"clustering supervisé"* (ou en anglais *"supervised clustering"*). La différence entre le clustering standard (non supervisé) et le clustering supervisé est donnée par la figure 5. Les algorithmes de clustering supervisé visent à former des clusters pures en termes de classe tout en minimisant le nombre de clusters K. Cette contrainte sur K va empêcher les algorithmes de clustering supervisé à découvrir la structure complète du concept cible. De ce fait, un seul cluster peut donc contenir un certain nombre de sous-groupes distincts (voir le cluster G de la figure 5 b)).

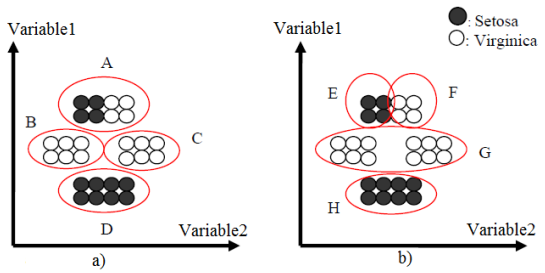

FIGURE 5 – La différence entre le clustering standard a) et le clustering supervisé b)

3.1 État de l'art

Les algorithmes de clustering supervisé les plus répandus dans la littérature sont :

• Al-Harbi *et al.* [15] proposent des modifications au niveau de l'algorithme des K-moyennes. Ils remplacent la distance euclidienne usuelle par une distance euclidienne pondérée. Le vecteur de poids est choisi de telle sorte que la confiance des partitions générées par l'algorithme des K-moyennes soit maximisée. Cette confiance est définie comme étant le pourcentage d'objets classés correctement par rapport au nombre total d'objets dans le jeu de données. Dans cet algorithme, le nombre de clusters est une entrée.

• Aguilar *et al.* [16] et Slonim *et al.* [17] ont proposé des méthodes basées sur l'approche agglomérative ascendante. Dans [16], les auteurs ont proposé un nouvel algorithme de clustering hiérarchique (S-NN) basé sur les techniques du plus proches voisins. Cet algorithme commence par N clusters où N est le nombre d'objets du jeu de données. Ensuite, il fusionne successivement les clusters ayant des voisins identiques (*i.e.,* objets proches ayant la même étiquette). Par conséquent, tous les voisins ayant les distances plus courtes que le premier ennemi (*i.e.,* l'objet qui n'a pas la même étiquette) seront collectés. Tishby *et al.* ont introduit dans [18] la méthode 'information bottleneck'. Basée sur cette méthode, ils ont proposé une nouvelle méthode de clustering (agglomérative) [17] qui maximise d'une manière explicite, l'information mutuelle entre les données et la variable cible par cluster.

• Dans [19], Cevikalp *et al.* ont proposé une méthode qui crée des clusters homogènes, nommée HC. Ces travaux sont effectués dans le but de trouver le nombre et l'emplacement initial des couches cachées pour un réseau RBF. Cevikalp *et al.* supposent que les classes sont séparables puisqu'ils cherchent des clusters purs en classe. Le nombre et l'emplacement des clusters sont déterminés en fonction de la répartition des clusters ayant des chevauchements entre les classes. L'idée centrale de l'algorithme HC est de partir d'un nombre de clusters égal au nombre de classes puis de diviser les clusters qui se chevauchent en tenant compte de l'information supplémentaire donnée par la variable cible.

• Eick *et al.* [20] proposent quatre algorithmes de clustering supervisés, basés sur des exemples représentatifs. Ce

genre d'algorithme a pour but de trouver un sous-ensemble de représentants dans l'ensemble d'entrées de telle sorte que le clustering généré en utilisant ce dernier minimise une certaine fonction de pertinence. Dans [20], les auteurs utilisent une nouvelle fonction pour mesurer la qualité de ces algorithmes. Cette fonction remplit les deux critères suivants : i) Minimisation de l'impureté de classe dans chaque cluster ii) Minimisation du nombre de clusters.

• Vilalta *et al.* [12, 21] mais aussi Wu *et al.* [22] utilisent eux la technique appelée "décomposition des classes". Cette technique repose sur deux étapes principales : (1) réalisation d'un clustering de type k-moyennes (où k est selon les auteurs une entrée ou une sortie de l'algorithme) par groupe d'exemples qui appartiennent à la même classe j. Le nombre de clusters peut différer par classe. On obtient alors P clusters au total ($P = \sum_j k_j$). (2) Entraînement d'un classifieur sur les P classes résultantes et interprétation des résultats. Cette technique permet aussi l'amélioration des classifieurs linéaires (simples).

4 Travaux passés et futurs

L'objectif de cet article est de chercher un modèle qui prend en considération les points énumérés en fin de Section 2 et que ne permettent pas totalement les algorithmes décrits dans la section précédente. Comme une première piste de travail, nous nous intéressons à modifier l'algorithme des K-moyennes. Cet algorithme figure parmi les algorithmes de clustering les plus répandus et le plus efficace en rapport temps de calcul et qualité [23]. Les différentes étapes de l'algorithme des K-moyennes modifié sont présentées dans l'algorithme 1.

1) Prétraitement des données
2) Initialisation des centres
3) Répéter un certain nombre de fois (R) jusqu'à convergence
   3.1 Coeur de l'algorithme
4) Choix de la meilleure convergence
5) Mesure d'importance des variables (après la convergence et sans réapprendre le modéle)
6) Prédiction de la classe des nouveaux exemples.

Algorithme 1 : K-moyennes.

4.1 Travaux réalisés

Etape 1 des k-moyennes (voir Algorithme 1) : Généralement, la tâche de clustering nécessite une étape de prétraitement non supervisé afin de fournir des clusters intéressants (pour l'algorithmes des K-moyennes voir par exemple [24] et [25]). Cette étape de prétraitement peut empêcher certaines variables de dominer lors du calcul des distances. En s'inspirant de ce résultat, nous avons montré dans [26] (à travers l'évaluation de deux approches : conditional Info et Binarization) que l'utilisation d'un prétraitement supervisé peut aider l'algorithme des K-moyennes standard à atteindre une bonne performance prédictive. La prédiction de la classe étant basée sur l'appartenance au cluster le plus proche après la fin de convergence puis sur un vote majoritaire dans le cluster concerné.

Etape 5 des k-moyennes (voir Algorithme 1) : Dans [27] nous avons proposé une méthode supervisée pour mesurer l'importance des variables après la convergence du modèle. L'importance d'une variable est mesurée par son pouvoir prédictif à prédire l'appartenance des instances aux clusters (*i.e.,* la partition générée par le modèle). Autrement dit, une variable est importante si elle est capable de générer une partition proche de celle obtenue en utilisant toute les variables. Dans cette étude, nous n'avons pas considéré les interactions qui peuvent exister entre les variables (*e.g.,* une variable n'est importante qu'en présence des autres). Ceci, fait l'objet des futurs travaux (parmi d'autres).

4.2 Travaux en cours

Etape 2 des k-moyennes (voir Algorithme 1) : Dans [28] nous avons proposé une nouvelle méthode d'initialisation des k-moyennes dans le cas supervisé : le cas où la valeur de K correspond au nombre de classes à prédire (C). Cette méthode est basée sur l'idée de la décomposition des classes après avoir prétraité les données à l'aide de la méthode proposée dans [26]. A l'avenir nous comptons étendre cette méthode lorsque $K \neq C$.

Etape 3.1 des k-moyennes (voir Algorithme 1) : La fonction objectif utilisée dans l'algorithme des k-moyennes standard consiste à minimiser l'inertie intra et par conséquent maximiser l'inertie inter cluster. L'élaboration d'un nouveau critère à optimiser pour pouvoir atteindre l'objectif d'un algorithme à base de clustering est à l'étude.

4.3 Travaux à venir

Pour la deuxième partie de la thèse en cours, on s'intéressera à traiter :

Etape 4 des k-moyennes (voir Algorithme 1) : L'algorithme des k-moyennes n'assure pas de trouver un minimum global. De ce fait, il est souvent exécuté plusieurs fois (on parle de "réplicates") et la meilleure solution en terme d'erreur est alors choisie. Dans le cadre de la classification à base de clustering, le critère utilisé pour choisir la meilleure réplicate doit prendre en considération le compromis homogénéité - pureté des clusters. Dans les travaux à venir, on cherchera à définir un nouveau critère pour le choix de la meilleure "réplicates" dans le cas supervisé.

Etape 6 des k-moyennes (voir Algorithme 1) : par défaut la classe prédite est la classe majoritaire présente dans le cluster. On cherchera à améliorer ce point comme cela existe déjà pour les arbres de décisions.

5 Conclusion

L'ensemble des travaux décrits dans cet article permettra d'obtenir un algorithme complet de "classification à base de clustering" basé sur un algorithme de k-moyennes qui aura été "supervisé" à chaque étape.

Références

[1] Pages, J.P., Cailliez, F., Escoufier, Y. Analyse factorielle : un peu d'histoire et de géométrie. *Revue de Statistique Appliquée, Vol XXVII*, pp. 5-28, 1979.

[2] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering : A review. *ACM* , pp. :264-323, 1999.

[3] Antoine Cornuéjols, Laurent Miclet. Apprentissage artificiel : concepts et algorithmes, Eyrolles 2010.

[4] John Ross Quinlan. C4. 5 : programs for machine learning, volume 1. Morgan kaufmann, 1993.

[5] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen. Classification and regression trees. *CRC press*, 1984.

[6] V. N. Vapnik. The Nature of Statistical Learning Theory. *Springer-Verlag New York, Inc.*, 1995.

[7] S. B. Kotsiantis. Supervised machine learning : A review of classification techniques. *In Proceedings of the 2007 Conference on Emerging Artificial Intelligence Applications in Computer Engineering*, pp. 3-24, 2007.

[8] S. Haykin. Neural Networks : A Comprehensive Foundation. *Prentice Hall PTR, Upper Saddle River, NJ, USA, 2nd edition*, 1998.

[9] R. Caruana and A. Niculescu-Mizil. An Empirical Comparison of Supervised Learning Algorithms. *Proceedings of the 23rd International Conference on Machine Learning*, pp. 161-168, 2006

[10] Monirul Kabir, Md Monirul Islam, and Kazuyuki Murase. A new wrapper feature selection approach using neural network. *Neurocomputing*, 73(16) pp. 3273-3283, 2010.

[11] Glenn Fung, Sathyakama Sandilya, and R. Bharat Rao. Rule extraction from linear support vector machines. *In Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, pp. 32-40. ACM, 2005.

[12] Ricardo Vilalta, and Irina Rish. A Decomposition of Classes via Clustering to Explain and Improve Naive Bayes. *ECML, volume 2837 of Lecture Notes in Computer Science*, pp. 444-455. Springer, 2003

[13] A. A. Gill, G. D. Smith, and A. J. Bagnall. Improving Decision Tree Performance Through Induction- and Cluster-Based Stratified Sampling. *IDEAL, volume 3177 of Lecture Notes in Computer Science*, pp. 339-344. Springer, 2004

[14] Hassan, Md R and Kotagiri, R. A new approach to enhance the performance of decision tree for classifying gene expression data. *BMC proceedings*, pp. S3, 2013

[15] Sami H Al-Harbi and Victor J Rayward-Smith. Adapting k-means for supervised clustering. *Applied Intelligence*, 24(3), pp. 219-226, 2006.

[16] Aguilar J., Roberto Ruiz, José C Riquelme, and Raúl Giràldez. Snn : A supervised clustering algorithm. *In Engineering of Intelligent Systems*, pp. 207-216. Springer, 2001.

[17] Noam Slonim and Naftali Tishby. Agglomerative information bottleneck. *MIT Press*, pp 617-623, 1999.

[18] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 1999.

[19] Hakan Cevikalp, Diane Larlus, and Frederic Jurie. A supervised clustering algorithm for the initialization of rbf neural network classifiers. *In Signal Processing and Communications Applications, 2007. SIU 2007. IEEE 15th, pages 1âĂŞ4. IEEE*, 2007.

[20] Christoph F Eick, Nidal Zeidat, and Zhenghong Zhao. Supervised clustering algorithms and benefits. *In Tools with Artificial Intelligence, 2004. ICTAI 2004. 16th IEEE International Conference on*, pp. 774-776. IEEE, 2004.

[21] Francisco Ocegueda-Hernandez and Ricardo Vilalta. An Empirical Study of the Suitability of Class Decomposition for Linear Models : When Does It Work Well ? *In SIAM* 2013.

[22] Junjie Wu, Hui Xiong and Jian Chen. COG : local decomposition for rare class analysis *In Data Min Knowl Disc (DMKD)*, pp. 191-220. Springer, 2009.

[23] Jain, Anil K. Data Clustering : 50 Years Beyond K-means. *Pattern Recogn. Lett, Elsevier Science Inc*, pp. 651-666, 2009

[24] Milligan, G., Cooper, M. A study of standardization of variables in cluster analysis. *In : Journal of Classi cation, Springer-Verlag.*, 5(2) pp.181-204,1988

[25] Celebi E. M., Hassan A. Kingravi, Patricio A. Vela. A Comparative Study of Efficient Initialization Methods for the K-Means Clustering Algorithm. *Journal of Expert Systems with Applications*, 40(1) pp.200-210, 2013

[26] Alaoui Ismaili O., Lemaire V., Cornuéjols A. Supervised pre-processings are useful for supervised clustering. *Springer Series Studies in Classification, Data Analysis, and Knowledge Organization*, 2015

[27] Alaoui Ismaili O., Lemaire V., and Cornuéjols A. A supervised methodology to measure the variables contribution to a clustering. *In International Conference on Neural Information Processing (ICONIP), Kuching, Sarawak, Malaysia*, 2014

[28] Vincent Lemaire, Oumaima Alaoui Ismaili, Antoine Cornuéjols "An Initialization Scheme for Supervized K-means", *to appear in International Joint Conference on Neural Networks (IJCNN), IEEE, Ireland*, 2015