

Un raisonnement approximatif pour l'apprentissage supervisé de règles

Amel Borgi

Unité de recherche SOIE, ENSI, Tunis
Institut National des Sciences Appliquées et de la Technologie
Centre Urbain Nord de Tunis
BP 676, 1080 Tunis Cedex, Tunisie
Amel.Borgi@insat.rnu.tn

Résumé. Le cadre de ce travail est celui de la méthode d'apprentissage supervisé SUCRAGE qui se base sur la génération automatique de règles de classification. Ces règles sont exploitées par un moteur d'inférence classique : seules les règles dont les prémisses sont vérifiées par la nouvelle observation à classer sont déclenchées. Ce moteur a été étendu à une inférence approximative qui permet de déclencher les règles pas trop éloignées de la nouvelle observation. Nous proposons une utilisation originale du raisonnement approximatif non plus comme un mode d'inférence mais comme un moyen d'affiner l'apprentissage. Le raisonnement approximatif est utilisé pour générer de nouvelles règles dont les prémisses sont élargies : les imprécisions des observations sont alors prises en compte et les problèmes liés à la discrétisation des attributs continus sont atténués. Notre approche a été testée avec différentes bases d'apprentissage et confrontée à une application réelle dans le domaine du traitement d'images.

1 Introduction

Le cadre général de notre travail est celui de l'apprentissage supervisé pour la fouille de données, et plus précisément l'apprentissage supervisé par génération de règles (Duch et al., 2004) (Zhou 2003). La simplicité et la facilité d'interprétation des règles de production font qu'elles constituent une forme de représentation des connaissances largement utilisée dans les systèmes d'apprentissage (Duch et al., 2004) (Haton et al. 1991) (Holmes et al., 2002) (Mikut et al., 2005) (Prentzas et al., 2005) (Rakotomalala, 2005).

Parmi les nombreux travaux portant sur des méthodes d'apprentissage inductif par génération de règles, nous pouvons citer ceux portant sur les arbres de décision (Quinlan, 1986, 1993) (Breiman et al., 1984) (Rakotomalala, 2005), ou les méthodes de type « graphes d'induction » (Zighed et al., 2002). La fonction de classement y est donnée sous forme d'arbre ou de graphe, elle se traduit aisément en une base de règles. Dans ces méthodes dites *mono-attributs* ou *monothétiques*, les prémisses des règles sont construites étape par étape. A chacune d'elles, on ajoute une condition sur un meilleur attribut, et le choix du meilleur attribut se fait selon le pouvoir discriminant relativement aux classes de cet attribut pris seul. D'autres méthodes, qualifiées de *multi-attributs* ou *polythétiques* présentent l'avantage de sélectionner en bloc les attributs qui apparaissent dans les prémisses des règles (DiPalma et al. 97). Parmi ces approches nous pouvons citer des méthodes symboliques explorant de

manière sélective le treillis de Galois d'une relation binaire comme le système CHARADE (Ganascia, 1987), ou également certaines méthodes d'apprentissage dites « disjonctives », par recherche dans l'espace des généralisations ou des hypothèses possibles comme l'algorithme de l'étoile (Michalski et al., 1983). La construction de règles de classification peut également se faire grâce à des approches évolutionnaires comme les algorithmes génétiques (Holmes et al, 2002) (Venturini, 1996), ou dans un contexte flou comme les systèmes d'inférence floue pour la construction de règles de classification floues (Mikut et al., 2005) (Ciliz, 2005) (Nozaki et al., 1994).

Nous avons proposé une méthode polythétique d'apprentissage à partir d'exemples par génération automatique de règles de classification, la méthode SUCRAGE¹ (Borgi, 1999) (Borgi et al. 2001). La fonction de classement est directement donnée sous la forme d'une base de règles de production et les attributs qui apparaissent dans les prémisses des règles sont sélectionnés en bloc. Cette sélection est réalisée par une recherche de corrélations linéaires entre les attributs des exemples d'apprentissage. La construction des règles passe également par une discrétisation des attributs continus. Les conclusions des règles sont des hypothèses sur l'appartenance à une classe, elles sont entachées d'incertitude. La phase de classification proprement dite est assurée par un moteur d'inférence classique qui exploite la base de règles générée et assure la gestion de l'incertitude. Ce moteur met en œuvre un raisonnement que nous qualifions d'« exact » et qui consiste à ne déclencher que les règles dont les prémisses vérifient *exactement* la nouvelle observation à classer.

Dans cet article, nous nous intéressons à un autre raisonnement : le raisonnement approximatif (Zadeh, 1979) (Haton et al. 1991) (El-Sayed, et al., 2003). Il permet d'introduire plus de souplesse et de pallier aux problèmes des frontières dus à la discrétisation. Un tel raisonnement se rapproche plus du mode de réflexion humaine qui n'exige pas toujours une parfaite adéquation entre les faits, ou les causes, pour pouvoir conclure. Le raisonnement approximatif étudié prend en compte l'imprécision des observations et les incertitudes liées aux conclusions. Il permet d'associer un degré de confiance final aux conclusions sur la base d'une adéquation imprécise entre les règles et les observations (Borgi et al. 2001).

L'originalité de notre approche réside dans l'utilisation du raisonnement approximatif, non plus uniquement comme un mode d'inférence, mais pour affiner l'apprentissage. Ce raisonnement permet de générer de nouvelles règles et d'atténuer de cette façon les problèmes liés à la discrétisation et aux imprécisions des observations. Dans notre modèle, le raisonnement approximatif n'a alors plus vocation d'être une méthode d'inférence dont l'objectif est de déclencher certaines règles mais s'inscrit dans le processus d'apprentissage même. L'avantage d'intégrer le raisonnement approximatif dans la phase d'apprentissage, comme nous nous proposons de le faire, est double : d'une part cela permet de prendre en compte les imprécisions des données et des règles (points de coupure stricts lors de la discrétisation), d'autre part cela permet d'avoir une base de règles et une trace de décision plus faciles à interpréter puisque cette fois les règles sont exploitées par un moteur d'inférence classique (inférence « exacte »).

L'article est organisé comme suit : la partie 2 est consacrée à la présentation de la méthode SUCRAGE. Plus précisément, nous y décrivons la phase d'apprentissage, c'est-à-dire la construction des règles, ainsi que la phase de classification, seul le moteur d'inférence exacte y est présenté. Dans la partie 3, nous étudions le raisonnement approximatif

¹ SUpervised Classification by Rules Automatic GEneration

également mis en œuvre dans le moteur d'inférence. La partie 4 s'attache à expliquer le recours au raisonnement approximatif pour générer de nouvelles règles de classification et contribuer ainsi au processus d'apprentissage. Avant de conclure cette étude, nous présentons dans la cinquième partie les principaux résultats expérimentaux obtenus avec différentes bases d'apprentissage notamment dans le cadre d'une application pour la segmentation d'images.

2 La méthode SUCRAGE

2.1 La génération des règles de classification

Dans cette partie, nous nous attachons à décrire la phase d'apprentissage de la méthode de classification SUCRAGE. L'ensemble d'apprentissage, désigné par Ω , est constitué d'exemples décrits par un ensemble d'attributs numériques notés $X_1, \dots, X_i, \dots, X_p$. Les exemples de l'ensemble d'apprentissage sont étiquetés par la classe à laquelle ils appartiennent. Les classes sont représentées par la variable Y qui prend ses valeurs dans un ensemble discret et de cardinal fini C . Les classes sont notées y_1, y_2, \dots, y_C . Les règles générées sont de la forme :

$$A_1 \text{ et } A_2 \text{ et } \dots \text{ et } A_k \longrightarrow y, \alpha$$

Avec

A_i : une condition du type : X_j est dans $[a,b]$. X_j est la $j^{\text{ème}}$ composante du vecteur représentant un objet et $[a,b]$ un intervalle issu de la discrétisation du domaine de variation de l'attribut X_j ,

y : une hypothèse sur l'appartenance à une classe,

α : un degré de croyance représentant l'incertitude de la conclusion.

Notre approche est polythétique dans la mesure où les attributs qui apparaissent dans les prémisses des règles sont sélectionnés en bloc et non pas de manière successive (Di Palma et al., 1997). Cette sélection est réalisée par une recherche de corrélations linéaires entre les attributs des exemples d'apprentissage (Borgi, 1999) (Borgi et al., 2001). La première étape consiste donc à calculer la matrice de corrélation linéaire $R=(r_{i,j})_{p \times p}$ (p : nombre total d'attributs), où $r_{i,j}$ est le coefficient de corrélation linéaire entre les composantes X_i et X_j des vecteurs de l'ensemble d'apprentissage. On décide que deux attributs X_i et X_j sont corrélés si la valeur absolue de $r_{i,j}$ est supérieure à un seuil θ que l'on se fixe.

L'étape suivante dans la construction des règles est celle de la discrétisation. La méthode de discrétisation retenue dans ce travail est la *discrétisation régulière*. Elle consiste à construire M intervalles d'amplitudes égales, notés rg_0, rg_1, \dots, rg_M (Borgi, 1999). Dans (Borgi, 1999) et (Borgi et al., 2001), nous avons étudié et implémenté d'autres méthodes de discrétisation. Il s'agit notamment d'approches de discrétisation supervisée comme la méthode basée sur le MDLPC (Minimum Description Length Principle Cut) introduite par Fayyad et Irani (1993) ou encore la méthode FUSINTER introduite par Zighed et al. (1998) et qui utilise une mesure d'incertitude sensible aux effectifs. Nous nous focalisons ici sur le raisonnement approximatif et ne nous étendons donc pas sur les différentes méthodes de discrétisation.

Nous disposons de plusieurs groupes d'attributs corrélés. Par ailleurs chaque attribut, après la phase de discrétisation, prend ses valeurs dans l'ensemble totalement ordonné $\{rg_0,$

Un raisonnement approximatif pour l'apprentissage supervisé de règles

$rg_1, \dots, rg_{(M-1)}$). En se référant à la théorie des CF (Shortliff et al., 1975) et à l'instar de Vernazza (Vernazza, 1993), nous décidons de regrouper dans une même prémisse tous les attributs corrélés. Les prémisses sont alors construites en considérant pour chaque groupe d'attributs corrélés, un sous-intervalle (rg_i) pour chaque attribut, et ce avec toutes les combinaisons possibles. En fait, les parties prémisses des règles constituent une partition de l'espace des attributs corrélés. La figure 1 illustre une telle partition dans le cas d'un groupe de deux attributs corrélés, par exemple X_4 et X_5 . La partition régulière est obtenue avec une taille de subdivision $M=4$. Dans ce cas, 4^2 prémisses sont générées, par exemple : X_4 est dans rg_3 ET X_5 est dans rg_2 .

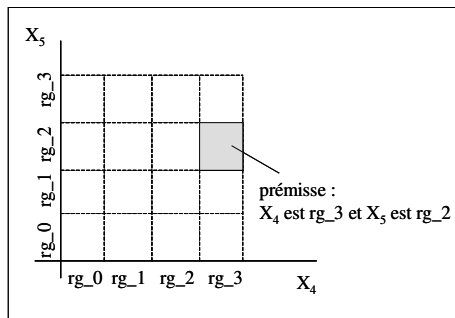


FIG. 1 – Exemple de partition de l'espace des attributs corrélés.

Chaque prémisse construite selon la méthode exposée ci-dessus conduit à la génération de C règles, C étant le nombre total de classes. Chaque conclusion est accompagnée d'un degré de croyance noté α que nous représentons par une probabilité classique (Pearl, 1990) (Shafer, 1996) : c'est la probabilité conditionnelle d'obtenir la conclusion quand la prémisse est vérifiée. Cette probabilité conditionnelle est estimée sur l'ensemble d'apprentissage selon une approche fréquentiste (Borgi, 1999).

Il est à noter que l'on pourrait également adopter une approche bayésienne naïve en considérant les probabilités des règles conditionnellement à la classe, et non plus les probabilités des classes conditionnellement aux prémisses des règles. Plus généralement, nous pouvons représenter l'incertitude des règles de différentes manières, ainsi dans (Borgi, 1999), nous proposons de la représenter par un coefficient de certitude ou Certainty Factors dans le cadre de la théorie de la confirmation (Shortliff et al., 1975).

2.2 Le moteur d'inférence exacte

L'étape suivant la phase d'apprentissage est celle de la reconnaissance ou de la généralisation. Elle consiste à classer de nouveaux objets n'appartenant pas à l'ensemble d'apprentissage. Pour atteindre cet objectif, il est évidemment nécessaire d'exploiter la base de règles. Nous utilisons pour cela un moteur d'inférence d'ordre 0+. Ce moteur reçoit en entrée la base de règles construite lors de l'apprentissage, ainsi qu'un vecteur décrivant l'objet à classer. Le moteur d'inférence associe alors une classe à ce vecteur. La méthode d'inférence utilisée est la méthode d'inférence classique du raisonnement par déduction. Les

règles déclenchées sont celles dont les prémisses sont vérifiées exactement par le nouveau vecteur à classer. Il s'agit du raisonnement que nous qualifions d'« exact ».

Le moteur doit gérer l'incertitude des règles et assurer sa prise en compte dans la dynamique inférentielle. Une fois les règles déclenchées, le problème est d'abord de définir comment les degrés de différentes règles concluant à la même hypothèse, i.e. à la même classe, sont combinés pour obtenir un degré de confiance final en cette classe.

Le choix d'un opérateur adéquat pour un problème donné peut être délicat. Ce problème entre dans le cadre plus général des concepts de l'agrégation des données (Dubois et al. 1985). Nous proposons d'utiliser, pour le traitement de l'incertitude, une co-norme triangulaire (Dubois et al. 1985) (Gupta et al. 1991). Les co-normes sont un exemple de fonctions d'agrégation, ce sont des opérateurs utilisés dans le cadre du traitement des connaissances incertaines et de la prise de décisions (Yager, 1985). Leur origine se retrouve dans l'étude des espaces métriques probabilistes. Ils ont ensuite été introduits dans la théorie des sous-ensembles flous (Gupta et al. 91).

Le coefficient de confiance final associé à chaque classe y_i est donc le résultat du calcul d'une co-norme sur les probabilités associées aux règles déclenchées et concluant à y_i . Deux co-normes ont été utilisées : la co-norme probabiliste ($S(p,q)=p+q-p*q$) et la co-norme de Zadeh (max). Les tests et résultats présentés dans cet article ont été réalisés avec la co-norme de Zadeh : elle permet d'obtenir des résultats similaires ou meilleurs qu'avec la co-norme probabiliste. Finalement, la classe attribuée au vecteur à classer est celle de coefficient de confiance final maximum.

3 Le raisonnement approximatif

Le raisonnement approximatif, de manière générale, fait référence à tout raisonnement qui manipule des connaissances imparfaites. Cette imperfection a de multiples facettes : la connaissance peut être vague, imprécise, incertaine... Malgré de telles imperfections, le raisonnement approximatif permet de traiter ces connaissances et d'aboutir à des conclusions et des prises de décisions. Dans (Haton et al. 1991), le raisonnement approximatif englobe aussi bien la représentation de l'imprécision et de l'incertitude que leur traitement et propagation dans un système à base de connaissances.

Le terme raisonnement approximatif a toutefois une acception particulière introduite par Zadeh dans le domaine de la logique floue (Zadeh 1975, 1979) (Yager, 2000). Dans ce cadre, le raisonnement approximatif correspond au Modus Ponens Généralisé qui est une extension du Modus Ponens à des données floues. Cette définition du raisonnement approximatif n'est pas en contradiction avec la première plus générale et qui englobe toutes les formes d'imperfections.

Le raisonnement approximatif que nous introduisons se situe à l'intersection de ces deux approches. Nous sommes toutefois plus proches des « flouistes » dans la mesure où nous restons fidèles au Modus Ponens Généralisé (Zadeh 1979) (De Baets et al., 1993) mais nous l'adaptions à un cadre symbolique (El-Sayed et al., 2003). Nous proposons un modèle de Raisonnement Approximatif qui permet d'associer un degré de confiance final aux conclusions (aux classes) sur la base d'une adéquation imprécise entre les règles et les observations (les données d'entrée). Ce raisonnement ne déclenche pas uniquement les règles dont les prémisses sont vérifiées exactement par la nouvelle observation, mais également celles qui ne sont pas *trop éloignées* de cette observation. Ainsi nous nous plaçons dans le

Un raisonnement approximatif pour l'apprentissage supervisé de règles

cas d'une règle incertaine : A_1 et A_2 et ... et $A_n \longrightarrow B$ avec un degré α , et d'une observation (A'_1, A'_2, \dots, A'_n) voisine de la prémisse (A_1, A_2, \dots, A_n). Dans ce contexte, illustré par le figure 2, notre raisonnement approximatif permettra de déduire la conclusion (B avec le degré α') en fonction de l'incertitude de la règle et de la proximité de l'observation à la règle.

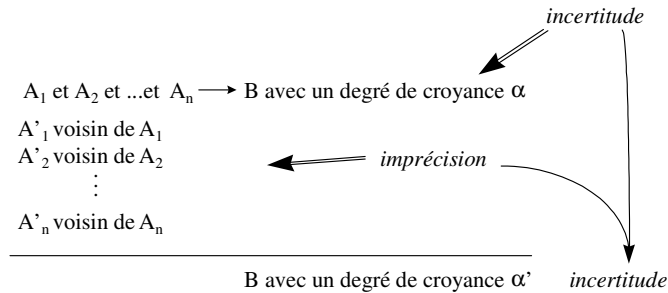


FIG. 2 – Cas particulier du Modus Ponens Généralisé.

Ce Raisonnement Approximatif permet de raisonner en présence d'observations qui sont « approximativement analogues » (autrement dit voisines) à celles des prémisses des règles. La prise en compte d'observations proches des prémisses des règles permet de déborder autour de ces prémisses. Plus précisément, cela permet de déborder autour des intervalles issus de la discrétisation et d'atténuer ainsi les problèmes de frontières liés à toute discrétisation. De plus, le raisonnement approximatif permet de traiter les données imprécises : d'éventuelles imprécisions dues par exemple aux erreurs de mesures des observations sont prises en compte.

Pour que notre raisonnement approximatif puisse devenir opérationnel, il faut formaliser tout d'abord la notion de voisinage. Ensuite, il faut modéliser l'inférence approximative, c'est à dire déterminer le degré de la conclusion finale (α') du schéma ci-dessus en fonction de l'imprécision de l'observation et de l'incertitude de la règle représentée par α .

3.1 Adéquation entre observation et prémisse

Dans le cadre des travaux sur le Raisonnement Approximatif, Zadeh (1971) a souligné la nécessité d'introduire une distance pour définir des faits voisins. Dans le contexte d'une logique multi-valente de Lukaciewicz, Akdag et Pacholczyk (1991) ont également introduit le degré de voisinage symbolique. De même que Ruspini (1991) a introduit le degré de similarité entre deux objets A et A' , qui peut être considéré comme le degré de vérité de la proposition vague « A est similaire à A' ».

Tout en respectant les principes de base incontournables de la notion de voisinage, nous introduisons une distance originale adaptée à notre cadre applicatif, le traitement d'images qui est à l'origine de ce travail. Cette distance va mesurer la proximité d'un élément d'une observation à un élément d'une prémisse (par exemple le niveau d'une couleur de l'observation par rapport à un intervalle de niveau de cette couleur). Ces distances, que nous qualifions de locales, seront ensuite agrégées pour obtenir une distance globale entre l'observation et la prémisse (Borgi, 1999) (Borgi et al., 2001).

3.1.1 Une distance locale

Nous considérons, par souci de clarté, la règle suivante :

$$X_1 \text{ dans } rg_r1 \text{ ET } X_2 \text{ dans } rg_r2 \text{ ET } \dots \text{ ET } X_n \text{ dans } rg_rn \longrightarrow y, \alpha$$

qui regroupe dans sa prémisse les attributs X_1, X_2, \dots, X_n . Cette règle ne perd pas en généralité : elle peut être obtenue à un renommage près des attributs.

Une observation est représentée par un vecteur de \mathbb{R}^p où p est le nombre total d'attributs ($n \leq p$), mais seuls les attributs apparaissant dans la prémisse seront retenus. Nous notons $V=(v_1, v_2, \dots, v_n)$ les éléments de l'observation concernés par la prémisse. Pour comparer V à la prémisse suivante : $X_1 \text{ dans } rg_r1 \text{ ET } X_2 \text{ dans } rg_r2 \text{ ET } \dots \text{ ET } X_n \text{ dans } rg_rn$, nous commençons par faire des comparaisons locales entre v_1 et $X_1 \text{ dans } rg_r1$, entre v_2 et $X_2 \text{ dans } rg_r2$, ...

Il s'agit de définir les distances locales d_1, d_2, \dots, d_n du schéma de Modus Ponens suivant :

$$A_1 \text{ et } A_2 \text{ et } \dots \text{ et } A_n \longrightarrow B \text{ avec un degré } \alpha$$

$$A'_1 \text{ } d_1\text{-distant de } A_1$$

...

$$A'_n \text{ } d_n\text{-distant de } A_n$$

$$B \text{ avec un degré } \alpha'$$

Cela revient plus précisément à déterminer les distances d_i suivantes : v_1 est d_1 -distant de rg_r1 , ..., v_n est d_n -distant de rg_rn

Les v_i ($i \in \{1, \dots, n\}$) sont des valeurs numériques, et les rg_ri sont des valeurs symboliques, des intervalles de valeurs. Pour les comparer, nous introduisons une interface numérique-symbolique (Borgi, 1999). Cette interface discrétise les observations avec par exemple la taille de la subdivision M utilisée pour construire les prémisses des règles (et obtenir les intervalles rg_k). Nous avons préféré discrétiser *chaque* intervalle rg_k en M sous-intervalles, cela permet d'une part d'avoir une distance plus fine, et d'autre part en gardant la même taille de subdivision d'avoir une homogénéité des unités de mesure. La partition de chaque rg_k se fait toujours avec la méthode de discrétisation régulière. Les intervalles issus de cette discrétisation sont notés $\sigma_0, \sigma_1, \dots$. Chaque intervalle rg_k (k variant de 0 à $M-1$) conduisant à M intervalles de type σ , nous obtenons donc $M \cdot M$ intervalles de type σ . Ces intervalles sont donc notés $\sigma_0, \sigma_1, \sigma_2, \dots, \sigma_{M \cdot M - 1}$. La figure 3 illustre les sous-intervalles σ_i obtenus avec une taille de subdivision $M=3$ dans le cas d'un domaine de variation $D=[0,255]$.

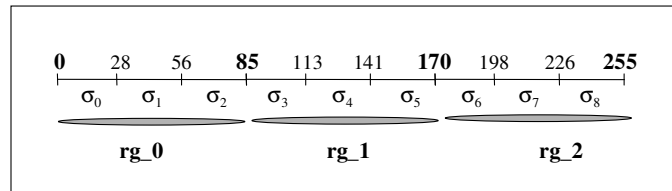


FIG. 3 – Un exemple d'interface numérique-symbolique.

Un raisonnement approximatif pour l'apprentissage supervisé de règles

On peut ainsi associer à chaque valeur numérique v_i le sous intervalle σ_i auquel elle appartient. La distance d_i entre v_i et rg_ri est alors définie par le nombre de sous intervalles de type σ séparant σ_i de rg_ri . Cette distance étant bien entendu nulle si σ_i est inclus dans rg_ri .

La distance locale entre v_i et rg_ri , et plus généralement entre v et rg_k pour un attribut X_i , est alors définie par la fonction $dist$ de la façon suivante :

soit \mathfrak{I} la fonction qui associe à chaque valeur de l'attribut X_i , de domaine de variation R_i , l'indice de l'intervalle de type σ auquel il appartient (et ce pour une taille de subdivision M fixée) :

$$\mathfrak{I} : R_i \longrightarrow [0..M*M-1]$$

$$v \longrightarrow j / v \in \sigma_j$$

Soit σ_i le sous intervalle de type σ auquel appartient v : $t = \mathfrak{I}(v)$, la fonction $dist$ est alors définie par :

$$dist : \begin{matrix} R_i \times \{rg_0, rg_1, \dots, rg(M-1)\} & \longrightarrow & [0..M*(M-1)] \\ (v, rg_k) & & \longrightarrow dist(v, rg_k) \end{matrix}$$

$dist(v, rg_k) =$ nombre de sous intervalles de type σ séparant $\sigma_i = \sigma_{\mathfrak{I}(v)}$ de rg_k .

$$\text{soit : } \begin{matrix} dist(v, rg_k) = 0 & \text{si } t \in [M*k .. M*(k+1)-1] \\ = \sup(t - (M*(k+1) - 1), M*k - t) & \text{sinon} \end{matrix}$$

avec $t = \mathfrak{I}(v)$

Ainsi à chaque couple (observation, prémisse) ou (observation, règle), on associe un vecteur de n composantes, n étant le nombre d'éléments de la prémisse de la règle. Ce vecteur distance à valeurs entières est noté :

$$D = \begin{pmatrix} d_1 \\ d_2 \\ \dots \\ d_n \end{pmatrix} \text{ avec pour tout } i \in [1..n], d_i \in [0..M*(M-1)]$$

3.1.2 Une distance globale

Nous désirons agréger les différentes distances locales d_i (i variant de 1 à n). Le résultat de cette agrégation est une distance globale, que nous notons g -distance, et à laquelle nous désirons conférer un certain nombre de propriétés (Borgi, 1999). Une propriété particulière que nous imposons à cette g -distance est qu'elle soit très sensible vis-à-vis des faits voisins. Cette g -distance doit soit mesurer la proximité entre deux faits voisins, soit indiquer, par la donnée d'une valeur maximale, qu'ils ne sont pas voisins. Notre but n'est pas de mesurer toute distance, mais seulement l'écart entre des vecteurs pas trop éloignés. C'est une mesure de proximité, et non d'éloignement. Cette distance est représentée par un nombre entier de l'intervalle $[0..M-1[$, de plus à partir d'un certain seuil, tous les vecteurs éloignés sont considérés à une distance maximale (égale à $M-1$).

Ainsi, nous imposons à la valeur agrégée $d=g$ -distance (d_1, d_2, \dots, d_n) les conditions suivantes :

$$\text{Propriété 1 : } d \in [0..M-1]$$

Propriété 2 : A partir d'un certain seuil, lorsque les distances locales sont suffisamment élevées, on a : $g\text{-distance}(d_1, d_2, \dots, d_n) = M-1$.

Propriété 3 : Si toutes les valeurs du vecteur distance sont égales entre elles et inférieures strictement à M , la valeur agrégée est égale à cette valeur commune. Ceci permet d'assurer une certaine homogénéité, en particulier lorsque la valeur commune vaut zéro, cela correspondra à deux faits très proches, voire identiques.

Propriété 4 : Si chaque vecteur contient les mêmes distances, le même nombre de fois, alors, ils ont le même agrégat d . Cela suppose que tous les éléments de la prémisse ont la même importance.

Propriété 5 : La g -distance est croissante relativement à chaque argument :

$g\text{-distance}(d_1, d_2, \dots, d_n) \geq g\text{-distance}(d'_1, d_2, \dots, d_n)$ si $d_1 \geq d'_1$
et ce quelle que soit la place de d_1 dans le vecteur.

Nous proposons une formule d'agrégation basée sur une fonction de « dispersion » S :

$$S : k \longrightarrow S(k) = \sum_{i=1}^n (d_i - k)^2$$

S dépend du vecteur distance (d_1, d_2, \dots, d_n) , nous préférons noter cette fonction simplement par S . $S(k)$ permet, de façon similaire à la variance, de mesurer la dispersion des distances locales d_i autour de k .

Nous définissons alors une distance globale de la façon suivante :

$$g\text{-distance} : [0..M*(M-1)]^n \longrightarrow [0..M-1]$$

$$(d_1, d_2, \dots, d_n) \longrightarrow \max[S^{-1}(\min_{k=0}^{M-1} S(k))]$$

Il est prouvé dans (Borgi 1999) (Borgi et al, 2001) que cette distance globale vérifie bien les 5 propriétés énoncées plus haut.

Il est à remarquer que la distance agrégée peut être nulle sans pour autant que les composantes du vecteur distance soient toutes nulles. En d'autres termes il est possible d'avoir une $g\text{-distance}$ nulle pour une observation qui ne vérifie pas exactement la règle.

3.2 L'inférence approximative

La mise en œuvre du Raisonnement Approximatif se base sur une méta-connaissance qui comporte deux volets complémentaires : une première hypothèse classique, qui fait qu'un faible écart ne modifie en rien l'incertitude de la conclusion, une deuxième hypothèse plus forte qui fait que plus l'écart est grand, plus l'incertitude augmente, de telle manière qu'un écart maximal conduise à une incertitude totale (Polya, 1958) (Borgi, 1999).

Pour calculer le degré de croyance final α' d'une conclusion via le Raisonnement Approximatif, étant donné la distance globale d (symbolique) entre les prémisses et l'observation et α le degré de croyance (numérique) de la conclusion de la règle déclenchée, nous proposons la formulation suivantes : $\alpha' = \alpha * (1 - \delta(d))$

Etant donné que α' et α sont de nature numérique, il est nécessaire de numériser l'écart d à l'aide d'une fonction δ :

$$\delta : [0..M-1] \longrightarrow [0,1] \text{ intervalle de IR}$$

$$d \longrightarrow \frac{d}{M-1}$$

Un raisonnement approximatif pour l'apprentissage supervisé de règles

Notons f la fonction de mise à jour du degré de croyance d'une règle en fonction de la distance globale d d'une observation à la prémisse de cette règle :

$$f : [0,1] \times [0..M-1] \longrightarrow [0,1]$$
$$(\alpha, d) \longrightarrow \alpha \cdot \left(1 - \frac{d}{M-1}\right)$$

Nous avons montré dans (Borgi, 1999) que la fonction f ainsi définie vérifie bien les deux hypothèses de travail que nous avons émises plus haut (relatives à la méta-connaissance). Nous avons par ailleurs montré que cette fonction de modification des degrés de croyance possède également la propriété d'une fonction génératrice de modus ponens, notion définie par Trillas et Valverde (1985).

4 Raisonnement approximatif pour l'apprentissage de nouvelles règles

Dans cette partie, nous présentons l'utilisation du raisonnement approximatif non plus comme un mode d'inférence pour exploiter les règles en phase de classification, mais comme moyen d'affiner l'apprentissage. L'utilisation du raisonnement approximatif lors de la phase d'apprentissage consiste à générer de nouvelles règles dont les prémisses sont élargies. La méthode consiste à générer des règles en utilisant le raisonnement exact puis à regarder « autour » de ces règles pour vérifier si on ne peut pas les améliorer ou ajouter de meilleures règles.

La nouvelle base de règles ainsi obtenue sera ensuite exploitée par un moteur d'inférence exacte plus facile à mettre en œuvre. Ainsi le coût de mise en œuvre du raisonnement approximatif est reporté de la phase de classification (utilisée plusieurs fois) à la phase d'apprentissage (construction du modèle). Intégrer le raisonnement approximatif dans la phase d'apprentissage va permettre de prendre en compte les imprécisions des données et des règles (points de coupure stricts lors de la discrétisation) comme cela était le cas lorsque le raisonnement approximatif était utilisé en phase de reconnaissance (comme mode d'inférence). De plus cela va permettre d'avoir une base de règles et une trace de décision plus faciles à interpréter puisque cette fois les règles sont exploitées par un moteur d'inférence classique (inférence « exacte »). Il ne faut toutefois pas perdre en performance : l'objectif est d'obtenir avec cette nouvelle base de règles des résultats proches d'une génération de règles exacte exploitée par un moteur approximatif.

Le raisonnement approximatif introduit dans le contexte des sous-ensembles flous y constitue un mode d'inférence et non pas d'apprentissage (Zadeh, 1975, 1979) (Yager, 2000). Par contre, de nombreux travaux portent sur la construction de règles floues comme les Systèmes d'Inférence Floue (Mikut et al., 2005) (Ciliz, 2005) (Nozaki et al., 1994). Ces approches permettent de prendre en compte les imprécisions et les incertitudes. Elles passent par une partition floue de l'espace des entrées : tous les attributs apparaissent dans les prémisses des règles, et les points de coupure ne sont pas stricts. Notre démarche tout en s'inspirant des ces méthodes (débordement autour des points de coupure et prise en compte des imprécisions) s'inscrit dans un cadre symbolique. Le recours à un raisonnement approximatif « symbolique » permet de gérer les imprécisions tout en assurant l'interprétabilité des résultats en termes de lisibilité de la trace et des inférences.

Nous proposons d'utiliser le raisonnement approximatif pour affiner l'apprentissage de la façon suivante : une observation O située aux alentours d'une règle R que l'on a générée avec

le raisonnement exact, va permettre de générer une nouvelle règle (la règle fille R_{fille}). Pour considérer qu'une observation O est aux alentours d'une règle R , il nous faut définir un g -seuil, c'est la valeur maximale autorisée de g -distance(O,R) :

si g -distance(O,R) \leq g -seuil Alors O est aux alentours de R .

Pour chaque observation O aux alentours d'une règle R (la règle mère) et ayant la même conclusion, on va construire une nouvelle règle (la règle fille R_{fille}) :

- la prémisse est celle contenant $\text{Prémisse}(R)$ et O la plus restrictive possible et convexe en utilisant les rangs et les sous-intervalles de type σ ,
- la classe de la conclusion ne change pas,
- le degré de croyance de la règle est recalculé sur l'ensemble d'apprentissage dans sa totalité avec la nouvelle prémisse.

La phrase « Prémisse contenant $\text{Prémisse}(R)$ et O la plus restrictive possible et convexe en utilisant les rang et les sous-intervalles de type σ » signifie que pour créer la nouvelle prémisse, on part de l'ancienne et on ajoute à toutes les conditions que O ne vérifie pas les intervalles de type σ qui lui permettraient de la vérifier. La règle fille contient dans sa prémisse les mêmes attributs que la règle mère mais avec des valeurs plus larges. La nouvelle règle R_{fille} « extrapole » ou « généralise » R .

De façon plus formelle, considérons, comme dans la présentation du raisonnement approximatif la règle R suivante :

$$X_1 \text{ dans } rg_r1 \text{ ET } X_2 \text{ dans } rg_r2 \text{ ET } \dots \text{ ET } X_n \text{ dans } rg_rn \longrightarrow y_t, \alpha$$

Et soit $V=(v_1, v_2, \dots, v_n)$ le vecteur représentant une observation O (plus précisément il s'agit de la projection du vecteur de dimension p (nombre d'attributs) représentant l'observation sur l'espace, de dimension n , de la prémisse de R). Nous nous plaçons dans le cas où V ne vérifie pas la prémisse de la règle R . Le raisonnement exact ne permettra pas de déclencher la règle R pour l'observation O .

Nous supposons également que la distance globale de V à la prémisse de R est inférieure à g -seuil, en d'autres termes que nous avons : g -distance(d_1, d_2, \dots, d_n) \leq g -seuil, avec d_1, d_2, \dots, d_n les distances locales respectives entre v_1 et rg_r1, v_2 et rg_r2, \dots, v_n et rg_rn .

La nouvelle règle R_{fille} construite à partir de la règle mère R possède dans sa partie prémisse les mêmes attributs que R , mais avec des valeurs plus larges. Sa conclusion est une hypothèse sur l'appartenance à la classe apparaissant dans la conclusion de R . Le degré de croyance est modifié : il est à nouveau estimé sur l'ensemble d'apprentissage. Cette nouvelle estimation du degré de croyance de la règle fille construite via le raisonnement approximatif permet d'intégrer ce raisonnement dans le processus même d'apprentissage.

La règle R_{fille} est définie par :

$$X_1 \in (rg_r1 \cup \sigma'_1) \text{ ET } X_2 \in (rg_r2 \cup \sigma'_2) \text{ ET } \dots \text{ ET } X_n \in (rg_rn \cup \sigma'_n) \longrightarrow y_t, \alpha'$$

avec $\forall i \in [1..n]$

on note σ_j le sous intervalle de type σ auquel appartient v_i , on rappelle que les intervalles de type σ sont obtenus par une discrétisation régulière de chaque intervalle rg_k en M sous-intervalles.

σ'_i est alors défini par :

si σ_j est adjacent à rg_ri , c'est à dire que $d_i = \text{dist}(v_i, rg_ri) = 1$, alors $\sigma'_i = \sigma_j$

sinon σ'_i est l'union des sous intervalles de type σ séparant rg_ri de σ_j (σ_j inclus).

En d'autres termes chaque intervalle rg_ri de la règle R est élargi jusqu'à l'intervalle de type σ contenant la valeur v_i de l'observation. Par exemple, dans le cas d'une discrétisation

Un raisonnement approximatif pour l'apprentissage supervisé de règles

d'ordre $M=3$, nous avons comme l'indique la figure 4, rg_0 composé de σ_0 , σ_1 et σ_2 , rg_1 composé de σ_3 , σ_4 et σ_5 , ... Si la donnée $O_i \in \sigma_4$ et la condition est X_i est dans rg_0 alors la nouvelle condition sera X_i est dans $rg_0 \cup \sigma_3 \cup \sigma_4$ (en supposant bien sûr que la condition de seuil sur la distance globale est vérifiée).

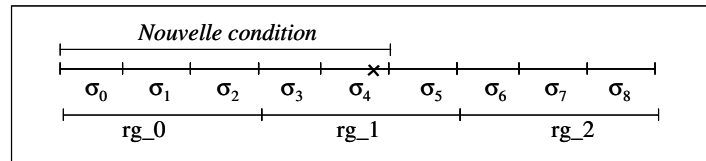


FIG. 4 – Un exemple de construction d'une nouvelle condition

4.1 Méthode à nombre de règles constant

Cette approche peut être résumée par : « A partir d'une observation située aux alentours des règles que l'on a générées avec le raisonnement exact, on vérifie si l'on ne peut pas étendre chaque règle à une règle de meilleure qualité ».

Dans la procédure de construction d'une nouvelle règle que nous avons présentée, c'est un couple (observation, règle) vérifiant certaines propriétés qui donne naissance à une nouvelle règle. Une même règle (règle mère) peut donc conduire à plusieurs règles filles. Avec la méthode à nombre de règles constant, seule la meilleure règle est conservée : sur toutes les règles filles que l'on peut générer et la règle mère, seule celle qui a le plus fort degré de croyance est retenue. Le nombre général de règles ne change donc pas.

4.2 Extension de la méthode à nombre de règles constant : méthode avec ajout de règles

Nous cherchons ici à étendre la méthode de génération d'une nouvelle base de règles afin que la meilleure règle ne soit pas la seule conservée. Pour cela, nous utilisons la « force brute » et nous ajoutons dans la base de règles toutes les règles que l'on peut générer à partir de chacune : une règle peut alors conduire à plusieurs nouvelles règles et non plus comme précédemment à une règle unique (celle de plus fort degré). Cette méthode permet de créer une large base de règles proche des données mais cette base, en raison de sa taille, devient illisible quant à la recherche d'interprétation par un expert.

Il devient alors nécessaire d'optimiser la taille de la base de règles (Mikut et al., 2005) (Prentzas et al., 2005) (Nozaki et al. 1994). Nous avons réalisé un travail dans ce sens et avons utilisé trois approches différentes pour la réduction du nombre de règles : les Algorithmes Génétiques, l'élimination des règles peu crédibles et la sélection des règles par oubli (Borgi, 2005). Notre objectif était d'obtenir un système de classification compact, sans trop perdre en performance. Ces approches testées dans le cas d'une génération exacte des règles ont conduit à des résultats expérimentaux très intéressants. Nous n'avons toutefois pas encore testé ces méthodes pour optimiser une base de règles construite via le raisonnement approximatif. Il serait donc intéressant d'exploiter ces approches dans le cadre d'une génération approximative de règles et plus généralement d'étudier d'autres méthodes

d'élimination de règles inutiles et de simplification de bases de règles (Mikut et al., 2005) (Ciliz, 2005).

5 Tests et résultats

Le système SUCRAGE initialement développé permet la génération de règles par la méthode présentée en 2.1. ainsi que leur exploitation par un moteur d'inférence selon un raisonnement exact ou approximatif (Borgi, 1999) (Borgi et al., 2001). Ce système a été complété par un module de génération de règles via le raisonnement approximatif. Nous avons testé cette nouvelle application sur deux bases d'apprentissage issues du serveur de l'Université d'Irvine², il s'agit des données IRIS et WINE. De plus des tests ont été réalisés avec des données issues de notre application initiale pour la segmentation d'images par classification de pixels (Borgi, 1999) (Borgi et al., 2001). Nous avons utilisé des données réelles provenant d'une image de coupe de cryo-section de cuisses humaines. Les pixels sont représentés par des vecteurs à 5 attributs. Une classification manuelle de l'image par un expert (médecin, anatomiste) a été faite. Quatre tissus ont été identifiés : graisse, os, moelle et muscle, ils correspondent aux classes. La seconde image sur laquelle nous avons testé notre approche représente un papillon coloré. Les pixels sont représentés par des vecteurs de 6 attributs : il s'agit des composantes Rouge, Vert, Bleu ainsi que Teinte, Saturation, Luminance. Les classes sont au nombre de 6 et correspondent à différentes zones de l'image caractérisées par leur couleur (Vert, Rose, Orange, Noir, Jaune et Blanc). Les six classes principales de couleur qui constituent cette image ne sont pas parfaitement séparées, certaines se chevauchent. Cette image permet de tester la robustesse de la méthode de classification vis à vis de ce chevauchement.

Nous avons utilisé la méthode de validation croisée stratifiée d'ordre 10 qui nous a permis d'estimer le taux de succès (taux d'exemples bien classés) de notre classifieur et d'évaluer ainsi sa capacité de généralisation (Kohavi, 1995). Les résultats obtenus sont présentés et analysés dans cette partie. Il s'agit de comparer d'une part les résultats obtenus avec une génération « exacte » des règles par SUCRAGE exploitées ensuite par une inférence exacte ou approximative, avec d'autre part une base de règles construites via le raisonnement approximatif et ensuite exploitée par un moteur d'inférence exacte.

Avant de présenter ces différents résultats, nous donnons ici un tableau comparatif des résultats obtenus avec SUCRAGE (version de base : génération et inférence « exactes ») et d'autres méthodes d'apprentissage. Il s'agit de l'approche par induction de graphe SIPINA (Zighed et al. 2002), ainsi que de trois méthodes de construction d'arbres de décision (ID3, C4.5 (Quinlan, 1986, 1993) et CART (Breiman et al., 1984)). Les données testées sont les bases IRIS et WINE. Chaque case du tableau Tab. 1 contient le taux de bonnes classifications suivi entre parenthèses du nombre de règles.

L'analyse des résultats consignés dans le tableau Tab. 1 montre que les taux de généralisation obtenus avec SUCRAGE sont satisfaisants et plus élevés que ceux obtenus avec les approches graphe d'induction ou arbres de décision considérées ici. Cette amélioration de performance se fait au prix d'un plus grand nombre de règles. Il est toutefois à noter que dans le cas des différentes approches considérées (SIPINA, ID3, C4.5, et CART), une retranscription des arbres de décision et graphes d'induction en règles est nécessaire. De

² ftp://ftp.ics.uci.edu/pub/machine-learning-databases/

Un raisonnement approximatif pour l'apprentissage supervisé de règles

plus, cette retranscription est parfois suivie d'une simplification de la base de règles, ce qui permet d'en diminuer le nombre. Une simplification de la base de règles obtenue à l'aide de SUCRAGE autoriserait une comparaison plus précise avec les autres approches.

Méthode Données	SUCRAGE	SIPINA	ID3	C4.5	CART
IRIS	97.33 (23.5)	94.7 (3.6)	94.7 (3.6)	95.13 (4.7)	93.3 (3.0)
WINE	92.05 (96.4)	86.1 (8.2)	82.8 (4.9)	87.7 (6.6)	81.6 (5.0)

TAB. 1 – Taux de généralisation moyen et nombre de règles moyen. Comparaison de SUCRAGE (génération et inférence « exactes ») avec les graphes d'induction et les arbres de décision.

5.1 Résultats de la méthode à nombre de règles constant

Nous avons testé la méthode de construction de nouvelles règles via le raisonnement approximatif selon l'approche à nombre de règles constant. Les premiers tests ont été effectués avec $g\text{-seuil} = 0$ ou $g\text{-seuil} = 1$ qui semblent les seules valeurs raisonnables. Des valeurs supérieures à 2 lanceraient une recherche que l'on ne pourrait pas considérer comme aux alentours de la règle. Par exemple, pour une taille de discrétisation $M=3$, et une règle *Si* X_1 dans rg_0 ET X_2 dans rg_0 Alors ..., un point de (σ_0, σ_7) qui est dans (rg_0, rg_2) serait accepté (la distance globale vaut 2).

L'analyse des résultats et l'émission d'hypothèses pour les expliquer peuvent se faire en examinant plus précisément la forme des règles générées. Nous distinguons deux cas en fonctions de $g\text{-seuil}$ ($g\text{-seuil} = 0$ ou 1).

Cas $g\text{-seuil}=0$. Ce cas donne des résultats (taux de bonnes classification) presque identiques à la génération exacte (suivie d'une inférence exacte). Ces résultats sont quasiment identiques à ceux de la colonne « Générateur Exact, Inférence Exacte » du tableau 2.

Sur les données testées, il n'y a que très peu de changements entre les règles générées exactement et approximativement. Ceci est dû principalement au constat suivant : il est impossible, pour une prémisse contenant un nombre strictement inférieur à 3 attributs d'avoir un point à une $g\text{-distance}$ nulle.

Toutes les règles de la forme X_i dans rg_ri ET X_j dans $rg_rj \rightarrow y, \alpha$ ne peuvent être améliorées. Une telle valeur de $g\text{-seuil}$ n'est intéressante que pour de longues prémisses.

Voici un exemple d'une règle mère et de la règle fille générée :

Règle mère :

```
regle { #22
si
X0 est { rang_2 }
X2 est { rang_2 }
X3 est { rang_2 }
alors IrisVirginica 0.95
}
```

Règle fille :

```
regle { #22
si
X0 est { rang_2 sigma_5}
X2 est { rang_2 }
X3 est { rang_2 }
alors IrisVirginica 0.96
}
```

Cet exemple illustre l'un des changements les plus importants que l'on puisse avoir avec une prémisse contenant 3 attributs : la nouvelle règle ne pourra « s'élargir » que jusqu'à un seul sous-intervalle de type σ . Si on suppose que la règle est améliorée (c'est-à-dire que le degré de croyance a augmenté), il n'est pas certain que cela puisse changer significativement les résultats.

Cas $g\text{-seuil}=1$. Le Tableau 2 présente les taux de bonnes classifications obtenus avec les données IRIS et WINE avec chacune des trois approches possibles :

- colonne « Générateur Exact, Inférence Exacte » : les règles ont été générées par SUCRAGE. La base de règles est ensuite exploitée par un moteur d'inférence exacte.
- colonne « Générateur Exact, Inférence Approximative » : les règles ont été générées par SUCRAGE. La base de règles est ensuite exploitée par un moteur d'inférence approximative. C'est les résultats de cette approche que nous espérons approcher (voire améliorer) en utilisant le raisonnement approximatif pour construire de nouvelles règles.
- colonne « Générateur Approximatif 1, Inférence Exacte » : les règles ont été générées par SUCRAGE puis de nouvelles règles ont été construites via le raisonnement approximatif, avec une valeur de $g_seuil = 1$. La base de règles est ensuite exploitée par un moteur d'inférence exacte.

Méthode	Générateur Exact, Inférence Exacte	Générateur Exact Inférence Approximative	Générateur Approximatif 1 Inférence Exacte
Données			
IRIS	97.33	97.33	98.00
WINE	92.05	92.71	93.27

TAB. 2 – Résultats sur les données IRIS et WINE. Méthode à nombre de règles constant, $g\text{-seuil}=1$

Avec les deux ensembles de données IRIS et WINE, la génération approximative des règles permet une légère amélioration des résultats par rapport à ceux obtenus avec une construction « exacte » des règles (que celles-ci soient ensuite exploitées par une inférence exacte ou approximative). Bien que notre objectif (ne pas perdre en performance par rapport à la génération exacte des règles suivie d'une inférence approximative) soit atteint, ces résultats ne sont pas concluants : ici une génération exacte des règles suivie par une inférence exacte conduit déjà à de bons résultats. Le raisonnement approximatif introduit en tant que mode d'inférence ou pour affiner l'apprentissage ne semble pas pertinent pour ces données.

Dans le cas des données issues des images du papillon coloré et de cryo section, la méthode de génération approximative de règles à nombre de règles constant (et $g\text{-seuil}=1$) donne des résultats similaires à ceux obtenus avec une génération exacte des règles suivie d'une inférence approximative (Tableau 4, colonne « Générateur Exact, Inférence Approximative »). Avec les deux approches (inférence approximative ou génération approximative à nombre de règles constant et $g\text{-seuil}=1$), il y a une faible perte en performance par rapport à une génération exacte suivie d'une inférence exacte (Tableau 4, colonne « Générateur Exact, Inférence Exacte »).

Pour $g\text{-seuil}=1$, l'observation « autour » des règles n'est plus ici insuffisante (cas $g\text{-seuil}=0$) mais peut l'être trop : on assiste parfois à la création de règles *doublon*. L'observation aux alentours d'une règle peut aller jusqu'à l'observation d'une autre règle

Un raisonnement approximatif pour l'apprentissage supervisé de règles

exacte qui a déjà été formulée, c'est alors la plus forte qui va l'emporter. Il peut y avoir ici une perte d'information. L'algorithme tend alors à créer une absorption des règles faibles par les règles fortes plutôt qu'une extension des règles fortes.

5.2 Résultats de la méthode avec ajout de règles

Le tableau 3 présente les résultats obtenus avec les données IRIS et WINE, et le tableau 4 ceux obtenus avec les données image (image de cryo-section et image du papillon), et ce avec la méthode étendue de génération de nouvelles règles via le raisonnement approximatif : cette fois chaque nouvelle règle générée est ajoutée à la base de règles. La colonne « Générateur Approximatif étendu, Inférence Exacte » de ces tableaux contient les résultats obtenus avec cette approche, l'intitulé des deux premières colonnes est inchangé par rapport au tableau 2. De plus chaque case contient le taux de bonnes classifications suivi entre parenthèse du nombre de règles (qui prend pour cette méthode étendue de l'importance).

Méthode	Générateur Exact Inférence Exacte	Générateur Exact Inférence Approximative	Générateur Approximatif étendu Inférence Exacte
Données			
IRIS	97.33 (23.5)	97.33 (23,5)	97.33 (61.4)
WINE	92.05 (96.4)	92.71 (96.4)	91.57 (343.8)

TAB. 3 – Résultats sur les données IRIS et WINE. Méthode avec ajout de règles.

Méthode	Générateur Exact Inférence Exacte	Générateur Exact Inférence Approximative	Générateur Approximatif étendu Inférence Exacte
Données			
Image cryo section	86.91 (68.5)	84.25 (68.5)	91.33 (879.7)
Image papillon	82.06 (107.9)	80.72 (107.9)	86.45 (937.8)

TAB. 4 – Résultats sur les données image de cryo-section et image papillon. Méthode avec ajout de règles.

Avec le générateur approximatif étendu, les performances sont maintenues (voire très légèrement dépréciées avec les données WINE) par rapport aussi bien au générateur exact suivi d'une inférence exacte qu'au générateur exact suivi d'une inférence approximative. La méthode de génération étendue procédant par ajout de règles, il y a donc augmentation de la taille de la base de règles. Dans le cas des données IRIS et WINE, cette approche ne semble donc pas très intéressante au vu des calculs supplémentaires effectués. Encore une fois, le raisonnement approximatif ne semble pas adapté à ces données.

Les résultats obtenus avec l'image du papillon coloré et celle de cryo-section sont très satisfaisants dans la mesure où la méthode de génération de règles approximative étendue améliore sensiblement les taux de bonnes classification par rapport aux deux autres approches (de l'ordre de 5%). Cette fois, avec les données image, le raisonnement approximatif semble adéquat. Cela est sans doute lié à la nature des données, les imprécisions, prises en compte par l'approche approximative, y sont plus importantes.

Avec cette méthode, le nombre de règles générées augmente par contre très largement. Des règles inutiles sont sans doute générées, voire des règles nocives entraînant une baisse des résultats. Un travail de diminution du nombre de règles devient ici indispensable, aussi bien pour éliminer les règles inutiles ou nocives, que pour des raisons de lisibilité de la base de règles (Nozaki et al. 1994) (Prentzas et al., 2005) (Ciliz 2005) (Borgi, 2005).

6 Conclusion

La méthode d'apprentissage supervisé SUCRAGE permet de générer des règles de classification puis des les exploiter par un moteur d'inférence qui met en œuvre soit un raisonnement exact soit un raisonnement approximatif. L'originalité de notre approche réside dans l'utilisation du raisonnement approximatif pour affiner l'apprentissage : ce raisonnement n'est plus considéré uniquement comme un second mode de fonctionnement du moteur d'inférence, en complément du raisonnement exact, mais est appréhendé comme un prolongement de la phase d'apprentissage. Le raisonnement approximatif va permettre de générer de nouvelles règles plus larges, plus générales, et d'atténuer, de cette façon, les problèmes liés à la discrétisation (points de coupure stricts). De ce point de vue, ce processus peut être considéré comme une alternative aux méthodes de discrétisation supervisée. Il serait alors intéressant de comparer les résultats obtenus par la génération approximative de règles à ceux obtenus avec des approches de discrétisation supervisée comme la méthode basée sur le MDLPC (Minimum Description Length Principle Cut) introduite par Fayyad et Irani (1993), la méthode FUSINTER introduite par Zighed et al. (1998) et qui utilise une mesure d'incertitude sensible aux effectifs, ou encore les méthodes basées sur une mesure d'entropie comme dans ID3 (Quinlan, 1986, 1993) et CART (Breiman et al., 1984). Le processus de raffinement de l'apprentissage que nous avons proposé permet en outre de prendre en compte l'imprécision des données. L'autre avantage d'intégrer le raisonnement approximatif dans la phase d'apprentissage, est que cela permet d'avoir une base de règles et une trace de décision plus faciles à interpréter puisque cette fois les règles sont exploitées par un moteur d'inférence classique (inférence « exacte »). De plus, le surcoût lié à la mise en œuvre du raisonnement approximatif est déplacé de la phase de reconnaissance à celle d'apprentissage.

Les différents tests réalisés conduisent à des résultats en généralisation satisfaisants dans la mesure où ils sont proches de ceux obtenus avec une génération de règles « exacte » exploitée par un moteur approximatif : l'objectif était de conserver les performances mais en déplaçant le raisonnement approximatif de la phase de reconnaissance à celle d'apprentissage. Toutefois, les résultats obtenus avec les données IRIS et WINE ne sont pas concluants : l'approche approximative ne semble pas adaptée à ces données (que ce soit pour l'inférence approximative ou la génération approximative de règles). Par contre les résultats obtenus avec les données images sont très intéressants, ils dépassent notre objectif initial puisque la génération des règles via le raisonnement approximatif permet d'améliorer les taux de bonnes de classification de façon significative.

Une perspective immédiate de notre travail est de compléter l'évaluation de notre approche en réalisant des tests sur d'autres benchmarks (contenant des imprécisions), et également en traitement d'images. Plus généralement, la suite du travail va s'orienter sur la première méthode de génération de nouvelles règles (à nombre de règles constant) pour la rendre plus proche de ce qui se passe lors de l'inférence approximative. La recherche d'autres

Un raisonnement approximatif pour l'apprentissage supervisé de règles

formes de *g-distance* peut s'avérer utile notamment pour pouvoir obtenir des résultats de génération entre les valeurs 0 (où l'on reste trop proches de l'observation) et 1 (où l'on s'éloigne trop des « alentours » de l'observation) de *g-seuil*. La seconde méthode, qui enrichit la base de règles avec toutes les nouvelles règles, est pénalisée par la taille finale de la base obtenue. Une perspective intéressante est de se pencher sur les façons de réduire le nombre de règles sans trop perdre en qualité de classification, et plus généralement de mettre à jour la nouvelle base de règles en gérant notamment les règles redondantes ou contradictoires (Nozaki et al. 1994) (Ciliz 2005) (Mikut et al., 2005).

Références

- Akdag H., Pacholczyk D. (1991). *Symbolic treatment of hierarchical questionnaires*. Proceedings of WOCFAI'91, M. De Glas et D. Gabbay Editors, Angkor, Paris.
- Borgi A., (2005). Différentes méthodes pour optimiser le nombre de règles de classification dans SUCRAGE, *3rd International Conference: Sciences of Electronic, Technologies of Information and Telecommunications*, SETIT 2005, 11 pages, Tunisia.
- Borgi A, Akdag H. (2001). Apprentissage supervisé et raisonnement approximatif, l'hypothèse des imperfections. *Revue d'Intelligence Artificielle*, vol 15, n°1, pp 55-85.
- Borgi A., Akdag. H. (2001). Knowledge based supervised fuzzy-classification : An application to image processing. *Annals of Mathematics and Artificial Intelligence, Special Issue : Representation of Uncertainty*, Vol 32, p 67-86, 2001.
- Borgi A. (1999). *Apprentissage supervisé par génération de règles : le système SUCRAGE*, Thèse de doctorat, Université Paris VI.
- Breiman L. , J.H. Friedman, R.A. Olshen, C. J. Stone (1984). *Classification and regression trees*. Chapman and Hall.
- Ciliz Kemal (2005). Rule base reduction for knowledge-based fuzzy controllers with application to a vacuum cleaner. *Expert Systems with Applications*, 28, p. 175-184.
- De Baets B. and E.E. Kerre (1993). The generalized modus ponens and the triangular fuzzy data model. *Fuzzy Sets and Systems*, Vol. 59, pp 305-317.
- Di Palma S., J. Da Rugna, D.A. Zighed (1997). Apprentissage supervisé polythétique : une évaluation. *Cinquièmes Rencontres de la Société Francophone de Classification*, Lyon.
- Dubois D., Prade H. (1985). *Théorie des possibilités. Application à la représentation des connaissances en informatique*. Masson, Paris.
- Duch W., Setiono R., Zurada J. (2004). Computational Intelligence Methods for Rule-Based Data Understanding. *Proceedings of the IEEE*, VOL. 92, NO. 5.
- El-Sayed M., Pacholczyk D. (2003). Towards a Symbolic Interpretation of Approximate Reasoning, *Electronic Notes in Theoretical Computer Science*, Vol. 82, Issue 4, 1-12.
- Fayyad U.M. and K. Irani (1993). Multi-interval discretization of continuous-valued attributes for classification learning. *Proceedings of The 13th International Joint Conference on Artificial Intelligence*, pp. 1022-1027. Morgan Kaufman.

- Ganascia J-G. (1987). CHARADE : A Rule System Learning System. *Proceedings of the Tenth Int. Joint Conf. on Artificial Intelligence, IJCAI 87*, Vol. 1, 345-347. Milan, Italy.
- Gupta M. M., Qi J. (1991). Connectives (And, Or, Not) and T-Operators in Fuzzy Reasoning. *Conditional Inference and Logic for Intelligent Systems*, 211-233, North-Holland.
- Haton J-P., Bouzid N., Charpillet F., Haton M-C., Lâasri B., Lâasri H., Marquis P., Mondot T., Napoli A. (1991). *Le raisonnement en intelligence artificielle. Modèles, techniques et architectures pour les systèmes à base de connaissances*, InterEditions.
- Holmes J.H., Lanzi P.L., Stolzmann W., Wilson S.W. (2002). Learning classifier systems: New models, successful applications, *Information Processing Letters* 82, 23–30, Elseviers.
- Kohavi R. (1995). A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection, *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, Vol. 2, Canada.
- Michalski R.S., Ryszard S. (1983). A theory and methodology of inductive learning. *Machine Learning : An Artificial Intelligence Approach*, Volume I, pages 83-134. R. S. Michalski, J. G. Carbonell et T. M. Mitchell editors, Morgan Kaufman Publishers.
- Mikut R., Jäkel J., Gröll L. (2005). Interpretability issues in data-based learning of fuzzy systems. *Fuzzy sets and systems*, 150, P. 179-197.
- Nozaki K., Ishibuchi H., Tanaka H. (1994). Selecting Fuzzy Rules with Forgetting in Fuzzy Classification Systems. *Proc. third IEEE Inter. Conf. on Fuzzy Systems* Vol. 1.
- Pearl. J. (1990). Numerical Uncertainty In Expert Systems. *Readings in Uncertain Reasoning*, Edited by G. Shafer and J. Pearl. Morgan Kaufman publishers. California.
- Polya G (1958). *Les mathématiques et le raisonnement plausible*. Gauthier-Villars, Paris.
- Prentzas J., Hatzilygeroudis I. (2005). Rules-based update methods for a hybrid rule base. *Data & Knowledge Engineering*, 55, 103-128, Elseviers.
- Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning* 1, 81–106.
- Quinlan J. R. (1993). *C4.5 : Programs for Machine Learning*. Morgan Kaufman.
- Rakotomalala R. (2005). Arbres de Décision, *Revue MODULAD*, n°33, pp. 163-187.
- Ruspini E. H. (1991). On the semantics of fuzzy logic. *International Journal of Approximate Reasoning*, Vol. 5, N. 1, pp. 45-88.
- Shafer G. (1996). *The Art of Causal Conjecture*, MIT Press, Cambridge, Massachusetts.
- Shortliff E. H., Buchanan B.G. (1975). A Model of Inexact Reasoning in Medicine, *Readings in Uncertain Reasoning*, p 259-273, 1975, Morgan Kaufman.
- Trillas E. and L. Valverde (1985). On Mode and Implication in Approximate Reasoning. *Approximate Reasoning in Expert Systems*, pp 157-166. M. M. Gupta, A. Kandel, W. Bandler, J. B. Kiszka editors, Elsevier Science Publishers.

Un raisonnement approximatif pour l'apprentissage supervisé de règles

- Venturini G. (1996). Algorithmes génétiques et apprentissage. *Approches Symboliques et numériques de l'apprentissage, Revue d'Intelligence Artificielle*, Vol. 10, 345-387.
- Vernazza G. (1993). Image Classification By Extended Certainty Factors. *Pattern Recognition*, vol. 26, n° 11, p. 1683-1694, Pergamon Press Ltd.
- Yager R.R. (2000). Approximate reasoning and conflict resolution. *International Journal of Approximate Reasoning*, 25, p. 15-42, Elsevier.
- Yager R. R. (1985). Forms of Multi-Criteria Decision Functions and Preference Information Type. *Approximate Reasoning in Expert Systems*, 167-177. Elsevier Science Publishers.
- Zadeh L. A. (1971). Similarity relations and fuzzy orderings. *Information Sciences*, 3, 177-200.
- Zadeh L. A. (1975). The Concept of a Linguistic Variable and its Application to Approximate Reasoning I, II, III. *Information Sciences*, vol 8, vol. 9.
- Zadeh L.A. (1979). A Theory of Approximate Reasoning. *Machine Intelligence*, J. Hayes, D. Michie et L. Mikulich Eds, vol. 9, p. 149-194.
- Zighed D. A. , S. Rabaséda, R. Rakotomalala (1998). FUSINTER : A Method for Discretization of Continuous Attributes. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*. Vol. 6, No. 3, pp 307-326.
- Zighed D. A., Rakotomalala R. (2002). Graphes d'Induction - Apprentissage automatique et Data Mining. *Base de Données et Statistiques*, Dunod.
- Zhou Z.H. (2003). Three perspectives of data mining. *Artificial Intelligence*, 143, p. 139-146.

Summary

This work is done within the framework of the supervised learning method SUCRAGE which is based on automatic generation of classification rules. These rules are then exploited by a classic inference engine: it fires only the rules with which the new observation to classify matches exactly. This engine was extended to an approximate inference which allows to fire rules not too far from the new observation. We propose an original use of approximate reasoning either as a mode of inference but as a means to refine the learning. The approximate reasoning is used to generate new rules with widened premises: imprecision of the observations are then taken into account and problems due to the discretization of continuous attributes are eased. The objective is then to exploit the new base of rules by an exact inference engine, easier to interpret. Experimental tests were carried out with different benchmarks and our method was confronted with a real application in the field of image processing.