Algorithme semi-interactif pour la sélection de dimensions

Lydia Boudjeloud, François Poulet

ESIEA Pôle ECD
38, rue des docteurs Calmette et Guérin
Parc Universitaire de Laval-Changé
53000 Laval
boudjeloud|poulet@esiea-ouest.fr

Résumé. Nous présentons un algorithme génétique semi-interactif de sélection de dimensions dans les grands ensembles de données pour la détection d'individus atypiques (outliers). Les ensembles de données possédant un nombre élevé de dimensions posent de nombreux problèmes aux algorithmes de fouille de données, une solution est d'effectuer un pré-traitement afin de ne retenir que les dimensions "intéressantes". Nous utilisons un algorithme génétique pour le choix du sous-ensemble de dimensions à retenir. Par ailleurs nous souhaitons donner un rôle plus important à l'utilisateur dans le processus de fouille, nous avons donc développé un algorithme génétique semi-interactif où l'évaluation des solutions n'élimine pas complètement la fonction d'évaluation mais la couple avec une évaluation de l'utilisateur. Enfin, l'importante réduction du nombre de dimensions nous permet de visualiser les résultats de l'algorithme de détection d'outlier. Cette visualisation permet à l'expert des données d'étiqueter les éléments atypiques (erreurs ou simplement des individus différents de la masse).

1 Introduction

Nous nous intéressons à la recherche d'outliers (individus atypiques) dans les ensembles de données ayant un grand nombre de dimensions. Pour pouvoir traiter de tels ensembles de données (par exemple les ensembles de données de fouille de texte ou de bio-informatique), la plupart des algorithmes de fouille de données actuels nécessitent un prétraitement permettant de réduire le nombre de dimensions (avec plus ou moins de perte d'information). L'approche la plus intuitive pour appréhender le problème des grandes dimensions est d'énumérer tous les sous-ensembles de dimensions possibles et de rechercher le sous-ensemble qui satisfait la problématique traitée. Cependant, le fait d'énumérer (rechercher) toutes les combinaisons possibles est un problème NP-difficile (Narenda et Fukunaga, 1977). Parmi les solutions proposées pour ce problème, on retrouve la réduction de dimensions (combinaison de dimensions, généralement linéaire) et la sélection de dimensions (on n'utilise qu'un sous-ensemble des dimensions originales). L'avantage de cette dernière solution est que nous ne perdons pas l'information que pourrait apporter la dimension, car elle est considérée individuellement non en combinaison (linéaire) avec d'autres dimensions. Les techniques de sélection de dimensions consistent donc à réduire l'ensemble des

dimensions considérées. L'objectif est de réduire la complexité, augmenter la précision de la prédiction et/ou réduire le temps de traitement des données, en sélectionnant le sous-ensemble de dimensions de taille minimale (Dash et al., 1997). L'étude du problème de sélection de dimensions se justifie facilement par le fait qu'une recherche exacte a un coût exponentiel en temps de calcul et en espace mémoire. En effet, la sélection d'un sous-ensemble de dimensions demanderait l'exploration de tout l'espace de recherche. Pour $|D|$ dimensions, la recherche exhaustive consiste à explorer $2^{|D|}$-1 sous-ensembles possibles. La recherche d'un sous-ensemble de s dimensions parmi D consiste à appliquer le critère d'évaluation $C_{|D|}^{s}$ fois. Si on trouve S' ensembles, on aura donc une complexité de $\sum_{s=0}^{S'} C_{|D|}^{s} = O(|D|^{s})$. Lorsque l'on s'attaque à des problèmes réels, il faut se résoudre à un compromis entre la qualité des solutions obtenues et le temps de calcul utilisé. Au milieu des années 1970 sont apparues des méthodes qui supervisent l'évolution de solutions fournies par des heuristiques. Ces méthodes assurent un compromis entre diversification (quand il est possible de déterminer que la recherche se concentre sur de mauvaises zones de l'espace de recherche) et intensification (on recherche les meilleures solutions dans la région de l'espace de recherche en cours d'analyse). Ces algorithmes ont été appelés métaheuristiques et ont pour objectif de trouver des solutions dont la qualité est au-delà de ce qu'il aurait été possible de réaliser avec une simple heuristique (Jourdan, 2003). Dans cet article, nous proposons un algorithme génétique pour le choix du sous-ensemble de dimensions à retenir.

Par ailleurs nous souhaitons donner un rôle plus important à l'utilisateur dans le processus de recherche et de sélection de l'algorithme génétique, pour cela nous avons choisi d'utiliser un algorithme génétique interactif. Nous présentons donc une nouvelle méthode interactive, proposant elle-même des solutions potentielles à l'utilisateur. Les solutions de l'algorithme génétique se présentent sous forme de sous-ensembles de dimensions. Puisque le nombre de dimensions utilisé est faible, on peut ensuite visualiser les éléments de l'ensemble de données sur ces sous-espaces de dimensions (à l'aide de matrices de scatter-plot (Carr et al., 1987) ou de coordonnées parallèles (Inselberg, 1985)) pour permettre à l'expert d'interpréter les résultats obtenus. Nous présentons donc des visualisations d'un ensemble de données projeté sur quelques sous-ensembles de dimensions à l'utilisateur, ce dernier pourra lui même juger de la pertinence de la visualisation présentée selon ses objectifs (repérer les dimensions les plus pertinentes pour la détection d'outlier). Pour cela, nous avons choisi d'utiliser un algorithme génétique interactif (AGI). D'une manière générale, cet algorithme fonctionne de la façon suivante : une première évaluation automatique se fait à l'aide d'un critère d'évaluation des sous-espaces de dimensions pour la détection d'outlier basé sur les distances, les données sont ensuite visualisées et présentées graphiquement à l'utilisateur pour une seconde évaluation visuelle. Ce dernier choisit celles qui lui semblent les plus pertinentes. Les caractéristiques visuelles des sous-espaces de données sélectionnés sont prises en compte par l'algorithme pour la génération suivante de sous-espaces, qui sont à nouveau présentés à l'utilisateur et ainsi de suite jusqu'à ce que la recombinaison des caractéristiques permette de générer une projection visuelle de données complètement satisfaisante pour l'utilisateur. Un des avantages de cette méthode est de faire collaborer deux méthodes, automatique et visuelle. La première automatique, à l'aide des critères d'évaluation permet d'éliminer les solutions redondantes ou bruitées et la seconde visuelle et interactive permet à l'utilisateur de participer au processus de recherche et d'aborder un

aspect d'évaluation des solutions présentées sous forme visuelle qui représente justement un nouveau domaine de recherche. Un autre avantage est que la méthode s'adresse plus particulièrement au spécialiste des données qui peut utiliser les connaissances du domaine pour l'interprétation visuelle des résultats tout au long du processus de fouille et ainsi apporter un aspect d'aide à la décision. La méthode lui permet d'étiqueter les éléments atypiques, par exemple est-ce que ce sont des erreurs ou simplement des individus différents de la masse. Nous détaillons dans une première partie notre algorithme puis commentons les résultats obtenus sur quelques ensembles de données, nous essayerons ensuite d'interpréter visuellement les résultats obtenus, puis nous terminons par la conclusion et les travaux futurs.

2 Algorithme : Viz-IGA

Face au problème de la prise en compte des préférences de l'utilisateur, des auteurs ont montré comment ce dernier pouvait sélectionner lui même directement les solutions qui le satisfont le plus, sans passer par une fonction automatique parfois impossible à définir (Takagi, 2001), (Hayashida et Takagi, 2000). Une nouvelle catégorie d'algorithmes génétiques est née, connue sous le nom d'Algorithmes Génétiques Interactifs. Ces algorithmes génétiques interactifs (AGIs) permettent ainsi des applications nouvelles comme l'obtention de belles images de synthèse ou de sons polyphoniques. Dans ces applications l'utilisateur note selon ses critères les individus qui représentent des images (ou des sons) et l'AGI fait évoluer les individus selon les préférences de l'utilisateur. Les algorithmes génétiques interactifs (AGIs) sont une extension des AGs dans lesquels a lieu une interaction entre la méthode de recherche et l'utilisateur, ce dernier guidant la méthode vers les solutions ayant les caractéristiques qu'il préfère. L'algorithme génétique standard doit être modifié comme suit : les étapes d'évaluation et de sélection automatique des individus sont remplacées par une présentation des individus à l'utilisateur qui sélectionne en un certain nombre. Cela implique notamment de limiter la population à un petit nombre d'individus. Les principales conditions d'application des AGIs citées précédemment sont particulièrement intéressantes dans notre cas d'évaluation visuelle des sous-espaces de dimensions sélectionnés. En effet, l'interprétation visuelle des résultats obtenus est importante pour la détection d'outlier et valider des sous-espaces qui présentent mieux les solutions attendues est aussi une étape importante dans le processus de recherche de sous-ensembles de dimensions. L'utilisateur peut ainsi intervenir dans le processus de recherche des sous-espaces de dimensions pertinents.

2.1 Initialisation

Nous considérons que l'utilisateur veut avoir des représentations graphiques de données dans un sous-espace de dimensions sur lesquels il peut voir des outliers facilement détectables. Le but recherché est d'aider l'utilisateur à comprendre et interpréter ses données à travers les résultats de l'algorithme. Pour cela, on va lui proposer des représentations graphiques en *k-D* des données avec une des méthodes de visualisation de données (coordonnées parallèles (Inselberg, 1985), matrices de scatter-plot (Carr et al., 1987) ou star plot (Card et al., 1999)).

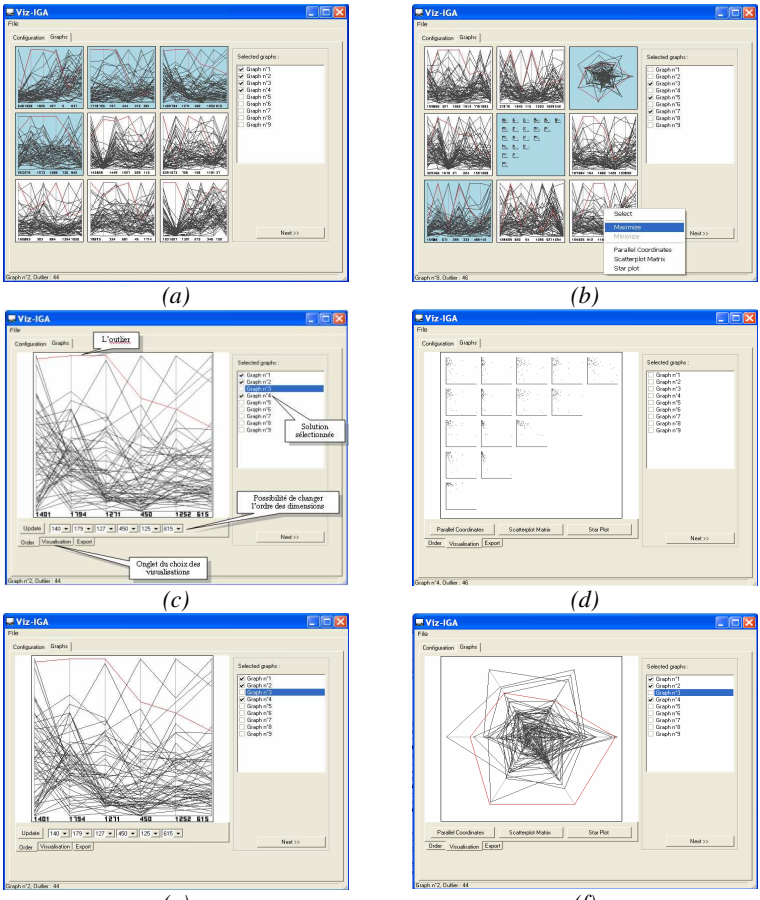


FIG. 1 – *Différentes interfaces de Viz-IGA.*

Le choix de *k*, de l'ensemble de données à traiter et de la méthode de visualisation revient à l'utilisateur. Nous proposons ces méthodes de visualisation car elles permettent à l'utilisateur d'interagir avec l'ensemble des données sous la forme d'une série de projections, l'utilisateur peut voir la pertinence d'une dimension et le comportement des éléments de l'ensemble de données.

2.2 Représentation de l'individu et opérateurs génétiques

Un individu est donc un sous-ensemble de dimensions représenté par une combinaison d'axes (Axe_1, Axe_2, …, Axe_k), ce sous-ensemble étant sélectionné par un algorithme génétique (Boudjeloud et Poulet, 2005). Le codage choisi consiste à fixer un nombre *s* (taille du sous-ensemble de dimensions à sélectionner), ainsi un individu de l'AG (ou de l'AGI)

représente une combinaison possible de *s* dimensions. Ce type de codage permet une meilleure interactivité avec les dimensions. La taille *s* est un paramètre d'entrée de l'algorithme génétique. Pour notre problème nous nous basons sur des petites tailles pour faciliter l'interprétation visuelle des résultats (typiquement inférieur à 10). L'opérateur de croisement échange des sous-groupes de dimensions des individus parents, en respectant la contrainte de non présence de clones (les individus qui ont le même sous-ensemble de dimensions, présentées dans un ordre différent ou pas, sont interdits dans notre algorithme de même que des dimensions identiques dans un même individu). L'opérateur de mutation échange aléatoirement un gène en respectant les mêmes conditions. Nous avons opté pour un point de coupe "optimisé" (Boudjeloud et Poulet, 2005), où l'on détermine dans ce cas le meilleur point avant d'opérer la coupe, ce qui implique une évaluation de chaque individu issu de chaque coupe possible.

2.3 Evaluation visuelle semi-interactive

Pour que l'utilisateur puisse évaluer visuellement les individus de la population et avoir des représentations visuelles de sous-espaces de données sur lesquelles il peut voir des outliers facilement détectables, nous présentons les individus à l'écran (figure 1-a) en faisant une première évaluation automatique à l'aide d'un critère d'évaluation à base de distance (Boudjeloud et Poulet, 2005). Neuf individus choisis aléatoirement dans la population de l'AG sont affichés simultanément pour ne pas surcharger l'interface et faciliter les comparaisons. Pour évaluer la qualité d'un individu, l'expert dispose de la représentation en coordonnées parallèles qu'il peut éventuellement changer en matrices de scatter-plot ou star-plot, selon ses préférences (figure 1-b). Il ne faut pas oublier que nous traitons des ensembles de données de grandes dimensions (de l'ordre de dix à cent milles dimensions, cf. figure 4), notre objectif principal est d'obtenir des visualisations de données pas trop surchargées, pour cela l'AG traite et présente des visualisations de données avec des sous-ensembles de petite taille (de 4 à 10 pour que les visualisations restent claires). L'utilisateur peut aussi agrandir ou faire un zoom sur la visualisation d'un individu en particulier, changer l'ordre des dimensions et changer la méthode de visualisation, (figure 1-c). Les individus peuvent être affichés individuellement avec les matrices de scatter-plots (figure 1-d), les coordonnées parallèles (figure 1-e) ou avec la méthode Star Plot (figure 1-f). Trois possibilités sont offertes à l'utilisateur pour sélectionner une solution potentielle en cliquant directement sur la visualisation de l'individu, en le sélectionnant sur la partie droite de l'interface (figures 1-c, d, e, f)) ou en faisant un clic droit sur une visualisation en particulier, le choix de la sélectionner est alors offert à l'utilisateur (figure 1-b). Une fois les sélections effectuées les solutions apparaissent de couleurs différentes, comme sur l'exemple de la figure (1-a) où les solutions 1, 2, 3 et 4 sont sélectionnées. L'utilisateur doit cliquer sur l'onglet "Next" pour relancer l'AG sur quelques générations avant d'avoir d'autres visualisations.

3 Déroulement de l'algorithme

3.1 Algorithme Viz-IGA

Pour une taille du sous-ensemble de dimensions *s* fixée
1- Génération aléatoire de la population initiale

2- Vérifier les conditions de la population (pas de clones, pas de dimensions identiques dans un même individu)

3- Evaluation :

1- Première évaluation automatique des individus selon le critère à base de distance pour la détection d'outlier

2- Seconde évaluation visuelle par l'utilisateur toutes les 100 générations

4- Sélection :

1- Première sélection par l'utilisateur (soit E' = ensemble des solutions sélectionnées)

2- Seconde sélection par tournoi (on sélectionne aléatoirement deux individus, on ne garde que le meilleur)

5- Croisement mutation (en respectant conditions du point 2)

6- Si stagnation pendant η générations :

1- Muter quelques individus de E'

2- Générer de nouveaux individus

7- Aller à 2 ou fin

3.2 Description des étapes

Notre objectif principal est d'obtenir des visualisations de données significatives et pas trop surchargées, il est préférable de fixer s à de petites tailles (<10 pour que les visualisations restent claires). Nous avons choisi de représenter les données à l'aide des coordonnées parallèles (Inselberg, 1985) et des matrices de scatter-plot (Carr et al., 1987), cependant, ceci n'est pas figé, on pourra remplacer ou introduire d'autres méthodes de visualisation de données, nous avons notamment la méthode Star Plot (Card et al., 1999) dans les exemples présentés. Nous utilisons un algorithme génétique pour la recherche de sous-ensembles de dimensions pertinentes. L'interaction avec l'utilisateur intervient sur certaines générations afin de ne pas faire converger l'AG trop rapidement. Nous faisons intervenir l'utilisateur dans le processus de recherche dans deux étapes : l'évaluation et la sélection.

Population initiale : les individus de l'algorithme génétique représentent des sous-espaces de dimensions constitués à partir des dimensions décrivant l'ensemble des données. Une fois la population de départ prête, nous l'évaluons une première fois à l'aide d'un critère d'évaluation à base de distance décrit dans (Boudjeloud et Poulet, 2005).

Evaluation automatique : vu le nombre important de combinaisons de dimensions possibles, il est nécessaire de faire une sélection de dimensions en utilisant ce critère de validité des sous-espaces avant de les présenter à l'utilisateur. Une fois cette présélection automatique faite, l'utilisateur peut intervenir interactivement selon que la visualisation générée le satisfait ou pas.

Evaluation interactive : une fois la population évaluée et triée selon les différents objectifs, nous présentons à l'utilisateur 9 visualisations. Ces visualisations représentent la projection des données dans des sous-ensembles de dimensions choisis aléatoirement dans la population de l'AG (un individu de l'AG représente un sous-ensemble de dimensions, 9 individus sont choisi aléatoirement et sont présentés visuellement). Notre choix s'est fixé à 9 représentations pour ne pas surcharger l'interface. Les solutions sont représentées par des projections en coordonnées parallèles ou d'autres méthodes de visualisation selon le choix de l'utilisateur (figure 1-b). Nous opérons un croisement et une mutation, puis, toutes les 100

générations, nous proposons à l'utilisateur d'autres visualisations, il peut en sélectionner certaines s'il le souhaite en cliquant dessus, selon qu'elles sont assez significatives pour lui. Pendant ces 100 générations l'AG travaille tout seul sans intervention de l'utilisateur. L'algorithme prend en compte le choix de l'utilisateur pour les prochaines générations dans le processus de recherche de deux manières. Nous avons choisi 100 générations pour que l'AG puisse éliminer les solutions redondantes, les moins intéressantes automatiquement et éviter d'avoir toujours les mêmes solutions qui seront affichées (présentées à l'utilisateur).

Sélection interactive : les solutions sélectionnées par l'utilisateur seront stockées dans une mémoire E' que nous faisons intervenir dans deux étapes de l'algorithme, la première étant la reproduction, la seconde pour remédier à la stagnation de la recherche.

Reproduction : nous faisons intervenir les solutions de E' (sélectionnées par l'utilisateur) dans la reproduction de la façon suivante : chaque nouvel enfant généré aura une partie des gènes d'un parent issu de E' et une partie des gènes d'un parent issu d'une sélection par tournoi où l'on sélectionne aléatoirement et uniformément 2 individus en ne gardant que le meilleur.

Stagnation : dès que la solution stagne (ne s'améliore pas pendant un certain nombre de générations) nous générons de nouvelles solutions à partir des solutions de E' (sélectionnées par l'utilisateur) en les faisant intervenir dans le processus de mutation. Lorsqu'un gène doit être muté, il sera changé par un gène d'un individu de E' (l'emplacement du gène et l'individu de E' sont aléatoires).

L'espace de recherche étant grand, il est important d'avoir une grande capacité d'exploration. Ces mécanismes permettent de maintenir une diversité dans la population en introduisant à certains moments de nouveaux individus. Ils permettent aussi d'éviter une convergence prématurée ou une stagnation des solutions. Nous utilisons ces mécanismes lorsque le meilleur individu est le même durant un certain nombre de mutations η_{mut} et de croisements η_{croi} (ces deux paramètres sont exprimés en pourcentage de la taille de la population). Alors, tous les individus de la population qui ont une valeur d'évaluation en dessous de la moyenne de la population (sous la médiane) sont remplacés par de nouveaux individus générés en respectant les conditions de non-présence de clones et de dimensions identiques dans un même individu. La différence entre notre AGI et les autres AGIs existants est que nous faisons coopérer les deux méthodes visuelle et automatique et que nous faisons intervenir l'utilisateur dans deux processus de l'AG : l'évaluation et la sélection.

4 Résultats et interprétation

Le système a été implémenté sous Windows 2000 dans un environnement très intuitif pour l'utilisateur. Les différentes possibilités de Viz-IGA ont été testées sur des ensembles de données du Kent Ridge Biomedical Dataset Repository (Jinyan et Huiquing, 2002). Les différentes figures (3a, 3b, 3c, 3d) sont créées à partir d'un exemple de détection d'outlier sur l'ensemble de données Breast cancer, Lung cancer, Ovarian et MLL. Notre méthode permet de retrouver des outliers sur des sous-espaces de dimensions identiques à ceux trouvés sur l'ensemble total des données détectés par LOCI (Papadimitriou et al., 2003) un algorithme récent, qui détecte les éléments outliers de l'ensemble des données que nous avons testé (Boudjeloud et Poulet, 2005). De plus, Viz-IGA permet de mettre en évidence les dimensions prépondérantes et souligner la pertinence et l'intérêt de certaines d'entre elles pour la détection d'outlier. Il permet d'isoler et de voir correctement l'élément outlier. Le

nombre de dimensions peut être réduit jusqu'à un facteur 1000 en moyenne sans perte significative d'information, puisque nous arrivons à retrouver le même outlier dans les sous-espaces de dimensions que sur l'ensemble total des données. Il faut entre 2 et 10 minutes pour arriver à la visualisation la plus pertinente pour les problèmes mentionnés précédemment, soit environ de 4 à 20 interventions de l'utilisateur. Au-delà de 15 minutes, l'expérience montre que l'utilisateur se lasse et commence à se fatiguer. Une autre difficulté qui peut lasser l'utilisateur est la stagnation des solutions. Il peut en effet avoir plusieurs fois les mêmes solutions proposées. Il peut par exemple penser qu'il a obtenu la solution finale alors que c'est juste un optimum local. C'est le cas par exemple lors des tests effectués sur l'ensemble de données Colon où à chaque intervention de l'utilisateur on voit bien sur la figure 2 l'amélioration de la courbe et la convergence de l'algorithme en comparaison avec l'AG (Boudjeloud et Poulet, 2005) aux mêmes générations (Viz-IGA converge en moins de générations que l'AG). Cependant, on voit sur la courbe de VIz-IGA des moments de stagnation, par exemple entre les générations 800 et 1000, les générations 1200 et 1400 (il y a deux niveaux de stagnation) et les dernières générations à partir de la génération 1600. Dans ces cas là, il peut arriver que les mêmes solutions soient présentées à l'utilisateur plusieurs fois à la suite.

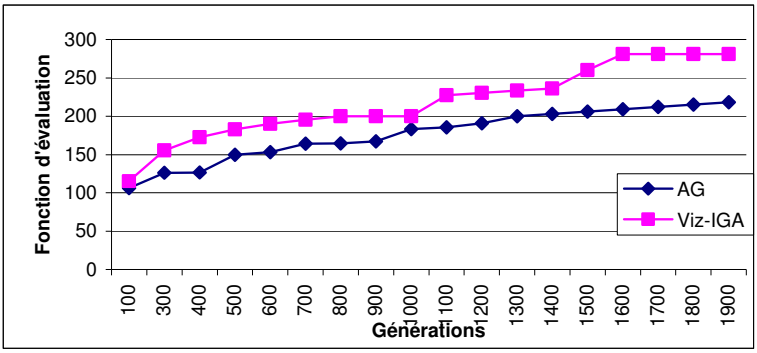
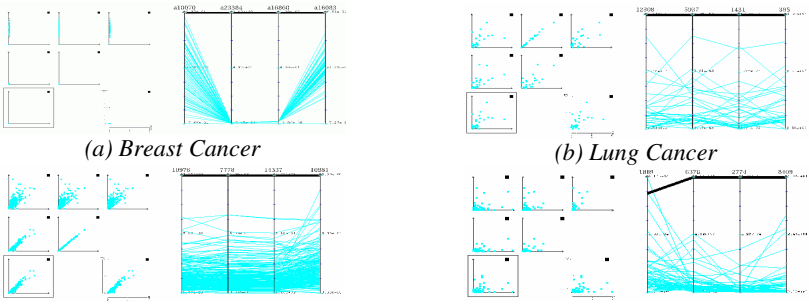


FIG. 2 – *Convergence de l'AG et de Viz-IGA.*

5 Modélisation de l'expertise

La visualisation des résultats obtenus sur quelques ensembles de données (figure 3) montre bien que les points détectés sont éloignés et présentent un comportement atypique par rapport au reste des données, néanmoins nous ne pouvons fournir plus d'explication sur le type des points détectés par notre algorithme (par exemple erreur ou "outlier réel"). En effet, dans le cas de valeurs extrêmes on ne sait pas dire si cette valeur est une valeur possible ou non. Seul l'expert des données peut répondre à cette question. Dans le cas où le point détecté est une erreur on l'élimine de l'ensemble des données et dans le cas contraire on le garde dans les données car il peut représenter à lui seul des informations importantes. Un des moyens de combler cette lacune est de créer un modèle des données permettant de qualifier les éléments détectés comme outliers ou erreurs. Ainsi, étant donné un nouvel élément introduit dans

l'ensemble des données, nous pourrons utiliser le modèle pour prédire son état : outlier, erreur ou donnée normale.



(a) Breast Cancer *(b) Lung Cancer*

(c) Ovarian *(d) MLL*

FIG. 3 – *Visualisation des résultats sur les différents ensembles de données.*

Nous proposons donc de construire un modèle de l'expertise de l'expert. Celui-ci doit tout d'abord étiqueter les éléments qui ont été détectés comme étant outliers (on peut supposer qu'il n'y a que 2 types d'éléments : les erreurs et les "vrais outliers"). A partir de cet ensemble de données étiquetées, on utilise un algorithme de classification supervisée (par exemple un algorithme d'induction d'arbre de décision) pour construire un modèle de l'expertise du spécialiste des données. Les nouveaux éléments outliers seront alors analysés avec le modèle construit et la présence de l'expert n'est plus indispensable pour qualifier ces outliers.

5.1 Construction du modèle

Concernant la partie expertise de la détection d'outlier, nous n'avons pas pu avoir accès à un ensemble de données avec un spécialiste pouvant étiqueter les éléments détectés. Nous avons donc décidé de le faire à partir de l'ensemble de données Colon Tumor (2000 dimensions, 62 éléments) de (Jinyan et al., 2002). Le nouvel ensemble de données est créé en rajoutant des éléments que nous avons étiqueté nous même d'erreur ou d'outlier. Par exemple des éléments qui présentent des valeurs extrêmes sont des erreurs et ceux qui présentent un comportement différent par rapport au reste des données sont des outliers. Nous obtenons donc l'ensemble Colon plus quelques nouveaux éléments. L'ensemble de données Colon a 62 éléments, 5 ont été détectés comme outlier par notre algorithme, nous les étiquetons comme tel, nous créons leurs clones (5), ces clones ont les mêmes valeurs que les originaux sur l'ensemble des dimensions mais sont présentés dans un ordre différant en permutant les valeurs de certaines dimensions. Nous rajoutons au nouvel ensemble de données 10 éléments avec plusieurs valeurs extrêmes qui vont être considérés comme erreurs. Nous obtenons un ensemble de données que nous avons appelé "Colon-Bis" de 2000 dimensions, 77 éléments et trois classes :

Classe 1 : données correctes (57 éléments).
Classe 2 : outliers (10 éléments).
Classe 3 : erreurs (10 éléments).

L'ensemble de données crée n'a qu'un petit nombre d'éléments erreurs et d'éléments outliers, s'ils étaient nombreux, ils ne seraient plus considérés comme tels. Nous avons pris

pour l'ensemble d'apprentissage 67 éléments choisis aléatoirement et les 10 éléments restants pour l'ensemble de test. Reste à choisir un algorithme d'apprentissage qui pourra prédire la classe des nouveaux individus. De nombreux algorithmes d'apprentissage automatique peuvent être utilisés, nous avons choisi comme algorithmes les k-PPV (Cover et Hart, 1967), C4.5 (Quinlan, 1993), CART (Breiman et al., 1984) et LibSVM (Fan et al., 2005). Nous avons effectué des tests dont nous présentons les résultats dans le tableau 1. Ces résultats sont très satisfaisants, nous arrivons à prédire les nouveaux éléments avec un taux de précision de 100% avec le modèle établi par LibSVM. Une fois le modèle établi, le besoin d'étiqueter par le spécialiste des données n'est plus nécessaire.

Algorithmes	Taux de bon classement (%)
LibSVM	100
CART	99
C4.5	98.5
k-PPV	90

TAB. 1 – *Résultats obtenus sur l'ensemble de données Colon Bis.*

6 Conclusion

Nous avons présenté un algorithme génétique semi-interactif pour la sélection de dimensions appliqué à la détection d'outlier. Nous avons introduit une nouvelle représentation de l'individu de l'algorithme génétique. Notre choix s'est fixé sur des petites tailles de sous-ensembles de dimensions pour faciliter l'interprétation visuelle des résultats et souligner la pertinence des dimensions pour chacune des applications, ajoutant ainsi un aspect d'aide à la décision. Cependant, l'utilisateur est libre de fixer la taille des sous-ensembles de dimensions. Il peut aussi intervenir sur le choix de la méthode visuelle utilisée et sur l'ordre des dimensions dans la visualisation proposée. Notre algorithme nous permet la détection d'outliers dans des ensembles de données ayant un grand nombre de dimensions en n'utilisant qu'un sous-ensemble de dimensions de l'ensemble initial. Puisque le nombre de dimensions utilisé est faible, on peut ensuite visualiser ces éléments (à l'aide de matrices de scatter-plot ou de coordonnées parallèles) pour permettre à l'utilisateur de choisir les solutions qui lui paraissent pertinentes. Elles sont alors utilisées pour générer et visualiser d'autres solutions et aussi pour expliquer et valider les résultats obtenus. Il ne faut pas oublier que l'on travaille sur des données de grandes dimensions. Cette étape n'est possible que parce que nous n'utilisons qu'un sous-ensemble restreint de dimensions de l'ensemble de données initial. Cette interprétation des résultats serait absolument impossible en considérant l'ensemble des dimensions comme le montre la figure 4.

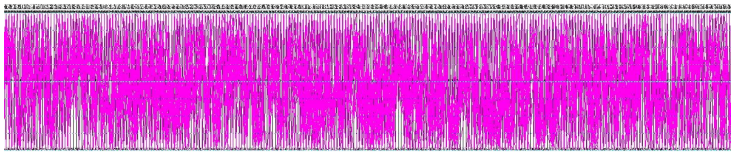


FIG. 4 – *Visualisation de quelques centaines d'attributs de l'ensemble de données Colon Tumor.*

Nous pensons étendre nos applications à des données symboliques et aussi améliorer notre méthode pour optimiser l'ordre des dimensions dans les visualisations en utilisant des critères de validité et étendre la méthode au clustering. Viz-IGA a permis de montrer que l'on peut gagner beaucoup en augmentant l'interaction entre l'expert du domaine et son outil de fouille de données. Pour finir, l'étude des AGI est certainement prometteuse dans d'autres domaines. Ces algorithmes proposent une interaction très simple et efficace pour un utilisateur non informaticien, ce qui peut leur assurer un certain succès dans les applications nécessitant une interaction homme/machine. Nous avons fait coopérer les méthodes automatiques et les méthodes de visualisation de données sur deux aspects, l'interaction avec l'utilisateur dans le processus de recherche en le faisant participer dans la sélection et l'évaluation des solutions proposées par l'AG et dans l'interprétation et la qualification des éléments détectés comme outlier à travers le modèle d'expertise du spécialiste des données. Nous avons proposé une partie expertise, à l'aide des visualisations présentées l'expert des données peut qualifier les outliers détectés (par exemple en deux classes : erreur ou élément significativement différent de la masse). Il ne faut pas oublier que l'on travaille sur des fichiers de grandes tailles. Cette étape n'est possible que parce que nous n'utilisons qu'un sous ensemble restreint de dimensions de l'ensemble de données initial. Cette qualification des outliers serait absolument impossible en considérant l'ensemble des dimensions comme l'illustre très bien l'exemple de la figure 4 où l'on ne peut détecter aucune information à propos des éléments de l'ensemble de données ou des dimensions. Une fois la qualification effectuée, nous utilisons un algorithme d'apprentissage pour créer un modèle de l'expertise du spécialiste des données. Les nouveaux outliers peuvent alors être qualifiés par le modèle construit sans la présence de l'expert des données. Cette étape souligne l'importance de la visualisation de données pour l'interprétation des résultats et son apport pour l'aide à la décision. Les tests effectués pour l'expertise ont été effectués sur un ensemble de données artificiel crée par nos soins car nous n'avons pas pu avoir accès à un ensemble de données et un spécialiste pouvant qualifier les éléments détectés d'erreur ou outlier réel. Nous avons obtenu des résultats satisfaisants par ce premier travail qui nous a permis de faire participer l'utilisateur dans le processus de recherche de sous-ensembles de dimensions pertinents pour détecter, interpréter visuellement et qualifier des éléments outliers.

Références

Boudjeloud, L. et F. Poulet (2005). Détection et interprétation visuelle d'outliers dans les grands ensembles de données. *Numéro spécial de la Revue des Nouvelles Technologie de l'Information : Visualisation et Extraction des Connaissances*, F. Poulet et P. Kuntz Eds, (à paraître).

Breiman, L., J. H. Friedman, R. A. Olshen, and C. J. Stone (1984). *Classification And Regression Trees*. New York: Chapman and Hall.

Card, S., J. MacKinlay and B. Shneiderman (1999). *Readings in information visualization : Using vision to think*. Morgan Kaufman.

Carr, D. B., R. J. Littlefield and W. L. Nicholson (1987). Scatter-plot matrix techniques for large n. *Journal of the American Statistical Association*, 82(398):424-436.

Cover, T. M. and P. E. Hart (1967). Nearest neighbor pattern classification. *In IEEE Transaction on information theory*. 13: 21-27.

Dash, M., H. Liu and J. Yao (1997). Dimensionality reduction for unsupervised data. *In Proceedings of 9th IEEE International conference on tools with artificial (ICTAI).*

Fan, R-E., P-H. Chen and C-J. Lin (2005). Working set selection using the second order information for training svm. *Technical report*, Department of Computer Science, National Taiwan University. Logiciel disponible en ligne (http ://www.csie.ntu.edu.tw/ cjlin/libsvm/) accédé en septembre 2005.

Hayashida, N. and H. Takagi (2000). Visualised IEC : Interactive evolutionary computation with multidimensional data visualization. *In Industrial electronics, control et instrumentation, IECON2000*, 2738-2743.

Inselberg, A. (1985). The plane with parallel coordinates. *In Special Issue on Computational Geometry*, 1:69-97.

Jinyan, L. and L. Huiqing (2002). Kent ridge bio-medical data set repository. http ://sdmc.-lit.org.sg/GEDatasets accédé en septembre 2005.

Jourdan, L. (2003). Métaheuristiques pour l'extraction des connaissances, application à la génomique. *Thèse de doctorat*, Université des Sciences et Technologies de Lille.

Narendra, P.M. and K. Fukunaga (1977). A branch and bound algorithm for feature subset selection. *In IEEE Transactions in Computers*, 26:914-922.

Papadimitriou, S., H.Kitawaga, P. B. Gibbons and C. Faloutsos (2003). LOCI: Fast Outlier Detection Using the Local Correlation Integral. *19th International Conference on Data Engineering, Sponsored by the IEEE Computer Society ICDE'03*.

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. San Mateo: Morgan Kaufmann.

Takagi, H. (2001). Interactive evolutionary computation : Fusion of the capacities of EC Optimization et human evaluation. *In Proceedings of the IEEE*, 89:1275-1296.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. Springer-Verlag, New York.

Summary

We present a semi-interactive genetic algorithm for dimensions selection in high dimensional data sets for outlier detection. Several data mining algorithms have problems with high dimensional data sets, one solution is to carry out a pre-processing in order to retain only "interesting" dimensions. In addition we want to give a more important role to the user in the search process, for that we chose to use an interactive genetic algorithm. In this type of approach the user replaces the genetic algorithm fitness function. Our approach does not completely eliminate this function but we use the user evaluation with it, then we introduce a semi-interactive genetic algorithm. Finally, the important reduction of the dimensions number enables us to display the algorithm results of the outlier detection. This visualization allows the data expert to label the atypical elements, for example if they are errors or simply individuals different from the mass.