Table de matière

Table des figures

Liste des tableaux

Glossaire

ARFF: Attribut Relation File Format

CAH: Classification Ascendante Hiérarchique

CDH : Classification Descendante Hiérarchique

CSV: Comma-Separated Value

HTML: Hypertext Markup Language

IA: Intelligence Artificial

Keel: Knowledge Extraction based en Evolutionary Learning

KNIME: Konstanz Information Miner

Knn: K-Nearest Neighbor

MATLAB: MATrix LABoratory

MLP : Multi Layer Perceptron

R : Révolution

SPSS: Statistical Package for the Social Sciences

SVM: support vecteur machine

Weka: Waikato Environment for Knowledge Analysis

Introduction générale

Depuis plusieurs années, grâce à l'émergence des nouvelles technologies de l'information et de la communication, l'information médicale est devenue de plus en plus disponible et accessible. Le domaine médical dispose aujourd'hui d'une très grande quantité de données permettant ainsi la recherche d'une information médicale quelconque. Cependant, l'exploitation de cette grande quantité de données rend la recherche et la classification des données médicales précises complexes et coûteuses en termes de temps. Cette difficulté a motivé le développement de nouveaux outils de classification des données adaptés, comme « MATLAB » , « TANAGRA » et « WEKA »ce sont trois logiciels de datamining ,matlab payant et les deux autres outil tanagra et weka sont gratuits. S'ils poursuivent le même objectif, permettre aux utilisateurs de définir une succession de traitements sur les données, ils présentent néanmoins des différences. C'est tout à fait normal. Leurs auteurs n'ont pas la même culture informatique, cela se traduit par des choix technologiques différents ; ils n'ont pas la même culture de la fouille de données, ce qui se traduit par un vocabulaire et par un mode de présentation des résultats parfois différents.

Le traitement des bases de données médicales est un enjeu important, alors on est curieux de savoir comment se comportent les différents logiciels dans ce contexte. Ils sont nombreux dans le datamining.

Nous essayons de suivre un peu leur évolution. La capacité à analyser des grands fichiers est un critère que nous regardons souvent pour situer nos propres implémentations. La plupart chargent l'ensemble des données en mémoire centrale. De ce fait, la différenciation en termes de performances repose essentiellement sur la technologie utilisée (compilé ou pseudo-compilé) et la programmation.

Depuis la venue de l'informatique, l'ensemble des données stockées sous forme numérique ne cesse de croitre de plus en plus rapidement partout dans le monde. Les individus mettent de plus en plus les informations qu'ils possèdent à disposition de tous via le web. De nombreux processus industriels sont également de plus en plus contrôlés par l'informatique. Les résultats d'analyses médicales sont aussi de plus en plus régulièrement conservés pour être analysés, et de nombreuses mesures météorologiques, remplissent aussi d'importances bases de données numériques [1].

En ce qui concerne les résultats d'analyses médicales, leur étude peut par exemple aider à mieux détecter les patients à risque pour certaines maladies, permettant ainsi de prévenir plutôt que de guérir [1].

L'intelligence artificielle est un sous-domaine de l'informatique. Ces derniers temps, l'expression « intelligence artificielle » est fréquemment utilisé dans le public car il s'agit d'un domaine en constante évolution notamment grâce aux progrès des technologies informatiques et entre autres grâce aux capacités toujours plus grandes des machines pour effectuer les calculs.

Le Machine learning ou apprentissage automatique est le fait qu'une machine dotée d'une intelligence artificielle puisse être capable de s'autogérer mais surtout d'être autodidacte.

Ainsi par un processus systématique, le Machine learning est capable de conceptualiser, de développer et d'assimiler des tâches de natures différentes. Cela consiste en la mise en place d'algorithmes ayant pour objectif d'obtenir une analyse prédictive à partir de données, dans un but précis.

La classification est une méthode basique de l'apprentissage automatique, l'objectif déborde le cadre strictement exploratoire. C'est la recherche d'une typologie, ou segmentation, c'est-à-dire d'une partition, ou répartition des individus en classes, ou catégories. Ceci est fait en optimisant un critère visant à regrouper les individus dans des classes, chacune le plus homogène possible et, entre elles, les plus distinctes possible.

En général, ce processus est associé à la possibilité de mesurer en un certain sens la qualité et la précision des résultats. Il est possible d'envisager plusieurs techniques d'apprentissage différentes : apprentissage supervisé, non supervisé, semi-supervisé, par transfert et apprentissage par renforcement dans le cadre de ce mémoire, on parlera surtout de l'apprentissage supervisé. En particulier, l'apprentissage supervisé vis la modélisation d'une relation entrées sorties à partir uniquement d'observation de paires entrées sortie issues de cette relation.

L'apprentissage statistique est aujourd'hui confronté à des données dont la nature est de plus en plus complexe (courbes, images) et qui prennent des valeurs dans des espaces dont la dimension est toujours plus élevée. En particulier, ces données peuvent se présenter sous la forme de fonctions, ou courbes, aléatoires. Néanmoins, le périmètre d'utilisation des méthodes statistique traditionnelles se limite souvent au cas où l'espace des observations est de dimension finie. Dans ce nouveau contexte, l'enjeu est alors de proposer des méthodes permettant de traiter ces données fonctionnelles. L'objectif du présent travail de mémoire consiste essentiellement à étudier et appliquer quelques techniques d'apprentissage supervisé. Plus précisément, notre travail pratique se divise en trois parties :

La première partie, intitulée de classifier les données médicales : « Pima », « Appendicite » et « Heart » par la méthode des k-plus proches voisins(Knn), en utilisant trois outils de classification « MATLAB » , « TANAGRA » et « WEKA » est

consacrée à l'étude du taux d'erreur, de la sensibilités, de la spécificité et du temps d'exécution total .Dans la deuxième parties de notre travail on vas faire la même chose en changeant la méthode de classification par support vecteur machine (SVM) et réseaux de neurones (RN) pour réaliser une étude comparative dans la troisième partie entre les résultats des trois outils.

Nous n'allons surtout pas tomber dans le piège des jugements lapidaires du style « qui est bien, qui n'est pas bien, qui est le meilleur… ». D'autres critères sont très importants pour évaluer les logiciels : la portabilité; l'ergonomie; la richesse de la bibliothèque des méthodes; les aspects pratiques qui permettent une prise en main facile; l'accessibilité et la compréhensibilité des résultats; etc.

Ce comparatif s'inscrit avant tout dans le cadre du traitement des grandes bases. Les critères de rapidité et surtout d'occupation mémoire deviennent alors critiques.

En résumé, la question posées s'énonce par : quelles sont les lacunes des outils utilisées « MATLAB », « TANAGRA » et « WEKA » pour la classification supervisé des données médicales ?

L'objectif est de cerner le comportement de quelques logiciels dans ce contexte, lors de l'induction du SVM, Knn, RN. Ils ont pour point commun en général de réaliser l'ensemble des traitements en mémoire centrale.

Le plan de ce projet de fin d'étude s'articule autour de quatre chapitres :

Chapitre 1 : présente les notions fondamentales de la classification des données ainsi que les différentes techniques adoptées de la classification.

Chapitre 2 : expose les différents outils de classification data mining et présente en détails les outils utilisé dans ce projet de fin d'étude.

Chapitre 3 : concerne les méthodes de l'apprentissage supervisé et présente en détails les trois méthodes utilisées dans ce projet de fin d'étude.

Chapitre 4 : présente dans la première partie l'implémentation des trois méthodes de classification supervisée sur les trois bases de données médicales « Pima », « Appendicite », « Heart » par les trois outils de classification « MATLAB », « TANAGRA » et « WEKA ». Dans la deuxième partie une étude comparative entre les mesures de performance des résultats obtenues.

En dernier lieu, une conclusion générale et des perspectives de ce travail seront présentées**.**

Chapitre1

Classification des données

I. Introduction

Depuis un demi-siècle, les chercheurs en intelligence artificielle travaillent à programmer des machines capables d'effectuer des tâches qui requièrent de l'intelligence. Nous citerons l'aide à la décision : l'aide au diagnostic médical ; la reconnaissance de formes : la reconnaissance de la parole ou la vision artificielle ; le contrôle de processus : la conduite de procédées industriels ; la prédiction : la prédiction de consommation électrique ou la prédiction de cours boursiers ; la conduite de robots ; l'exploration de grandes bases de données on peut dire aussi la fouille de données ou data mining en anglais. Le datamining désigne l'ensemble des techniques qui permettent d'analyser et d'interpréter des données volumineuses, contenues dans une ou plusieurs bases de données afin de dégager des tendances. Il peut également être définit comme un domaine interdisciplinaire qui utilise des techniques d'apprentissage automatique, classification des bases de données et de la visualisation pour l'extraction d'informations à partir de bases de données volumineuses cependant dans notre mémoire, nous focalisons notre attention sur les méthodes de classification parmi celle-ci, les méthodes de classification supervisé.

Ce chapitre dresse les notions fondamentales du domaine de classification des données le domaine de notre travail. Il commence par décrire l'apprentissage automatique (machine learning) pour ensuite aborder ses différents types : la classification, clustering et régression. Il présente aussi la classification nous nous intéressons particulièrement à exposer les différentes techniques adoptées de la classification de l'apprentissage supervisé et non supervisé.

Enfin, il expose les limites actuelles de la classification et les difficultés rencontrées en domaine restreint, notamment le domaine médical, le domaine de notre étude.

II. L'intelligence artificielle

L'IA fait exécuter par l'informatique des opérations que l'être humain réalise naturellement : reconnaître un visage, transcrire une parole vocale en parole écrite, jouer aux échecs, trier des messages, détecter des comportements suspects ou des fraudes, etc. La puissance de l'ordinateur (mémoire, rapidité) lui permet de les accomplir avec une performance hors de la portée de l'intelligence humaine.

Chacune de ces opérations consiste en un classement : un visage est classé sous l'identité d'une personne ; un message est classé dans le dossier des spam ; une parole vocale est classée sous un mot écrit ; le prochain coup aux échecs, est classé comme « meilleur coup possible », etc.

Il faut, pour pouvoir classer un être, disposer a priori d'une nomenclature qui définisse des classes. Dans la vie courante chacun de nous utilise plusieurs nomenclatures : lorsque nous rencontrons une personne, nous nous comportons envers elle en fonction de la catégorie psychosociologique dans laquelle nous la rangeons selon son âge, son habillement, son langage, etc. Lorsque nous sommes au volant nous inférons le comportement prévisible des autres conducteurs selon leur apparence et celle de leur voiture, etc. : nous interprétons ainsi des symptômes pour parvenir à un diagnostic [2].

L'apprentissage est un processus mentaux de haut niveau et considéré une des tâches de L'IA. Dans le cadre de ce travail de mémoire nous nous intéressons principalement au problème d'apprentissage pour le domaine médical.

Au début on peut définir l'apprentissage comme une formation professionnelle en alternance, méthodique et complète, dispensée d'une part dans l'entreprise et d'autre part dans un centre de formation pour apprentis (CFA).

Auprès du maître d'apprentissage, l'apprenti découvre le monde du travail et acquiert les bases techniques et pratiques de son futur métier. Au CFA, l'apprenti bénéficie d'enseignements généraux et professionnels complémentaires. L'apprentissage permet au jeune d'acquérir très tôt une expérience professionnelle associée à un diplôme et lui donne ainsi toutes les chances de trouver rapidement un emploi. Parmi les types d'apprentissage on distingue plusieurs types :

a) Naturel,
b) Artificiel,
c) Automatique…

a) Apprentissage naturel

L'apprentissage est une modification durable des potentialités de comportement résultant d'une interaction répétées avec l'environnement [3]. Parmi les modalités de l'apprentissage naturel on peut trouver :

- ✓ Apprentissage par cœur,
- ✓ Apprentissage par instruction,
- ✓ Apprentissage par généralisation,
- ✓ Apprentissage par découverte,
- ✓ Apprentissage plus ou moins supervisé,
- ✓ Apprentissage autonome.

b) Apprentissage artificiel

Un programme possède des capacités d'apprentissage si ses potentialités de comportement sur les données se modifient en fonction de ses performances au fur et à mesure qu'il traite les données.

Un programme possède des capacités d'apprentissage si au cours du traitement d'exemples représentatifs de données il est capable de construire et d'utiliser une représentation de ce traitement en vue de son exploitation [3].

On distingue quelques notions voisines de l'apprentissage artificiel :

- ✓ Extraction d'un concept,
- ✓ Catégorisation, classification,
- ✓ Acquisition de connaissances,
- ✓ Prédiction,
- ✓ Généralisation,
- ✓ Compréhension,
- ✓ Régression,
- ✓ Fouille de Données,
- ✓ Reconnaissance des Formes.

c) Les types d'erreurs en apprentissage

Les sources d'erreur en apprentissage par généralisation sont de trois types :

- ✓ Les données peuvent être bruitées, fausses, mal étiquetées, erreur intrinsèque,
- ✓ L'espace H ou l'on cherche une hypothèse est trop restreint erreur d'approximation, biais inductif,
- ✓ L'algorithme de recherche dans H ne fonctionne pas bien erreur d'estimation, variance **[03].**

III. L'Apprentissage automatique (machine learning)

Le machine learning ou « apprentissage automatique » en français est un concept qui fait de plus en plus parler de lui dans le monde de l'informatique, et qui se rapporte au domaine de l'intelligence artificielle. Encore appelé « apprentissage statistique », ce terme renvoie à un processus de développement, d'analyse et d'implémentation conduisant à la mise en place de procédés systématiques. Pour faire simple, il s'agit d'une sorte de programme permettant à un ordinateur ou à une machine un apprentissage automatisé, de façon à pouvoir réaliser un certain nombre d'opérations très complexes.

L'objectif visé est de rendre la machine ou l'ordinateur capable d'apporter des solutions à des problèmes compliqués, par le traitement d'une quantité astronomique d'informations. Cela offre ainsi une possibilité d'analyser et de mettre en évidence les corrélations qui existent entre deux ou plusieurs situations données, et de prédire leurs différentes implications.

III.1 Les différents types de l'apprentissage automatique

L'Apprentissage automatique se décompose en 2 étapes: une phase d'entraînement (on apprend sur une partie des données) et une phase de vérification (on teste sur la seconde partie de données).

Nous aurons donc 3 phases: la représentation, l'évaluation, l'optimisation. La phase de représentation consiste à trouver le modèle mathématique le plus adapté. Il existe un nombre important de modélisations. L'évaluation mesure l'écart entre le modèle et la réalité des données de tests. Enfin, l'optimisation vise à amenuiser cet écart.

Nous pouvons dénombrer 3 méthodes basiques:

– la **Classification**: modélisation de plusieurs groupes de données dans des classes existantes. Par exemple: la classification des types d'orchidées, la tendance d'un parti politique…

– le **Clustering**: ressemble à la classification mais ce ne sont pas des classes connues.

– la **Régression**: les données sont liées à d'autres données numériques par une corrélation (une droite, une courbe, une tendance).

III.2 Classification

La classification est d'abord employée pour désigner le partage d'un ensemble d'individus en classes de telle sorte que tout individu appartienne à une classe et une seule. Mais le terme classification sert aussi à désigner des systèmes emboités de classes alors on peut dire que c'est une opération statistique qui consiste à regrouper des objets (individus ou variables ou observations) en un nombre limité de groupe (classes, segments), et à classer des individus en fonction de certaines de leurs caractéristiques. Il existe différents types de classification, mais un des plus intuitifs et des plus utilisés est la classification supervisée. L'objectif global de la classification est d'identifier les classes auxquelles appartiennent des objets à partir de traits descriptifs (attributs, caractéristiques).

La classification de données est un problème délicat qui apparaît dans de nombreuses sciences telles que l'analyse du datamining ainsi plusieurs secteur d'application parmi celles si on intéressé dans notre mémoire le domaine médicale.

Domaine commercial: classification répartissant l'ensemble des magasins d'une enseigne en établissements homogènes d'un point de vue de type de clientèle…

Marketing: classification appelée plus fréquemment segmentation. Permettant la recherche des différents profils de clients constituant la clientèle. Après avoir détecté les classes de la clientèle, l'entreprise peut adapter sa stratégie marketing à chaque profil.

Domaine médical: classification permettant de déterminer des groupes de patients susceptibles d'être soumis à des protocoles thérapeutiques, chaque groupe regroupant tous les patients réagissant identiquement **[4].**

IV. Approche Paramétrique versus non-paramétrique

IV.1 Non paramétrique

Les approches dites non paramétriques (classification hiérarchique, méthode des centres mobiles basée sur l'hypothèse : plus deux individus sont proches, plus ils ont de chances de faire partie de la même classe, en plus ce que distingue cette approche est qu'on ne fait aucune hypothèses sur le modèle que suivent les données, C'est le cas des plus proches voisins, donc il suffit de trouver les propriétés de convergence quand le nombre de données est grand.

IV.2 Paramétrique

La seconde grande famille des méthodes de classification, ce sont les approches probabilistes, utilisent une hypothèse sur la distribution des individus à classifier, c'est-à-dire, on suppose que l'on connaît la forme du modèle qui a généré les données. Par exemple, on peut considérer que les individus de chacune des classes suivent une loi normale. Le problème qui se pose, est de savoir déterminer ou estimer les paramètres des lois (moyenne, variance) et à quelle classe les individus ont le plus de chances d'appartenir à partir de l'ensemble d'apprentissage.

V. Les Techniques de classification

Les algorithmes d'apprentissage peuvent se catégoriser selon le mode d'apprentissage qu'ils emploient.

Nous pouvons classer les types d'apprentissages en plusieurs catégories très distinctes : apprentissage supervisé, non supervisé, semi-supervisé, par transfert et apprentissage par renforcement dans le cadre de cette mémoire, on parlera surtout de l'apprentissage supervisé.

V.1 L'apprentissage supervisé

Il s'agit d'une méthode d'apprentissage qui utilise un ensemble de classes d'apprentissage connues afin d'ajuster un modèle statistique qui pourra être utilisé ultérieurement pour le déploiement. Cette méthode s'oppose aux méthodes non supervisées, dans lesquelles les classes ne sont pas connues (pas étiquetées). L'objectif est de déterminer à quel groupe l'individu a le plus de chances d'appartenir.

Un expert doit préalablement étiqueter des exemples. Le processus se passe en deux phases. Lors de la première phase (hors ligne, dite d'apprentissage), il s'agit de déterminer un modèle des données étiquetées. La seconde phase (en ligne, dite de test) consiste à prédire l'étiquette d'une nouvelle donnée, connaissant le modèle préalablement appris. Parfois il est préférable d'associer une donnée non pas à une classe unique, mais une probabilité d'appartenance à chacune des classes prédéterminées (on parle alors d'apprentissage supervisé probabiliste).

Le but d'un algorithme d'apprentissage supervisé est donc de généraliser pour des entrées inconnues ce qu'il a pu « apprendre » grâce aux données déjà traitées par des experts, ceci de façon « raisonnable ».

Il y a plusieurs problèmes de l'apprentissage supervisé car la classification supervisée nécessite beaucoup de moyens humains, quand le nombre de documents augmente. On pourrait donner l'exemple le plus connu : problèmes d'aide ou diagnostic médical, où les superviseurs sont généralement les médecins afin de noter la classe des objets de l'ensemble d'apprentissage à partir des remarques constatées. Ou bien l'exemple d'un tableau où le dernier descripteur (Jouer) représente la classe des exemples.

Parmi les méthodes d'apprentissage supervisé on distingue les méthodes suivantes :

- ✓ Boosting,
- ✓ Machine à vecteurs de support,
- ✓ Mélanges de lois,
- ✓ Réseau de neurones,
- ✓ Méthode des k plus proches voisins,
- ✓ Arbre de décision,
- ✓ Classification naïve bayesienne,
- ✓ Inférence grammaticale,
- ✓ Espace de versions.

V.2 L'apprentissage non-supervisé

Vise à caractériser la distribution des données, et les relations entre les variables, sans discriminer entre les variables observées et les variables à prédire. La tâche revient à la machine de procéder toute seule à la catégorisation des données. Pour ce faire, le système va croiser les informations qui lui sont soumises, de manière à pouvoir rassembler dans une même classe les éléments présentant certaines similitudes. Ainsi, en fonction du but recherché, il reviendra à l'opérateur ou au chercheur de les analyser afin d'en déduire les différentes hypothèses. L'algorithme doit découvrir par lui-même la structure en fonction des données. On distingue deux catégories de classifications non-supervisées : hiérarchiques et non-hiérarchiques.

Par exemple : pour un épidémiologiste qui étudie les victimes du cancer du foie et veut tenter de faire émerger des hypothèses explicatives. L'ordinateur pourrait différencier plusieurs groupes, pour ensuite les associer à divers facteurs explicatifs.

L'objectif de l'apprentissage non supervisé regroupé (classer) les individus qui se ressemblent le plus qui ont des caractéristiques semblables.

Ce regroupement peut avoir des buts divers : tenter de séparer des individus appartenant à des sous-populations distinctes, décrire les données en procédant à une réduction du nombre d'individus pour communiquer, simplifier, exposer les résultats.

Différentes tâches sont associées à l'apprentissage non supervisé par exemple le clustering (segmentation, regroupement) : construire des classes automatiquement en fonction des exemples disponibles. Les techniques de clustering cherchent à décomposer un ensemble d'individus en plusieurs sous ensembles les plus homogènes possibles, règles d'association consiste à analyser les relations entre les variables ou détecter des associations et la réduction de dimensions **[5].**

L'apprentissage non supervisé peut aussi être utilisé en conjonction avec une inférence bayésienne pour produire des probabilités conditionnelles pour chaque variable aléatoire étant donné les autres.

Une autre forme d'apprentissage non supervisé est le partitionnement de données qui n'est pas toujours probabiliste.

V.3 L'apprentissage semi-supervisé

C'est une classe de techniques d'apprentissage automatique qui utilise un ensemble de données étiquetées et non-étiquetés. Il se situe ainsi entre l'apprentissage supervisé qui n'utilise que des données étiquetées et l'apprentissage non-supervisé qui n'utilise que des données non-étiquetées

Il vise à faire apparaître la distribution sous-jacente des « exemples » dans leur espace de description. Il est mis en œuvre quand des données (ou « étiquettes ») manquent. Le modèle doit utiliser des exemples non-étiquetés pouvant néanmoins renseigner .Il est souvent associé au concept d'apprentissage transductif. Il s'effectue sur les données de la base d'apprentissage dans le but de faire des prédictions sur les observations de la base de test, et uniquement celles ci. Le but n'est donc pas de déterminer la fonction qui minimise l'erreur en généralisation, mais celle qui minimise l'erreur moyenne sur la base de test.

Un exemple d'apprentissage semi-supervisé est le co-apprentissage, dans lequel deux classifieurs apprennent un ensemble de données, mais en utilisant chacun un ensemble

de caractéristiques différentes, idéalement indépendantes. Si les données sont des individus à classer en hommes et femmes, l'un pourra utiliser la taille et l'autre la pilosité par exemple.

Nous distinguons trois approches en classification semi-supervisée. D'abord les méthodes générales qui peuvent être appliquées à n'importe quelle méthode de classification supervisée. Ensuite, les méthodes spécifiques aux modèles prédictifs. Enfin les méthodes spécifiques aux modèles génératifs.

Il y a quatre algorithmes de la classification semi-supervisée on peut citer:

1. Co-traing.
2. Transductive SVM's
3. Semi-supervised.
4. Graph based algorithms

L'apprentissage semi supervisé vise les problèmes avec relativement peu de données étiquetées et une grande quantité de données non étiquetées. Ce type de situation peut se produire quand l'étiquetage des données est coûteux, comme dans le cas de la classification de pages internet.

V.4 L'apprentissage par transfert

L'apprentissage par transfert peut être vu comme la capacité d'un système à reconnaître et appliquer des connaissances et des compétences, apprises à partir de tâches antérieures, sur de nouvelles tâches ou domaines partageant des similitudes.

V.5 L'apprentissage par renforcement

L'apprentissage par renforcement est né de la rencontre entre la psychologie expérimentale et les neurosciences computationnelles. Il tient en quelques concepts clés simples basés sur le fait que l'agent intelligent observe les effets de ses actions et déduit de ses observations la qualité de ses actions.

L'apprentissage par renforcement se manifeste tous les jours dans notre quotidien, que ce soit lorsque nous marchons, lorsque nous apprenons un nouveau langage de programmation ou lorsque nous pratiquons un sport.

L'algorithme apprend un comportement étant donné une observation. L'action de l'algorithme sur l'environnement produit une valeur de retour qui guide l'algorithme d'apprentissage [6].

VI. La différence entre l'apprentissage supervisé et non-supervisé

La différence entre ces deux principes de fonctionnement réside dans le fait que l'apprentissage supervisé peut être influencé par des à priori au moment de l'étiquetage des données. Ce n'est pas le cas de l'apprentissage non-supervisé, qui se révèle ainsi beaucoup plus fiable dans la mesure où les réponses obtenues vont plus loin que la compréhension humaine des faits. Par ailleurs, il faut noter que la machine learning peut également faire intervenir un mode de fonctionnement mixte qui utilise les deux types d'apprentissage pour arriver à des résultats plus précis.

VII. Domaines d'application

La classification comme dit préalablement joue un rôle dans presque toutes les sciences et techniques qui font appel à la statistique multidimensionnelle. Il existe toute une pléthore de domaines dans lesquels la classification intervient, à savoir la finance, la sécurité, la médecine, l'industrie automobile et la technologie dans tout son ensemble. Pour citer quelques cas pratiques d'utilisation de l'apprentissage automatique ou statistique, on peut énumérer entre autres l'intégration de Watson au centre de cancérologie Mémorial Sloan Kettering de New York.

On peut citer les voitures à conduite autonome, dont le modèle le plus représentatif est celui mis au point par Google. La reconnaissance (vocale, faciale, d'objets ou de caractères) dans la sécurité ou l'information.

Il s'agit d'une intelligence artificielle qui s'appuie sur une énorme base de données pour établir des diagnostics médicaux plus poussés, en moins de temps et à moins de frais qu'avec des spécialistes humains, même les plus qualifiés et la détection de fraudes en finance et la classification des séquences d'ADN en médecine sont autant de domaines d'intervention du machine learning.

VIII. Conclusion

Ce qu'il faut retenir de tout ce qui précède, c'est tout simplement que l'intelligence artificielle semble promise à un bel avenir, au vu de la portée technologique de la machine learning. La machine Learning offre un certain nombre de méthodes statistiques avancées pour traiter des tâches de régression et de classification avec plusieurs variables dépendantes et indépendantes. Parmi ces méthodes, citons la méthode des Séparateurs à Vaste Marge (SVM - Support Vector Machines) pour des problèmes de régression et de classification, la méthode des Réseaux Bayésiens Naïfs pour des problèmes de classification, et les méthodes des K Plus Proches Voisins pour des problèmes de régression et de classification. Vous trouverez une présentation de ces techniques et d'autres techniques de classification dans les chapitres suivants.

Chapitre2

Environnements et outils de classification

I. Introduction

Depuis plusieurs années, data mining a été un vaste domaine de recherche pour de nombreux chercheurs en raison de la quantité énorme de données et d'informations disponibles dans les bases de données. Avec une telle quantité de données, il existe un besoin de techniques et d'outils puissants qui peuvent gérer les données de meilleure façon et extraire la connaissance pertinente.

Dans ce chapitre nous présentons quelques outils de data mining aux différentes catégories, et nous exposons aussi en détails les outils que nous avons utilisé dans notre étude comparative.

II. Définition

Les outils de fouille de données sont des programmes spécialisés dans l'analyse et extraction de connaissance à partir de grande quantités des données informatisées, pour objectif aide l'analyste en exploration de données : extraction d'un savoir ou d'une connaissance par des méthodes automatique ou semi automatique.

L'exploration des donnés (fouille des données) se propose d'utiliser un ensemble d'algorithmes au différents disciplines scientifiques telle que les statistiques, l'intelligence artificiel ou l'informatique pour construire des modèles à partir des données, afin de trouver des structures intéressantes ou des motifs selon des critères fixés au préalable, et d'en extraire un maximum de connaissances.

On peut classer les outils en deux catégories très distinctes :

A. **Les logiciels commerciaux** : proposent une interface graphique conviviale, ils sont destinés à la mise en œuvre de traitements sur des données en vue du déploiement des résultats. Les méthodes disponibles sont souvent peu référencées, il est de toute manière impossible d'accéder à l'implémentation

B. **Les outils libres** : sont constitués d'un assemblage de bibliothèques de programmes. Un chercheur peut facilement accéder au code source pour vérifier les implémentations, ajouter ses propres variantes, et mener de nouvelles expérimentations comparatives. Ces plates-formes ne sont guère accessibles à des utilisateurs non informaticiens.

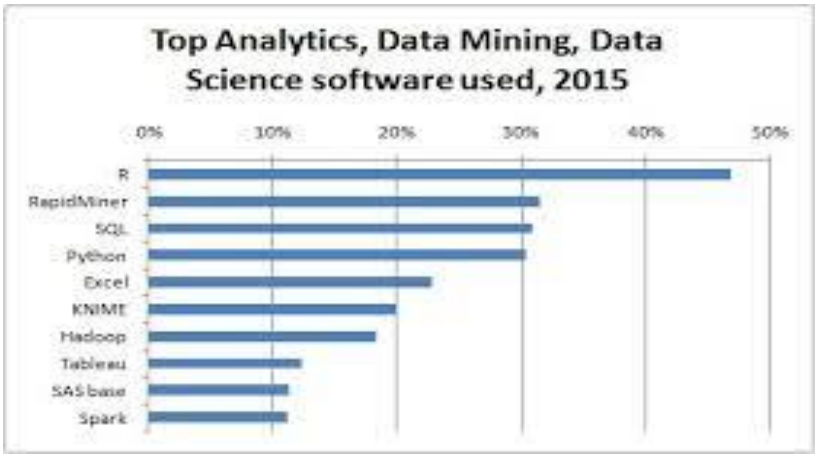

Figure 2.1: top outils de Fouille de Données en 2015

III. Les logiciels commerciaux :

Les logiciels commerciaux sont édités par des sociétés bien connues sur le marché, parmi ces logiciels on trouve :

III.1 SAS :

SAS (Société par Actions Simplifiée), est une forme de société très récente, outils d'extraction de données SAS est très répandu dans le monde professionnel, particulièrement dans le monde de la santé (laboratoires pharmaceutiques, hôpitaux) SAS est particulièrement « robuste » et peut traiter des jeux de données très volumineux (plusieurs millions d'individus).

Figure 2.2 logo de logiciel SAS

Parmi les caractéristiques de SAS on trouve :

- SAS est disponible sous Microsoft Windows, Linux, Unix
- Il est écrit en C.
- Le langage SAS est basé sur trois parties :

A) les étapes DATA. Elles permettent de répondre aux besoins correspondant à la création et à la gestion de tables de données ;

B) une étape crée ou modifie une table d'observations. En colonne se trouvent le plus souvent un identifiant et des indicateurs et en lignes, les sujets observés ;

C) les procédures ont chacune leur objectif : trier une table, compter les lignes ou synthétiser les indicateurs,

III.2. SPSS :

SPSS (Statistical Package for the Social Sciences) est un logiciel utilisé pour l'statistique. Il permet de manipuler et d'analyser des fichiers de données, implémentant la plupart des algorithmes d'intelligence artificielle comme les arbres de décision et les réseaux de neurones, etc.

Il est écrit en java, Différentes versions de SPSS existent pour Windows, Mac OSX et Unix.

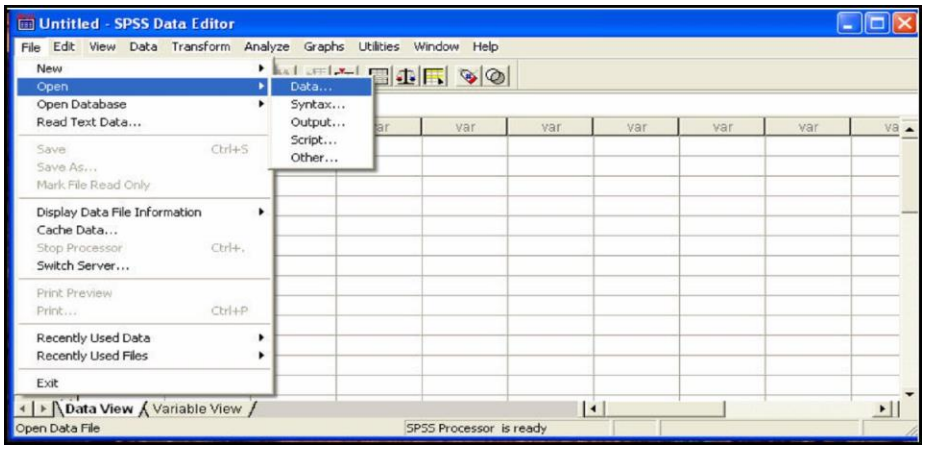

Figure 2.3 la fenêtre principale de SPSS

Il a plusieurs caractéristiques qui sont **:**

- Il est fréquemment utilisé dans les sciences sociales.
- SPSS est accessible via les menus déroulants ou peuvent être programmées avec un langage en ligne de commande appelé 4GL (licence propriétaire).
- SPSS dispose de quatre fenêtres : éditeur de données; Afficheur de sortie; Éditeur de syntaxe; Fenêtre de script.

- De nombreuses tâches peuvent être effectuées avec les menus et les boîtes de dialogue, mais certaines fonctionnalités très puissantes ne sont disponibles qu'avec la syntaxe des commandes.
- Graphs command est utilisé exclusivement dans SPSS pour créer des graphiques. SPSS génère généralement des graphiques couramment utilisés dans les domaines des sciences sociales, tels que les histogrammes, les diagrammes de dispersion et la ligne de régression, etc.
- Il est utilisé par des chercheurs appartenant à déférents domaine scientifiques (économie, science de la santé, éducation nationale, etc.). En plus de l'analyse statistique, la gestion des données (sélection de cas, reformatage de fichier, création de données dérivées) et la documentation des données sont deux autres caractéristiques de ce logiciel.

IV. Les outils libres

Parmi les logiciels libres on trouve :

IV.1. RapidMiner [7]

RapidMiner est également l'un des outils open source dans l'exploration de données développé par Ingo Mierswa et RalfKlinkenberg. Rapid Miner également connu sous le nom de YALE (encore un autre outil d'apprentissage) basé sur XML.

Figure 2.4 logo d'outil Rapidminer

Caractéristiques :

- Rapid Miner possède une collection de fonctionnalités, il possède une bonne connectivité.
- Rapid Miner se compose de nombreux algorithmes d'apprentissage de WEKA.
- Ensemble compact et complet.
- Il lit et écrit simplement des fichiers Excel et des bases de données diverses.

Avantage:[8]

• Possède l'installation complète pour l'évaluation du modèle en utilisant des ensembles de validation croisée et de validation indépendante.

• Plus de 1 500 procédés pour l'intégration des données, la transformation, l'analyse et la modélisation des données, ainsi que la modélisation et la visualisation - aucune autre solution sur le marché n'offre plus de procédures et donc plus de possibilités de définir les processus d'analyse optimale

• RapidMiner propose de nombreuses procédures, en particulier dans le domaine de la sélection des attributs et de la détection des valeurs aberrantes, qu'aucune autre solution ne propose

IV.2. ORANGE

Orange est un logiciel libre d'exploration de données (data mining). Il propose des fonctionnalités de modélisation à travers une interface visuelle, une grande variété de modalités de visualisation et des affichages variés dynamiques. Développé en Python, il existe des versions Windows, Mac et Linux.

Le logiciel libre orange a plusieurs caractéristique comme :

- Il contient un ensemble de modules pour le prétraitement des données, la caractérisation des fonctionnalités et le filtrage, la modélisation, le modèle évaluation et techniques d'évaluation.
- Il est également très utile pour les processus analytiques
- Il est très efficace pour le complément de bioinformatique

IV.3. R

Le logiciel R est un logiciel de statistique créé par Ihaka & Gentleman **[9].** Il est à la fois un langage informatique et un environnement de travail: les commandes sont exécutées grâce à des instructions codées dans un langage relativement simple, les résultats sont affichés sous forme de texte et les graphiques sont visualisés directement dans une fenêtre qui leur est propre.

R est un logiciel gratuit et à code source ouvert (open source). Il fonctionne sous UNIX (et Linux), Windows et Macintosh. C'est donc un logiciel multi plates-formes. C'est aussi un outil très puissant et très complet, particulièrement bien adapté

pour la mise en œuvre informatique de méthodes statistiques. Il est plus difficile d'accès que certains autres logiciels du marché (comme SPSS par exemple), car il n'est pas conçu pour être utilisé à l'aide de «clics» de souris dans des menus.

R a double avantages majeurs:

- L'approche est pédagogique puisqu'il faut maitriser les méthodes statistiques pour parvenir à les mettre en œuvre.
- L'outil est très efficace lorsqu'on domine le langage R puisque l'on devient alors capable de créer ses propres outils, ce qui permet ainsi d'opérer des analyses très sophistiquées sur les données

Le logiciel R est particulièrement performant pour la manipulation de données, le calcul et l'affichage de graphiques. Il se caractérise par les points suivants :

- Un système de documentation intégré très bien conçu des procédures efficaces de traitement des données et des capacités de stockage de ces données
- Une suite d'opérateurs pour des calculs sur des tableaux et en particulier sur des matrices
- Une vaste et cohérente collection de procédures statistiques pour l'analyse de données des capacités graphiques évoluées
- Un langage de programmation simple et efficace intégrant les conditions, les boucles, la récursivité, et des possibilités d'entrée-sortie **[10].**

IV.4 KNIME

KNIME (prononcer NAÏM), acronyme de Konstanz Information Miner, est un logiciel libre édité par un laboratoire de l'université de Constance dénommé Nycomed Chair for Bioinformatics and Information Mining. Il intègre tous les modules d'analyse de Weka et permet de créer des scripts en langage R.

KNIME est une plate-forme d'exploration de données modulaire qui permet à l'utilisateur de créer visuellement des flux de données, exécuter de manière sélective certaines ou toutes les étapes de l'analyse, et ensuite enquêter sur les résultats grâce à des vues interactives sur les données et des modèles.

Figure 2.5 logo d'outil KNIME

IV.5 PYTHON

Python est un langage de programmation qui peut s'utiliser dans de nombreux contextes et s'adapter à tout type d'utilisation grâce à des bibliothèques spécialisées. Il est un logiciel gratuit et à code source ouvert (open source). Il fonctionne sur la plupart des plates-formes informatiques sous Linux, Windows et Macintosh. C'est donc un logiciel multi plates-formes.

IV.6 KEEL [11]

Keel (Knowledge Extraction based en Evolutionary Learning) software est un logiciel libre programmé sous java permet à l'utilisateur d'évaluer les techniques évolutives et d'autres problèmes de fouille de données classification, régression.

Keel est un outil pour évaluer les algorithmes évolutifs pour les problèmes d'exploration de données.

Cet outil a plusieurs avantages qui sont:

- Réduis le travail du programmeur et gagner le temps.
- Une vaste bibliothèque de méthodes et bases de données prêt à tester et facile à utiliser.
- On peut l'utiliser sur n'importe quelle machine avec java

Par contre on trouve que L'efficacité de Keel est limitée par le nombre d'algorithmes qu'elle prend en charge par rapport à d'autres outils.

Figure 2.6 Interface de Keel software2.0

Voici la structure de Keel figure 3.6 contient une interface bien organisée, facile à manipuler, il contient des méthodes récentes pour divers problèmes de datamining.

V. Description détaillé sur les outils de notre étude comparative

V.1 Matlab

Le nom **MATLAB** [12] est en fait l'abréviation de MATrix LABoratory c'est un logiciel interactif qui fournit à l'utilisateur un environnement lui permettant de réaliser un grand nombre de calculs, en particulier ceux où les matrices interviennent. L'élément de base est un tableau qui ne demande pas de dimensionnement préalable. Ceci vous permet de résoudre de nombreux problèmes techniques de calcul numérique bien plus rapidement que si vous deviez écrire un programme dans un langage tel que C ou Fortran. MATLAB s'opère depuis une session de commandes en ligne, celles-ci peuvent être exécutées une à une ou bien être sauvées dans un script afin de l'exécuter comme un programme. Il existe un grand nombre de fonctions et commandes MATLAB qui permet de réaliser :

- Des opérations vectorielles ou matricielles
- Des calculs statistiques
- De la visualisation de données et images

Le logiciel MATLAB s'utilise généralement avec les fenêtres suivantes:

- ✓ données en mémoire,
- ✓ éditeur de script,
- ✓ lignes de commandes,
- ✓ historique des commandes,
- ✓ adresse de navigation

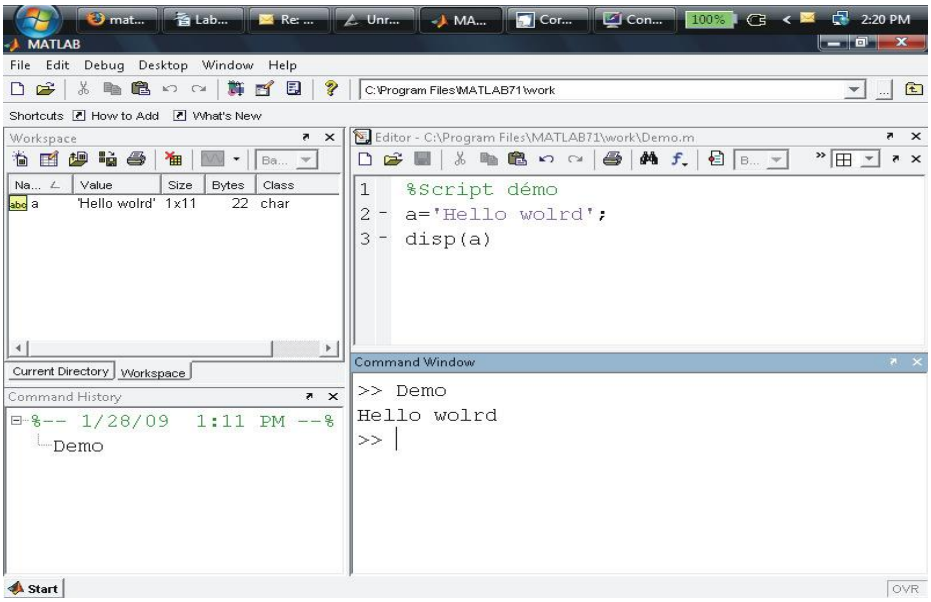

Figure 2.7 la fenêtre principale de matlab

Caractéristique :

- ➢ Matlab peut s'utiliser seul ou bien avec des toolbox « boîte à outils ».
- ➢ MATLAB, environnement dédié à la manipulation des matrices, permet d'accéder à des fonctionnalités de haut niveau de segmentation, de classification, de prédiction et d'association.
- ➢ Les outils MATLAB et ses Toolbox concernant le Data Mining: outils d'analyses factorielles outils de classification, arbres de décisions, réseaux de neurones, régressions linéaires et nonLinéaires au sens large.

Figure 2.8 réseau de neurone par matlab

V.2 Weka

Weka (Waikato Environment for KnowledgeAnalysis) est un ensemble d'outils permettant de manipuler, d'analyser des fichiers de données et pour implémenter différents algorithmes d'apprentissage artificielle(les arbres de décision, les réseaux de neurones, etc.).Il est écrit en java, développée à l'université de Waikatoen Nouvelle-Zélande 1992 [13].

Parmi tous les outils d'exploration de données disponibles, Weka est le plus couramment utilisé en raison de ses performances et de son support rapide pour l'algorithme de classification et de clustering.

Weka fournit à la fois une GUI et une CLI pour effectuer l'exploration de données et fait un bon travail pour fournir un support pour toutes les tâches d'exploration de données [Sol16]. Weka prend en charge une variété de formats de données comme CSV (Comma-separated Value), ARFF **(**Attribut Relation File Format) et Binaire.

Weka se concentre plus sur la représentation textuelle des données plutôt que sur la visualisation, même si elle fournit un support pour afficher une certaine visualisation, mais celles-ci sont très génériques. De plus, Weka ne fournit pas de

représentation visuelle des résultats du traitement de manière efficace et compréhensive comme Rapid Miner.

Weka effectue une précision lorsque la taille de l'ensemble de données n'est pas grande. Si la taille est grande, Weka connaît des problèmes de performance. Weka fournit un support pour filtrer des données ou des attributs **[13].**

Weka prend en charge les trois interfaces graphiques suivantes **[14]** :

Figure 2.9: Interface Graphique de Weka

1. **L'Explorateur (Explorer) :**

Cette interface est la plus couramment utilisée dans Weka pour implémenter des algorithmes d'exploration de données. Il supporte l'analyse des données exploratoires pour effectuer le prétraitement, la sélection des attributs, l'apprentissage et la visualisation. Cette interface se compose d'onglets différents pour accéder à différents composants pour effectuer l'exploration de données, ce qui peut être vu dans la Figure 2.10 Les différents onglets sont:

A. **Prétraitement (Preprocessing) :**

En utilisant cet onglet, nous pouvons charger des fichiers de données d'entrée et effectuer un prétraitement sur ces données à l'aide de filtres.

B. **classification (classify) :**

Cet onglet est utilisé pour implémenter différents algorithmes de classification et de régression. Nous pouvons le faire en sélectionnant un classificateur particulier à partir de cet onglet. Par exemple, l'algorithme SVM ou MLP peut être implémenté en utilisant cet onglet.

C. Associé (Associate) :

Cet onglet permet de trouver toutes les règles d'association entre les différents attributs des données et qui peuvent être utilisés pour l'extraction ultérieure. Par exemple, l'extraction des règles d'association.

D. **Cluster :**

En utilisant cet onglet, nous pouvons sélectionner un algorithme de clustering particulier à mettre en œuvre pour notre ensemble de données. Les algorithmes de cluster comme K-means peuvent être implémentés à l'aide de cet onglet.

E. Sélectionnez les attributs :

Cet onglet permet de sélectionner des attributs particuliers à partir du jeu de données utile pour la mise en œuvre de l'algorithme.

F. Visualiser :

Cet onglet permet de visualiser les données chaque fois que cela est disponible ou pris en charge par un algorithme particulier sous forme de matrice de diagramme de dispersion

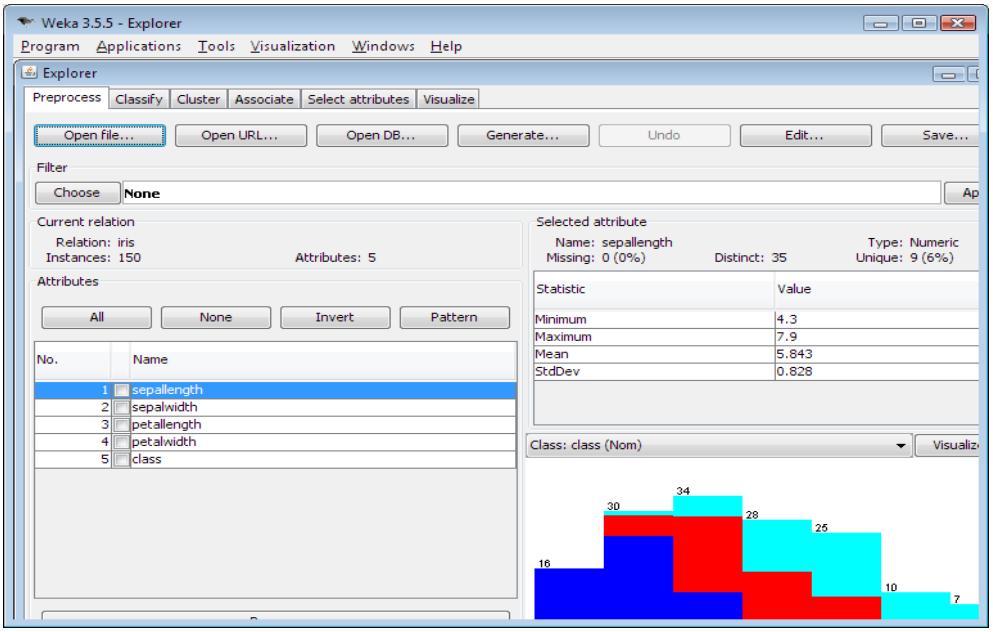

Figure 2.10 différents onglets de l'exploration de données

2. **L'expérimentateur (Experimenter)**

Cette interface utilisateur fournit un environnement expérimental pour tester et évaluer les algorithmes d'apprentissage machine.

3. **Le flux de connaissances (KnowledgeFlow)**

Le flux de connaissances est essentiellement une interface basée sur les composants, similaire à celle de l'explorateur. Cette interface est utilisée pour les nouvelles évaluations de processus

Weka se compose principalement :

- De classes Java permettant de charger et de manipuler les données.
- De classes pour les principaux algorithmes de classification supervisée ou non supervisée.
- D'outils de sélection d'attributs, de statistiques sur ces attributs.
- De classes permettant de visualiser les résultats

Nous pouvons l'utiliser weka à trois niveaux **:**

- Via l'interface graphique, pour charger un fichier de données, lui appliquer un algorithme, vérifier son efficacité.
- Invoquer un algorithme via la ligne de commande.
- Utiliser les classes définies dans ses propres programmes pour créer d'autres méthodes, implémenter d'autres algorithmes, comparer ou combiner plusieurs méthodes.

Les principaux points forts de Weka sont :

- Il permet de sélectionner la méthode la mieux adaptée ou la plus efficace.
- Il est également adapté au développement de nouveaux systèmes d'apprentissage [15].
- Il est portable car il est entièrement implémenté en Java et donc fonctionne sur quasiment toutes les plateformes modernes, et en particulier sur quasiment tous les systèmes d'exploitation actuels.
- Il est facile à utiliser par un novice en raison de l'interface graphique qu'il contient.

- Il contient une collection complète de préprocesseurs de données et de techniques de modélisation.

V.3 TANAGRA :

Tanagra est un logiciel gratuit et open source de Data Mining écrit par Rico Rakotomal a du laboratoire ERIC (Université Lyon 2 Lumière). Il implémente une série de méthodes de fouilles de données issues du domaine de la statistique exploratoire, de l'analyse de données, de l'apprentissage automatique et des bases de données. Tanagra offre aux chercheurs et aux étudiants une plate-forme de Data Mining facile d'accès, respectant les standards des logiciels du domaine, notamment en matière d'interface et de mode de fonctionnement, et permettant de mener des études sur des données réelles et/ou artificielles.

A. **Objectifs de TANAGRA**

Son premier objectif est d'offrir aux étudiants et aux experts d'autres domaines (médecine, bioinformatique, marketing, etc.) une plate-forme facile d'accès, respectant les standards des logiciels actuels, notamment en matière d'interface et de mode de fonctionnement, il doit être possible d'utiliser le logiciel pour mener des études sur des données réelles. Le second objectif est de proposer aux chercheurs une architecture leur facilitant l'implémentation des techniques qu'ils veulent étudier, de comparer les performances de ces algorithmes. TANAGRA se comporte alors plus comme une plateforme d'expérimentation qui leur permettrait d'aller à l'essentiel en leur épargnant toute la partie ingrate de la programmation de ce type d'outil, notamment la gestion des données. Point très important à nos yeux, la disponibilité du code source est un gage de crédibilité scientifique, elle assure la reproductibilité des expérimentations publiées par d'autres chercheurs et, surtout, elle permet la comparaison et la vérification des implémentations. **[16]**

B. Fonctionnalité de TANAGRA :

a. Organisation des traitements

La fenêtre principale du logiciel est subdivisée en trois grandes zones sont représente comme suit figure 2.11 :

A : Palette de composants : une série des ensembles composants, qui sont regroupés en catégories (data visualisation, svp Learning...)

B : Chaine de traitement : sur la gauche, le diagramme de traitements, représentant l'analyse courante

C : Résultat : dans le cadre de droite, l'affichage des résultats consécutifs à l'exécution de l'opérateur sélectionné. Il est bien sûr possible de sauvegarder, soit sous un format binaire, soit sous la forme d'un fichier texte

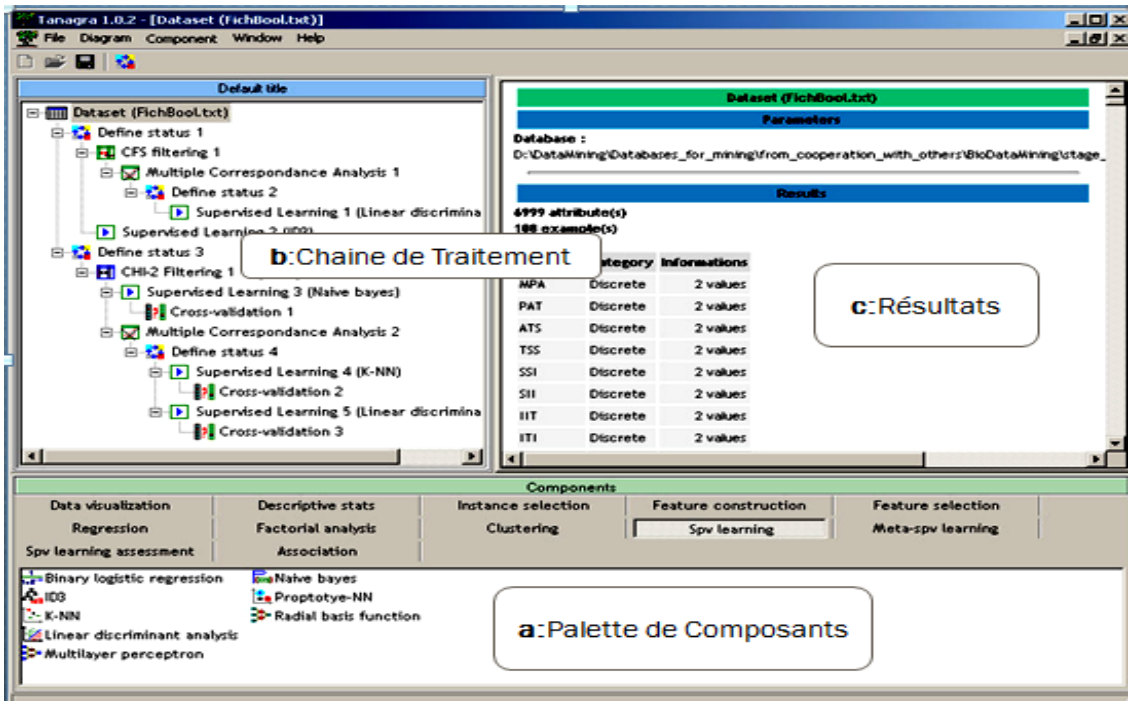

Figure 2.11 : La fenêtre principale de TANAGRA

b. Accès aux données :

L'accès et l'analyse du fichier de données sont réalisés lors de la définition d'une nouvelle chaîne de traitements. Les données sont chargées en mémoire après avoir été codées en interne. La rapidité est un critère clé lors de cette importation.

A l'heure actuelle, seuls les fichiers au format texte, séparateur "tabulation" sont importés. Le nom des variables est récupéré sur la première ligne, leur type (discret ou continu) est déterminé à partir de la ligne suivante.

c. Algorithmes de traitement (palette de composants) :

- Les algorithmes et les méthodes de fouille de données (les icônes, les composants) sont regroupés en catégorie dans la fenêtre du bas du logiciel.
- Chaque composant représente un algorithme de traitement de données.

- On distingue deux grandes superfamilles, à savoir les algorithmes d'obédience statistique :

1. Statistique descriptive, analyse de données et économétrie.
2. les algorithmes en apprentissage automatique et bases de données : filtrage d'individus et de variables, apprentissage supervisé, règles d'association.

d. Résultat :

Les opérateurs de TANAGRA produisent, la plupart du temps, des sorties au format **HTML**. Cette standardisation permet d'exporter facilement les résultats vers un logiciel d'édition, EXCEL par exemple, pour un éventuel post-traitement.

Les sorties comportent généralement deux parties : la description des paramètres du traitement demandé, et les résultats associés.

Le choix du format **HTML** à une seconde conséquence, l'exportation des résultats pour une lecture en dehors du logiciel est simplifiée. Il en est de même pour les impressions.

Lorsque cela est nécessaire, il reste possible de produire des résultats dans une fenêtre dans laquelle l'utilisateur peut agir de manière interactive. Il en est ainsi par exemple pour l'opérateur "Graphique X-Y", l'utilisateur peut modifier à la souris les variables en abscisses et ordonnées pour mieux comprendre la distribution des points.

C. Structures internes :

TANAGRA est implémenté en PASCAL OBJET, il est développé avec le langage de programmation DELPHI la version 6. Une version gratuite du compilateur est disponible sur le site de BORLAND. Le programme est donc compilé, il est distribué tel quel, son exécution ne nécessite aucune bibliothèque supplémentaire. En revanche, il ne fonctionne que sous Windows.

VI. Conclusion

Les outils data mining existe donc sous diverse type de langage, chacun a une méthode de fonctionnement et caractère spécifique. Dans ce chapitre nous avons exposé quelques outils brièvement pour les deux catégories. Ensuite, nous avons présenté une description détaillée sur les outils que nous utilisons dans notre étude comparative. Chaque outil propose des méthodes de classification, nous allons les voir en détail dans le chapitre suivant.

Chapitre3

Méthodes de classification supervisé

I. Introduction

La classification consiste à classer des individus en fonction de certaines de leurs caractéristiques. Il existe différents types de classification, mais un des plus intuitifs et des plus utilisés est la classification supervisée. L'idée de la classification supervisée est d'apprendre une règle de classement à partir d'un ensemble de données dont le classement est déjà connu. Une fois la règle apprise, il est possible de l'appliquer pour catégoriser de nouvelles données, dont le classement est inconnu.

Dans ce chapitre nous allons présenter les notions fondamentales de quelques méthodologies de classification supervisée nous nous intéresserons par la suite aux méthodes de classification supervisé : réseaux de neurones, support vecteur machines et K plus proche voisin on va donner plus de détails sur leurs principes de fonctionnement.

II. Les méthodes de classification supervisée

II.1 Classification naïve bayésien

C'est un type de classification Bayésienne probabiliste simple basée sur le théorème de Bayes avec une forte indépendance (dite naïve) des hypothèses et les probabilités conditionnelles. Le buts de cette méthode est de formaliser les méthodes et les intuitions, préciser la notion de 'plus probable' et d'analyse d'autres algorithmes ne manipulant pas explicitement des probabilités donc à chaque hypothèse on associe une probabilité (probabilité d'être la solution), l'observation d'une (ou de plusieurs) instances peut modifier cette probabilité on peut parler de l'hypothèse la plus probable, au vu des instances [17].

Les Réseaux Bayésiens Naïfs sont particulièrement efficaces lorsque le nombre de dimensions de l'espace des variables indépendantes (c'est-à-dire le nombre de variables en entrée) est élevé (un problème connu sous le nom du "problème" de dimensionnalité).

Un avantage de cette méthode est la simplicité de programmation, la facilité d'estimation des paramètres et sa rapidité.

II.2 Inférence grammaticale :

L'inférence grammaticale est un sous-domaine de l'apprentissage automatique et un ensemble fini de chaînes n'appartenant pas au langage dont le but est d'apprendre des modèles de langages (ensembles de mots).

L'inférence grammaticale a été étudiée depuis le développement de la théorie des grammaires formelles. Outre son intérêt théorique, elle offre un ensemble d'applications potentielles, en particulier dans les domaines de la Reconnaissance des Formes Syntaxiques et Structurelles, le Traitement de la Langue Naturelle et le Traitement de la Parole.

II.3 Arbre de décision

En anglais « decision trees », ensemble de règles de classification basant leur décision sur des tests associés aux attributs, organisés de manière arborescente. Leur structure arborescente les rend également lisibles par un être humain, contrairement à d'autres approches où le prédicteur construit est une « boîte noire » car ils sont employés pour détecter des critères permettant de répartir des individus d'une population en k classes prédéfinies **[18].**

Les arbres de décision fonctionnent a base des heuristiques qui, tout en satisfaisant l'intuition, donnent des résultats remarquables en pratique surtout lorsqu'ils sont utilisés en forêts aléatoires. L'arbre débute par la racine qui se divise en banches. Les branches conduisent à des nœuds qui peuvent à leur tour se diviser en branches. Les nœuds de l'arbre testent les attributs. Un nœud est défini par le choix conjoint d'une variable test parmi les explicatives et d'une division qui induit une partition en deux classes. Il y a une branche pour chaque valeur de l'attribut testé, Les feuilles spécifient les catégories et contiennent les décisions de classement final **[18]**

Un algorithme "arbre de décision" estime un concept cible par une représentation d'arbre, où chaque nœud interne correspond à un attribut, et chaque nœud terminal (ou feuille) correspond à une classe. La condition d'arrêt influe sur la profondeur et la précision du prédicteur produit .Il existe un très grand nombre de variantes algorithmes classiquement utilisés parmi ces algorithmes en peut citez :ID3 ,CHAID,CART,C4.5,C5,SLIQ, Exhaustive CHAID,QUEST,VFDT,UFFT .

III. Description détaillé des techniques de classification utilisées dans notre étude comparative

III.1 Réseaux de neurones

a. Définition

Un réseau de neurones artificiels est une technique de calcul qui se base sur le fonctionnement des neurones. Les neurones biologiques (ou neurones formels) opèrent mécaniquement à partir de règles précises [19]. Les réseaux sont fortement connectés de processeurs élémentaires fonctionnant en parallèle, chaque processeur élémentaires calcule une sortie unique sur la base des informations qu'il reçoit .Toute structure hiérarchique de réseaux est évidemment un réseau. Un réseau de neurones se présente comme un graphe où les nœuds sont les différentes unités de réseau et les arcs représentent les connexions entre ces unités. Le nombre de couches, le nombre de neurones par couche et les interconnexions entre les différentes unités du réseau définissent l'architecture (encore appelée topologie) de celui-ci. Un neurone peut être appelé unité ou cellule [20].

b. Réseaux multi couches

Les réseaux de neurones multicouches sont habituellement bâtis selon le modèle « normalisé » et comprennent 3 ou 4 couches en tout (donc 1 ou 2 couches cachées). S'il est théoriquement possible de construire des réseaux avec un très grand nombre de couches cachées, les réseaux comprenant plus de couches cachées sont très rares, étant donné que chaque nouvelle couche augmente la quantité de calculs d'une manière exponentielle. La plupart des réseaux de neurones multicouches sont, dans la pratique, des perceptrons multicouches (PMC**)** [21].

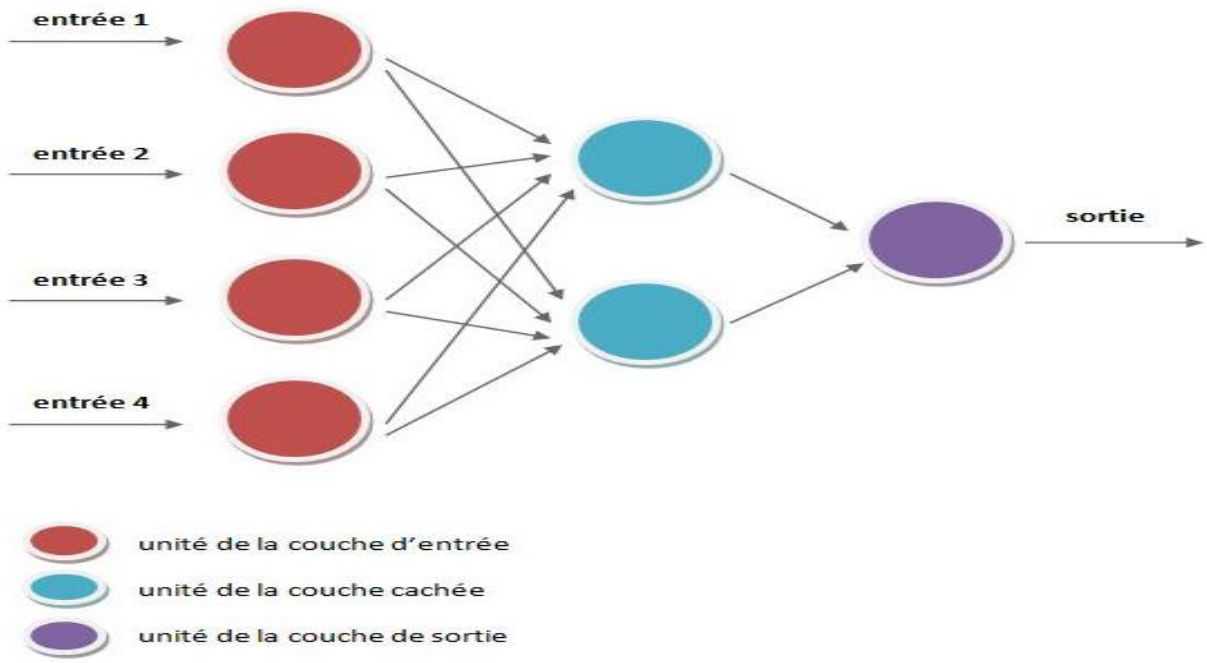

Figure 3.1: Simple réseau de neurones multicouche

Les réseaux de neurones constituent, tant pour analyser des données que pour prédire une variable dépendante. On distingue deux types de réseaux de neurones : les réseaux non bouclés et les réseaux bouclés.

c. Les réseaux de neurones bouclés (ou récurrents)

Un réseau de neurones bouclé est schématisé par un graphe des connexions qui est cyclique. Lorsqu'on se déplace dans les réseaux en suivant le sens des connexions, il est possible de trouver au moins un chemin qui revient à son point de départ (un tel chemin est désigné sous le terme de cycle).

d. Les réseaux de neurone non bouclés

Un réseau de neurones non bouclé réalise une (ou plusieurs) fonctions algébriques de ses entrées, par composition des fonctions réalisées par chacun des neurones. Il est représenté graphiquement par un ensemble de neurones « connectés » entre eux, l'information circulant des entrées vers les sorties sans « retour en arrière ».

e. Architecture

L'architecture générale des réseaux de neurones semblable, en première approximation à celle du cerveau humain, elle consiste en la représentation des neurones en couches (layers) successives, la première représentant la couche d'entrée (input layer), la dernière étant la couche de sortie (output layer), les couches intermédiaires étant les couches cachées (hidden layers) du réseau. Ces couches sont dites cachées car de l'extérieur du réseau, on ne peut analyser clairement leur fonctionnement. Le réseau reçoit les informations sur une couche réceptrice de neurones, traite ces information avec ou sans l'aide d'une ou plusieurs couche cachées contenant un ou plusieurs neurones et produit un signal ou plusieurs signales de sorties. Chaque neurone qu'il appartienne à la première couche, aux couches cachées ou à la couche de sortie est lié aux autres neurones par des connexions auxquelles sont affectés des poids [22].

f. Algorithmes d'apprentissage

Loi de Hebb: Cette règle très simple émet l'hypothèse que lorsqu'un neurone « A » est excité par un neurone « B » de façon répétitive ou persistante, l'efficacité (ou le poids) de l'axone reliant ces deux neurones devrait alors être augmentée.

Loi de Hopfield : Cette loi se base sur la même hypothèse que la loi de Hebb mais ajoute une variable supplémentaire pour contrôler le taux de variation du poids entre les neurones avec une constante d'apprentissage qui assure à la fois la vitesse de convergence et la stabilité du réseaux de neurones artificiel .

Loi Delta : Cette loi est aussi une version modifiée de la loi de Hebb. Les poids des liens entre les neurones sont continuellement modifiés de façon à réduire la différence (le delta) entre la sortie désirée et la valeur calculée de la sortie du neurone. Les poids sont modifiés de façon à minimiser l'erreur quadratique à la sortie du réseau de neurones artificiel. L'erreur est alors propagée des neurones de sortie vers les neurones des couches inférieures, une couche à la fois [23].

L'algorithme d'entraînement pour un perceptron à un seul neurone:

1- Initialisation

Mettre les poids initiaux w_1, w_2, …, w_n ainsi que le seuil θ à des valeurs aléatoires de l'intervalle [-0,5 , 0,5]. Mettre le taux d'apprentissage □ à une petite valeur positive.

2- Activation

Activer le perceptron en appliquant les intrants $x_1(p)$, $x_2(p)$. …, $x_n(p)$ et l'extrant désiré $Y_d(p)$. Calculer l'extrant actuel à l'itération p = 1.

$$Y(p) = étage\left[\sum_{i=1}^{n} x_i(p) w_i(p) - \theta\right]$$

où n est le nombre de signaux intrants, et étage est la fonction d'activation par étage.

3- Entraînement des poids

Mettre à jour les poids du perceptron

$$w_i(p+1) = w_i + \Delta w_i$$

où $\Delta w_i(p)$ est la correction de poids à l'itération p.

La correction de poids est calculée par la loi delta :

$$\Delta w_i = \alpha \times x_i(p) \times e(p)$$

où □ est le taux d'apprentissage et e(p) l'erreur à l'itération p (la différence entre l'extrant désiré et l'extrant actuel du perceptron)

4- Itération

Augmenter p de 1, retourner à l'étape 2 et répéter le procédé jusqu'à convergence.

III. 2 Les machines à vecteurs de support SVM

a) Définition

Les machines à vecteurs de support (SVM) appelés aussi «maximum margin classifier est une méthode de classification binaire par apprentissage supervisé introduite par Vapnik (1995).Pour résoudre des problèmes de classification, cette méthode repose sur l'existence d'un classifieur linéaire dans un espace approprié ,et sur l'utilisation de fonction noyau qui permettent une séparation optimale des données [24].

Les SVM impliquent plusieurs notions mathématiques, dont la théorie de la généralisation, peu abordée ici, la théorie de l'optimisation, et les méthodes d'apprentissage bastées sur des fonctions noyau. Le gros intérêt des noyaux est que tout ce qu'on vient de voir sur la séparation linéaire s'applique en fait très facilement à des séparations non linéaires, sous réserve de bien faire les choses [25].

b) Objectif

C'est de trouver l'hyperplan séparateur optimal qui maximise la marge entre les classes dans un espace de grande dimension d'une autre façon résoudre les problèmes de discrimination à deux classes. Définissons la marge d'un hyperplan comme étant la distance entre l'hyperplan et la donnée la plus proche.

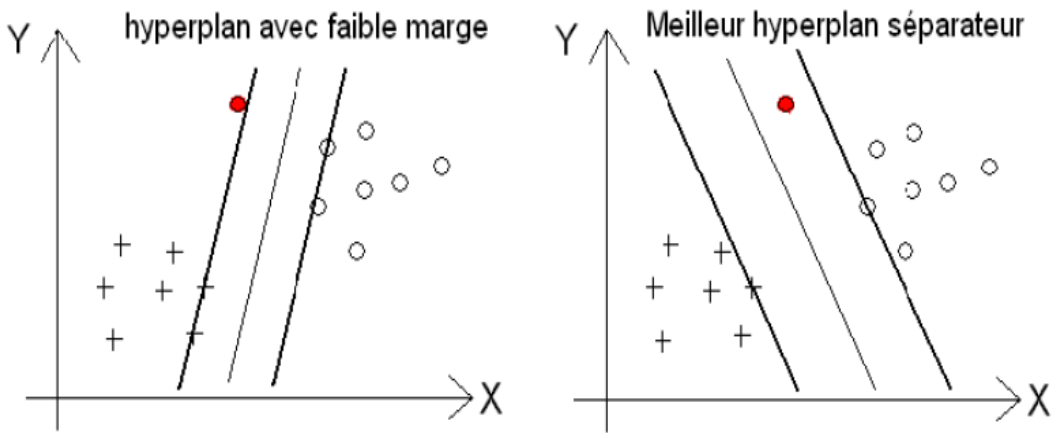

Figure 3.2 : hyperplan faible marge et optimale [25]

c) Principe de la maximisation de la marge

Il est évident qu'il existe une multitude d'hyperplan valide mais la propriété remarquable des SVM est que cet hyperplan doit être optimal. L'hyperplan optimal doit maximiser la distance entre la frontière de séparation et les points de chaque classe qui lui sont le plus proche.

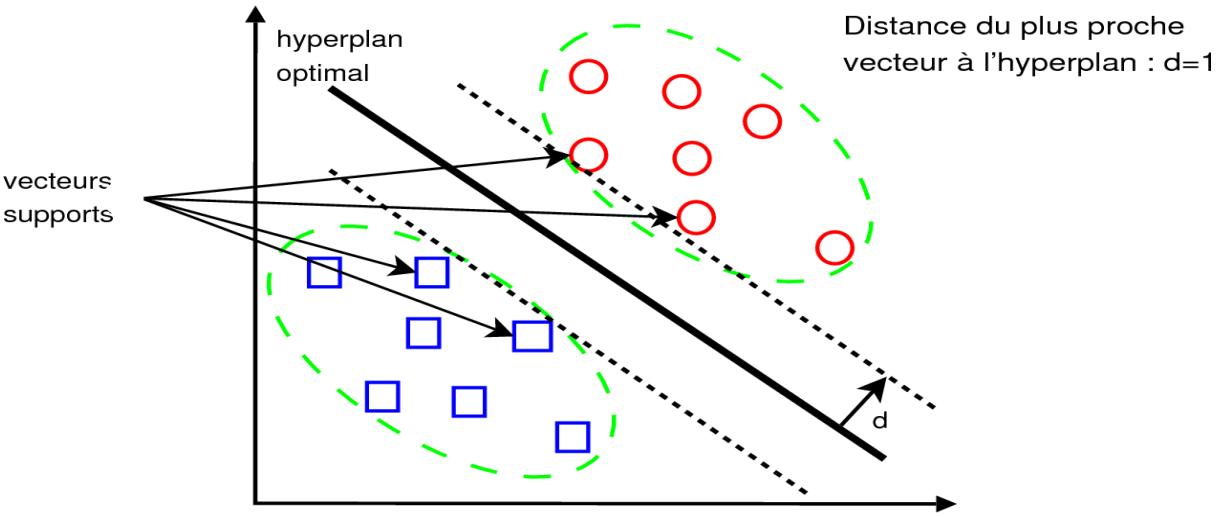

Figure 3.3 : hyperplan optimale et vecteurs supports

d) Modèle SVM

Les cas linéairement séparable : ils permettent de trouver facilement le classificateur linéaire. Dans la plupart des problèmes réels il n'y a pas de séparation linéaire possible entre les données **[25].**

Les cas non linéairement séparables : on doit changer l'espace des données. C'est-à-dire avoir une nouvelle dimension appelée espace de re-description car cette transformation non linéaire est effectuée via une fonction noyau **.**

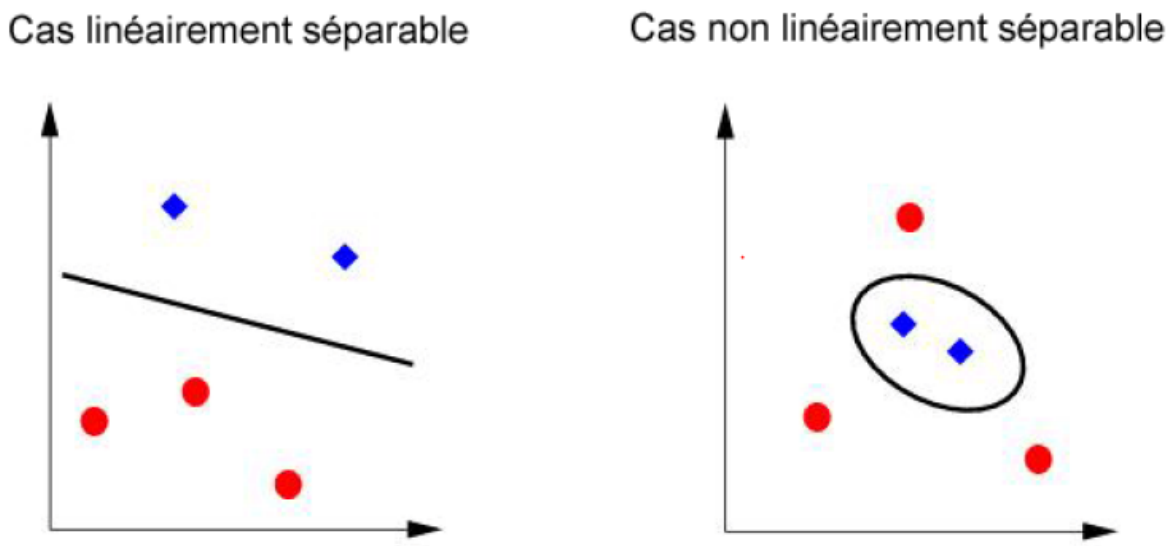

Figure3.4 : Les SVM linéairement séparable et non linéairement séparable

e) Fonctions Noyaux

SVM vous permet d'utiliser un certain nombre de noyaux dans vos modèles de Support Vecteur Machines. Ces noyaux peuvent être de plusieurs type linéaire, polynomial, fonction gaussienne radiale (RBF) et sigmoïde.

La fonction gaussienne radiale (RBF) est de loin le type de noyaux le plus fréquent en Séparateurs à Vaste Marge (SVM - Support Vecteur Machines). Ceci, en raison de ses réponses localisées et finies sur toute l'étendue de l'axe réel x.

III. 3 k-plus proches voisins KNN

a. Définition

La méthode k-plus proches voisins (k-neighrest neighbors en anglais) est une méthode non paramétrique très intuitive où une nouvelle observation est classée dans la classe d'appartenance de l'observation de l'échantillon d'apprentissage qui lui est la plus proche, au regard des covariables utilisées. La détermination de leur similarité est basée sur des mesures de distance. L'algorithme des k-plus proches voisins est un algorithme intuitif, aisément paramétrable pour traiter un problème de classification avec un nombre quelconque d'étiquettes[26].

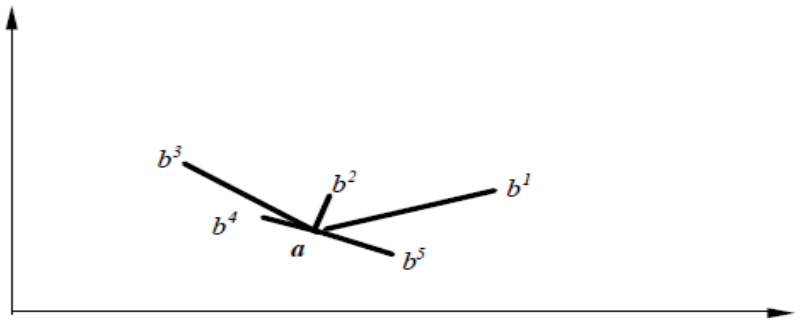

Figure 3.5: Méthode des 3 plus proche voisins

Dans l'exemple de la figure 3.5 les trois plus proches voisins de **a** sont b4, b2 et b5, donc **a** sera affecté à la classe majoritaire parmi ces trois points.

La méthode des k- plus proche voisins a l'avantage d'être très simple à mettre en œuvre et d'utiliser directement l'ensemble d'apprentissage. Elle ne fait aucune hypothèse a priori sur les données. La qualité de la discrimination par cette méthode dépend du choix du nombre k de voisins considérés.

b. Quelques règles sur le choix de k

Le paramètre k doit être déterminé par l'utilisateur : k ϵ N. En classification binaire, il est utile de choisir k impair pour éviter les votes égalitaires. Le meilleur choix de k dépend du jeu de données [25].

c. L'algorithme des k-plus proches voisins

L'algorithme des k-plus proches voisins est un des algorithmes de classification les plus simples d'apprentissage automatique supervisé. Le seul outil dont on a besoin est une distance entre les éléments que l'on veut classifier. Si on représente ces éléments par des vecteurs de coordonnées, il y a en général pas mal de choix possibles pour ces distances, partant de la simple distance usuelle (euclidienne) en allant jusqu'à des mesures plus sophistiquées pour tenir compte si nécessaire de paramètres non numériques comme la couleur, la nationalité, etc.

Voici l'algorithme de k plus poche voisin :

Input: Données d'apprentissage ; $\mathbf{X}^{\text{train}} = (\mathbf{x}_1^{\text{train}}, \ldots, \mathbf{x}_n^{\text{train}})$; classes des données d'apprentissage $\mathbf{z}^{\text{train}} = (z_1^{\text{train}}, \ldots, z_n^{\text{train}})$; $\mathbf{X}^{\text{test}} = (\mathbf{x}_1^{\text{test}}, \ldots, \mathbf{x}_m^{\text{test}})$

Algorithme `Knn` :

for $i \leftarrow 1$ ***to*** m **do**
  for $j \leftarrow 1$ ***to*** n **do**
    Calculer la distance euclidienne entre $\mathbf{x}_i^{\text{test}}$ et $\mathbf{x}_j^{\text{train}}$ en utilisant l'équation
    $\boldsymbol{d}_j \leftarrow d(\mathbf{x}_i^{\text{test}}, \mathbf{x}_j^{\text{train}})$
  end
  Calculer la classe z_i^{test} du $i$ème exemple qui vaut la classe de son pvv :

  `trouver l'indice du ppv de` $\mathbf{x}_i^{\text{test}}$:

  $ind\_ppv_i \leftarrow \arg\min_{j=1}^{n} \boldsymbol{d}_j$

  `trouver la classe du ppv de` $\mathbf{x}_i^{\text{test}}$ (qui est $\mathbf{x}_{ind\_ppv_i}^{\text{train}}$) :

  $z_i^{\text{test}} = z_{ind\_ppv_i}^{\text{train}}$
end

Result: classes des données de test $\mathbf{z}^{\text{test}} = (z_1^{\text{test}}, \ldots, z_n^{\text{test}})$

IV. Conclusion

Dans ce chapitre nous avons tenté de présenter de manière simple et complète le concept de quelques méthodes de classification supervisée et non supervisée, nous avons donné une vision générale sur les différentes méthodologies de la classification supervisé utilisé dans notre étude comparative telle que : Les réseaux de neurones (RN), les k plus proche voisin (Kppv), et le support vecteur machines (SVM). Dans le chapitre suivant, nous présenterons les différents résultats de chaque outil en utilisant les méthodes que nous avons détaillé précédemment.

Chapitre4

Expérimentation et Résultats

I. Introduction

Dans les chapitres précédents nous avons exposé les méthodes d'apprentissage et les différents outils de classification data mining. Nous présentons dans cette dernière partie notre étude comparative, nous avons réalisé une étude comparative entre les résultats de trois outils de classification des données médicales. Plusieurs méthodes sont utilisées dans ce domaine nous nous sommes intéressés à trois méthodes qui sont: Knn, SVM et RN. Nous avons choisi ces trois méthodes car elles sont très utilisées dans la littérature.

Pour que les résultats soient réellement comparables, il aurait fallu subdiviser les données en deux parties (apprentissage et test) puis de lancer les logiciels sur les mêmes ensembles de données mais ici on va utiliser la technique de ré-échantillonnage validation croisé.

La figure 4.1 donne une vue générale de notre travail.

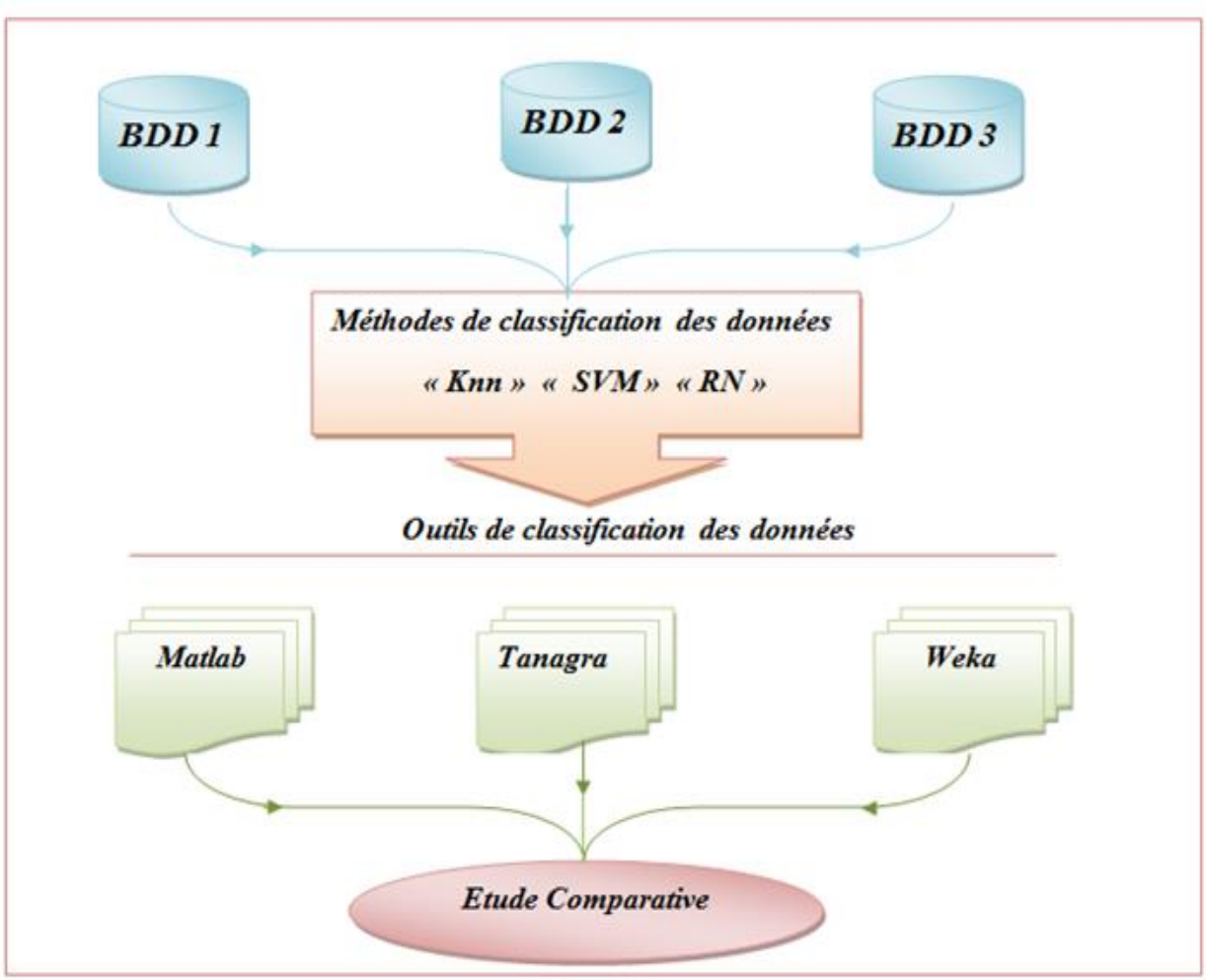

Figure 4.1-schéma synoptique de notre étude

II. Bases de données

Pour mener à bien notre étude comparative, nous avons choisi trois bases de données médicales réelles extraites du dépôt d'UCI [27] qui sont: Pima, Appendicite et Heart. Le choix de ces trois bases est justifié par les critères suivants :

1. La taille de la base.
2. Nombre des attributs.
3. Nombre de classe.

II.1 Description de la base de données Pima

La base Pima indien Diabète est constitué de 768 cas dont 268 sont diabétique et 500 non diabétique. Chaque cas est formé de 9 attributs, dont 8 représentent des facteurs de risque et le 9eme représente la classe du patient.

Les huit descripteurs cliniques sont :

1. Npreg : nombre de grossesses,
2. Glu : concentration du glucose plasmatique,
3. BP : tension artérielle diastolique
4. SKIN : épaisseur de pli de peau du triceps,
5. Insuline : dose d'insuline,
6. BMI : index de masse corporelle,
7. PED : fonction de pedigree de diabète (l'hérédité)
8. Age : âge

Et 2 classes (1 ET 0)

- Si classe =1 implique Présence de la maladie
- Si classe =0 implique Absence de la maladie

II.2 Description de la base de données Appendicits

Appendicits est une base de données crée par Shalom Weiss de l'université Rutgers, elle est composée de 106 individus avec sept attributs. Ces individus sont répartis en deux classes : 80.2% des données appartiennent à la classe « 1 » (diagnostic positive) et 19.8% appartient à la classe « 2 » (diagnostic négative). [28]

Les sept attributs (tests de laboratoire) sont :

1. WBC1
2. MNEP
3. MNEA
4. MBAP
5. MBAA
6. HNEP
7. HNEA

II.3 Description de la base de données Heart

La base Heart représente les maladies cardiovasculaires, elle contient 270 Individus décrit par 13 attributs. La base est regroupée en 2 classes, la classe 1 représente l'absence de la maladie et la classe 2 la présence d'une maladie.

Les 13 descripteurs cliniques sont :

-- 1. age [29,77]
-- 2. sex [homme et femme] [0,1]
-- 3. chest pain type (4 values)
-- 4. resting blood pressure [94, 200]
-- 5. serum cholestoral in mg/dl [126, 564]
-- 6. fasting blood sugar > 120 mg/dl
-- 7. resting electrocardiographic results (values 0,1,2)
-- 8. maximum heart rate achieved
-- 9. exercise induced angina
-- 10. oldpeak = ST depression induced by exercise relative to rest
-- 11. the slope of the peak exercise ST segment
-- 12. number of major vessels (0-3) colored by flourosopy
-- 13. thal: 3 = normal; 6 = fixed defect; 7 = reversable

Et 2 classes : 1 et 2 qui représentent l'absence ou la présence de la maladie cardiovasculaire.

III. Critères d'évaluation

Le critère d'évaluation est un facteur clé à la fois dans l'évaluation de la performance de classification et guidance de la modélisation de classificateur. Pour comparer de façon synthétique les performances des différentes méthodes et de différents outils retenues pour notre étude nous avons calculé : le taux de classification(**TC**), la sensibilité(**SE**) et la spécificité(**SP**), leurs définitions respectives sont les suivantes :

$$TC = 100*[(VP + VN)/ (VP + VN + FP + FN)]$$
$$Se = 100*[(VP/VP + FN)]$$
$$Sp = 100*[(VN/VN + FP)]$$

- Vrai positive (**VP**) : les cas positives classé positives
- Vrai négative (**VN**) : les cas négatif classé négative
- Faux positive (**FP**) : les cas positive classé négative
- Faux négative (**FN**) : les négatives classées positive
- Taux de classification : le pourcentage des exemples classé correctement
- Sensibilité (**Se**) : le pourcentage des exemples positive classé correctement

- Spécificité (**Sp**) : le pourcentage des instances négative classé correctement

IV. Résultats et Discussions

Cette partie relate les résultats de l'étude comparative entre les trois outils de classification des données en utilisant les trois algorithmes (RN, SVM, Knn appliqués aux trois bases de données sélectionnées (Pima, Appendicite, Heart).

L'objectif de chaque algorithme est de parvenir à mieux classer les futures observations en minimisant l'erreur de classification.

La question que nous nous sommes posé est : Quel est le meilleur outil? Ou quel outil donne les meilleurs résultats ?

Nous utilisons une méthode de ré-échantillonnage pour obtenir une estimation plus fidèle de la vraie valeur de l'erreur, dans le but de faciliter le travail et d'améliorer la performance du classifieur on applique la technique de k-cross validation (k=10) car on fait l'apprentissage et le test pour tous les éléments de la base. La validation croisée est une procédure couramment utilisée pour évaluer les performances des méthodes d'apprentissage supervisé, surtout lorsque les bases sont de taille réduite.

Nous constatons qu'avec des stratégies différentes, ces trois logiciels permettent de la mettre en œuvre assez facilement.

Les expérimentations ont été réalisées sur 3 bases de données médicales : Pima, Appendicite et Heart. Les principales caractéristiques de ces ensembles de bases sont représentées dans le tableau 4.1.1 :

Base de données	Nombre de cas	Nombre des attributs	Nombre de classe
Pima	768	8	2
Appendicits	106	7	2
Heart	270	13	2

TABLEAU 4.1.1-Les trois bases de données utilisées dans cette étude

Les logiciels que nous avons mis en compétition sont les suivants :

Logiciel	Version	URL
TANAGRA	1.4	http://eric.univ-lyon2.fr/~ricco/tanagra/
WEKA	3-6	http://www.cs.waikato.ac.nz/ml/weka/
Matlab	R2013a	http://www.marocpolis.com/education/23-mathematiques/1700-telecharger-mathworks-matlab-r2013a

TABLEAU 4.1.2-Les trois logiciels utilisées dans cette étude

IV.1 Résultats de la méthode Knn

Le tableau suivant résume les résultats obtenus par l'algorithme de k plus proche :

		Pima			**Appendicits**			**Heart**		
Test d'évaluation	Nombre de k	matlab	Tanagra	Weka	matlab	tanagra	weka	matlab	Tanagra	weka
Taux de classification(%)	K=1	0.6719	0.6763	0.6998	0.8182	0.8400	0.8558	0.5889	0.5926	0.6592
	K=5	0.7252	0.7053	0.7311	0.8582	0.8700	0.8558	0.6370	0.6481	0.7851
	K=9	0.7278	0.7303	0.7442	0.7200	0.8800	0.8270	0.6481	0.6593	0.8000
	K=13	07462	0.7382	0.7429	0.8510	0.8800	0.8462	0.6556	0.6630	0.8149
	K=17	**0.7447**	**0.7645**	**0.7520**	**0.8582**	**0.8800**	**0.8750**	**0.6593**	**0.6704**	**0.8260**
Sensibilité(%)	K=1	0.7410	0.5396	0.5798	**1**	***0.9024***	*0.8888*	*0.6542*	*0.5433*	*0.7705*
	K=5	0.8351	0.5955	0.6313	*0.8750*	*0.8965*	***0.8913***	*0.7362*	*0.6146*	*0.7891*
	K=9	0.8381	0.6422	0.6551	1	0.8977	*0.8453*	*0.7322*	*0.6320*	*0.8048*
	K=13	0.8606	0.7420	0.6591	0.8889	0.8977	*0.8709*	*0.7138*	*0.6330*	*0.8198*
	K=17	**0.8665**	**0.7794**	**0.6805**	1	0.8977	*0.8804*	***0.7521***	***0.6504***	***0.8209***
Spécificité(%)	K=1	0.5493	0.7494	0.7537	0	*0.5555*	*0.6428*	*0.5037*	*0.6282*	*0.8100*
	K=5	0.5228	0.7514	0.7754	*0.5000*	0.6470	*0.7500*	***0.5703***	*0.6708*	*0.8173*
	K=9	**0.8381**	0.7656	0.7827	0	0.7500	*0.5714*	*0.5590*	*0.6768*	*0.8301*
	K=13	0.5361	**0.8098**	**0.7827**	0	**0.7500**	***0.8804***	*0.5447*	***0.6832***	*0.8348*
	K=17	0.5218	0.7975	0.6805	**0.7500**	0.7500	*0.6667*	*0.5284*	*0.6826*	***0.8425***

TABLEAU 4.2 Tableau comparatif des résultats de classification des différents outils sur les trois bases de données par l'algorithme Knn.

- L'expérience a été faite sur des bases de taille différente. Le taux de classification est optimiste et les résultats très proche pour les différentes valeurs de k sur les trois bases de données avec les trois outils (les meilleurs résultats sont en gras). d'un point de vue temps d'exécution, on observe que le classifieur k plus proche voisin a bien classé les données de la base appendicits, comme remarque importante plus le nombre de K augmente plus le taux de classification augmente, ceci montre que le choix de K est important et influe sur les résultats, lorsque le K est grand signifie que le classifieur KNN exploite plus d'information sur le voisinage qui peut améliorer les résultats classification.

-En plus nous remarquons que la spécificité du système est élevée pour les trois bases de données avec les trois outils utilisée ça veut dire que le système a fait un bon apprentissage pour les données négative. Concernant la sensibilité presque la majorité des résultats sont satisfaisants, ce qui veut dire que le système a fait une bonne reconnaissance des données positive. Avec ces performances, nous pouvons dire que le classifieur a fait une bonne classification.

IV.2 Résultats de la méthode SVM

Le tableau suivant résume les résultats obtenus par l'algorithme support vecteur machine: Données par l'algorithme SVM.

		Pima			**Appendicites**			**Heart**		
Test d'évaluation	Degré polynomial	matlab	tanagra	weka	matlab	tanagra	weka	matlab	Tanagra	weka
Taux de classification(%)	K=1	**0.7527**	**0.7748**	**0.7645**	**0.8654**	**0.8800**	**0.7748**	**0.8370**	**0.6026**	0.7852
	K=5	0.6654	0.7514	0.7566	0.8365	0.8300	0.7514	0.7704	0.6481	0.8000
	K=7	0.6575	0.7553	0.7526	0.8270	0.8200	0.7553	0.8037	0.6593	0.8149
	K=9	0.6316	0.7305	0.7355	0.8173	0.7900	0.7305	0.7407	0.6630	0.8260
	K=10	0.6250	0.7305	0.7382	0.8173	0.7700	0.7305	0.5556	0.6704	**0.8296**
Sensibilité(%)	K=1	**0.7792**	**0.7435**	0.7291	**0.8817**	0.8977	0.7435	0.6542	0.5433	0.7705
	K=5	0.7352	0.7251	**0.7500**	0.8333	**0.9220**	0.7251	0.7362	0.6146	0.7891
	K=7	0.7431	0.7469	0.7423	0.8252	0.9210	**0.7469**	0.7322	0.6320	0.8048
	K=9	0.7219	0.6942	0.7062	0.8173	0.9178	0.6942	0.7138	0.6330	0.8198
	K=10	0.7328	0.6918	0.7023	0.8173	0.9154	0.6918	**0.7520**	**0.6504**	**0.8209**
Spécificité(%)	K=1	**0.6937**	**0.7853**	**0.7764**	0.7272	**0.7500**	**0.7853**	0.5037	0.6282	0.8100
	K=5	0.5318	0.7587	0.7583	**1**	0.5217	0.7587	0.5703	0.6708	0.8173
	K=7	0.5030	0.7574	0.7554	**1**	0.5000	0.7574	0.5590	0.6768	0.8301
	K=9	0.4667	0.7390	0.7433	0	0.4444	0.7397	**0.7361**	**0.7770**	0.7574
	K=10	0.4208	0.7405	0.7483	0	0.4137	0.7405	0.5284	0.6826	**0.8425**

TABLEAU 4.3 Tableau comparatif des résultats de classification des différents outils sur les trois bases de

-Les résultats de cette expérimentation pour les trois bases de données avec les différents outils sont satisfaisants surtout pour un degré polynôme égale « 1 » sauf le résultat de la base de données Heart avec tanagra.

-Comme remarque importante : pour la spécificité on trouve entre les résultats deux résultats très faible (valeur nulle) pour la base de données appendicits sous matlab pour un degré polynôme k=9,10, cela veut dire que le système n'a fait aucune reconnaissance pour les données négative par contre les autres résultats sont meilleurs.

- En plus nous remarquons que la sensibilité est élevé ce qui veut dire que le système a fait un bon apprentissage pour les données négatives. Avec ces performances, nous pouvons dire que le modèle par les trois outils a donné une bonne classification.

IV.3 Résultats de la méthode RN :

Le tableau suivant résume les résultats obtenus par l'algorithme réseau neurone:

		Pima			**appendicits**			**Heart**		
Test d'évaluation	Nombre de neurone	matlab	tanagra	weka	matlab	tanagra	weka	matlab	Tanagra	weka
Taux de classification(%)	K=5	0.7598	0.7670	0.7658	0.8691	0.7670	0.8700	0.8333	0.8112	0.8141
	K=10	0.7622	0.7670	0.7632	**0.8782**	0.7670	0.8700	0.8296	0.8186	0.8111
	K=15	0.7688	**0.7683**	**0.7737**	0.8691	**0.7696**	**0.8800**	0.8407	0.8075	**0.8259**
	K=20	0.7638	0.7683	0.7684	0.8400	0.7683	0.8700	0.8444	**0.8449**	0.8259
	K=25	**0.7727**	0.7696	0.7684	0.8664	0.7696	0.8700	**0.8481**	0.8038	0.8037
Sensibilité(%)	K=5	0.8808	0.7250	0.7050	**1**	0.7250	0.9156	0.8839	0.8235	0.8017
	K=10	0.8741	**0.7250**	0.6891	**1**	**0.7250**	0.9156	0.8740	0.8216	0.7948
	K=15	0.8772	0.7198	**0.7235**	**1**	0.7198	**0.9166**	0.8775	**0.8355**	0.8016
	K=20	0.8747	0.7163	0.6926	0.9689	0.7163	0.9156	0.8579	0.8289	**0.8173**
	K=25	**0.8935**	0.7198	0.6894	0.9664	0.7198	0.9176	**0.8977**	0.8211	0.7815
Spécificité(%)	K=5	0.5226	0.7830	**0.7900**	0.3333	0.7830	0.6470	0.8004	0.7948	0.8129
	K=10	0.5422	0.7830	0.7881	0.3333	0.7830	0.6470	0.7852	**0.8141**	0.8235
	K=15	**0.5628**	**0.7878**	0.7974	0.3333	**0.7878**	0.6875	0.7890	0.7881	**0.8456**
	K=20	0.5457	0.7875	0.7859	0.2500	0.7875	0.6470	**0.8153**	0.7966	0.8322

	K=25	0.5381	0.7878	0.8268	**0.5000**	0.7878	**0.7333**	0.7993	0.7815	0.8212

TABLEAU 4.4 Tableau comparatif des résultats de classification des différents outils sur les trois bases de données par l'algorithme RN.

-D'après les résultats de cette expérimentation, nous observons que les valeurs taux de classification pour les trois bases de données avec tous les outils sont satisfaisants avec une perturbation parfois le taux augmente et parfois il diminue par une petite différence lorsqu'on change le nombre de neurone ce qui indique que le changement d'architecture du PMC peut influencer les résultats.

-La sensibilité du système est élevée pour les trois bases de données avec les trois outils utilisés ça veut dire que le système a fait un bon apprentissage pour les données positives. Donc lorsqu'un individu non malade notre modèle le détecte avec succès. Même remarque pour la spécificité les résultats sont meilleurs sauf les résultats de la base de données appendicits sous matlab ils sont très faibles ce qui veut dire que le système a fait une mauvaise reconnaissance sur les données négatives.

V Comparaisons des résultats

Il existe dans la littérature un paramètre qui mesure de manière précise la variance de performance qui s'appelle paramètre de Friedman, on va calculer ce paramètre en utilisant la formule suivante :

$$Z = \frac{\mathrm{Ri} - \mathrm{Rj}}{\sqrt{\mathrm{k}(\mathrm{k}+1)/6\mathrm{n}}}$$

Ri moyenne de classement du premier outil, **Rj** moyenne de classement du deuxième outil, **k** nombre de colonne et **n** nombre de ligne. [29]

Pour calculer le paramètre de Friedman on va suivre les étapes suivantes :

- Choisir le classement correspondant à chaque test d'évaluation (meilleur résultat =1).
- Calculer la moyenne de classement pour chaque test d'évaluation.
- Appliquer la formule du paramètre de Friedman pour chaque deux outil et finalement on compare les résultats.

V.1 Comparaison des résultats de la méthode KNN

A. Taux de classification

Le tableau suivant résume les résultats de taux de classification des bases de données pour les trois outils pour l'algorithme Knn :

	Test d'évaluation	k	matlab	tanagra	weka
Pima	**Taux de classification(%)**	K=1	0.6719 3	0.6763 2	0.6998 1
		K=5	0.7252 2	0.7053 3	0.7311 1
		K=9	0.7278 3	0.7303 2	0.7442 1
		K=13	07462 1	0.7382 3	0.7429 2
		K=17	0.7447 3	0.7645 1	0.7520 2
Appendicites	**Taux de classification(%)**	K=1	0.8182 3	0.8400 2	0.8558 1
		K=5	0.8582 3	0.8700 2	0.8750 1
		K=9	0.7200 2	0.8800 1	0.8270 3
		K=13	0.8582 2	0.8800 1	0.8462 3
		K=17	0.8510 3	0.8800 1	0.8558 2
Heart	**Taux de classification(%)**	K=1	0.5889 3	0.5926 2	0.7851 1
		K=5	0.6593 2	0.6481 3	0.8000 1
		K=9	0.6556 3	0.6593 2	0.8149 1
		K=13	0.6370 3	0.6630 2	0.8260 1
		K=17	0.6481 3	0.6704 2	0.6592 1
Moyenne de classement (Pima, Appendicits, Heart)			2.6	1.93	1.46

TABLEAU 4.5.1 résultats de taux de classification avec le classement des outils pour toutes les bases de données.

comparaison	z
Matlab vs. Weka	1.4516
Matlab vs. Tanagra	2.3829
Weka vs. tanagra	0.9312

TABLEAU 4.5.2 résultats de la comparaison des outils par paramètre de Friedman

- Grace à la moyenne de classement nous pouvons calculer le paramètre de Friedman par la formule précédente et les résultats sont résumés dans le tableau 4.5.2 :

-D'après les résultats obtenus, si on veut comparer matlab vs tanagra on remarque que la mesure de Z est élevé cela signifie qu'il y a une différence de performance entre les deux, par contre l'autre mesure de Z est faible cela signifie qu'il n y a pas une différence de performance.

B. La sensibilité

Le tableau suivant résume les résultats de la sensibilité des bases de données de trois outils pour l'algorithme Knn :

	Test d'évaluation	k	matlab	tanagra	weka
Pima	**Sensibilité (%)**	K=1 K=5 K=9 K=13 K=17	0.7410 1 0.8351 1 0.8381 1 0.8606 1 0.8665 1	0.5396 3 0.5955 3 0.6422 3 0.7420 2 0.7794 2	0.5798 2 0.6313 2 0.6551 2 0.6591 3 0.6805 3
Appendicits	**Sensibilité (%)**	K=1 K=5 K=9 K=13 K=17	1 1 *0.8750 3* 1 1 0.8889 2 1 1	*0.9024 2* *0.8965 1* 0.8977 2 0.8977 1 0.8977 2	*0.8888 3* *0.8913 2* *0.8453 3* *0.8709 3* *0.8804 3*
Heart	**Sensibilité (%)**	K=1 K=5 K=9 K=13 K=17	*0.6542 2* *0.7362 2* *0.7322 2* *0.7138 2* *0.7521 2*	*0.5433 3* *0.6146 3* *0.6320 3* *0.6330 3* *0.6504 3*	*0.7705 1* *0.7891 1* *0.8048 1* *0.8198 1* *0.8209 1*
Moyenne de classement (Pima, Appendicits, Heart)			1.53	2.40	2.06

TABLEAU 4.5.3 résultats de sensibilité avec le classement des outils pour toutes les bases de données.

comparaison	z
Matlab vs. Weka	3.8345
Matlab vs. Tanagra	2.5400
Weka vs. tanagra	1.2873

TABLEAU 4.5.4 résultats de la comparaison des outils par paramètre de Friedman

En comparant les résultats obtenus dans le tableau 4.5.4 pour la sensibilité, nous remarquons clairement que la valeur entre weka et tanagra montre que la différence de performance de ce test est faible par rapport la valeur entre matlab vs tanagra et matlab vs weka.

C. La spécificité

Le tableau suivant résume les résultats de spécificité classification des bases de données pour les trois outils pour l'algorithme Knn :

	Test d'évaluation	K	matlab	tanagra	Weka
Pima	**Spécificité(%)**	K=1 K=5 K=9 K=13 K=17	0.5493 3 0.5228 3 0.8381 1 0.5361 3 0.5218 3	0.7494 2 0.7514 2 0.7656 3 0.8098 1 0.7975 1	0.7537 1 0.7754 1 0.7827 2 0.7827 2 0.6805 2
Appendicits	**Spécificité(%)**	K=1 K=5 K=9 K=13 K=17	0 3 *0.5000 3* 0 3 0 3 0.7500 1	*0.5555 2* 0.6470 2 0.7500 1 0.7500 2 0.7500 1	*0.6428 1* *0.7500 1* *0.5714 2* *0.8804 1* *0.6667 2*

Heart	**Spécificité(%)**	K=1 K=5 K=9 K=13 K=17	*0.5037 3* *0.5703 3* *0.5590 3* *0.5447 3* *0.5284 3*	*0.6282 2* *0.6708 2* *0.6768 2* *0.6832 2* *0.6826 2*	*0.8100 1* *0.8173 1* *0.8301 1* *0.8348 1* *0.8425 1*
Moyenne de classement (Pima, Appendicits, Heart)			2.73	1.8	*1.33*

TABLEAU 4.5.5 résultats de sensibilité avec le classement des outils pour toutes les bases de données.

comparaison	z
Matlab vs. Weka	1.8351
Matlab vs. Tanagra	3.1224
Weka vs. tanagra	1.2873

TABLEAU 4.5.6 résultats de la comparaison des outils par paramètre de Friedman

-Dans cette comparaison les résultats obtenue pour la spécificité est élevée pour chaque deux outil ce qui signifie qu'il y a une déférence remarquable dans les résultats.

V.2 Comparaison des résultats de la méthode de SVM

A. Taux de classification

Le tableau suivant résume les résultats du taux de classification des bases de données pour les trois outils pour l'algorithme SVM :

	Test d'évaluation	k	matlab	tanagra	weka
Pima	**Taux de classification (%)**	K=1 K=5 K=7 K=9 K=10	*0.7527 3* *0.6654 3* *0.6575 3* *0.6316 3* *0.6250 3*	*0.7748 1* *0.7514 2* *0.7553 1* *0.7305 2* *0.7305 2*	*0.7645 2* *0.7566 1* *0.7526 2* *0.7355 1* *0.7382 3*
Appendicits	**Taux de classification (%)**	K=1 K=5 K=7 K=9 K=10	*0.8654 2* *0.8365 1* *0.8270 1* *0.8173 1* *0.8173 1*	*0.8800 1* *0.8300 2* *0.8200 2* *0.7900 2* *0.7700 2*	*0.7748 3* *0.7514 3* *0.7553 3* *0.7305 3* *0.7305 3*
Heart	**Taux de classification (%)**	K=1 K=5 K=7 K=9 K=10	*0.8370 1* *0.7704 2* *0.8037 2* *0.7407 2* *0.5556 3*	*0.5926 3* *0.6481 3* *0.6593 3* *0.6630 3* *0.6704 2*	*0.7852 2* *0.8000 1* *0.8149 1* *0.8260 1* *0.8296 1*
Moyenne de classement (Pima, Appendicits, Heart)			*2.06*	2.06	*1.86*

TABLEAU 4.5.7 résultats de Taux de classification avec le classement des outils pour toutes les bases de données.

comparaison	z
Matlab vs. Weka	0
Matlab vs. Tanagra	0.5477
Weka vs. tanagra	0.5477

TABLEAU 4.5.8 résultats de la comparaison des outils par paramètre de Friedman

Les résultats obtenus dans le tableau 4.5.8 du paramètre Z est faible pour chaque deux outils cela signifie qu'il n'y a pas une différence de performance, les résultats du taux de classification sont proches.

B. La sensibilité

Le tableau suivant résume les résultats de la sensibilité de la classification des bases de données pour les trois outils pour l'algorithme SVM :

	Test d'évaluation	k	matlab	tanagra	weka
Pima	**Sensibilité(%)**	K=1	*0.7792 1*	*0.7435 2*	*0.7291 3*
		K=5	*0.7352 2*	*0.7251 3*	*0.7500 1*
		K=7	*0.7431 3*	*0.7469 1*	*0.7423 2*
		K=9	*0.7219 1*	*0.6942 3*	*0.7062 2*
		K=10	*0.7328 1*	*0.6918 3*	*0.7023 2*
Appendicits	**Sensibilité(%)**	K=1	*0.8817 2*	*0.8977 1*	*0.7435 3*
		K=5	*0.8333 3*	*0.9220 1*	*0.7251 2*
		K=7	*0.8252 2*	*0.9210 1*	*0.7469 3*
		K=9	*0.8173 2*	*0.9178 1*	*0.6942 3*
		K=10	*0.8173 2*	*0.9154 1*	*0.6918 3*
Heart	**Sensibilité(%)**	K=1	*0.6542 2*	*0.5433 3*	*0.8100 1*
		K=5	*0.7362 2*	*0.6146 3*	*0.8173 1*
		K=7	*0.7322 2*	*0.6320 3*	*0.8301 1*
		K=9	*0.7138 2*	*0.6330 3*	*0.8348 1*
		K=10	*0.7520 2*	*0.6504 3*	*0.8425 1*
Moyenne de classement (Pima, Appendicits, Heart)			1.93	*2.13*	1.93

TABLEAU 4.5.9 résultats de sensibilité avec le classement des outils pour toutes les bases de données.

comparaison	z
Matlab vs. Weka	3.6428
Matlab vs. Tanagra	2.3829
Weka vs. tanagra	1.2599

TABLEAU 4.5.10 résultats de la comparaison des outils par paramètre de Friedman.

Même remarque pour les valeurs de Z entre les outils pour le test d'évaluation de la sensibilité, Z est faible donc la différence de performance est faible.

C. La spécificité

Le tableau suivant résume les résultats de la spécificité de la classification des bases de données pour les trois outils pour l'algorithme RN :

	Test d'évaluation	k	matlab	tanagra	weka
Pima	**Spécificité(%)**	K=1	*0.6937 3*	*0.7853 1*	*0.7764 2*
		K=5	*0.5318 3*	*0.7587 1*	*0.7583 2*
		K=7	*0.5030 3*	*0.7574 1*	*0.7554 2*
		K=9	0.4667 3	0.7390 2	0.7433 1
		K=10	*0.4208 3*	*0.7405 2*	*0.7483 1*
Appendicits	**Spécificité(%)**	K=1	*0.7272 3*	*0.7500 2*	*0.7853 1*
		K=5	*1 1*	*0.5217 3*	*0.7587 2*
		K=7	*1 1*	*0.5000 3*	*0.7574 2*
		K=9	*0 3*	*0.4444 2*	*0.7397 1*
		K=10	*0 3*	*0.4137 2*	*0.7405 1*
Heart	**Spécificité(%)**	K=1	*0.5037 3*	*0.6282 2*	*0.8100 1*
		K=5	*0.5703 3*	*0.6708 2*	*0.8173 1*
		K=7	*0.5590 3*	*0.6768 2*	*0.8301 1*
		K=9	*0.7361 3*	*0.7770 1*	*0.7574 2*
		K=10	*0.5284 3*	*0.6826 2*	*0.8425 1*
Moyenne de classement (Pima, Appendicits, Heart)			2.73	*1.86*	*1.4*

TABLEAU 4.5.11 résultats de spécificité avec le classement des outils pour toutes les bases de données.

comparaison	z
Matlab vs. Weka	0.5477
Matlab vs. Tanagra	0
Weka vs. tanagra	0.5477

TABLEAU 4.5.12 résultats de la comparaison des outils par paramètre de Friedman

Les résultats de la mesure de Z pour la spécificité est différent par rapport les mesures de taux de classification et sensibilité car le z est élevé.

V.3 Comparaison des résultats de la méthode de RN

A. Taux de classification

Le tableau suivant résume les résultats du taux de classification des bases de données pour les trois outils pour l'algorithme de RN:

	Test d'évaluation	k	matlab	tanagra	weka
Pima	**Taux de classification(%)**	K=1	0.2402 3	0.2330 1	0.2342 2
		K=5	0.2378 3	0.2330 1	0.2368 2
		K=10	0.2312 2	0.2304 1	0.2263 3
		K=15	0.2362 3	0.2317 2	0.2316 1
		K=20	0.2273 1	0.2304 2	0.2316 3
Appendicits	**Taux de classification(%)**	K=1	0.1309 2	0.2330 3	0.1300 1
		K=5	0.1218 2	0.2330 3	0.1300 1
		K=10	0.1309 2	0.2304 3	0.1200 1
		K=15	0.1600 2	0.2317 3	0.1300 1
		K=20	0.1336 2	0.2304 3	0.1100 1
Heart	**Taux de classification(%)**	K=1	0.1667 1	0.1888 3	0.1852 2
		K=5	0.1704 1	0.1814 3	0.1889 2
		K=10	0.1593 1	0.1925 3	0.1741 2
		K=15	0.1556 1	0.1851 3	0.1741 2
		K=20	0.1519 1	0.1962 2	0.1963 3
Moyenne de classement (Pima, Appendicits, Heart)			1.8	2.4	1.8

TABLEAU 4.5.13 résultats de taux de classification avec le classement des outils pour toutes les bases de données.

Comparaison	z
Matlab vs. Tanagra	1.6433
Matlab vs. Weka	0
Weka vs. tanagra	1.6433

TABLEAU 4.5.14 résultats de la comparaison des outils par paramètre de Friedman

En comparant les résultats obtenus dans le tableau 4.5.14 pour le taux de classification, on remarque clairement qu'il y a une faible mesure de z entre Matlab vs. Weka ce qui signifie qu'il n y a pas une différence de performance.

B. La sensibilité

	Test d'évaluation	k	matlab	tanagra	weka
Pima	**Sensibilité (%)**	K=1	0.8808 1	0.7250 2	0.7050 3
		K=5	0.8741 1	0.7250 2	0.6891 3
		K=10	0.8772 1	0.7198 3	0.7235 2
		K=15	0.8747 1	0.7163 2	0.6926 3
		K=20	0.8935 1	0.7198 2	0.6894 3
Appendicits	**Sensibilité (%)**	K=1	1 1	0.7250 3	0.9156 2
		K=5	1 1	0.7250 3	0.9156 2
		K=10	1 1	0.7198 3	0.9166 2
		K=15	0.9689 1	0.7163 3	0.9156 2
		K=20	0.9664 1	0.7198 3	0.9176 2
Heart	**Sensibilité (%)**	K=1	0.8839 1	0.8235 2	0.8017 3
		K=5	0.8740 1	0.8216 2	0.7948 3
		K=10	0.8775 1	0.8355 2	0.8016 3
		K=15	0.8579 1	0.8289 2	0.8173 3
		K=20	0.8977 1	0.8211 2	0.7815 3
Moyenne de classement (Pima,Appendicits, Heart)			1	2.4	2.6

TABLEAU 4.5.15 résultats de sensibilité avec le classement des outils pour toutes les bases de données.

Comparaison	z
Matlab vs. Tanagra	3.8340
Matlab vs. Weka	4.3823
Weka vs. tanagra	0.5477

TABLEAU 4.5.16 résultats de la comparaison des outils par paramètre de Friedman

D'après les résultats obtenus, si on veut comparer matlab vs tanagra on remarque que les résultats de paramètre Z varient pour chaque deux outil, car on a trouvé une valeur faible de Z entre weka vs. Tanagra cela signifie qu'il y n a pas une différence de performance entre les deux, par contre l'autre mesure de Z est élevé cela signifie qu'il y a une différence de performance.

C. La spécificité

	Test d'évaluation	k	matlab	tanagra	weka
Pima	**Spécificité(%)**	K=1	0.5226 3	0.7830 2	0.7900 1
		K=5	0.5422 3	0.7830 2	0.7881 1
		K=10	0.5628 3	0.7878 2	0.7974 1
		K=15	0.5457 3	0.7875 1	0.7859 2
		K=20	0.5381 3	0.7878 2	0.8268 1
Appendicits	**Spécificité(%)**	K=1	0.3333 3	0.7830 1	0.6470 2
		K=5	0.3333 3	0.7830 1	0.6470 2
		K=10	0.3333 3	0.7878 1	0.6875 2
		K=15	0.2500 3	0.7875 1	0.6470 2
		K=20	0.5000 3	0.7878 1	0.7333 2
Heart	**Spécificité(%)**	K=1	0.8004 2	0.7948 3	0.8129 1
		K=5	0.7852 3	0.8141 2	0.8235 1
		K=10	0.7890 2	0.7881 3	0.8456 1
		K=15	0.8153 2	0.7966 3	0.8322 1
		K=20	0.7993 3	0.7815 2	0.8212 1
Moyenne de classement (Pima, Appendicits, Heart)			2.8	1.8	1.4

TABLEAU 4.5.17 résultats de spécificité avec le classement des outils pour toutes les bases de données.

Comparaison	z
Matlab vs. Tanagra	2.7389
Matlab vs. Weka	3.8345
Weka vs. tanagra	1.0955

TABLEAU 4.5.18 résultats de la comparaison des outils par paramètre de Friedman

Pour la même façon de comparaison, on remarque que les résultats obtenue montrent qu'il y a une différence performance entre matlab et weka.

VI. Discussion

TANAGRA, MATLAB et WEKA sont trois logiciels de data mining. S'ils poursuivent le même objectif, permettre aux utilisateurs de définir une succession de traitements sur les données, ils présentent néanmoins des différences. C'est tout à fait normal. Leurs auteurs n'ont pas la même culture informatique, cela se traduit par des choix technologiques différents; il ne procède pas de la même manière dans la fouille de données, ce qui se traduit par un vocabulaire et par un mode de présentation des résultats parfois différents.

A la lumière de l'objectif de notre étude comparatif et d'après l'interprétation et les remarques précédentes on peut dire que les résultats de chaque test d'évaluation de différent outil sont proches par un pourcentage. Puis, nous avons distingué la variance de performance mesurée par le paramètre de Friedman car elle n'est pas vraiment remarquable.

a. Les données :

Pour une comparaison plus juste et plus crédible, il était évident qu'il fallait opter pour les mêmes données. Notre souci était, d'éviter la lourdeur de la phase de préparation des données en optant pour des données sans erreurs, sans valeur manquante, sans valeurs nulles et sans redondance.

Un tel choix, nous permet d'éliminer une éventuelle influence des données sur les résultats produits par l'algorithme. Notre choix de données synthétisées et de grande qualité réduit encore plus un quelconque impact négatif. L'absence d'erreurs, de redondances, de valeurs manquantes et de valeurs erronées atteste de la bonne qualité des données, qui d'une part facilite la tâche de la classification et rehausse la qualité des résultats et d'autre part permet une neutralité parfaite.

Pour bien visualiser la différence entre les outils il faut vérifier d'autres critères comme :

b. Le temps de traitement :

Certains logiciels donnent directement une indication sur le temps de traitements, nous la privilégierons. Dans le cas contraire, nous on peut utiliser simplement un chronomètre. Ce n'est pas très précis. Mais nous ne sommes pas non plus en train de juger le 100 mètres de la finale olympique. Le plus important est d'avoir un ordre d'idées pour positionner les logiciels. Nous mesurons principalement le temps dévolu à l'importation des données et à la création de la méthode.

Il joue un rôle très important dans l'évaluation parce que le logiciel qui donne un résultat satisfaisant dans un temps minimum est considéré le meilleur et le plus rapide.

c. **Occupation mémoire** :

Concernant l'occupation mémoire, pour obtenir une mesure globale, indiquant à la fois les allocations interne et l'espace nécessaire à l'environnement global, nous privilégierons les indications du gestionnaire de tâches de WINDOWS.

Son importance est très visible avec les données volumineuses, les bases que nous avons utilisées dans notre étude sont de taille réduite.

d. La richesse de la bibliothèque des méthodes:

Matlab est un logiciel commercial de calcul interactif. Il permet de réaliser des simulations numériques basées sur des algorithmes d'analyse numérique. Il peut donc être utilisé pour la résolution approchée d'équations différentielles, d'équations aux dérivées partielles ou de systèmes linéaires.

Matlab est composé d'une suite d'instructions et toolbox dédié pour réalisé les opérations vectorielles ou matricielles et des calculs statistiques, il est de plus en plus utilisé dans l'industrie et les banques pour développer des prototypes de logiciels et tester de nouveaux algorithmes.

Tanagra est un outil **qui** propose une interface suffisamment conviviale, et accessible aux utilisateurs non spécialistes qui veulent effectuer des études sur des données réelles. D'autre part il définisse une architecture simplifiée à l'extrême, les efforts de développement portent sur l'essentiel, à savoir la mise au point et l'intégration d'algorithmes de fouille de données, les chercheurs peuvent ainsi mener des expérimentations sur les méthodes. Mais son inconvénient est très limité par exemple si on fait l'exécution par réseau de neurone on ne peut pas faire l'exécution seulement avec une couche caché, son principe de fonctionnement est de créer un diagramme de traitement et d'importer les données.

Weka est un logiciel incontournable Il intègre un très grand nombre de techniques, essentiellement d'obédience « machine learning ». Il propose peu de techniques issues de la statistique. Il est également possible de l'utiliser en ligne de commande.

WEKA a énormément progressé, il était impossible de charger un fichier un tant soit peu conséquent. Comme les classes de gestion de données n'ont pas été modifiées entre temps, il faut y voir surtout une amélioration drastique de la machine virtuelle JAVA.

WEKA impressionne par la richesse de sa bibliothèque de méthodes, mais paradoxalement c'est également son principal défaut : les possibilités sont immenses mais la présentation est très touffue, il est difficile de repérer les bonnes fonctionnalités et les composants à mettre en œuvre.

VII. Conclusion

Quel que soit le logiciel utilisé, nous constatons à travers cette étude qu'ils obéissent à la même logique de fonctionnement. Le tout est de trouver les composants, les menus et les boîtes de dialogues adéquats pour définir les bons paramètres de l'analyse.

La conclusion qu'on peut tirer de cette étude comparative est qu'il est difficile d'affirmer qu'un tel outil est meilleur par rapport à un autre, le choix dépend du problème où il va être appliqué. D'autres critères sont aussi très importants pour évaluer les outils : le temps de traitement, occupation mémoire, l'ergonomie, la richesse de la bibliothèque des méthodes.

Conclusion générale et perspectives

Diagnostiquer une maladie chez un patient sain ne produit pas les mêmes conséquences que de prédire la bonne santé chez un individu malade. Dans le premier cas, le patient sera soigné à tort, ou peut être demandera-t-on des analyses supplémentaires superflues; dans le second cas, il ne sera pas soigné, au risque de voir son état se détériorer de manière irrémédiable, et pour cela on a étudié la classification des données médicales afin d'éviter cette difficulté.

Nous avons introduit dans ce mémoire un nouveau paradigme qui a la problématique des lacunes des outils utilisées « MATLAB », « TANAGRA » et « WEKA » pour la classification supervisée des données médicales. Nous nous sommes intéressé plus particulièrement aux méthodes d'apprentissage supervisée qui visent à retourner une réponse précise à la question quel est le meilleur outil? Ou quel outil donne les meilleurs résultats ?

Ce travail nous a amené au développement d'une étude comparative entre les trois outils de classification des données en utilisant les trois algorithmes RN, SVM et Knn appliqués aux trois bases de données sélectionnées (Pima, Appendicite, Heart) du domaine médical.

L'objectif est de donner la possibilité aux professionnels d'étudier et d'appliquer les différentes techniques d'apprentissage supervisé à travers plusieurs outils de classifications.

Pour ce faire, nous avons tout d'abord étudié les approches et les notions fondamentales de la classification des données.

En premier lieu, nous avons présenté l'intelligence artificielle ainsi que les différentes techniques intervenant dans la classification. Puis, nous avons distingué les différentes approches adoptées pour l'apprentissage automatique et les méthodes d'apprentissage supervisé et non supervisé. Cette analyse nous a permis de constater l'importance de chaque méthode de classification surtout pour la classification supervisé pour le bon fonctionnement des bases de données dans le domaine médical, pour ensuite aborder les différents outils de classification, nous nous intéressons particulièrement à exposer les trois outils utilisés : « MATLAB », « TANAGRA » et « WEKA ».

Enfin, nous présentons les limites actuelles de notre étude comparative entre les résultats des trois outils de classification des données médicales et les difficultés rencontrées car on a conclue qu'il est difficile d'affirmer qu'un tel outil est meilleur par rapport à un autre.

Ce travail nous a également apporté une certaine vision globale de notre problème et nous a permis de définir le type d'approche à employer qui nous semble la plus adaptée à notre problématique.

Globalement, cette étude a permis d'exposer concrètement la problématique de classification de données dans le domaine médical, pouvant en cela contribuer a posteriori à trouver les solutions adéquates aux questions auxquelles est confronté ce domaine de recherche.

Notre travail est une nouvelle contribution apportée aux nombreuses études sur la classification. Pour une évaluation plus complète, et d'après ce qu'on a pu remarquer à travers ce travail on pourrait envisager quelques perspectives afin de noter ce qu'on peut améliorer dans celui-ci, il serait judicieux dans le futur d'élargir le volume et le type des données. Comme, il serait pertinent de tester d'autres types de classifieurs, surtout ceux qui supportent le modèle non supervisé et d'utiliser d'autres logiciels comme Python ou Keel pour voir le comportement de ces algorithmes.

Bibliographie

[1] Laurent Candillier, Contextualisation , visualisation et évaluation en apprentissage non supervisé , pp05, 15septembre 2006.
[2] http://www.iconomie.org/comprendre-lintelligence-artificielle/.
[3] Laurent Miclet, Apprentissage Artificiel : Méthodes et Algorithmes, pp 03-42, 30 novembre 2004.
[4] Pauline Le Badezet ,Alexandra Lepage , Classification Exemple : Enquête d'opinion sur les OGM ,pp05-06.
[5] Françoise Fessant, Apprentissage non supervisé , pp03, 28 septembre 2006
[6] http://blog.octo.com/apprentissage-par-renforcement-de-la-theorie-a-la-pratique/.
[7] A.Komathi, T.Ramya , M. Shanmugapriya, V. Sarmila "Comparative Study of Data Mining Tools», International Journal of Advanced Research in Computer Science and Software Engineering, novembre 2016.
[8] Kalpana Rangra, Dr. K. L. Bansa, "Comparative Study of Data MiningTools", International Journal of Advanced Research in ComputerScience and Software Engineering ,2014
[9] Ross Ihaka and Robert Gentleman. R : a language for data analysis and graphics. Journal of computational and graphical statistics, 5(3) :299–314, 1996.
[10] PierreLafaye de Micheaux, Rémy Drouilhet, and Benoît Liquet.Présentation du logiciel r. In Le logiciel R, Statistique et probabilités appliquées, pages 1–6. Springer Paris, 2011.
[11] Hadji Zahra, "Apprentissage en distribution déséquilibrée par les méthodes d'ensemble", université de Tlemcen.pp 44,2015.
[12] Professeurs Quentin Louveaux, Olivier Bruls, et Frédéric Nguyen, "INTRODUCTION A MATLAB", Université Liége, 2008.
[13]Harshwardhan Solanki. "Comparative Study of Data Mining Tools and AnalysiswithUnified Data MiningTheory". *International Journal of Computer Applications* (0975-8887), vol. 75, no.16, pp. 23-28, August 2013.
[14]Akshay VishwanathBhinge: "A COMPARATIVE STUDY ON DATA MINING TOOLS", université California, pp.14-17. 2015.
[15] Witten, I.H., Frank, E.: "Data Mining: Practical machine Learning tools and techniques", 2nd addition, Morgan Kaufmann, San Francisco(2005).
[16] Rakotomalala « TANAGRA : un logiciel gratuit pour l'enseignement et la recherche », ERIC – Université Lumière Lyon 25, av Mendès France, pp 1.
[17] Arnaud Liefooghe, Classification supervisée, pp30, 31.
[18] https://www.researchgate.net/publication/309731330 page 12-13.
[19]http://sante-medecine.journaldesfemmes.com/faq/22419-reseau-de-neurones-artificiels-definition.
[20] Norbert Tsopzé1, EngelbertMephuNguifo, Gilbert Tindo, Une étude des algorithmes de construction d'architecture des réseaux de neurones multicouche, CRIL-CNRS, IUT de Lens, SP 16 Rue de l'Université 62307 Lens Cedex {tsopze,mephu}@cril.univ-artois.fr ¤¤Département d'Informatique - Université de Yaoundé I BP 812 Yaoundé tsopze.norbert@gmail.com, gtindo@uycdc.uninet.cm.
[21] Marc-Olivier.,Résumé : Réseaux de neurones ,pp06, Mai 2002 .
[22]DenidThuillier , PRINCIPALES ET APPLICATIONS DES RESEAUX DE NEURONNES : Deux illustrations sur l'habitat au Maroc, pp 03.
[23] Andrei Doncescu , Les réseaux de neurones artificiels pp 19.

[24] Mohamadally Hasan. Fomani Boris , SVM : Machines a Vecteurs de Support ou Separateurs a Vastes Marges ..BD Web, ISTY3 .Versailles St Quentin, France . pp01, 16 janvier 2006.
[25]Herv_eFrezza-Buet, Sup_elec , Machines à Vecteurs Supports Didacticiel , pp 4,20 ,Octobre 2013.
[26]Eve Mathieu-Dupas ,Algorithme des k plus proches voisins pondères et application en diagnostic, Marseille, France, France. pp02,04 , 2010.
[27] UCI Machine Learning Repository. http://www.ics.uci.edu .
[28] Ben Mazzouz Maamar, Khouani Amin, 'optimisation paramétrique d'un classifieur neuronale par métaheuristique Application Données médicales ', mémoire master, Université Abou bekr belkaid Tlemce-n, 16 juin 2015.
[29] W.H. Kruskal, W.A. Wallis, Use of ranks in one-criterion variance analysis, Journal of the American Statistical Association 47 (1952) 583–621.