الجمهورية الجزائرية الديمقراطية الشعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم العالي و البحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جامعة أبي بكر بلقايد – تلمسان –

Université Aboubakr Belkaïd – Tlemcen –

THESE

Présentée pour l'obtention du **grade** de **DOCTORAT 3ème Cycle**

En : Génie biomédicale

Spécialité : Signaux et Images en Médecine

Par : BELAROUCI Sara

Sujet

Traitement et classification des données médicales non équilibrées

Soutenue publiquement, le 30/06/2016 , devant le jury composé de :

Mr. BESSAID Abdelhafid	Professeur	Université de Tlemcen	Président
Mr. CHIKH Mohammed Amine	Professeur	Université de Tlemcen	Directeur de thèse
Mr. RAHMOUN Abdelatif	Professeur	ESI de Sidi BelAbbes	Examinateur 1
Mr. MERAD Lotfi	MCA	Ecole préparatoire de Tlemcen	Examinateur 2

Dédicace

Je dédie cette thèse

A la mémoire de mon cher père

A ma très chère mère

A mes frères

A toute ma famille

A tous mes amies de l''Université et d'ailleurs

A tous ceux qui m'aiment et qui ont cru en moi

Qu'ils trouvent dans ce modeste travail l'expression de ma reconnaissance, mon amour, mon amitié et mon estime.

Remerciements

Je tiens à remercier avant tous, le bon Dieu qui m'a donné la force et la patience tout au long de la préparation de cette thèse.

Je tiens à exprimer mes sincères remerciements et ma très profonde gratitude à mon directeur de Thèse Mr. CHIKH Mohammed Amine, Professeur à l'université de Tlemcen, pour ses précieux conseils ainsi que son aide tout au long de cette Thèse. Il a su diriger avec enthousiasme et son esprit critique a été une aide importante pour la réalisation de ces travaux de recherches. Encore plus que ses grandes qualités scientifiques, j'ai beaucoup apprécié ses qualités humaines, en particulier l'écoute, la sympathie, la clairvoyance, la générosité, la gentillesse, le partage et la compréhension. Ce fut une grande fierté et un immense honneur pour moi de travailler sous la houlette d'un homme pour lequel j'ai un grand respect.

J'exprime ma plus vive gratitude à Mr. RAHMOUN Abdelatif, professeur à l'école supérieur d'informatique de Sidi BelAbbes ainsi qu'à Mr. MERAD Lotfi, Maitre de Conférences classe A à l'école préparatoire de Tlemcen, pour avoir accepté de juger mes travaux en tant que rapporteurs. Je tiens également à exprimer ma plus vive reconnaissance à Mr. BESSAID Abdelhafid, professeur à l'Université de Tlemcen, pour avoir accepté de juger mes travaux et présider ce jury. Je les remercie pour le soin qu'ils ont apporté à la lecture de ce manuscrit. Leurs remarques et leurs suggestions m'ont permis d'améliorer le manuscrit et de soulever des questions scientifiques importantes.

Je tiens également à remercier les membres du Laboratoire de génie biomédical pour leur soutien durant ces années. Plus particulièrement à Souaid, Hoda, Houaria et Bachir.

Une thèse est un travail assez personnel qui s'inscrit toutefois dans une équipe. Je remercie donc tous les membres de l'équipe CREDOM pour leur soutien et nos nombreux échanges. Je pense en particulier à Sarra, Amine, Djazia, Asma, Khalida, Nesma, Yasmine, Amina, Amel, Imane, Mansouria, Amina...

Merci également à mes amis qui m'ont apportée soutien et détente durant cette thèse ; Meriem, Sara, Fatima, Sara, et tous ceux qui s'y reconnaitront.

J'aimerais remercier du fond du cœur ma mère pour leur soutien moral. Ma famille qui a toujours porté un intérêt à ce que je faisais ...

Enfin je remercie tous ceux qui ont contribué de près ou de loin à l'aboutissement de ce travail.

Sara BELAROUCI

Résumé

Dans cette thèse nous traitons un problème majeur rencontré par les méthodes d'apprentissage supervisé dans beaucoup d'application du monde réelle telles que la détection de fraudes ou d'intrusions, le diagnostic médical, la classification de textes, etc. Ce problème concerne le déséquilibre des classes, lorsqu'au moins une classe est sous représentée par rapport aux autres. C'est particulièrement le cas du domaine de diagnostic médical, en effet nous rencontrons souvent des classes de pathologies minoritaires qui sont mal représentées lors de la phase d'apprentissage. Ce problème perturbe les algorithmes d'apprentissage, qui ils présentent une grande précision sur les classes majoritaires et une faible précision sur les classes minoritaires. Afin de remédier à ce problème, nous proposons dans le cadre de cette thèse de doctorat une méthode de pondération basée sur l'algorithme des moindres carrés moyens (LMS), qui pénalise les erreurs des différents échantillons par des poids différents. Cette méthode affecte des poids forts aux différents échantillons de la classe minoritaire et des poids faibles aux différents échantillons des classes majoritaires. Après cette phase d'équilibrage, nous testons plusieurs approches de classification (RNMC, SVM et K-PPV) sur les nouveaux ensembles de données équilibrés. Dans ce travail plusieurs contributions ont été réalisées dans le but d'améliorer les performances de la classification des données déséquilibrées et de focaliser la classification sur les classes minoritaires qui sont d'un grand intérêt lors du traitement des ensembles de données médicales déséquilibrées.

Les travaux réalisés durant cette thèse ont confirmé clairement que l'utilisation de la méthode de pondération LMS est effectivement très pertinente pour équilibrer les ensembles de données médicales, ainsi que les résultats obtenus sont très prometteurs et ils sont comparables aux travaux existants dans la littérature.

Mots clés : *Données déséquilibrées, Réseaux de Neurones, Séparateurs à Vastes Marges, K- plus proche voisin, Méthodes d'échantillonnage, Algorithme LMS.*

Abstract

In this thesis, we treat a major problem related to supervised learning methods in many real world applications such as fraud or intrusion detection, medical diagnosis, the text classification, etc. This problem concerns the imbalance of classes, when at least one class is underrepresented compared to other classes. This is particularly the case of the medical diagnostic field, indeed we often encounter classes of minority diseases that are poorly represented in the learning phase. This problem disrupts learning algorithms that can present high accuracy on the majority class and poor accuracy on the minority classes. To remedy this problem, we propose in this thesis a weighting method based on a Least Mean Square (LMS) algorithm, which penalizes errors of different samples with different weights. This method affects strong weights for the different samples of the minority class and low weights for the different samples of the majority classes. After this balancing phase, we test several classification approaches (NN, SVM and k-NN) on new balanced datasets. In this work, several contributions were made in order to improve the performances of the classification of imbalanced data and focus the classification on minority classes that are most important in the treatment of imbalanced medical datasets.

The work realized during this thesis has clearly confirmed that the use of the LMS weighting method is actually very relevant to balance medical datasets. The obtained results are very promising and they are comparable to the existing works in the literature.

Keywords: *Imbalanced data, Neural Networks, Support Vector Machine, K-Nearest Neighbors, Sampling methods, LMS algorithm.*

الملخص

في هذه الأطروحة نحن نتعامل مع مشكلة كبيرة تتعلق بأساليب التعلم تحت المراقبة التي واجهتها العديد من التطبيقات في العالم حاليا، مثل الكشف عن الغش أو التسلل، التشخيص الطبي، تصنيف النص، الخ. هذه المشكلة تتمثل في عدم التوازن بين الفئات، عندما تكون فئة واحدة على الأقل ممثلة تمثيلا ضعيفا بالمقارنة مع الفئات الأخرى. هذا هو الحال بصفة خاصة في مجال التشخيص الطبي حيث ان المعطيات المتعلقة بالأمراض تمثل دائما فئة الاقلية في معظم المعطيات الطبية وتظهر بشكل ضعيف في مرحلة التعلم. هذه المشكلة تؤثر في طريقة التعلم التي يمكن أن تقدم دقة عالية على معطيات الأغلبية ودقة رديئة على معطيات الأقلية. لمعالجة هذه المشكلة، نقترح في هذه الأطروحة طريقة الترجيح على أساس خوارزمية (LMS)، التي تقوم بتخفيض الاخطاء لعينات مختلفة بأوزان مختلفة. هذه الطريقة تحدد اوزان مرتفعة لعينات مختلفة من معطيات الأقلية واوزان منخفضة لعينات مختلفة من معطيات الأكثرية. بعد هذه المرحلة التي تتمثل في تحقيق التوازن، نحن نقوم باختبار ثلاث مصنفات (RNMC، SVM، K-PPV) على مجموعات المعطيات المتوازنة الجديدة. في هذا العمل قدمت عدة مساهمات من أجل تحسين أداء عملية تصنيف المعطيات الغير متوازنة وتركيز التصنيف على معطيات الأقلية التي هي الأكثر أهمية في معالجة مجموعات المعطيات الطبية الغير متوازنة.

العمل المنجز خلال هذه الأطروحة أكد بوضوح أن استخدام طريقة الترجيح LMS هو في الواقع مهم جدا لتحقيق التوازن في مجموعات المعطيات الطبية، والنتائج التي تم الحصول عليها واعدة جدا وقابلة للمقارنة مع الاعمال المنجزة حاليا في هدا المجال.

الكلمات المفتاحية: المعطيات الغير متوازنة، الشبكات العصبية، آلة دعم المتجه، الجار الأقرب، خوارزمية LMS.

Table des matières

Table des figures

Liste des tableaux

Liste d’abréviations

AdaOUBoost: Adaptive Over-sampling and Under-sampling Boost)
ADASYN: Adaptive synthetic sampling approach
AIRS: Artificial Immune Recognition System
ANN: Artificial Neural Networks
AUC: Area under the ROC Curve
BMPM: Biased Minimax Probability Machine
BPNN: Back-Propagation Neural Network
BSM: Borderline Synthetic Minority oversampling
BSMAIRS: Borderline Synthetic Minority oversampling - Artificial Immune Recognition System oversampling
CCW-kNN: Class Confidence Weighted - k-Nearest Neighbour
CenKNN: Centroid-based dimension reduction - k-Nearest-Neighbor
CF-kNN: Coupled Fuzzy k-Nearest Neighbour
CNMC: Classifieur Neuronal Multicouche
DCIL-IncLPSVM: Dynamic Class Imbalance Learning- Incremental Linear Proximal Support Vector Machines
DGC+: weighted Data Gravitation Classification
EDA: Estimation of Distribution Algorithms
ELM: Extreme Learning Machine
ENN: Edited Nearest Neighbor
FFNN: Feed Forward Neural Networks
FIR : Finite-duration Impulse Response
FN : Faux Négatifs
FP : Faux Positifs
FS: Feature Selection
FSVM-CIL: Fuzzy Support Vector Machines for Class Imbalance Learning
GA: Genetic Algorithm
GD: Gradient Descent
GDMV: Gradient Descent with Momentum and Variable learning rate
GI-KNN: Globally Informative k-Nearest Neighbour
Gmean: Geometric mean
GSVM: Geometric (mean) Support Vector Machine
HC-kNN: Hybrid Coupled k-Nearest Neighbour
IDELM: Integrating the Data selection and Extreme Learning machine
IIR : Infinite-duration Impulse Response
IPF: Iterative- Partitioning Filter
IR: Imbalance Ratio
kENN: k Exemplar based Nearest Neighbor
k-NN: k-Nearest Neighbour
K-PPV : K-Plus Proche Voisin

LI-KNN: Locally Informative k-Nearest Neighbour
LM: Levenberg-Marquardt
LMS: Least Mean Square
LPSVM: Linear Proximal Support Vector Machines
MLP: Multi-Layer Perceptron
MNN: Modular Neural Network
MPSVM: Modified Proximal Support Vector Machine
NBSVM: Near-Bayesian Support Vector Machine
NN: Neural Networks.
OS: Over-Sampling
OSS: One-Side Selection
PIMA: Pima Indian Diabetes
PNN: Positive-biased Nearest Neighbor
PSO: Particle Swarm Optimization
PSS: Parallel Selective Sampling
PSVM: Proximal Support Vector Machines
QBC: Query-By-Committee
RBF: Radial Basis Function
RBFN: Radial Basis Function Network
REA: Resampling Ensemble Algorithm
RICA: Real-coded Immune Clone Algorithm
RNA : Réseaux de Neurones Artificiels
RNMC : Réseau de Neurones Multicouche
RPROP : Resilient back propagation
S- AIRS : Synthetic minority oversampling technique – Artificial Immune Recognition System
SE : Sensibilité
SMOTE : Synthetic Minority Oversampling Technique
SMOTE+C4.5: Synthetic Minority Oversampling Technique + C4.5
SP: Spécificité
SUNDO: Similarity- based Under Sampling and Normal Distribution- based Oversampling
SVM : Support Vector Machines
TC : Taux de Classification correcte
TSEA: Two-Stage Evolutionary Algorithm
TSEAFS: Two-stage Evolutionary Algorithm Feature Selection
US: Under- Sampling
UCI: University California Irvine
VN : Vrais Négatifs
VP : Vrais Positifs
WBC: Wisconsin Breast Cancer
WDBC: Wisconsin Diagnostic Breast Cancer
WLTSVM: Weighted Lagrangian Twin Support Vector Machine
WSVM : Wavelet Support Vector Machine

Introduction générale

Contexte

L'approche par classification est très sollicitée par le domaine d'analyse des données (Data Mining). Elle consiste à regrouper des ensembles d'exemples similaires en classes selon certains critères. Chaque groupe est composé d'objets similaires et qui sont dissimilaires avec les objets des autres groupes où chaque groupe étant bien différencié des autres. Lors d'une tâche de classification, la phase d'apprentissage est très importante parce que cette phase doit faire face à certaines difficultés lorsqu'elle est confrontée aux particularités des données contenues dans des ensembles de données réelles. Parmi les difficultés les plus connues, nous citerons les problèmes de complexité dus à un grand nombre de cas traités, les contraintes liées aux grandes dimensions, la sensibilité des algorithmes aux données bruitées et enfin le traitement des données lorsque la composante des classes présentées est déséquilibrée, ce qui est souvent le cas pour les données réelles [1]. Dans notre travail de thèse, nous sommes beaucoup plus intéressés au problème de classification des données déséquilibrées.

Dans le monde réel, les classes des données sont déséquilibrées lorsqu'au moins une classe est sous représentée par rapport aux autres, les classes les plus fréquentes sont dites majoritaires et les classes les plus rares sont dites minoritaires. La classe d'intérêt représente généralement la classe minoritaire dans la plupart des problèmes de classification, car dans la réalité les événements rares sont fréquemment ceux qui ont le plus d'intérêt. Le déséquilibre des données peut se manifester dans de nombreuses applications du monde réel telles que la détection de fraudes ou d'intrusions, la gestion du risque, la classification de textes, le diagnostic médical, le suivi médical, … etc. Et aussi ce problème est lié avec d'autres problèmes existants dans la littérature comme les petits disjoints, les données bruitées et frontières, le chevauchement entre les classes, le manque de densité…etc.

Problématique, Motivation et Positionnement

Le déséquilibre dans la répartition des données, cause beaucoup de problèmes pendant la phase d'apprentissage des classifieurs. En effet, nous rencontrons souvent des classes minoritaires qui sont mal représentées lors de la phase d'apprentissage, lorsqu'un algorithme d'apprentissage standard est appliqué aux données déséquilibrées, les règles d'induction qui décrivent la classe minoritaire sont souvent rares et plus faibles que celles des classes majoritaires, puisque les classes

minoritaires sont souvent moins nombreuses et sous représentées. Le problème de déséquilibre des classes perturbe les algorithmes d'apprentissage qui peuvent présenter une grande précision sur les classes majoritaires et une mauvaise précision sur les classes minoritaires qui sont en général intéressantes [2, 3]. Le but de l'analyse est de donner une approche qui offrira une grande précision pour les classes minoritaires sans pour autant compromettre le taux de reconnaissance des classes majoritaires [4]. La plupart des algorithmes pénalisent d'autant plus la classe minoritaire que sa fréquence est faible, ce qui montre l'importance de l'équilibrage de données pour la conception d'un bon classifieur.

La réduction de l'erreur de généralisation est l'une des principales motivations de la recherche en apprentissage automatique, car un coût induit par une mauvaise classification peut être élevé [1]. Dans la littérature, plusieurs solutions sont proposées pour traiter le problème d'apprentissage des données déséquilibrées qui est devenu un axe de recherche d'une grande importance. Nous distinguons trois grandes familles principales selon le type de problème traité. La première famille concerne les approches de modification des algorithmes de classification, elles consistent à adapter les techniques de classification standards pour obtenir une classification correcte de la classe minoritaire. La deuxième concerne les approches de pondération des données (les méthodes d'échantillonnage) qui consistent à rétablir l'équilibre entre les classes soit par l'ajout des instances à la classe minoritaire (sur échantillonnage), soit par élimination des instances présentés dans des classes majoritaires ou soit par la combinaison des deux techniques d'échantillonnage c.-à-d. nous augmentons le nombre des instances minoritaires et nous diminuons le nombre des instances majoritaires simultanément. Et la troisième famille concerne les approches sensibles aux coûts qui consistent à minimiser le coût d'erreurs de mauvaise classification.

Dans le monde réel, ces approches d'équilibrage ont été appliquées à plusieurs domaines de recherche afin de réaliser une classification correcte des données minoritaires. Chacune de ces trois approches possède des avantages et des inconvénients. Les approches de modification des algorithmes de classification ne changent pas l'ensemble d'apprentissage mais elles peuvent laisser des lacunes qui doivent être examinées. Ces techniques dépendent de l'algorithme ou de l'application ciblée [5]. Par conséquent, elles ne sont efficaces que dans une situation bien limitée et aussi certaines techniques de classification sont complexes et nécessitent une meilleure compréhension des fonctions d'apprentissage automatique [6]. Par contre les approches de

pondération sont des techniques faciles à appliquer aux domaines de classification des données déséquilibrées. Elles sont indépendantes de l'algorithme de classification où les techniques d'apprentissage ne sont pas modifiées. Mais elles sont à l'origine de quelques inconvénients qui doivent être pris en charge, nous citons à titre d'exemple les techniques de sur-échantillonnage qui introduisent un risque de sur-apprentissage. Par contre les techniques de sous échantillonnage provoquent un risque de perte d'information où elles peuvent supprimer des individus importants de la classe majoritaire, nous pouvons dire que les techniques d'échantillonnage sont efficaces seulement sur les ensembles de données binaires [7]. Les approches sensibles aux coûts sont des techniques étroitement liées à la phase d'apprentissage. Elles présentent trois limites principales, la première limite concerne la nécessité de définir les coûts de mauvaise classification, qui ne sont pas généralement disponibles sur tous les ensembles de données. La deuxième concerne les coûts de mauvaise classification pour les prédictions incorrectes de classe qui doivent être habituellement déterminées a priori, ces coûts peuvent être un problème spécifique et ils nécessitent un long processus d'essai et d'erreur afin de configurer le système [8, 9]. La troisième limite concerne l'amélioration des indicateurs de performance dans la fonction objective (fitness**)** qui peuvent augmenter considérablement le temps d'apprentissage en raison de la surcharge de calcul nécessaire pour calculer ces mesures, en particulier sur les grands ensembles de données [4, 10, 11].

En raison de ces limitations, nous proposons une méthode de pondération basée sur la technique des moindres carrées moyennes appelée Least Mean Square (LMS). L'objectif principal de cette approche consiste à pénaliser les erreurs des différents échantillons par des poids différents où les données minoritaires et majoritaires sont pondérées. Cette technique permet d'affecter des poids forts aux différents échantillons de la classe minoritaire et des faibles poids aux différents échantillons des classes majoritaires afin de créer un équilibre entre les classes. Cette technique permet de renforcer aussi la pertinence des attributs de chaque classe afin de minimiser l'erreur pour atteindre une meilleure classification. Nous pouvons dire que cette méthode de pondération (LMS) des données minoritaires est une technique de prétraitement, elle permet de créer un équilibre entre les classes. Elle est aussi sensible aux coûts car elle minimise le coût d'erreurs de mauvaise classification.

Dans notre travail de thèse, nous appliquons la technique de pondération LMS sur des différents ensembles de données médicales déséquilibrées (binaires et multi classes) dans le but d'examiner l'efficacité et de voir exactement l'intérêt de cette méthode d'équilibrage. Ce problème de déséquilibre est souvent rencontré dans le domaine de diagnostic médical. Dans les ensembles de données médicales, les pathologies atypiques et rares sont minoritaires par rapport aux cas normaux. Ce domaine est plus sensible, car si un classifieur fait une erreur sur les classes majoritaires c.-à-d. classer comme malade un individu sain (Faut Positive - FP) est coûteuse en termes d'examens inutiles et de stress pour le patient. Mais si ce calssifieur fait une erreur sur la classe minoritaire c.-à-d. ne pas détecter la maladie chez un patient ou d'une autre façon classer comme sain un individu malade (Faut Négative - FN) est bien plus grave, elle peut entraîner des complications. L'objective principale de notre travail consiste à créer un équilibre entre les classes par l'application de la technique de pondération LMS sur les différents ensembles de données dans le but de minimiser le taux d'erreur de FN afin d'obtenir une classification correcte de la classe minoritaire.

Plusieurs techniques de classification supervisée ont été proposées pour traiter le problème de classification des données déséquilibrées. Parmi les techniques les plus utilisées dernièrement sont les Réseaux de Neurones Artificiels (RNA), Support Vector Machines (SVM), K-Plus Proche Voisin (K-PPV). Ces techniques sont sélectionnées pour être appliquées dans nos différentes expérimentations, ce choix est justifié par la simplicité de leurs algorithmes d'apprentissage [6,12].

Contributions

Le but majeur de cette recherche est de créer une méthode plus efficace pour gérer et traiter le problème d'apprentissage des données déséquilibrées. L'objectif principale de cette thèse concerne en particulier l'application de la méthode d'équilibrage sur plusieurs algorithmes de classification sans modifier les classifieurs et de la tester sur des différentes ensembles de données binaires et multi classes sans modifier les ensembles. Les trois contributions principales proposées dans le cadre de sujet de doctorat sont :

- Dans la première contribution, nous étudions l'intérêt de la pondération des données minoritaires par l'algorithme de pondération LMS, et son impact sur l'amélioration à la performance de la classification avec les différents degrés de déséquilibre.

- Notre deuxième contribution concerne la pondération des données médicales en utilisant d'autres classifieurs. Dans lequel nous validons l'impact de la technique de pondération LMS sur les différentes techniques de classification (RNA, SVM, K-PPV) avec l'utilisation de plusieurs ensembles de données (binaires et multi classes), et nous comparons les performances obtenues par les classifieurs avant et après l'équilibrage.
- La troisième contribution montre l'intérêt de la méthode de pondération LMS adoptée par rapport à d'autres méthodes existantes dans la littérature. Nous comparons les performances obtenues par la méthode de pondération LMS avec les performances obtenues par les autres méthodes d'équilibrage les plus répondues dans la littérature c.-à-d. les méthodes d'échantillonnage (sous-échantillonnage, sur-échantillonnage et SMOTE (Synthetic Minority Oversampling Technique)).

Le travail présenté dans cette thèse de doctorat s'inscrit dans le contexte d'aide au diagnostic médical. Nous avons donné un intérêt plus particulier à l'apprentissage supervisé pour traiter le problème des classes déséquilibrées, dans le but d'améliorer les performances de la classification et de focaliser la classification vers les classes minoritaires qui sont plus importantes dans le traitement des données médicales.

Organisation du mémoire

Notre thèse de doctorat est organisée en quatre principaux chapitres en plus de l'introduction générale et la conclusion générale.

Le 1^{ier} chapitre est consacré à une présentation générale du problème des données déséquilibrées qui est devenu un problème majeur pour l'apprentissage automatique, nous présenterons une vue d'ensemble des domaines de données déséquilibrées et les différents problèmes liés à ce problème.

Le chapitre 2 est dédié aux différentes techniques de classification, nous citons en particulier les techniques adoptées dans notre travail : RNA, SVM et K-PPV. Après quelques définitions de base, nous présentons les principales approches existantes dans la littérature qui traitent le problème de déséquilibre.

Le chapitre 3 présente la théorie de l'algorithme LMS (Least Mean Square). Nous exposons quelques définitions de base du filtrage adaptatif où nous citerons le principe de base et quelques caractéristiques. Nous présentons ensuite l'algorithme le plus utilisé dans le filtrage adaptatif qui est l'algorithme LMS, où nous présentons le fondement théorique, quelques définitions de base et quelques caractéristiques.

Chapitre 4 présente nos différentes contributions et toutes les expérimentations que nous avons réalisé au cours de notre thèse, ce chapitre présente aussi les discussions des résultats obtenus et une comparaison avec l'état de l'art.

Une conclusion générale qui synthétise toutes les idées extraites de nos différentes expérimentations et présente aussi les pistes de recherche ouvertes à partir de ce travail de thèse de doctorat.

Chapitre 1 : Données déséquilibrées

I. Introduction

En général, les classes des données ne sont pas équilibrées et des fois la classe minoritaire qui est la classe d'intérêt. Souvent le déséquilibre dans la répartition des données, cause beaucoup de problèmes pendant la phase d'apprentissage des classifieurs. En effet, nous rencontrons souvent des données minoritaires qui sont mal représentées lors de la phase d'apprentissage. La plupart des algorithmes pénalisent d'autant plus la classe minoritaire que sa fréquence est faible, ce qui montre l'importance de l'équilibrage de données pour la conception d'un bon classifieur.

Notre chapitre est structuré comme suit. La section 2 présente les données déséquilibrées. La section suivante introduit une vue d'ensemble des domaines de données déséquilibrées. Nous présenterons ensuite les différents problèmes des données déséquilibrées, avec une première partie consacrée à la description des problèmes liés aux caractéristiques intrinsèques de données, et une seconde où nous décrivons les problèmes liés à l'asymétrie des coûts. Nous terminons ce chapitre par une conclusion.

II. Présentation des données déséquilibrées

Les données sont dites déséquilibrées lorsqu'au moins une classe est sous représentée par rapport aux autres. La classe d'intérêt est généralement la classe minoritaire, car dans la réalité les événements rares sont souvent ceux qui ont le plus d'intérêt. Le problème des données déséquilibrées est ainsi lié à celui des coûts de mauvaise classification asymétriques suivant la classe. Le déséquilibre des données peut se manifester dans des problèmes du monde réel tels que la détection de fraudes ou d'intrusions, le diagnostic médical, le suivi médical, la bio-informatique, la catégorisation de textes etc… En outre, la distribution des données de test peut être différente de celle de l'échantillon d'apprentissage et les performances de la classification peuvent être inconnues lors de l'apprentissage [1].

Si le problème du déséquilibre des données a été soulevé dans de nombreux travaux parmi lesquels on citera ceux issus de deux ateliers consacrés à ce sujet [13, 14], ainsi que d'un numéro spécial de SIGKDD [15], bon nombre de problèmes restent encore ouverts, les plus souvent rencontrés dans le traitement des données réelles. Afin de bien cerner le problème du déséquilibre

des classes, nous essayons de comprendre comment les données déséquilibrées sont considérées comme une problématique majeure pour les méthodes d'apprentissage automatique [1].

Selon l'étude de Provost [16], un classifieur conventionnel ne peut pas être utilisé avec satisfaction sur des données déséquilibrées, puisqu'il cherche avant tout à minimiser l'erreur de prédiction, et il aura donc tendance à catégoriser les nouvelles données dans la classe majoritaire [1].

Dans le cas d'un ensemble composé de 100 individus, dont 90 appartiennent à la classe majoritaire et 10 à la classe minoritaire. Prédire systématiquement la classe majoritaire donne un taux d'erreur de 10% seulement. Le résultat semble être satisfaisant alors que la performance du classifieur sur les exemples de la classe majoritaire compte nef fois plus que sa performance sur la classe minoritaire. De ce fait, l'utilisation de classifieurs construits par les algorithmes d'apprentissage standards sans ajuster le seuil de classification peut être considérée comme une erreur de lourde conséquence [1].

III. Domaines de données déséquilibrées

Plusieurs techniques de Data Mining ont été utilisées pour résoudre le problème de la classification des données déséquilibrées rencontré dans plusieurs domaines d'application.

III.1 Détection de la fraude

Sachant que la fraude de carte de crédit et la fraude cellulaire, est un problème coûteux pour de nombreuses organisations d'affaires. Aux États-Unis, la fraude cellulaire coûte à l'industrie des télécommunications des centaines de millions de dollars par an. Une méthode pour détecter la fraude est de vérifier les modifications suspectes dans le comportement de l'utilisateur. Le comportement d'achat de quelqu'un qui vole une carte de crédit est probablement différent de celui du propriétaire initial. Les entreprises tentent de détecter les fraudes en analysant différents modèles de consommation dans leurs ensembles de données de transaction. Cependant, dans leurs collections de transaction, il y a beaucoup d'utilisateurs plus légitimes que des exemples frauduleux [17], ce qui constitue un handicap majeur lors de la conception d'une détection de fraude.

III.2 Diagnostic médical

Les ensembles de données médicales stockent de grandes quantités de données sur les patients. Les techniques de fouille de données appliquées à ces ensembles de données permettent de découvrir les relations entre les données et les modèles cliniques et pathologiques, aussi elles aident à comprendre la progression et les caractéristiques de certaines maladies. La connaissance extraite peut être utilisée pour le diagnostic précoce. C'est un facteur important pour sauver la vie d'un patient. Dans les ensembles de données médicales, en général les pathologies atypiques et rares sont minoritaires par rapport aux cas normaux [17]. Ce phénomène représente une difficulté majeure lors de la conception des systèmes de diagnostic.

III.3 Détection d'intrusion

Les systèmes informatiques basés sur le réseau sont d'un grand besoin pour la société moderne, les attaques contre ces systèmes informatiques et ces réseaux informatiques sont devenus des cas fréquents. L'apprentissage des règles de prédiction à partir des données du réseau est une approche de détection d'anomalie efficace pour automatiser et simplifier le développement manuel de signatures d'intrusions. Différents types d'attaques réseau sont présents - certaines sont fréquentes, d'autres rares. À titre d'exemple, les données de la compétition KDD-CUP'99 contient quatre catégories d'attaques réseau : denial-of-service (dos), surveillance (probe), remote-to-local (r2l), et user-to-root (u2r). Parmi ces 4 types d'attaques, les catégories u2r et r2l sont intrinsèquement rare [17]. Ces deux dernières peuvent constituer une difficulté pour les identifier.

III.4 Détection des déversements de pétrole à partir d'images radar de la surface de l'océan

Seulement environ 10% des déversements de pétrole proviennent de sources naturelles, telles que les fuites de fonds marins. La plus répandue est la pollution causée intentionnellement par les navires qui veulent disposer les résidus d'huile dans leurs réservoirs. Les images radar satellitaires permettent de surveiller les eaux côtières. Un système de détection des déversements de pétrole à partir d'images satellitaires pourrait être un système efficace d'alerte précoce, et peut-être un effet dissuasif sur les déversements illégaux, et pourrait avoir un impact significatif sur l'environnement.

Bien que les satellites présentent continuellement des images, mais les images contenant des déversements de pétrole sont minoritaires que ceux sans déversements de pétrole [17].

III.5 Les établissements de fabrication moderne

Dans une usine de fabrication moderne, comme une ligne d'assemblage de Boeing, les processus sont de plus en plus contractés par des cellules automatisés ou semi-automatisés, une alarme sera déclenché à chaque détection d'anomalie. Par contre le nombre de cas défectueux signalés est nettement inférieur à celui des procédures ordinaires [17]. Ils constituent un ensemble de données minoritaires à identifier correctement.

III.6 Gestion des risques

Chaque année, l'industrie des télécommunications est en face de plusieurs milliards de dollars de dettes irrécouvrables. Par conséquent, le contrôle des factures irrécouvrables est un problème important dans l'industrie. Une solution consiste à utiliser de grandes quantités de données historiques pour construire des modèles d'évaluation des risques par client ou par la base des transaction, afin de soutenir les politiques de gestion des risques qui réduisent le niveau de la dette irrécouvrable. Dans un ensemble de données contenant des milliers de dossier client, les clients qui ne payent pas représentent une minorité de l'ensemble [17]. Ceci constitue une difficulté supplémentaire pour les identifier.

En plus de ces exemples, ils existent d'autres applications comme la classification de textes et la commercialisation directe. Certaines de ces applications, telles que la détection de la fraude, la détection d'intrusion, le diagnostic médical, etc., sont également reconnus comme des problèmes de détection d'anomalies. L'objectif principal dans la détection d'anomalies est de trouver des objets qui sont différents de la majorité existante. Sachent que les objets anormaux et normaux constituent deux classes distinctes, une partie non négligeable des systèmes de détection d'anomalies considère la détection d'anomalies comme un problème de partitionnement de données dichotomiques dans laquelle des échantillons de données sont classés soit comme anormales ou normales. En général les anomalies sont minoritaires par rapport aux observations normales, le problème de déséquilibre de classes est fortement lié aux applications de détection d'anomalie [17].

IV. Les différents problèmes des données déséquilibrées

Nous présentons dans cette partie les différents problèmes liés avec des données déséquilibrées :

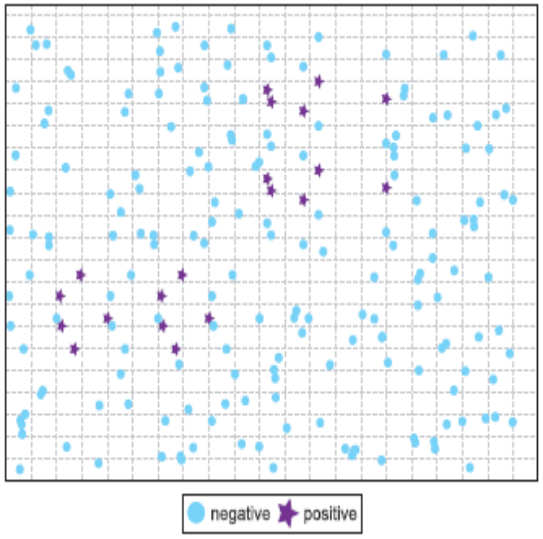
IV.1 Rôles des caractéristiques intrinsèques dans la classification des données déséquilibrées

Nous soulevons dans cette section le problème de déséquilibre en focalisant notre étude sur plusieurs caractéristiques intrinsèques de données qui ont une forte influence sur la classification des données déséquilibrées [18].

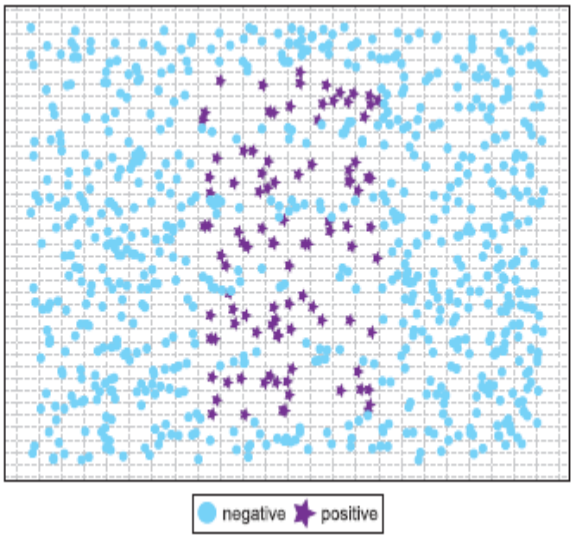
IV.1.1. Petits disjoints

La présence de classes déséquilibrées est étroitement liée au problème des petits disjoints. Cette situation concerne les concepts représentés au sein de petits groupes, ils sont considérés comme un résultat direct des sous-concepts sous-représentés [19, 20]. Bien que ces petits disjoints soient implicites dans la plupart des problèmes, l'existence de ce type de phénomène augmente fortement la complexité du problème dans le cas des classes déséquilibrées, car il devient difficile de savoir si ces exemples représentent un sous-concept réel ou simplement un bruit [21].

Cette situation est représentée sur la Figure 1.1, où nous exposons un ensemble de données générées artificiellement avec de petits disjoints pour la classe minoritaire et le problème «Subclus» créée dans [22]. Où nous trouvons des petits disjoints pour les deux classes : les échantillons négatifs sont sous-représentés par rapport aux échantillons positifs dans la région centrale de zones rectangulaires positives, tandis que les échantillons positifs ne couvrent qu'une petite partie de l'ensemble des données et ils sont placés à l'intérieur de la classe négative. Nous devons souligner que, dans toutes les figures de cette partie, les cas positifs sont représentés avec des étoiles sombres alors que les instances négatives à identifier sur la figure avec des cercles de lumière.



(a) Ensemble de données artificiels : petits disjoints pour la classe minoritaire



(b) Ensemble de données 'Subclus' : petits disjoints pour les deux classes

Figure 1. 1. Exemple de petits disjoints sur les données déséquilibrées.

Le problème des petits disjoints est traité en générale par les algorithmes de classification fondés sur le concept "diviser et conquérir" [23]. Cette approche consiste à diviser le problème d'origine en petites parties, comme le cas des arbres de décision [24], ou nous obtenons des partitions de classes de données. Ce traitement sera difficile dans le cas d'un degré de déséquilibre des données élevé.

Plusieurs techniques pour traiter le problème de petits disjoints ont été citées dans les travaux de Weiss [25, 26] :

1. Obtenir des données d'apprentissage supplémentaire. Le manque de données peut induire l'apparition de petits disjoints, en particulier dans la classe minoritaire, et ces zones peuvent être mieux couvertes tout en utilisant un plan d'échantillonnage clair et riche [27].
2. Utilisez un biais inductif plus approprié et plus convenable pour assurer une détection correcte des petits disjoints. Nous devons utiliser des mécanismes plus poussés. A titre d'exemple, les auteurs dans [28] ont modifié CN2[1] où le biais de généralité maximale est utilisé uniquement pour les grands disjoints, et le biais de spécificité maximale a été ensuite utilisé pour les petits disjoints. Cependant, cette approche risque de dégrader également les

[1] Méthode Clark and Niblett (CN2) introduit en 1987.

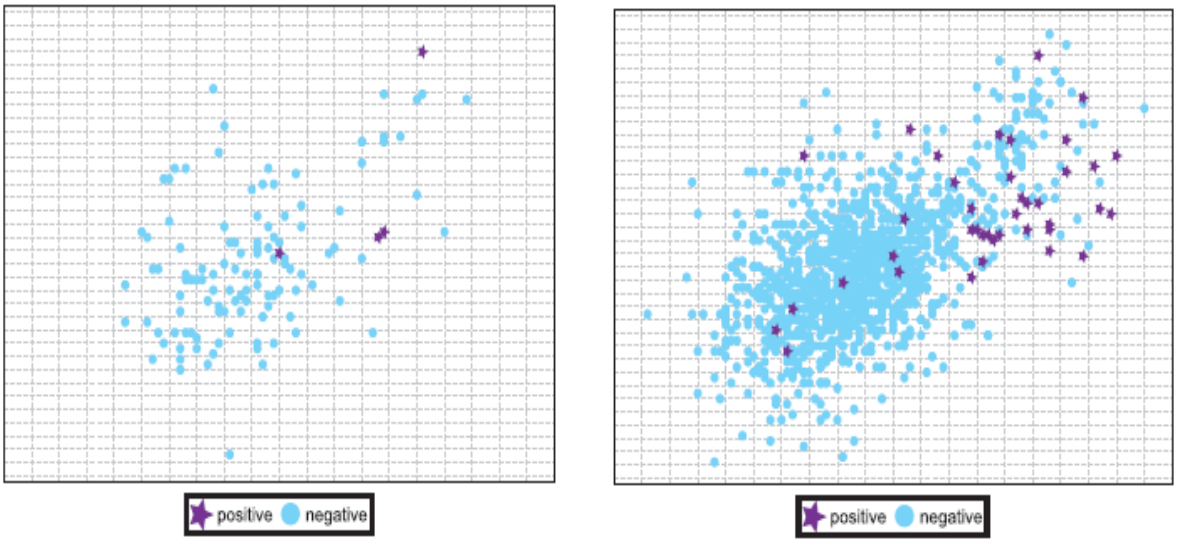
performances des petits disjoints, et certains auteurs ont proposé d'affiner la recherche et d'utiliser différents types d'apprentissage à partir des exemples séparés (grands disjoints et petits disjoints) [29, 30].

3. Utilisation des paramètres plus appropriés. Cette situation est liée au problème précédent qui concerne le processus d'extraction de données, il est recommandé d'utiliser des mesures spécifiques pour les données déséquilibrées, de telle sorte que les classes minoritaires dans les petits disjoints sont positivement pondérées lors de la conception du modèle de classification. Par exemple, l'utilisation respective de la précision et du rappel pour les classes minoritaires et majoritaires peut générer des règles plus précises pour la classe positive [31].
4. La désactivation de l'élagage (*Disabling pruning*). L'élagage a tendance à éliminer la plupart des petits disjoints par une généralisation des règles obtenues. Cette méthode est en général écartée de l'utilisation [18].
5. L'utilisation du boosting. Les algorithmes de boosting comme l'algorithme AdaBoost, sont des algorithmes itératifs qui affectent des différent poids aux données de l'apprentissage à chaque itération [32]. Après chaque itération, l'algorithme boosting augmente les poids associés aux exemples incorrectement classés et diminue les poids associés aux exemples correctement classés. Sachant que les exemples dans les petits disjoints sont difficiles à prédire, logiquement l'algorithme de boosting va améliorer leur performance de classification. Suivant cette idée, de nombreuses approches ont été développées en modifiant les paramètres de l'algorithme de boosting pour la mise à jour de poids de données afin d'améliorer la performance de la classe minoritaire et les petits disjoints [18].

IV.1.2. Le manque de densité

Un problème qui peut survenir lors de la classification est la petite taille l'ensemble d'apprentissage [33]. Cette question est liée généralement au manque d'information, où les algorithmes d'induction ne disposent pas suffisamment de données pour faire des généralisations sur la distribution d'échantillons, une situation qui devient plus difficile en présence de données de haute dimension ou déséquilibrées. Une représentation visuelle de ce problème est illustrée sur la Figure 1.2, où nous présentons un diagramme de dispersion pour les données d'apprentissage du problème de l'ensemble yeast4 (attributs mcg vs. gvh) avec seulement un taux de 10% de l'ensemble des données originales (Figure 1.2a) et avec l'ensemble des données (Figure 1.2b).

Nous remarquons qu'il devient très difficile pour l'algorithme d'apprentissage d'obtenir un modèle qui est capable d'effectuer une bonne généralisation lorsqu'il n'y a pas assez de données qui représente tous les cas possible du problème, où le cas le plus défavorable concerne les données qui sont considérées comme bruit.



(a) 10% Instances d'apprentissage *(b) 100% Instances d'apprentissage*

Figure 1. 2. Manque de densité ou la petite taille d'ensemble d'apprentissage sur la base yeast4.

Le problème lié aux traitements et la prise en charge des données minoritaires des données déséquilibrées constitue aujourd'hui un nouveau défi à la communauté des chercheurs dans le domaine [34]. Dans ce cadre, la classe minoritaire peut être mal représentée et les connaissances extraites de ces données deviennent trop spécifiques, conduisant à un sur-apprentissage. En outre, comme nous avons indiqué dans la section précédente, le manque de densité dans l'ensemble d'apprentissage peut provoquer l'introduction de petits disjoints. Deux ensembles de données ayant le même ratio de déséquilibre (IR : Imbalance Ratio) ne présentent pas obligatoirement le même degré de complexité, il est important aussi de voir comment les données minoritaires sont représentées dans un ensemble d'apprentissage.

Dans [30], les auteurs ont analysé l'effet de la distribution des classes et la taille de l'ensemble d'apprentissage sur la performance de classifieur en utilisant l'algorithme d'apprentissage C4.5. Cette analyse propose la variation simultanée des données d'apprentissage

disponibles et le degré de déséquilibre dans une variété d'ensembles de données et ils ont calculé les différentes métriques de l'AUC (Area Under the ROC Curve) obtenues pour chaque cas.

La première remarque tirée de ces expérimentations, est que plus le nombre de données d'apprentissage est élevé et plus les performances sont indépendants de la distribution de la classe. La dernière remarque importante est que le rapport de déséquilibre donne les meilleures performances. Ces dernières varient d'un ensemble de données à une autre, ce qui montre clairement l'intérêt d'une meilleure distribution des données en tenant compte des classes minoritaires lors de la phase d'apprentissage. Un algorithme d'échantillonnage progressif peut être utile pour fixer la meilleure distribution des classes. Cela va permettre d'améliorer les performances des classifieurs.

IV.1.3. Chevauchement entre les classes

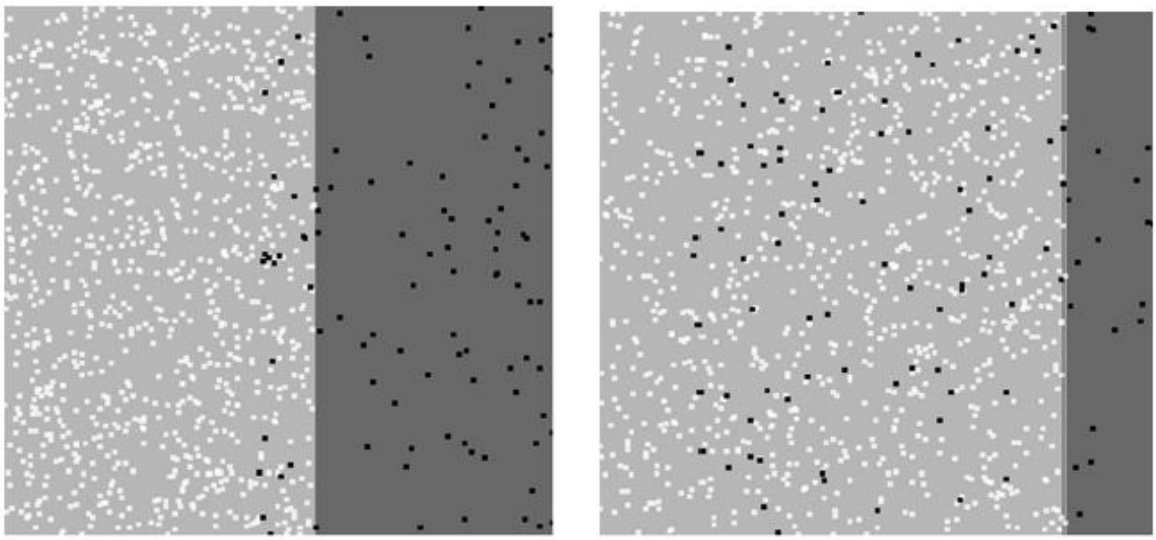
Le problème du chevauchement entre les classes concerne une région de l'espace de données avec une quantité égale de données d'apprentissage pour chaque classe. Dans ce cas nous obtenons une inférence avec pratiquement les mêmes probabilités a priori dans cette zone de chevauchement, Ce qui rend très difficile, voire impossible de faire la distinction entre les deux classes. En effet, tout problème « linéairement séparable » peut être résolu par un classifieur simple indépendant de la répartition de la classe.

Ils existent plusieurs travaux dans la littérature qui traitent le problème lié avec le chevauchement et le déséquilibre de classes. En particulier, dans [35], nous trouvons une étude où les auteurs proposent plusieurs expérimentations avec les ensembles de données synthétiques qui modifient le rapport de déséquilibre et le chevauchement existant entre les deux classes. Les résultats obtenus ont confirmé que les probabilités de classes ne sont pas le principal facteur qui diminue la performance de classification, mais plutôt le degré de chevauchement entre les classes qui peut être responsable.

Pour expliquer cette situation, ils ont créé un ensemble de données artificielles avec 1.000 exemples ayant un rapport de déséquilibre (IR) égale à 9, c'est à dire une instance positive pour 10 cas anormaux. Ensuite, ils ont varié progressivement le degré de chevauchement pour les valeurs

de caractéristiques individuelles jusqu'à atteindre 100% de chevauchement, et ils ont utilisé le classifieur C4.5 afin de montrer l'influence de chevauchement par rapport à un IR fixe.

La Figure 1.3 explique ce problème, nous observons que l'arbre de décision est non seulement incapable d'obtenir une discrimination correcte entre les deux classes quand ils se chevauchent, mais aussi que la classe préférée est celle de la majorité, conduisant à des performances faibles en utilisant la métrique AUC.



(a) 20% de chevauchement *(b) 80% de chevauchement*

Figure 1. 3. Exemple de chevauchement des ensembles de données déséquilibrées : frontières détectés par C4.5.

Aussi dans [36], une étude similaire avec plusieurs algorithmes dans des situations différentes de déséquilibre et de chevauchement a été présentée, ils ont développé principalement l'algorithme de K- Plus Proche Voisin (K-PPV). Dans ce cas, les auteurs ont proposé deux scénarios différents : d'une part, ils cherchent de trouver la relation lorsque le rapport de déséquilibre dans la zone de chevauchement est similaire au ratio global de déséquilibre alors que, d'autre part, ils cherchent la relation lorsque le rapport de déséquilibre dans la zone de chevauchement dans le cas inverse i.e. la classe positive est localement plus dense que la classe négative dans la région de chevauchement. Ils ont montré que lorsque le chevauchement des données n'est pas équilibré, le rapport de déséquilibre dans la partie de chevauchement peut être plus important que le taux de chevauchement. En outre, les classifieurs utilisant une procédure d'apprentissage globale ont

obtenu de bon taux de VP (Vrai Positive) alors que les modèles d'apprentissage en local obtiennent de bon taux de VN (Vrai Négative) en comparant aux précédents classifieurs.

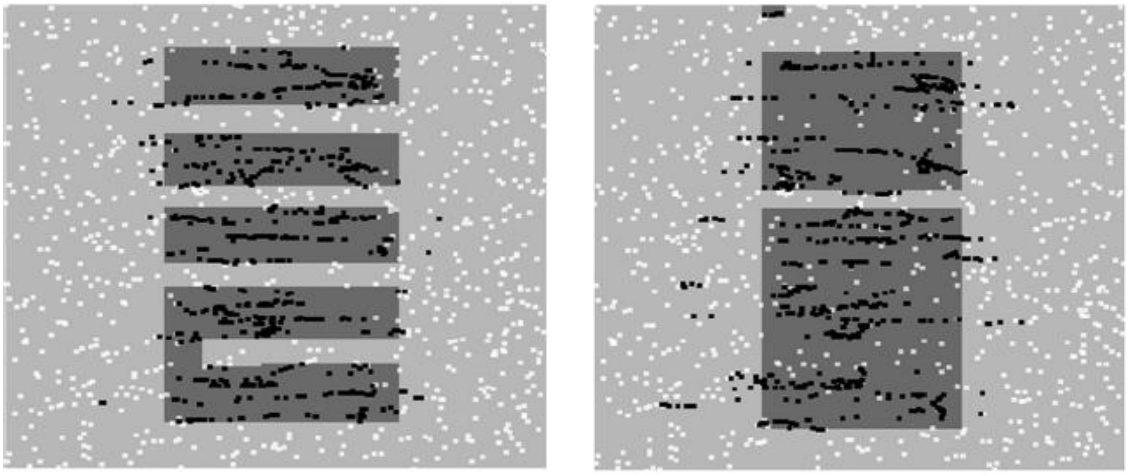
Dans [37], les auteurs ont examiné les effets de chevauchement et de déséquilibre sur la complexité du modèle appris, et ils ont montré que le chevauchement est un facteur beaucoup plus grave que le déséquilibre dans cette situation. Ils ont utilisé des ensembles de données synthétiques et un classifieur de type SVM (*Support Vector Machines*), ils ont varié le rapport de déséquilibre et le chevauchement entre les classes. Leurs résultats obtenus montrent que lorsque la taille de l'ensemble d'apprentissage est faible, un taux élevé de déséquilibre provoque une baisse importante des performances de classifieur, ce qui est expliqué par la présence de petits disjoints. Les classes que se chevauchent provoquent une baisse constante dans la performance quelle que soit la taille de l'ensemble d'apprentissage. Cependant, avec la présence simultanée de chevauchement et de déséquilibre, le rendement du classifieur est dégradé de manière significative au-delà de toute prévision.

Récemment dans [38], les auteurs ont tiré des connaissances intéressantes à partir des ensembles de données standards. Ils ont représenté les performances des différentes bases de données ordonnées selon les différentes mesures de la complexité des données afin de trouver quelques régions de certaines caractéristiques intéressantes, ils n'ont pas pu tirer un phénomène intéressant lié à IR, mais par contre ils ont tiré quelques indicateurs qui caractérisent le chevauchement entre les classes.

Une autre approche a été présentée dans [39], les auteurs ont combiné dans l'ordre le prétraitement et la sélection d'attributs. La phase de prétraitement traite le cas de la répartition des classes et les petits disjoints, la sélection d'attributs permet de réduire le degré de chevauchement. D'une manière générale, l'idée derrière de cette approche est de surmonter les différentes sources de la complexité des données telles que le chevauchement de classe, caractéristiques non pertinentes et redondantes, des échantillons bruités, le déséquilibre de la classe, faibles ratios de la taille de l'échantillon à dimensionnalité etc…, en faisant appel aux différentes approches existantes pour répondre à chaque problème à part.

IV.1.4. Cas des données bruitées

En général les données bruitées affectent un modèle de classification ou d'extraction de connaissances à base d'apprentissage [40, 41], ces types de données sont à l'origine des problèmes rencontrés par les systèmes de Data Mining. Dans le cas des données déséquilibrées, la présence de bruit a un impact important sur les classes minoritaires que sur les classes majoritaires [23] ; sachant que la classe positive a peu d'exemples pour commencer, il faudra donc peu d'exemples bruités à l'issue du sous-concept appris. Ce problème est présenté dans la Figure 1.4, dans lequel nous observons les frontières de décision obtenus avec SMOTE+C4.5 (*Synthetic Minority Oversampling Technique + C4.5*) dans le problème 'Subclus' sans données bruitées (Figure 1.4a) et comment les frontières entre les classes sont à générer aléatoirement par l'introduction d'un bruit gaussien de 20% (Figure 1.4b).



(a) Problème d'origine et les fonctions de décision

(b) Les instances bruitées et les nouvelles fonctions de décision

Figure 1. 4. Exemple de l'effet du bruit dans les ensembles de données déséquilibrées pour SMOTE+ C4.5 dans l'ensemble de données Subclus.

Selon les travaux de [23], ces zones bruitées peuvent être en quelque sorte considérées comme de petites disjointes et afin d'éviter l'extraction erronée des modèles de discrimination pour ces exemples, certaines techniques de gestion de sur-apprentissage doivent être employées, en particulier l'élagage. Cependant, le handicap de cette approche est que certaines classes minoritaires correctes seront ignorées, et de cette façon, le biais de l'apprentissage doit être réglé

sur place afin d'être en mesure de fournir un comportement global correct pour les deux classes du problème.

Par exemple, Batuwita et Palade ont développé l'algorithme FSVM-CIL (*Fuzzy Support Vector Machines for Class Imbalance Learning*) [42], une combinaison entre les SVMs et la logique floue visant à montrer l'importance au sein de la classe de différents exemples d'apprentissage afin de supprimer l'effet des valeurs aberrantes et du bruit. L'idée est d'attribuer des différentes valeurs d'appartenance floue pour les exemples positifs et négatifs et d'intégrer ces informations dans l'algorithme d'apprentissage de SVM, visant à réduire l'effet des valeurs aberrantes et le bruit lors de la recherche de l'hyperplan de séparation.

Dans [43] nous trouvons une étude empirique sur l'effet du déséquilibre des classes et le bruit sur les différents algorithmes de classification et les techniques d'échantillonnage de données. Les auteurs ont tiré trois informations importantes :

1. Les algorithmes de classification sont plus sensibles au bruit qu'au problème de déséquilibre. Cependant, comme de déséquilibre participe à la dégradation des performances alors il joue un rôle plus important dans la performance des classifieurs et les techniques d'échantillonnage en général.
2. Concernent les méthodes de prétraitement, des techniques simples de sous-échantillonnage tel que sous-échantillonnage aléatoire et ENN (*Edited Nearest Neighbor*) permettent de diminuer l'effet du bruit et du déséquilibre. Lorsque le niveau de déséquilibre est grand, l'algorithme ENN a prouvé sa robustesse en présence de bruit. En outre, l'algorithme OSS (*One-Sided Selection*) est relativement peu affecté lors d'une augmentation du niveau de bruit. D'autres techniques comme le sur-échantillonnage aléatoire, SMOTE et Borderline-SMOTE ont obtenu des bons résultats en moyenne, mais elles sont moins performances que l'algorithme de sous-échantillonnage.
3. Les classifieurs les plus robustes testés sur les données déséquilibrées et bruitées sont les classifieurs à base des réseaux bayésiens et des SVMs, ils donnent des résultats meilleurs par rapport aux algorithmes d'induction de règles. Hors la plupart des algorithmes montrent seulement de petits changements dans AUC lorsque le déséquilibre a été augmenté, par contre les performances des fonctions à base radiale (*Radial Basis Functions (RBF))* sont dégradées d'une manière significative lorsque le rapport de déséquilibre augmente. La

présence du bruit dégrade plus les performances des algorithmes d'extraction de règles en comparaison à d'autres algorithmes existants dans la littérature.

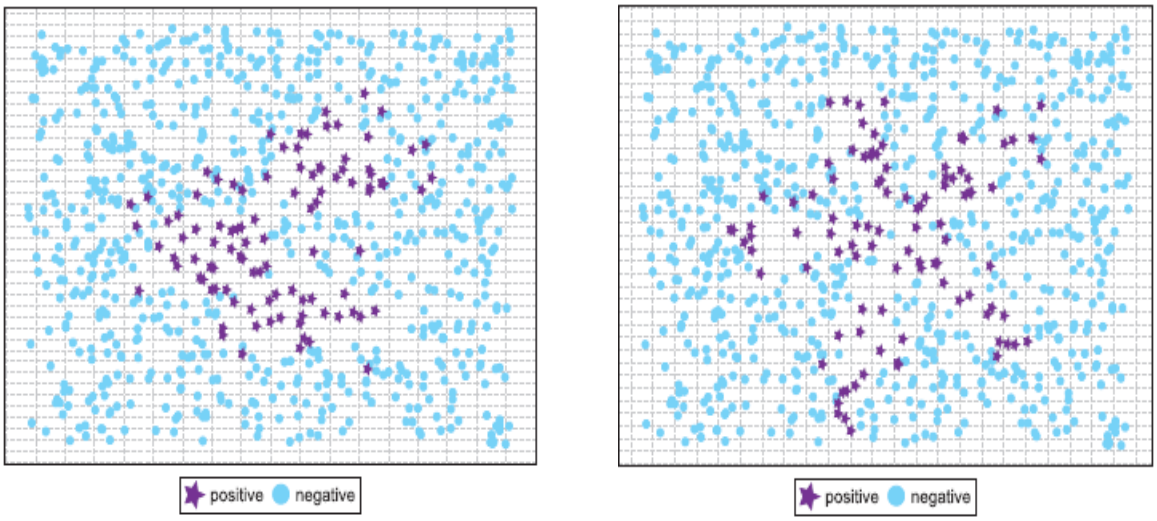
Les auteurs dans [44] ont présenté une étude similaire sur l'importance des données bruitées et déséquilibrées en utilisant les techniques de bagging et de boosting. Leurs résultats montrent l'intérêt de l'approche bagging, les auteurs ont proposé l'utilisation de techniques de réduction de bruit avant l'application des procédures de boosting.

Les auteurs dans [22] ont montré aussi une étude expérimentale, les effets du bruit sur un problème de déséquilibre spécifique tel que l'ensemble de données 'Subclus'.

IV.1.5. Cas des exemples de frontière

D'après les travaux de [45], nous distinguons une zone de sécurité entre les classes, des exemples bruités et frontières. Via les exemples bruités, nous connaissons les individus d'une classe situés dans des zones de sécurité de l'autre classe. Enfin, des exemples de frontières sont situés dans les frontières de classes de la zone environnante, où les classes minoritaires et majoritaires se chevauchent.

La Figue 1.5 représente deux exemples donnés par [22], nommé 'Paw' et 'Clover' respectivement. Dans le premier cas, la classe minoritaire est décomposé en 3 sous-régions elliptiques, où deux d'entre eux sont situés à proximité les uns des autres, et le reste sous-région plus petite est séparée (pôle supérieur droit). Ce dernier représente également un cadre non-linéaire, où la classe minoritaire ressemble à une fleur avec des pétales elliptiques, ce qui rend difficile la détermination les exemples de frontière afin d'effectuer une bonne discrimination des classes.



(a) Ensemble de données 'Paw' *(b) Ensemble de données 'Clover'*

Figure 1. 5. Exemple de données avec des exemples de frontière difficiles.

Le problème des données bruitées et la gestion des exemples de frontière sont étroitement liés, et la plupart des techniques de nettoyage peut être utilisé pour détecter et marquer ces instances de frontière, il est important de les différencier des instances bruitées, ces dernières peuvent dégrader la classification en général. Il est intéressant de définir des zones de frontières d'une manière précise lors de la discrimination entre les classes positives et négatives [46].

D'autres techniques connexes telles que l'algorithme Borderline-SMOTE [47] existant dans la littérature et qui vise à sur échantillonner les instances de la classe minoritaire dans les zones de frontière, nous citons la base d'exemples de 'Danger', c.-à-d. des exemples qui sont les plus susceptibles d'être mal classés, car ils apparaissent dans les zones de frontières, à partir de laquelle SMOTE génère des échantillons minoritaires synthétiques dans la zone des frontières.

D'autres approches comme Safe-Level-SMOTE [48] et ADASYN (*Adaptive Synthetic sampling approach*) [49] fonctionnent de manière similaire. La première approche est basée sur le même principe des approches précédentes, comme SMOTE et Borderline-SMOTE, elles peuvent générer des instances synthétiques dans des zones inappropriés, tels que les régions de chevauchement ; par conséquent, les auteurs calculent la valeur de 'niveau de sécurité' pour chaque instance positif avant de générer des instances synthétiques et les rapprocher au plus grand niveau de sécurité. D'autre part, l'idée principale de l'algorithme ADASYN consiste à utiliser une distribution de densité comme un critère de décision automatique du nombre d'échantillons

synthétiques, et qui doivent être générés pour chaque exemple minoritaire, cela en changeant de façon adaptative les poids des différents exemples minoritaires pour compenser les distributions asymétriques.

Dans [22], les auteurs ont présenté une série d'expérimentations où ils montrent que la dégradation de la performance d'un classifieur est fortement liée avec le nombre d'exemples de frontière. Ils ont montré que les mécanismes de ré-échantillonnage ciblées fonctionnent bien lorsque le nombre d'exemples de frontière est assez grand, alors que dans le cas contraire, les méthodes de sur-échantillonnage permettent d'améliorer la précision de la classe minoritaire.

Les auteurs dans [50] utilisent une approche d'apprentissage à base des règles floues hiérarchiques, qui définissent une granularité plus élevé pour ces sous-espaces de problème dans les zones de frontière. Les résultats ont montré l'intérêt de l'approche lorsque les ensembles de données sont très déséquilibrées.

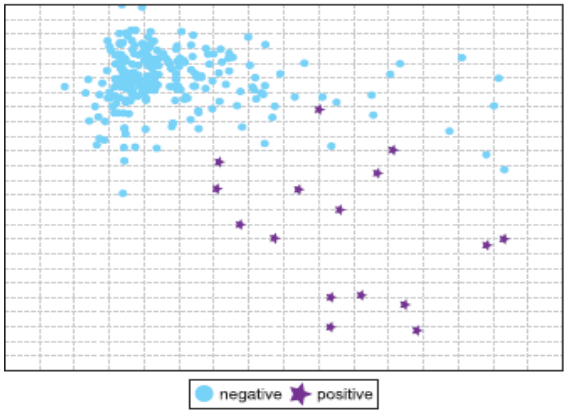
IV.1.6. Changement d'ensemble de données

Le problème de changement de l'ensemble de données [51, 52, 53] est défini comme un phénomène où les données d'apprentissage et de test sont distribuées différemment. Ceci est un problème commun qui peut affecter tous les types de problèmes de classification, il est souvent lié aux problèmes de biais de sélection échantillonné. Un faible taux de changement du jeu de données est présent dans la plupart des problèmes de données réels, mais en général les classifieurs sont souvent capables de gérer ce problème sans perte de performance importante.

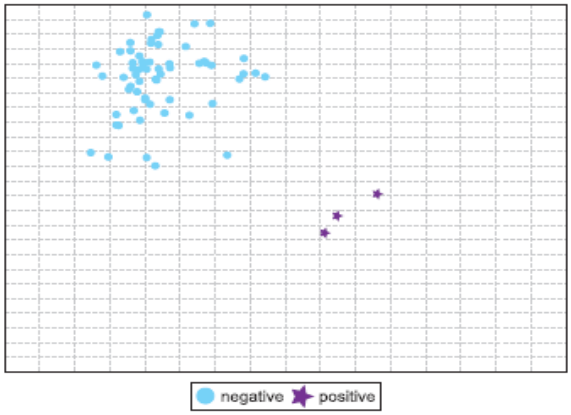
Cependant, le problème de changement du jeu de données est particulièrement pertinent lors de la classification des données déséquilibrées parce que dans ce cas la classe minoritaire est particulièrement sensible aux erreurs de classification singulières, en raison de nombre faible d'exemples présentés [54]. Dans les cas les plus extrêmes, un seul exemple mal classé de la classe minoritaire peut créer une dégradation significative de la performance.

Les figures 1.6 et 1.7 présentent deux exemples qui montrent l'influence du changement de jeu de données dans la classification déséquilibrée. Dans le premier cas (Figure 1.6), il est facile de voir une séparation similaire entre les classes dans l'ensemble d'apprentissage et l'ensemble de test. Toutefois, dans le deuxième cas (Figure 1.7), nous remarquons quelques exemples de la classe

minoritaire dans l'ensemble de test, qui se trouvent dans les zones les plus à droite et en bas alors qu'ils sont localisés dans d'autres zones dans l'ensemble d'apprentissage, conduisant à un écart entre les performances d'apprentissage et de test. Ces problèmes sont représentés dans un espace à deux dimensions au moyen d'une transformation linéaire des variables d'entrée, selon la technique présentée dans [54].

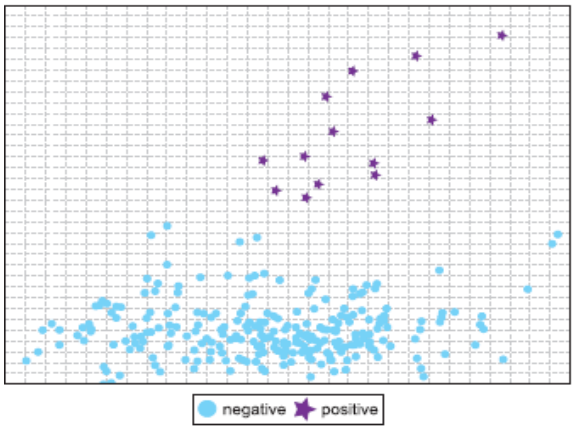


(a) Données d'apprentissage. AUC = 0.9043

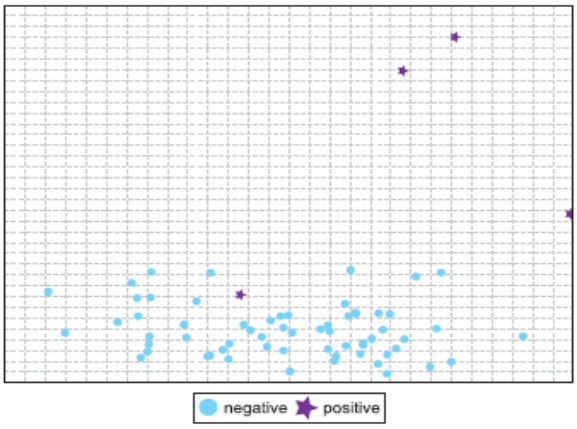


(b) Données de test. AUC =1.000

Figure 1. 6. Exemple de bonne conduite (pas de changement de données) dans des zones déséquilibrées : ensemble de données 'ecoli4', 5ème partition.



(a) Données d'apprentissage. AUC = 1.000



(b) Données de test. AUC =0.8750

Figure 1. 7. Exemple de mauvais comportement causés par le changement du jeu de données dans les zones déséquilibrées : ensemble de données 'ecoli4', 1ère partition.

Depuis le changement de jeu de données est une question très pertinente dans la classification des données déséquilibrées, il est facile de voir pourquoi il serait une perspective de recherche

intéressante. Ils existent deux approches différentes lors de l'étude de changement du jeu de données dans le domaine des données déséquilibrées :

1. La première approche se focalise sur le changement de données intrinsèques, qui sont les données d'intérêt, comprennent un certain degré de changement qui produit une dégradation importante des performances. Dans ce cas, nous développons des techniques à découvrir et à mesurer la présence du changement de données [55], mais en les concentrant uniquement sur la classe minoritaire. En outre, nous pouvons concevoir des algorithmes qui sont capables de travailler dans des conditions de changement de jeu de données, soit par le biais de techniques de prétraitement [56] ou avec des algorithmes ad hoc [57, 58]. Dans les deux cas, nous ne connaissons pas les techniques existantes dans la littérature qui traitent le problème de la classification déséquilibré en présence de changement de données.
2. La deuxième approche qui concerne le changement de données dans la classification des données déséquilibrées est liée au changement de jeu de données induites. La plupart des travaux de l'état d'art actuel est validée par des techniques de validation croisée stratifié, qui représentent une autre source potentielle de changement dans le processus d'apprentissage. Une technique de validation plus adapté doit être développée afin d'éviter l'introduction artificielle de problèmes de décalage de données.

IV.2 Asymétrie des coûts

L'apprentissage sensible aux coûts répond à une problématique un peu différente que l'apprentissage sur données déséquilibrées, mais les deux sont très liés. L'objectif est de prendre en compte l'asymétrie des classes en termes d'importance, ou de coûts des erreurs. Ainsi dans le cas d'aide au diagnostic médical, malgré qu'une erreur sur la classe majoritaire (classer comme malade un individu sain) est coûteuse en termes d'examens inutiles et de stress pour le patient, par contre faire une erreur sur la classe minoritaire (ne pas détecter la maladie chez un patient) est bien plus grave : elle peut entraîner des complications, voir le décès de la personne. Cette asymétrie des coûts n'est pas prise en compte par les modèles d'apprentissage de base. De plus les coûts ne sont pas précisément connus avec précision [59].

Ce problème est lié à celui du déséquilibre, car bien souvent les classes rares sont les plus importantes, et les erreurs sur ces dernières sont plus coûteuses. Nous verrons que les problèmes comme les méthodes de l'apprentissage sensible aux coûts (*Cost-sensitive learning*) peuvent être adaptées dans le cadre de l'apprentissage sur données déséquilibrées, comme le préconise par exemple Maloof [60] en observant que l'échantillonnage, l'ajustement d'une matrice de coûts et le déplacement du seuil de décision ont des effets similaires [59].

Ils existent différents types d'asymétries des coûts en apprentissage, que Turney [61] a regroupé en dix catégories. Le type de coûts le plus traité dans la littérature concerne les coûts de mauvaises classifications, mais d'autre méthodes existent pour tenir compte des coûts de tests (coût d'acquérir la valeur d'une variable pour un individu [62]), ainsi que du couplage des deux [59].

IV.2.1. Autres problèmes connexes

Nous citons quelques problèmes qui peuvent être mis en parallèle aux problèmes de l'asymétrie [59] :

– La variabilité intra-classe peut perturber l'apprentissage de la classe minoritaire si le concept est scindé en plusieurs sous-espaces [63]. Ce problème est à mettre en relation avec celui des cas rares.

– Le chevauchement des classes est également problématique [64]. La marge d'induction devra être appropriée.

– La duplication des données peut amplifier le problème du déséquilibre [65]. Ces problèmes peuvent devenir catastrophiques pour l'apprentissage lorsqu'ils sont en conjonction avec le déséquilibre. Dans de nombreux cas ils sont même plus perturbants que le déséquilibre lui-même [64, 66]. Les études de Japkowicz essaient d'étudier les problèmes qui rendent les modèles plus ou moins perturbés par le déséquilibre des classes [67, 68, 69].

V. Conclusion

Dans ce chapitre, nous sommes intéressés aux différents types de données déséquilibrées, cette problématique est souvent posée dans les données réelles. Ce phénomène est devenu un problème majeur en apprentissage automatique et en Data Mining. Sachant que la majorité des algorithmes ne s'adaptent pas avec la classe minoritaire, même si elle est porteuse d'une information pertinente. Dans ce contexte, plusieurs chercheurs se sont intéressés à ce genre de problème et plusieurs travaux se sont proposés dans la littérature.

Dans le chapitre suivant, nous nous intéressons aux différents travaux dans l'état d'art sur les différentes méthodes proposées pour surmonter ce problème de déséquilibre.

Chapitre 2 : Les techniques de classification et Les méthodes d’équilibrage

I. Introduction

La classification des données déséquilibrées nécessite des stratégies bien spécifiques pour obtenir une classification correcte de la classe minoritaire qui a été mal représenté lors de la phase d'apprentissage. La première section de ce chapitre présente les différentes techniques de classification des données déséquilibrées (Réseaux de Neurones Artificiels (RNA), Support Vector Machine (SVM) et K-Plus Proche Voisin (K-PPV). La deuxième section concerne les principales approches proposées dans la littérature pour traiter le problème de déséquilibre des données. Nous expliquons tout d'abord les méthodes de modification des algorithmes de classification, puis les méthodes de pondération (stratégies d'échantillonnage), ensuite les méthodes sensibles aux coûts. Nous terminons ce chapitre par une conclusion.

II. Techniques de classification des données déséquilibrées

II.1 Présentation du problème

L'approche par classification est très sollicitée par le domaine d'analyse des données (Data Mining). Elle consiste à regrouper des ensembles d'exemples similaires en classes selon certains critères. Chaque groupe est composé d'objets similaires et qui sont dissimilaires avec les objets des autres groupes où chaque groupe étant bien différencié des autres. Le processus de classification se déroule en trois phases ; la première concerne la phase d'apprentissage qui permet de constituer l'ensemble de données utilisées pour générer le modèle d'apprentissage. Alors que, la phase de test permet de constitué les données sur lesquels sera appliqué le modèle d'apprentissage pour objectif de tester et corriger l'algorithme. Et la troisième concerne la phase de validation, cette dernière peut être utilisé lors de l'apprentissage comme une sous population de l'ensemble d'apprentissage afin de valider le modèle et d'éviter le problème de sur-apprentissage.

On peut distinguer deux grands types d'algorithmes de l'apprentissage automatique, suivant que l'on veut structurer les données par l'apprentissage non supervisé ou prédire un modèle par l'apprentissage supervisé à partir d'autres modèles plus facilement accessibles qui lui sont liés (prédire la classe ou l'étiquette des exemples à partir de leurs attributs descriptifs). C'est à ce dernier type d'apprentissage que nous somme intéressé dans ce travail [1].

Dans le processus de classification, la phase d'apprentissage est très importante parce que cette phase doit faire face à certaines difficultés lorsqu'il est confronté aux particularités des données contenues dans des ensembles de données réelles. Parmi les difficultés les plus connues, nous citons les problèmes de complexité résultant du grand nombre de cas traités, les contraintes liées aux grandes dimensions, la sensibilité des algorithmes aux données bruitées et enfin le traitement des données lorsque la composante des classes présentées est déséquilibrée, ce qui est souvent le cas pour les données réelles [1].

Dans notre travail, nous sommes intéressés au problème de classification des données déséquilibrées parce que les classes minoritaires sont mal représentées lors de la phase d'apprentissage. Lorsqu'un algorithme d'apprentissage standard est appliqué aux données déséquilibrées, les règles d'induction qui décrivent la classe minoritaire sont souvent rares et plus faibles que celles des classes majoritaires, puisque les classes minoritaires sont souvent à la fois en infériorité numérique et sous-représentées. Le but de l'analyse est de produire une approche qui offrira une grande précision pour les classes minoritaires sans pour autant compromettre sérieusement l'exactitude des classes majoritaires [70].

Afin de résoudre le problème de classification des données déséquilibrées, plusieurs techniques de classification supervisée ont été proposées. Parmi les techniques les plus utilisées dernièrement sont les Réseaux de Neurones Artificiels (RNA), Support Vector Machines (SVM), K-Plus Proche Voisin (K-PPV). Ces techniques sont sélectionnées pour être appliquées dans nos expérimentations ultérieures parce qu'elles sont basées sur les algorithmes de classification de l'apprentissage automatique supervisé les plus utilisés dans la littérature [6, 12].

II.2 Réseaux de Neurones Artificiels (RNA)

Les Réseaux de Neurones Artificiels (RNA) connus en anglais sous l'appellation Artificial Neural Networks (ANN) sont des techniques basées sur la structure neuronale du cerveau qui imite la capacité d'apprentissage de l'expérience. Cela signifie que si un réseau neuronal est entrainé à partir de données passées, il sera en mesure de générer de nouvelles classes basées sur les motifs extraits à partir des données d'apprentissage [6]. Il est aussi appelé réseau connexionniste. Deux éléments caractérisent cette méthode, une organisation appelée réseau comprenant un certain nombre d'automates aux fonctionnalités relativement simples appelés neurones. L'information se

propage dans les réseaux sur des connexions pondérées par des paramètres souvent appelés poids. Le deuxième élément est l'algorithme d'apprentissage dont l'objectif est d'ajuster les poids du réseau de neurones de manière à obtenir un comportement global intéressant [71].

De nombreux projets de recherche ont montré qu'un réseau de neurone artificiel est basé sur une technique puissante pour la classification. L'utilisation un RNA présente plusieurs avantages lors de la classification. Tout d'abord, il peut s'adapter aux données sans faire des hypothèses antérieures sur les fonctions. Deuxièmement, un RNA est un modèle non linéaire qui peut être mis en œuvre pour la plupart des applications complexes du monde réel. Enfin, un RNA peut estimer les probabilités a posteriori, ce qui permet de développer une règle de classification et de mener une analyse statistique [6].

Ils existent de nombreuses applications pratiques en utilisant RNA dans l'industrie, les entreprises et la recherche. Par exemple, la prédiction de la faillite, la reconnaissance d'écriture, la détection de défauts et le diagnostic médical. En outre, de nombreux travaux de recherche liés au problème de données déséquilibrées ont utilisé RNA comme modèle de reconnaissance clé dans leurs études [6, 72, 73].

Ils existent plusieurs types de modèles de réseaux neuronaux utilisés pour traiter les problèmes de classification des données déséquilibrées, par exemple, le réseau de neurones à rétro-propagation, en anglai Back-Propagation Neural Network (BPNN) [74].

BPNN est un modèle simple et efficace d'un RNA. Il est appelé aussi un réseau de neurones multicouche (RNMC) ou un perceptron multi couche, en anglais Multi-Layer Perceptron (MLP) parce que les neurones sont organisés de manière à constituer plusieurs couches. Il contient trois ou plusieurs couches, une couche d'entrée, une couche de sortie et une ou plusieurs couches cachées comme illustré dans la Figure 2.1 [75]. En pratique, le nombre de couches varie de 3 à 4 dont une couche d'entrée, une couche de sortie et une ou deux couches cachées. La couche d'entrée contient autant de neurones que la dimension de l'espace d'entrée. La couche de sortie contient un neurone unique pour la valeur de la prévision. Enfin, une ou plusieurs couches cachées contiennent un nombre variable de neurones [6].

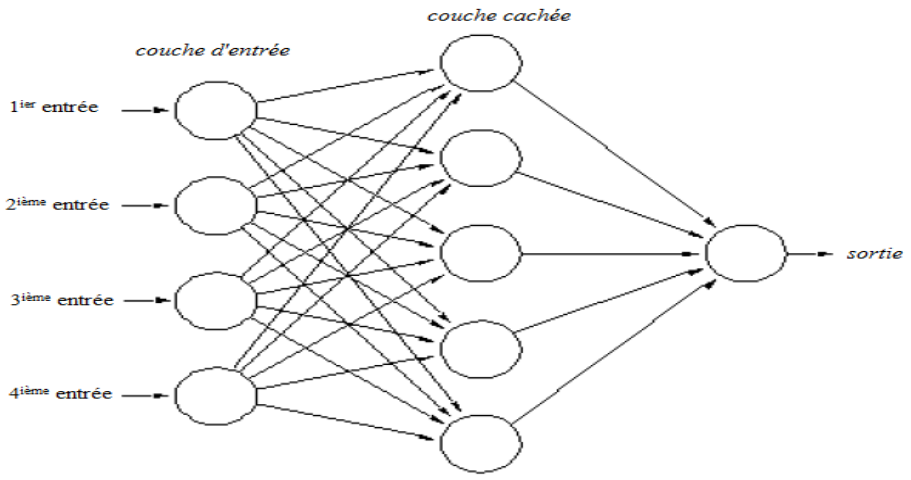


Figure 2. 1. Réseau de Neurones à rétro-propagation.

Dans le processus d'apprentissage de RNMC présenté à la Figure 2.2, chaque nœud dans la couche d'entrée, la couche cachée et la couche de sortie calcule et ajuste le poids approprié entre les nœuds et génère des valeurs de sortie de la somme résultante. Les valeurs de sorties réelles sont ensuite comparées avec les valeurs de sorties cibles. Les erreurs entre ces sorties sont recalculés et ensuite propagées vers la couche cachée afin de mettre à jour le poids de chaque nœud. Au cours de la phase de test, une entrée de test est introduite dans la couche d'entrée, et le réseau à propagation avant ou en anglai Feed Forward Neural Networks (FFNN) peut générer des résultats à partir du réseau d'apprentissage [76]. Permet les types de réseaux à propagation avant : le Perceptron Multi Couche [6].

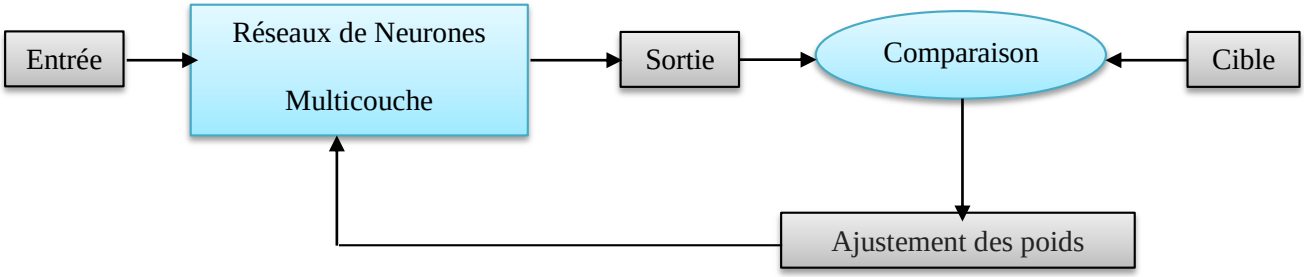


Figure 2. 2. Processus d'apprentissage d'un RNMC [77].

RNMC est un réseau de neurones robuste qui peut être testé facilement dans différents domaines d'application [74]. Cependant, il y a aussi des limites pour les réseaux de neurones à rétro-propagation. Ce réseau nécessite beaucoup de paires d'entrée et de cibles pour l'apprentissage du réseau [6].

Dans notre étude expérimentale, un réseau de neurones multicouche (RNMC) est utilisé. Sa phase d'apprentissage est réalisée par l'algorithme de rétro-propagation des erreurs. L'apprentissage par rétro-propagation fait appel à un algorithme de descente de gradient itératif visant à minimiser les erreurs quadratiques moyennes entre la sortie réelle d'un RNMC et la sortie désirée, et aussi consiste à rétro-propager le gradient de l'erreur de la couche de sortie vers l'entrée. Un pseudo-algorithme de rétro-propagation est donné par l'algorithme suivant [78] :

Algorithme 2. 1 : Principe de l'algorithme de rétro-propagation

1 : Initialisation : Affecter à tous les poids des valeurs aléatoires réelles.

2 : Présentation des entrées et la sortie désirée :

Présenter le vecteur d'entrée x(1), x(2), …, x(N) et leurs correspondantes sorties désirées d(1), d(2), …,d(N), une paire à la fois, où N est le nombre d'exemple d'apprentissage.

3 : Calcul des sorties réelles : calcule les sorties $y_1, y_2, \dots, y_{NM}$

$$y_i = \phi\left(\sum_{j=1}^{N_{M-1}} w_{ij}^{(M-1)} x_j^{(M-1)} + b_i^{(M-1)}\right), \quad i = 1, \dots, N_{M-1}.$$

4 : Adaptation des poids (w_{ij}) et les biais (b_i) :

$$\Delta w_{ij}^{(l-1)}(n) = \mu x_j(n)\, \delta_i^{(l-1)}(n)$$

$$\Delta b_i^{(l-1)}(n) = \mu\, \delta_i^{(l-1)}(n)$$

Où :

$$\delta_i^{(l-1)}(n) = \begin{cases} \varphi'(net_i^{(l-1)}[d_i - y_i(n)], & l = M \\ \varphi'(net_i^{(l-1)} \sum_k w_{ki}\, \delta_k^{(l)}(n) & 1 \le l \le M \end{cases}$$

Avec :

- $x_j(n)$ représente la sortie du nœud j à l'itération n,
- l est la couche,
- K est le nombre de nœuds de sortie du réseau de neurones,
- M est la couche de sortie,
- φ est la fonction d'activation.
- Le pas d'apprentissage est représenté par μ.

II.3 Séparateurs à Vastes Marges (SVM)

II.3.1. Approche SVM à deux classes

Les machines à vecteur de support ou séparateurs à vastes marges, ou S*upport Vector Machine* (SVM) sont une classe de techniques d'apprentissage supervisé. Elles ont été développées originellement par Vapnik en 1995 [79], Les SVMs sont dans leur origine des méthodes de classification binaire et de régression au départ mais ils existent d'autres versions de SVMs proposées pour traiter le problème multi-classe. Elles sont basées sur la notion de la marge. La marge signifie la distance minimale à partir des points de données les plus proches d'un hyperplan qui sépare les deux classes de données (positifs et négatifs). L'algorithme SVM vise à trouver la marge maximale qui peut créer la plus grande distance entre l'hyperplan séparateur et les points de données de classes différentes (voir Figure 2.3) [12]. Ces points de données sont connus en tant que vecteurs de support [6].

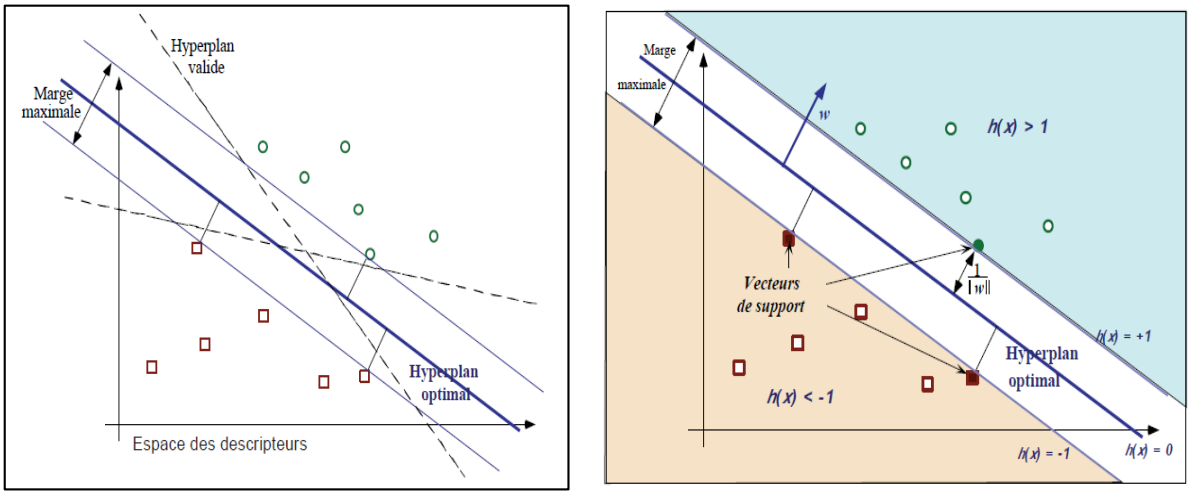


Figure 2. 3. Représentation schématique d'un SVM (Hyperplan optimal, marge maximale et vecteurs de support).

L'algorithme SVM peut être expliqué en quatre concepts de base : hyperplan séparateur, hyperplan de marge maximale, la marge souple, et la fonction du noyau. L'avantage du SVM est qu'il peut fournir une bonne généralisation, lorsque les paramètres du SVM sont choisis de manière appropriée. Il est également un modèle robuste, même si les instances d'apprentissage ont un certain biais. En outre, l'algorithme SVM se comporte normalement, même si les données d'apprentissage possèdent un grand nombre de caractéristiques. Cependant, l'algorithme SVM a

besoin d'un ensemble d'apprentissage de grande dimension afin d'obtenir une grande précision de classification [6].

Actuellement, ils existent plusieurs techniques appliquées pour modifier les performances du model SVM afin d'augmenter sa performance. Certaines versions de SVM possèdent un processus d'apprentissage plus rapide ou fournissent une complexité de calcul moins que la version classique, nous citons en particulier : *Proximal Support Vector Machines* (PSVM), *Modified Proximal Support Vector Machine* (MPSVM), *Linear Proximal Support Vector Machines* (LPSVM), …etc. En termes d'application, il y a un certain nombre de problèmes de classification traités avec succès par SVMs. Comme la classification d'images, la reconnaissance de l'écriture, la reconnaissance vocale, la classification des données médicales…etc. Elles sont utilisées dans des différents domaines de recherche et d'ingénierie tels que le domaine de santé, le marketing, la biologie, la reconnaissance de caractères manuscrits et en biométrie. En outre, ils existent de nombreuses études de recherche portant sur les problèmes de données déséquilibrées utilisant les modèles SVMs [48, 80, 81].

Parmi les modèles des SVM, nous citons les cas linéairement séparable et les cas non linéairement séparable. Les premiers modèles permettent de trouver facilement le classifieur linéaire. Dans la plupart des problèmes réels il n'y a pas de séparation linéaire possible entre les données, la notion de marge maximale ne peut pas être utilisé car il fonctionne seulement si les classes de données d'apprentissage sont linéairement séparables [82].

Dans le cas non linéaire, le principe consiste à projeter les données de l'espace d'entrée non linéairement séparables dans un espace de dimension plus grande appelé espace de redescription dans lequel on espère trouver un séparateur linéaire et les données deviennent linéairement séparables. En effet, intuitivement, plus la dimension de l'espace de redescription est grande, plus la probabilité de pouvoir trouver un hyperplan séparateur entre les exemples est élevée [82, 83, 84]. Ce principe est illustré par le schéma de la Figure 2.4 :

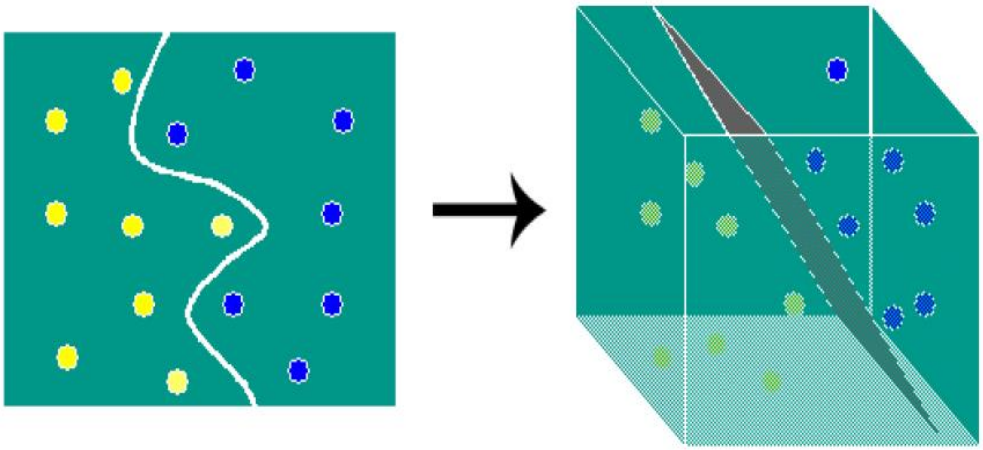


Figure 2. 4. Hyperplan séparateur dans le cas non linéairement séparable.

Nous procédons pour cela à une transformation des données d'un espace de départ vers un espace d'arrivée de manière qu'elle soit linéairement séparable dans l'espace d'arrivée. Cette transformation non linéaire est réalisée *via* une fonction noyau. En pratique, quelques familles de fonctions noyau paramétrables sont connues et il revient à l'utilisateur de l'approche SVM d'effectuer des tests pour déterminer celle qui convient le mieux pour son application. Nous citons les exemples de noyaux suivants : linéaire, polynomial, gaussien et Laplacien [82, 84].

II.3.2. Approche SVM multi-classes

A l'origine, Les machines à vecteur support (SVM) ont été conçues essentiellement pour les problèmes à 2 classes. Cependant, les problèmes du monde réel sont dans la plupart des cas multi-classes (N classes). La généralisation dans le cas multi-classes peut se faire de trois façons différentes. Les deux premières méthodes sont basées sur une multiplication des classifieurs bi-classes tandis que la dernière propose une résolution globale [83].

− **Un-contre-tous :** l'approche la plus naturelle est d'utiliser cette méthode de discrimination binaire et d'apprendre N fonctions de décision $\{f_m\}_{m=1\ldots N}$ permettant de faire la discrimination de chaque classe à partir des autres (chaque classe est opposée à toutes les autres). Il faut donc poser N problèmes binaires. L'affectation d'un nouveau point x à une classe C_i se fait par la relation :

$$i = \text{argmax}_{m=1\ldots N} F_m(x) \qquad (2.1)$$

− **Un-contre-un** : la deuxième méthode est une méthode dite de *un contre un*. Au lieu d'apprendre N fonctions de décisions, ici chaque classe est discriminée d'une autre. Ainsi, N (N-1)/2 fonctions

de décisions sont apprises et chacune d'entre elles effectue un vote pour l'affectation d'un nouveau point x. La classe de ce point x devient ensuite la classe majoritaire après le vote.

− **Méthode globale** : la dernière méthode est une approche qui étende la notion de marge aux cas multi-classes. Le problème fait intervenir N fonctions de décision et il est très gourmand en temps de calcul et en espace mémoire il est peu utilisé dans les cas réels.

II.4 K-Plus Proche Voisin (K-PPV).

La méthode des k-plus proches voisins (K-PPV) connue en anglais sous l'appellation (k-Nearest Neighbour ou k-NN) est l'une des techniques d'apprentissage supervisé la plus simple à mettre en œuvre et qui a vu son origine avec [85]. En effet, le processus d'induction ou de généralisation est effectué uniquement lorsque la classification est réalisée [12]. Par conséquent, K-PPV nécessite moins de temps de traitement pendant la phase d'apprentissage, mais il faut plus de temps de traitement pendant la phase de test. Les instances d'apprentissage de l'approche K-PPV sont représentées par un point en n attributs numériques dimensionnelles. Lorsqu'une instance inconnue doit être classée, le classifieur K-PPV recherche les k instances d'apprentissage qui sont les plus proches des cas inconnus. La distance entre deux instances peut être définie par la méthode de distance euclidienne. La distance euclidienne entre les instances $X = (x_1, x_2, \ldots . x_n)$ et $Y = (y_1, y_2, \ldots y_n)$ peut être calculée par l'équation 2.2 [6].

$$d(X,Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \qquad (2.2)$$

Pour classer les cas inconnus, une instance est affectée à une étiquette de classe par un vote majoritaire de ses k voisins. L'étiquette de la classe la plus fréquente des k instances les plus proches est affectée comme la classe prédite à une instance inconnue [6]. On peut dire que pour prédire la classe d'un nouvel objet, on le compare à ses voisins les plus proches par une mesure de similarité. Il est recommandé de ne pas considérer le voisin le plus proche de l'objet à classer, mais ses k-plus proches voisins afin de minimiser les risques d'erreurs [84].

L'opérateur de distance le plus souvent utilisé est la distance Euclidienne, cependant, en fonction du problème, on peut encore utiliser les distances de *Hamming*, de *Mahalanobis*, etc.

La méthode de K-PPV 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 et

ne tente pas de créer un modèle pour représenter les classes [84]. Certains inconvénients doivent être considérés. Tout d'abord, la technique K-PPV nécessite un temps assez long pour faire les différents calculs dans la phase de classification parce que chaque instance inconnue doit calculer la distance entre lui-même et toutes les instances d'apprentissage, afin de définir les k plus proches voisins. Par conséquent, cette technique nécessite un espace mémoire très important pour stocker les données de traitement. Deuxièmement, il n'existe pas toujours une règle claire pour définir la meilleure valeur de k. La valeur de k sélectionnée peut affecter les performances de classification du classifieur K-PPV. Une grande valeur de k peut convenir au problème de classification des données bruitées par contre une petite valeur de k peut convenir au problème de classification avec une petite région ou un petit fragment de classes de données [6]. La qualité de la discrimination par cette méthode dépend du choix du nombre k de voisins considérés et de la façon de calculer la distance entre les instances. Il est cependant souvent nécessaire de faire varier ce nombre k pour obtenir les meilleurs résultats possibles [84] et aussi il faut choisir une valeur impaire de k, pour éviter les cas d'égalité (pour le vote majoritaire). Pour le cas des données déséquilibrées, beaucoup de travaux ont été cité dans la littérature, en citant l'algorithme K-PPV comme outil de classification [86, 87].

Sur la Figure 2.5, nous pouvons voir l'effet du choix de la valeur de k sur le résultat de la classification. Dans la figure 2.5, si k prend les valeurs 1, 2 ou 3 alors la classe prédite est "**X**" par contre si k=5 alors la classe prédite est "**O**".

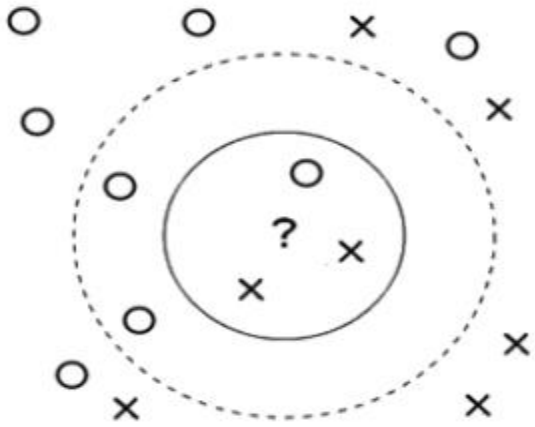


Figure 2. 5. Exemple de classification par la méthode K-PPV.

Un pseudo-algorithme des k plus proches voisins est donné par l'algorithme suivant [88] :

Algorithme 2. 2. Algorithme des k-plus proches voisins

Début
On cherche à classer le point x
Pour chaque exemple (y, w) de l'ensemble d'apprentissage **faire**
calculer la distance $D(y, x)$ entre y et x
Fin pour
Dans les k points les plus proches de x
compter le nombre d'occurrences de chaque classe
Attribuer à x la classe qui apparaît le plus souvent ; / la classe la plus fréquente/
Fin

III. Méthodes d'équilibrage

Vie l'importance du problème de déséquilibre des jeux de données, plusieurs techniques ont été développées dans la littérature pour traiter ce problème. Elles sont regroupées en trois grandes familles principales, selon le type du problème traité [89] :

1) *Modification au niveau des algorithmes :* elle concerne d'adaptation des algorithmes d'apprentissage de classification existants pour orienter l'apprentissage vers les données minoritaires. Ces méthodes nécessitent des connaissances particulières a priori sur le classifieur correspondant et le domaine d'application, sachant qu'en général le classifieur échoue lorsque la distribution des classes est inégale.

2) *Pondération des données :* les techniques utilisées permettent de rééquilibrer la répartition des classes par l'échantillonnage des jeux de données. c'est une étape de prétraitement qui permet d'éviter l'étape de modification des algorithmes d'apprentissage. Ces techniques sont indépendantes de l'algorithme de classification utilisée.

3) *Méthodes sensibles aux coûts :* elles concernent de la modification simultanée des données et des algorithmes d'apprentissage. Cette approche intègre les transformations au niveau de données (en ajoutant les coûts pour les cas) et les modifications au niveau des algorithmes (en modifiant le processus d'apprentissage à accepter des coûts). Elles orientées

les classifieurs vers la classe minoritaire. L'inconvénient majeur de cette approche est la nécessité de définir les coûts de mauvaise classification, qui ne sont pas généralement disponibles sur tous les ensembles de données.

III.1 Modification des algorithmes de classification

La modification au niveau des algorithmes met l'accent sur la proposition de nouvelles versions des algorithmes de classification existants afin de renforcer leurs capacités d'apprentissage pour apprendre la classe minoritaire, c'est une manière de les rendre moins sensibles au déséquilibre. La plupart des algorithmes de cette famille sont basés sur la méthode SVM *(Support Vector Machine)*, K-PPV *(K- Plus Proche Voisin)* et RNA (*Réseaux de Neurones Artificiels*).

III.1.1. Modification au niveau de l'algorithme SVM

L'un des algorithmes les plus utilisés en apprentissage est le Support Vector Machine (SVM). Dans sa forme traditionnelle, lorsque les ensembles de données traitées sont déséquilibrés, cet algorithme augmente le taux des échantillons mal classés de la classe minoritaire. Pour cette raison, plusieurs tentatives ont été faites pour modifier en interne ce classifieur pour prendre en charge les distributions déséquilibrées [90, 91, 92]. Joachims [90] a présenté un algorithme SVM pour optimiser directement une grande classe de mesures de performances (F_1 - Score, précision / rappel au point Breakeven) afin de formuler le problème comme une prédiction multivariée de tous les exemples. Wu et Chang [91, 92] ont modifié la matrice du noyau en fonction de la distribution des données déséquilibrées. Cette modification a pour but de compenser le décalage associé à des jeux de données déséquilibrées, elle pousse l'hyperplan vers une position plus proche de la classe positive [93].

Huang et al. [94] ont proposé une autre méthode appelée *Biased Minimax Probability Machine (BMPM)* pour résoudre le problème d'apprentissage des ensembles de données déséquilibrées. Qu'étant donné des matrices de moyennes et de covariance fiables des classes majoritaires et minoritaires, cet algorithme peut formuler un problème d'optimisation pour trouver la décision hyperplan en ajustant la limite inférieure de la précision de la classification des données futures. Par exemple, si la fonction objective est de maximiser la précision de classification pour la classe minoritaire, l'optimisation cherche à maximiser en définissant la limite inférieure de la précision de classification pour les deux classes. La réalisation de la mauvaise précision de cas

minoritaire peut être évitée, tout en maintenant le niveau de précision acceptable de la classe majoritaire dans l'apprentissage de données déséquilibrées.

Xiao-yan et Hong-bing [95] ont présenté une modification proximale de l'algorithme SVM appelée *Modified Proximal Support Vector Machine (MPSVM)* qui affecte différents coefficients de pénalité pour les échantillons positifs et négatifs respectivement, par l'ajout d'une nouvelle matrice diagonale dans le processus d'optimisation primaire. Et plus la fonction de décision est obtenue. L'algorithme de clone immunitaire réel codé (*Real-coded Immune Clone Algorithm - RICA-)* est utilisé pour sélectionner les paramètres optimaux globaux pour obtenir la performance de généralisation la plus élevée.

Wang et Japkowicz [96] ont proposé un autre algorithme basé sur les méthodes d'ensemble appelé *Boosting SVM* pour traiter le problème d'apprentissage des données déséquilibrées. Dans cet algorithme, le classifieur est développé à partir des poids actuels. Pour un cas donnée, la fonction de prédiction de la classe qui est développée à base de la fonction du noyau K. L'algorithme Boosting SVM permet de calculer le G-mean du classifieur par application de différents poids et il génère un nouvel ensemble de classifieurs. Le poids est calculé lors de l'exécution de l'algorithme boosting. G-mean est utilisé pour la prédiction d'un bon classifieur à partir d'un ensemble de classifieurs. L'algorithme *boosting SVM* reste toujours incapable de traiter le problème des ensembles de données déséquilibrées.

Pour la classification en ligne de flux de données avec une distribution déséquilibrée de classes, Lei at al. [97] ont proposé un algorithme d'apprentissage incrémental LPSVM *(Linear Proximal Support Vector Machines)* appelé DCIL-IncLPSVM *(Dynamic Class Imbalance Learning- Incremental Linear Proximal Support Vector Machines),* cet algorithme possède une bonne performance d'apprentissage des données déséquilibrées. L'algorithme LPSVM, comme les arbres de décision et l'algorithme SVM classique, etc., n'est pas à l'origine conçus pour traiter les données déséquilibrées. Lors de l'apprentissage des flux de données déséquilibrées, l'algorithme d'apprentissage incrémental est capable de s'auto ajuster pour traiter les données déséquilibrées *(Dynamic Class Imbalance Learning – DCIL).* Lei et al. ont simplifié l'algorithme LPSVM appelé LPSVM pondéré *(weighted LPSVM - wLPSVM),* ils ont proposé un nouvel algorithme d'apprentissage incrémental de wLPSVM pour DCIL, où le problème d'extraction de données de flux déséquilibré non stationnaire est gérer comme l'apprentissage de blocs de données déséquilibrées. Ces dernières sont traitées d'une manière incrémental. Lorsqu'un bloc de données

est présenté ou supprimé, l'algorithme DCIL-IncLPSVM proposé met à jour ses poids et l'algorithme LPSVM d'une manière simultanée.

Liu et al. [98] ont présenté un algorithme de classification SVM basé sur les ondelettes appelé *Wavelet Support Vector Machine* (WSVM), où la base d'ondelettes a été introduite pour construire la fonction du noyau de SVM. Pour les données déséquilibrées, un filtre à base d'une technique de sélection de fonctions est réalisé pour éliminer les informations redondantes et non pertinentes. Notons que l'algorithme PSO *(Particle Swarm Optimization)* a été appliqué pour optimiser les paramètres de l'algorithme WSVM.

Datta et Das [99] ont remarqué que lorsque l'algorithme SVM classique déplace l'hyper-plan de séparation vers la classe minoritaire comme étant la classe majoritaire, sera plus intéressant pour bien maitriser la région de chevauchement. En partant de cette remarque, ils ont proposé une nouvelle version de l'algorithme SVM pour la classification des données déséquilibrées appelée *NBSVM (Near-Bayesian Support Vector Machine).* Ils ont combiné le déplacement de l'hyperplan frontière avec les différentes pénalités de mauvaise classification. Alors on prévoit que le changement pour obtenir de meilleures performances inductives, même pour l'ensemble d'apprentissage qui est linéairement séparables ; les pénalités de régularisation inégales (pénalité plus élevé pour la classe minoritaire) devraient faire en sorte que ce changement ne résulte pas dans une mauvaise classification indue de la classe minoritaire. L'algorithme NBSVM consiste à utiliser les probabilités a posteriori bayésienne pour atteindre le décalage frontière, ainsi que les coûts de régularisation inégaux.

III.1.2. Modification au niveau de l'algorithme K-PPV

L'algorithme SVM n'est pas le seul algorithme qui a été modifié pour résoudre le problème des données déséquilibrées. Malgré que l'algorithme K- Plus Proche Voisin (K-PPV) a été considéré comme l'un des dix meilleurs algorithmes de Data Mining les plus efficaces en raison de sa simplicité et sa haute performance [100]. L'algorithme K-PPV classique n'est pas en mesure de prendre en charge une distribution déséquilibrée des données, en particulier lorsque les cas minoritaires sont répartis entre les cas majoritaires [13]. L'algorithme K-PPV classe chaque échantillon parmi ses k voisins majoritaires les plus proches dans le jeu de données d'apprentissage.

Dans la littérature plusieurs solutions ont été proposées pour ajuster cet algorithme. Par exemple, Barandela et al. [87] ont essayé de compenser le déséquilibre du jeu de données d'apprentissage sans altérer la distribution des classes. Ils ont proposés l'utilisation d'une distance

pondérée dans la phase de classification de l'algorithme k-plus proches voisins. Contrairement à la règle de K-PPV habituellement pondérée, les poids sont affectés aux classes respectives et non pas aux individus prototypes. De cette façon, le facteur de pondération est plus important pour la classe majoritaire que pour la classe minoritaire, sachant que la distance par rapport des prototypes de classe minoritaire positifs devient beaucoup plus faible que la distance par rapport des prototypes de la classe majoritaire. Cela crée une tendance aux nouveaux modèles pour trouver leurs voisins les plus proches parmi les prototypes de la classe minoritaire.

Yang et al. [101] ont proposé deux versions différentes de l'algorithme K-PPV basées sur le principe Informatif. Selon eux, un échantillon est traité pour être informatif, s'il est proche de l'instance de requête et plus loin des échantillons avec des étiquettes différents. LI-KNN (*Locally Informative k-Nearest Neighbour)* est une des versions proposées, elle prend deux paramètres k et I. Il trouve d'abord le k plus proche voisin de l'instance de requête, puis parmi eux il trouve les échantillons I les plus informatifs. L'étiquette de classe est attribuée à l'instance de requête basée sur ses échantillons d'informatifs. Ils ont également montré que les valeurs de k et de I affectent moins le résultat final. Une version GI-KNN (*Globally Informative k-Nearest Neighbour*) fonctionne sur l'hypothèse que certains échantillons sont plus informatifs que les autres. Elle trouve d'abord des échantillons informatifs globaux, puis elle attribue un poids à chaque échantillon dans l'ensemble de données d'apprentissage en fonction de son degré d'information. Ensuite, il utilise une métrique euclidienne pondérée pour calculer les distances.

Pour améliorer les performances de l'algorithme K-PPV pour la classification des données déséquilibrées, kENN *(k Exemplar based Nearest Neighbor)* [102] et CCW-kNN (*Class Confidence Weighted - kNN)* [103] ont été proposées. kENN propose une étape d'apprentissage où les instances d'apprentissage positives exemplaires sont identifiées et généralisées en boules gaussiennes comme concepts pour la classe minoritaire. Lors de la classification d'une instance de requête à l'aide de son k plus proche voisin, les concepts positifs formulés à l'étape d'apprentissage montrent que la classification est plus sensible à la classe minoritaire. Cette approche est basée sur l'extension de la frontière de décision pour la classe minoritaire. Liu et Chawla [103] ont proposé une nouvelle stratégie de pondération de k-plus proche voisin pour traiter le problème des classes déséquilibrées appelée CCW-kNN. Ils ont proposé CCW (*Class Confidence Weights)* qui affecte les valeurs probabilistes des attributs des classes d'étiquettes aux poids prototypes dans l'algorithme K-PPV. Ils ont utilisé des probabilités conditionnelles de classes, mais pas les

probabilités d'étiquettes de classe au voisinage de l'instance de requête. Le principal avantage de CCW est qu'il est capable de corriger la distorsion inhérente à la classe majoritaire dans les algorithmes K-PPV existants sur une mesure de distance. Ces méthodes exécutent plus de précision que l'algorithme K-PPV. Toutefois, les deux algorithmes kENN et CCW-kNN sont destinés aux données numériques et ils nécessitent une étape d'apprentissage pour trouver des échantillons d'apprentissage exemplaires, afin d'agrandir les frontières de décision pour la classe positive, ou d'apprendre le poids de la classe pour chaque échantillon d'apprentissage par le model des mélanges et le réseau bayésien.

Zhang et Li ont proposé une autre version de l'algorithme de plus proche voisin appelée *Positive-biased Nearest Neighbor* (PNN) [104]. L'objectif principal de cet algorithme consiste à renforcer ENN en comparant d'une manière dynamique la distance entre le $k^{ième}$ plus proche voisin local et la requête de la distance entre le $r^{ième}$ plus proche forme de POS (Classe positive) et la requête. Selon la règle de PNN, l'un des deux paramètres k et r est sélectionné pour équilibrer la distribution locale des modèles de classes binaires. Cela signifie que la zone de recherche de motif de requête dépend de la valeur des deux paramètres. PNN peut traiter les formes de l'ensemble de test d'une manière plus rapide.

Liu et al. [105] ont proposé une nouvelle approche de classification basée sur la similarité floue couplée pour répondre à la différence entre les classes par une appartenance floue et les couplages par une similarité d'objet couplée. Cette approche est appelée k-NN Floue Couplée (*Coupled Fuzzy k-Nearest Neighbour - CF-kNN*). Ils ont utilisé pour classer les données catégoriques déséquilibrées avec de fortes relations entre les objets, les attributs et les classes. Cette approche comprend la taille d'appartenance à une classe avec un poids d'attribut dans une mesure de similarité couplée, qui extrait efficacement l'inter-couplage et intra-couplage des relations dans les données catégoriques.

Liu et Yu. [106] ont proposé un algorithme de classification hybride couplé de k-plus proche voisin appelé *Hybrid Coupled k-Nearest Neighbour (HC-kNN)* pour surmonter le problème de classification des ensembles de données déséquilibrées de type mixte, en abordant à la fois les relations entre les classes et entre les caractéristiques. HC-kNN se compose de cinq parties : affectation d'appartenance (membership assignment), discrétisation de données, fonction de pondération, calcul de similarité et d'intégration. Ils ont fait une discrétisation sur les caractéristiques numériques afin d'adapter la similarité inter-couplage comme ils le font sur les

caractéristiques catégorielles, puis ils ont combiné cette similarité couplée à la similarité d'origine ou de la distance.

Pang et al. [107] ont proposé un classifieur hybride évolutif et efficace pour surmonter le problème des ensembles de données déséquilibrées et bruitées appelé CenKNN (*Centroid-based dimension reduction -K-Nearest-Neighbor*), qui combine deux techniques largement utilisées dans la classification de texte, le classifieur k-plus proche voisin et une méthode de réduction de dimension à base de centroïdes de classe. Le classifieur CenKNN projette des documents de grande dimension dans un espace de faibles dimensions engendrées par centroïdes de classe avec la méthode de réduction de dimension, puis ils utilisent la structure d'arbre k-d pour trouver efficacement les K voisins les plus proches. CenKNN surmonte deux problèmes liés au classifieur K-PPV pour la classification de texte, à savoir la sensibilité de la distribution des classes déséquilibrées et les caractéristiques non pertinentes ou bruitées. Cette approche permet de réduire considérablement le temps de calcul coûteux en algorithme K-PPV.

III.1.3. Modification au niveau des algorithmes à base des Réseaux de neurones

La plupart des classifieurs RNA classiques trouvent des difficultés majeures lors de l'apprentissage des ensembles de données déséquilibrées parce que les classifieurs sont conçus pour apprendre dans les ensembles de données équilibrées. Une étude réalisée par Murphey et al. [108] a mis en évidence les problèmes des ensembles de données déséquilibrées qui sont les instances fréquentes d'apprentissage de la classe majoritaire, le réseau tend à ignorer la classe minoritaire et la traite comme bruit. Par conséquent, les algorithmes d'apprentissage existants pour prendre en charge les problèmes de jeu de données déséquilibrées ont été proposés pour améliorer les classifieurs RNA classiques.

Par l'utilisation d'une seule couche feed-forward de RNA, Anand et al. [109] ont proposé une version modifiée de l'algorithme de rétro-propagation classique. L'algorithme se focalise sur le calcul de la direction des changements de poids, ce qui diminue l'erreur pour chaque classe.

Ensuite, une étude a été réalisée par l'utilisation de trois architectures différentes de RNA, *Fuzzy ARTMAP*, *Multi-layered backpropagation*, et *Radial Based Functions* (RBF) [110]. Pour chacune des trois architectures de réseau, trois méthodes d'apprentissage ont été utilisés : simple train -test, duplication des échantillons d'apprentissage de la classe minoritaire et la méthode

Snowball. En conclusion, les auteurs ont constaté que le *Fuzzy ARTMAP* a le potentiel de donner des bonnes solutions pour les problèmes de jeu de données déséquilibrées.

Par ailleurs, Fu et al. [111] ont proposé un algorithme d'apprentissage modifié pour le réseau de neurones RBF (*Radial Basis Function*) pour améliorer les performances de prédiction du réseau de neurones RBF classique. L'algorithme d'apprentissage modifié met l'accent sur l'amélioration de la précision de la classe minoritaire avec le maintien de la performance globale du classifieur.

Alejo et al. [112] ont proposé une méthode pour améliorer la performance de classification de RBF et MLP (*Multilayer Perceptron*) pour les ensembles de données déséquilibrées. Dans cette approche, les réseaux RBF et MLP sont appliqués avec une technique de filtrage pour réaliser le prétraitement de données, elle est basée sur une règle de plus proche voisin. Par conséquent, cette méthode a amélioré la performance de classifieur RBF. Par contre, la performance est mauvaise lorsque cette méthode a été utilisée avec le classifieur MLP.

Une autre approche appelée *Modular Neural Network (MNN)* [113], elle est basée sur le concept diviser et gagner [113]. Cette approche est adoptée comme une nouvelle technique de décomposition. Le réseau est combiné avec plusieurs méthodes d'intégration telles que la moyenne et l'algorithme génétique (*Genetic Algorithm -GA*) pour combiner les décisions par chaque réseau.

Giang et al. [114] ont modifié quatre principaux algorithmes d'apprentissage feedforward ANN, nommés GD (*Gradient Descent*), GDMV (*Gradient Descent with Momentum and Variable learning rate*), RPROP (*Resilient back propagation*) et LM (*Levenberg-Marquardt*). Cette recherche a montré que les algorithmes d'apprentissage modifiés sont en mesure d'obtenir une meilleure précision de classification en comparant aux algorithmes d'apprentissage classiques de RNA.

Adam et al. [115] ont proposé un autre algorithme d'apprentissage modifié de réseaux de neurones artificiels pour résoudre les problèmes des ensembles de données déséquilibrées. Cet algorithme est utilisé pour d'améliorer les performances de prédiction de classifieur RNA standards, il se focalise sur l'optimisation de la limite de décision de la fonction à pas à la couche de sortie de RNA par l'utilisation de l'algorithme d'optimisation par essaim particulaire nommé *Particle Swarm Optimization* (PSO). Dans son étude, Un réseau de neurones à rétro-propagation est choisi. Tout d'abord, un algorithme classique de back propagation est utilisé pour entrainer le

réseau ANN. Ensuite, l'algorithme PSO est appliqué pour apprendre la vraie sortie prédite des données d'apprentissage de ce réseau d'apprentissage.

Adam et al. [116] ont proposé un algorithme d'apprentissage supervisé en deux étapes d'une seule couche feedforward de RNA pour résoudre les problèmes d'apprentissage de jeu de données déséquilibrées. L'algorithme d'apprentissage de rétro-propagation de *Levenberg Marquart* est utilisé dans la première étape d'apprentissage, alors que la deuxième étape de mécanisme apprentissage est introduite par l'optimisation de seuil de décision de la fonction à pas à la couche de sortie de RNA, en utilisant l'optimisation PSO. A la fin des étapes d'apprentissage, ils ont obtenu des meilleurs poids et la valeur de seuil de décision à utiliser pour le processus de test.

Pérez-Godoy et al. [117] ont comparé la performance de trois modèles modifiés de *RBF Network* (RBFN) dans la classification des données déséquilibrées qui sont RBF à base de clustering, RBFN incrémental, RBFN à base évolutive, parmi lesquels l'algorithme de la moyenne des moindres carrés et la décomposition en valeurs singulières sont pris en compte dans la phase d'apprentissage des poids.

Umi et al. [118] ont proposé une modification d'un autre algorithme de réseau de neurones artificiels appelé *Extreme Learning Machine (ELM)* pour surmonter les problèmes de données déséquilibrées par l'intégration d'un processus de sélection de données, nommée IDELM (*Integrating the Data selection and Extreme Learning machine*). Cet algorithme permet d'appliquer une méthode de sélection de données au cours du processus d'apprentissage avec ELM.

III.2 Pondération des données : techniques d'échantillonnage

Les techniques d'échantillonnage sont parmi les plus simples et les plus faciles à mettre en œuvre. Ils sont utilisés pour traiter le problème de déséquilibre des jeux de données et elles sont également connues comme des méthodes de prétraitement des ensembles de données. L'objectif principale d'échantillonnage consiste à rétablir l'équilibre entre l'effectif associe à chaque classe. Ces techniques sont indépendantes du classifieur sous-jacent et peuvent être réalisés de trois manières, sous-échantillonnage (Under- Sampling –US) de la classe majoritaire, sur-échantillonnage (Over-Sampling –OS) de la classe minoritaire, ou par la combinaison des deux techniques d'échantillonnage (Méthodes hybrides).

III.2.1. Sur-échantillonnage (OS)

C'est une méthode très simple à mettre en œuvre et elle est parmi les méthodes les plus utilisées dans la littérature pour traiter le problème de déséquilibre des jeux de données. L'objectif principal de cette méthode consiste à ajouter des instances à la classe minoritaire pour créer un équilibre entre les classes.

Dans sa version aléatoire, il s'agit d'ajouter des instances au hasard à la classe minoritaire. De cette façon, la méthode de sur-échantillonnage aléatoire permet de répliquer aléatoirement des individus de la classe minoritaire que l'on ajoute à l'ensemble d'apprentissage initial jusqu'à obtenir un nombre d'exemples identique pour les deux classes. Certains individus de la classe minoritaire se trouvent donc plusieurs fois dans l'échantillon équilibré, d'autant plus que le déséquilibre est important. La pertinence de la réplication aléatoire peut être ici aussi remise en cause. En effet, cette solution court le risque de forcer le classifier à apprendre sur des zones très spécifiques de l'espace de représentation, et introduit par conséquent un fort biais d'apprentissage (risque de sur-apprentissage), ce qui rend la règle de classification à être trop spécifique ; même si la précision de jeu d'apprentissage est élevé, la performance de classification pour les nouveaux jeux de données de test sera probablement mauvaise [45]. Pour éviter ce problème de sur apprentissage, Chawla et al. [119] proposent une nouvelle méthode de sur- échantillonnage appelée SMOTE (*Synthetic Minority Oversampling Technique*). L'objectif principal de cette méthode est de générer des individus synthétiques dans la classe minoritaire. Le principe de cette méthode est présenté comme suit : on cherche les k plus proches voisins de chaque individu de la classe minoritaire. Les individus synthétiques sont construits en ayant une représentation correspondante à une localisation au hasard sur la ligne reliant le point de classe minoritaire concerné et un de ses k plus proches voisins. Ces individus ayant une représentation différente des individus de la classe minoritaire déjà présents. L'apprentissage utilise ensuite un classifieur standard sur le nouvel échantillon ainsi constitué [1]. Ainsi le problème du sur-apprentissage est évité et la frontière de décision de la classe minoritaire tend à se rapprocher de l'espace de la classe majoritaire [120].

Chawla et al. [121] ont proposé une nouvelle approche pour l'apprentissage des classes minoritaires. Cette approche basée sur une combinaison de l'algorithme SMOTE et la procédure de boosting (AdaBoost). Ils ont nommée SMOTEBoost. Contrairement à boosting standard où tous les exemples mal classés ont des poids égaux, la nouvelle approche SMOTEBoost consiste à créer

des exemples synthétiques dans la classe minoritaire par la méthode SMOTE au lieu de simplement augmenter le poids des exemples de la classe minoritaire, ainsi elle changeant de manière adaptative les poids des différents exemples pour compenser les distributions asymétriques. Donc SMOTEBoost consiste à utiliser l'algorithme SMOTE avant d'appliquer AdaBoost sur le nouvel échantillon construit par SMOTE.

Han et al. [47] ont développé une modification de SMOTE appelée Borderline-SMOTE, elle concerne deux nouvelles méthodes de sur-échantillonnage de la classe minoritaire ; borderline-SMOTE1 et borderline-SMOTE2, dans laquelle seuls les exemples minoritaires près de la frontière sont sur-échantillonnés.

D'autres approches comme Safe-Level-SMOTE [48] et ADASYN (*Adaptive Synthetic sampling approach*) [49] fonctionnent de manière similaire. La première approche est basée sur le même principe des approches précédentes, comme SMOTE et Borderline-SMOTE, elles peuvent générer des instances synthétiques dans des zones inappropriés, tels que les régions de chevauchement ; par conséquent, les auteurs calculent la valeur de 'niveau de sécurité' pour chaque instance positif avant de générer des instances synthétiques et les rapprocher au plus grand niveau de sécurité. D'autre part, l'idée principale de l'algorithme ADASYN consiste à utiliser une distribution de densité comme un critère de décision automatique du nombre d'échantillons synthétiques qui doivent être générés pour chaque exemple minoritaire, et cela en changeant de façon adaptative les poids des différents exemples minoritaires pour compenser les distributions asymétriques. La méthode ADASYN peut non seulement réduire le biais d'apprentissage introduit par la distribution des données déséquilibrées, mais peut également déplacer d'une manière adaptative la frontière de décision de classification de se concentrer sur les échantillons difficiles à apprendre.

Gao et al. [122] ont proposé dans leurs article une technique puissante utilisée pour traiter le problème de classification des données déséquilibrées à deux classes, ils ont combiné la méthode SMOTE et un algorithme d'optimisation par essaims particulaires (PSO) basé sur une fonction de base radial (RBF). Afin de renforcer l'importance de la petite région spécifique appartenant à la classe positive dans la région de décision. Les paramètres de noyaux RBF sont déterminés par l'utilisation d'un algorithme de PSO basé sur le critère de minimisation du taux de mauvaise classification de leave-one-out.

Kung et al. [123] ont étudié un ensemble de nouveaux algorithmes pour améliorer l'efficacité de classification pour les 5 ans de survie des patients atteints de cancer du sein à partir d'un ensemble de données massives avec la propriété déséquilibrée. Les algorithmes de classification étudiés sont combinés avec les techniques SMOTE et PSO, il intègre des classifieurs bien connues, telles que la régression logistique, arbre de décision C5, et le k-plus proche voisin. Les résultats expérimentaux montrent que l'algorithme hybride de SMOTE + PSO + C5 est le meilleur pour la classification parmi tous les algorithmes de combinaison utilisés.

Récemment, José et al. [124] proposent une extension de l'algorithme SMOTE à l'aide d'un nouvel élément ; basé sur un ensemble itératif du filtre bruité appelé *Iterative- Partitioning Filter* (IPF). Il peut surmonter les problèmes produits par les exemples bruyants et borderline dans des ensembles de données déséquilibrées. Cette approche est nommée SMOTE-IPF. Cette approche obtient des meilleures performances que d'autres techniques de ré-échantillonnage dans les scénarios considérés.

III.2.2. Sous- échantillonnage (US)

La solution inverse de sur-échantillonnage est le sous échantillonnage. C'est une méthode très simple à mettre en œuvre et parmi les plus utilisées dans la littérature pour traiter le problème de déséquilibre des jeux de données. L'objectif principal de la méthode de sous-échantillonnage consiste à éliminer des instances appartenant à la classe majoritaire et garde toutes les instances de la classe minoritaire pour créer un équilibre entre les classes.

Dans le cas de sous-échantillonnage aléatoire, la solution de base consiste à sélectionner et supprimer aléatoirement des exemples appartenant à la classe majoritaire jusqu'au le nombre d'individus de cette dernière égal au nombre des exemples de la classe minoritaire. Cette méthode est très simple à mettre en œuvre et rapide dans leur exécution, mais elle risque de supprimer des individus importants de la classe majoritaire. Donc la méthode de sous-échantillonnage aléatoire à un inconvénient majeur c'est le risque de perte d'information. Pour éviter ce problème plusieurs auteurs proposent de sélectionner des exemples de la classe majoritaire d'une façon plus intelligente. Par exemple, Kubat et Matwin [45] ont présenté dans leur article une méthode de sous échantillonnage appelée '*One-Side Selection'*. L'objectif principal de cette méthode consiste à supprimer les individus redondants de la classe majoritaire ainsi que les individus existant à la

frontière de décision entre les classes [1]. Les individus à la frontière sont détectés à l'aide des liens de Tomek [125], le principe de liens de Tomek se déroule comme suit : il existe deux échantillons s_i et s_j appartenant à des classes différentes. Une paire (s_i, s_j) est appelée un lien Tomek, si il n'existe pas un échantillon s_w tel que $d(s_i, s_w) < d(s_i, s_j)$ ou $d(s_j, s_w) < d(s_i, s_j)$ où $d(.,.)$ est la distance entre les deux échantillons considérés. Si ces deux échantillons forment un lien Tomek, c'est qui signifie que l'un des deux est du bruit, ou que tous les deux sont des points frontières. L'objectif principal de cette méthode de sous-échantillonnage est de supprimer les échantillons appartenant à la classe majoritaire qui sont éloignés de la frontière de décision afin d'éliminer les échantillons moins pertinente pour l'apprentissage [126].

Zhang et Mani [127] ont testé la méthode de K-PPV après avoir appliqué cinq approches de sous-échantillonnage. En plus de l'approche aléatoire (Random), et de la sélection des exemples les plus éloignés des exemples positifs (Distant), ces auteurs proposent au contraire trois méthodes de sélection des exemples frontières : la première (NearMiss1) sélectionne les exemples négatifs ayant la distance moyenne aux trois plus proches exemples positifs les plus faibles. La seconde méthode (NearMiss2) sélectionne les exemples négatifs ayant la distance moyenne aux trois plus éloignés exemples positifs. Enfin la troisième (NearMiss3) sélectionne les k plus proches exemples négatifs, garantissant ainsi que chaque exemple positif soit entouré par quelques exemples négatifs. L'idée de cette étude est de sélectionner les individus de la classe majoritaire dont la distance moyenne aux K plus proche voisins de la classe minoritaire est plus faible.

Barandela et al. [128] ont proposé une méthode qui consiste à appliquer un classifieur de type K-PPV aux données d'apprentissage et à en supprimant les individus mal étiquetés. Cependant comme la règle K-PPV est également soumise au problème du déséquilibre, ils biaisent la mesure de distance en la pondérant en fonction de la classe de l'exemple considéré. Ainsi, la distance à des individus de classe minoritaire diminue par rapport à la distance à des individus de classe majoritaire [129].

Yen et Lee [130] proposent une approche de clustering basée sur le sous-échantillonnage appelée *cluster based under-sampling*. Cette approche de clustering regroupe tous les échantillons d'apprentissage dans certains clusters. L'idée principale est qu'il existe différents clusters dans un ensemble de données, et chaque cluster semble avoir des caractéristiques distinctes. Si un cluster a plus d'échantillons de classe majoritaire et moins d'échantillons de classe minoritaire, il se

comporte comme les échantillons de classe majoritaire. D'autre part, si cluster a plus d'échantillons de classe minoritaire et moins d'échantillons de classe majoritaire, il ne tient pas les caractéristiques des échantillons de la classe majoritaire et il se comporte plus comme les échantillons de classe minoritaire. Par conséquent, cette approche sélectionne un nombre approprié d'échantillons de classe majoritaire de chaque cluster en considérant le rapport entre le nombre d'échantillons de classe majoritaire et le nombre d'échantillons de classe minoritaire dans le cluster .

Rahman et Davis [131] proposent une autre technique de clustering basée sur le sous-échantillonnage qui résout le problème de déséquilibre de classe pour les données cardiovasculaires. Leur approche de sous-échantillonnage est différente de l'approche du Yen et Lee [130]. Leur approche consiste à séparer les données dans deux ensembles, un ensemble contenant tous les échantillons de la classe majoritaire et l'autre contenant tous les échantillons de la classe minoritaire. Ensuite, ils ont regroupé les échantillons de la classe majoritaire à K clusters (K> 1) alors ils font K sous-ensembles d'échantillons de la classe majoritaire, où chaque cluster est considéré comme un sous-ensemble de la classe majoritaire. L'objectif était de ne pas établir un rapport de classe majoritaire sur classe minoritaire de 1:1; ils voulaient juste réduire l'écart entre le nombre d'échantillons de la classe majoritaire et le nombre d'échantillons de classe minoritaire. Tous les sous-ensembles de la classe majoritaire sont combinés séparément avec les échantillons de la classe minoritaire pour faire K ensembles différents de données d'apprentissage (La valeur de K dépend du domaine de données, dans leur mise en œuvre K est égal à 3). Tous les jeux de données combinés sont classés avec un arbre de décision et un algorithme de *Fuzzy Unordered Rule Induction*.

Annarita et Rosalia [132] ont décrit une nouvelle approche appelée *Parallel Selective Sampling* (PSS), qui sélectionne des instances de la classe majoritaire pour réduire le déséquilibre dans les grands ensembles de données. PSS est une méthode de filtrage qui a été combiné avec le classifieur SVM. Elle est basée sur l'idée que seulement les données d'apprentissage près de la limite de séparation (pour la classification) sont pertinentes. De cette façon, les informations d'ensemble de données d'apprentissage près de la frontière de séparation sont conservées tandis que la taille de l'ensemble d'apprentissage est effectivement diminuée. Des exemples pertinents de la classe majoritaire sont sélectionnés et utilisés dans l'étape de classification successive de SVM. En raison des exigences de calcul complexes, PSS est conçu et développé pour le calcul parallèle

et distribué. Enfin, PSS-SVM permet de préciser des prédictions statistiques en maintenant le temps de calcul.

III.2.3. Méthodes hybrides

Les méthodes hybrides combinent les deux techniques d'échantillonnage (sur-échantillonnage avec sous-échantillonnage). Elles consistent à établir d'une manière simultanée l'augmentation des échantillons de la classe minoritaire par les approches de sur-échantillonnage et la diminution des échantillons de la classe majoritaire par les approches de sous-échantillonnage pour équilibrer la répartition des classes. Par exemple Peng et Yao proposent une approche hybride appelée AdaOUBoost (*Adaptive Over-sampling and Under-sampling Boost*) [133]. Cette approche permet de sur-échantillonner les instances positives de la classe minoritaire d'une manière adaptative et de sous- échantillonner les instances négatives de la classe majoritaire pour former des sous classifieurs différents et de combiner ces sous classifieurs en fonction de leur précision pour créer un classifieur fort, qui vise à utiliser pleinement l'ensemble des données d'apprentissage et d'améliorer les performances de classification des jeux de données déséquilibrées. Dans l'algorithme AdaOUBoost, une approche de clustering basée sur le sous-échantillonnage consiste à diviser les exemples négatifs majoritaires dans certains sous-ensembles disjoints. Ensuite, pour chaque sous-ensemble d'exemples négatifs, un algorithme de sur-échantillonnage appelé « borderline-SMOTE » permet de sur-échantillonner les exemples positifs avec des tailles différentes. Puis, chaque sous–ensemble crée, il est utilisé dans la phase d'apprentissage de sous-classifieurs et ils ont obtenu un classifieur par la fusion de ces sous-classifieurs avec des poids différents. Enfin, cette approche consiste à combiner ces classifieurs dans chaque sous-ensemble d'exemples négatifs pour créer un classifieur fort.

Cateni et al. [126] présentent une nouvelle approche de ré-échantillonnage appelée *Similarity- based Under Sampling and Normal Distribution- based Oversampling (SUNDO)* pour surmonter le problème du déséquilibre de classe, qui combine une technique de sur-échantillonnage basée sur la distribution normale et une technique de sous-échantillonnage basée sur la similarité. Ils ont testés leurs approche sur quatre classifieurs de base (Support Vector Machine, Arbre de décision, Labelled Self-Organizing Map et les classifieurs bayésiens).

Qian et al. [134] proposent un algorithme d'ensemble de ré-échantillonnage appelé *Resampling Ensemble Algorithm* (REA) pour résoudre les problèmes de classification des jeux de

données déséquilibrées. Dans cet algorithme, la classe minoritaire est sur-échantillonnée par la technique de SMOTE et la classe majoritaire est sous-échantillonnée d'une façon aléatoire. Ensuite, plusieurs méthodes d'apprentissage automatique sont sélectionnées pour construire un ensemble. La combinaison des apprenants de base est effectuée selon la stratégie de bagging. Les échelles de sur-échantillonnage et de sous-échantillonnage sont analysées et les équations empiriques sont dérivées.

Qiang wang [135] a proposé une approche d'échantillonnage hybride basée sur support vector machine appelée *hybrid sampling SVM* pour résoudre le problème de déséquilibre de classe. Premièrement, cette méthode permet d'utiliser l'algorithme de classification SVM pour générer les hyperplans de classification et d'appliquer une technique de sous-échantillonnage pour réduire les échantillons négatifs contenant moins d'informations de classification. Ensuite, il permet de diviser l'ensemble de données d'apprentissage en plusieurs sous- ensembles et de synthétiser de nouveaux échantillons positifs par l'utilisation d'une technique de sur-échantillonnage. Une fois la classe majoritaire a été sous–échantillonnée et la classe minoritaire a été sur-échantillonnée, un nouveau ensemble de données d'apprentissage équilibré est créé et il est utilisé pour l'apprentissage de classifieur SVM.

III.3 Méthodes sensibles aux coûts

La principale différence entre les méthodes de ré-échantillonnage et ceux sensibles aux coûts, est que les premières sont principalement utilisées comme routines de prétraitement avant la phase d'apprentissage actuelle, tandis que les deuxièmes sont étroitement liées à la phase d'apprentissage. L'objectif principal de la classification sensible aux coûts est de minimiser le coût d'erreurs de mauvaise classification, qui peut être réalisé en choisissant la classe avec le risque conditionnel minimum [93].

Classification sensible aux coûts considère les coûts variables de différents types d'erreurs de mauvaise classification. On représente généralement ces coûts par une matrice carrée C de taille k × k, où C (i, j) représente le coût de classer un individu de la classe i vers la classe j. Avec cette notation, C (+, -) est le coût de mauvaise classification d'un individu positif (classe rare) comme individu négatif (classe courante) et C (-, +) est le coût du cas contraire. Dans le traitement de problème des classes déséquilibrées, l'importance de la reconnaissance des cas positifs est plus

élevée que celui des cas négatifs. Par conséquent, le coût de la mauvaise classification d'un individu positif emporte sur le coût de la mauvaise classification d'un individu négatif (à savoir, C (+, -)> C (-, +)) ; faire une bonne classification présente habituellement un coût nul c.à.d. C (+, +) = C (-, -) = 0. Le processus d'apprentissage sensible aux coûts cherche alors à minimiser le nombre d'erreurs de coûts élevés et le coût d'une mauvaise classification totale [17]. On définit alors le risque coût de choisir la classe i par [59] :

$$\text{Coût} = \sum_{i=1}^{k} C(i,j) * P(\text{c}_j) \qquad (2.3)$$

Les approches sensibles aux coûts cherchent à générer des modèles qui minimisent ce risque, en choisissant la classe j ayant le risque minimum, au lieu de minimiser le nombre d'erreurs [59].

Dans les approches sensibles aux coûts, les méthodes qui prennent en compte les coûts sont séduisantes, mais elles nécessitent en théorie de connaitre la matrice de coûts, ce qui est assez rare. Par exemple, Domingos [136] propose METACOST, une méthode générale qui permet d'introduire les coûts de mauvaise classification dans un algorithme d'apprentissage supervisé en ré-étiquetant chaque individu par la classe qui permet de minimiser le coût final grâce à une approche bootstrap pour ensuite construire le modèle final sur l'échantillon ainsi ré-étiqueté [1].

Fan et al. [137] ont proposé une autre version d'Adaboost sensible au coût appelée Adacost. Dans cet algorithme, les exemples appartenant à la classe rare qui sont mal classés sont affectés des poids plus élevés que ceux appartenant à la classe commune. Il est empiriquement démontré que le système proposé produit des coûts de mauvaise classification cumulatifs inférieurs à ceux AdaBoost.

Joshi et al. [138] ont proposé un algorithme amélioré de boosting, qui met à jour les poids des prédictions positives différemment des poids des prédictions négatives. Il échelle des exemples de faux positifs en proportion de la façon dont ils se distinguent des exemples vrais positifs et il échelle des exemples de faux positifs en proportion de la façon dont ils se distinguent des exemples de vrais négatifs, ce qui permet l'algorithme de se concentrer sur les deux, rappel et de précision aussi. C'est un nouvel algorithme qui peut parvenir à une meilleure prédiction pour la classe minoritaire.

Zhou et al. [7] ont appliqué l'apprentissage sensible aux coûts dans la rétro-propagation d'ANNs. Cette étude détermine également les effets de l'échantillonnage de données (sous-échantillonnage et sur-échantillonnage) et Threshold- moving (seuillage - mouvement) pendant l'apprentissage. Au lieu d'utiliser une technique d'échantillonnage de données, cette approche

trouve un seuil de décision optimal en utilisant un algorithme de Threshold- moving à la couche de sortie du réseau pour obtenir des meilleures performances de prédiction. En fin de compte, les auteurs ont déduit que le threshold- moving est le meilleur algorithme d'apprentissage pour un RNA sensible aux coûts.

Sun et al. [139] ont examiné les algorithmes de boosting sensibles aux coûts pour améliorer la classification des données déséquilibrées. Ils ont proposé trois algorithmes de boosting sensibles aux coûts appelés AdaC1, AdaC2 et AdaC3 par l'introduction des éléments de coûts dans le cadre de l'apprentissage dans la formule de mise à jour de poids d'AdaBoost.

He et Garcia [140] ont regroupé toutes les techniques sensibles aux coûts en trois classes. La première classe consiste à appliquer des coûts de mauvaise classification à l'ensemble de données comme une forme de pondération de l'espace de données. La deuxième classe consiste à appliquer des techniques de minimisation des coûts aux schémas de combinaison de méthodes d'ensemble. Et la dernière classe consiste à intégrer des fonctions sensibles aux coûts directement dans les paradigmes de classification pour ajuster essentiellement le cadre sensible aux coûts dans ces classifieurs.

Zhou et Liu [141] ont proposé une approche générale de mise à l'échelle de l'apprentissage sensible aux coûts multi-classe appelée $\text{Rescale}_{\text{new}}$ qui a montré la nécessité de séparer les coûts de mauvaise classification du taux de déséquilibre pendant la phase d'apprentissage.

Adel et al. [142] ont proposé un ensemble en ligne de classifieur réseau de neurones. Les modèles d'ensemble sont les méthodes les plus fréquentes qui sont utilisés pour classer les ensembles de données streams non-stationnaires et déséquilibrées. Ils ont appliqué une approche à deux couches pour le traitement des classes déséquilibrées et non-stationnaires. Dans la première couche, un apprentissage sensible aux coûts est intégré dans la phase d'apprentissage des réseaux neuronaux, et dans la deuxième couche une nouvelle méthode de pondération des classifieurs de l'ensemble est utilisée.

Zhang et Wang [143] ont proposé une méthode d'ensemble sensible aux coûts basée sur un classifieur SVM sensible aux coûts et requête par comité *(query-by-committee –QBC-)* pour résoudre le problème de classification des données déséquilibrées. Au début cette méthode divise le jeu de données de la classe majoritaire en plusieurs sous-ensembles en fonction de la proportion

d'échantillons déséquilibrés et elle entraine les sous classifieurs en utilisant la méthode AdaBoost. Ensuite, la méthode génère des échantillons d'apprentissage candidats par la méthode d'apprentissage actif QBC et elle utilise l'algorithme SVM sensible aux coûts pour apprendre les échantillons d'apprentissage.

Cao et al. [144] et Duan et al. [145] ont proposé des méthodes pour estimer les coûts de mauvaise classification lorsque les coûts sont inconnus basés respectivement sur les entropies d'information des classes individuelles et sur la maximisation d'une fonction de remise en forme appropriée par l'utilisation de l'algorithme d'optimisation PSO. L'efficacité de boosting pour la classification des données déséquilibrées est confirmée par Galar et al. [89] et elle a été combinée avec l'algorithme SVM sensible aux coûts introduit par Wang et Japkowicz [96].

IV. Conclusion

Dans ce chapitre, nous avons présenté les trois techniques de classification supervisée standards RNA, SVM et K-PPV. Ces techniques sont les plus utilisées dans la littérature. Nous avons cité aussi les différentes méthodes d'équilibrage proposées dans la littérature pour traiter le problème de classification des données déséquilibrées, elles sont regroupées en trois grandes familles : les techniques de modification des algorithmes de classification, les algorithmes de pondération des données, et les méthodes sensibles aux coûts.

Nous proposons dans le chapitre suivant une méthode de pondération des données basée sur les moindres carrés moyens (LMS), qui pénalise les erreurs des différents échantillons par des poids différents. Cette pondération permet de traiter le problème de classification des données déséquilibrées.

Chapitre 3 : Principe théorique de l'algorithme LMS

I. Introduction

Les techniques d'analyse de données permettent de comprendre des phénomènes et de tirer des informations pertinentes à partir de plusieurs ensembles de données médicales, biologiques ou autres. L'obtention d'une information pertinente à partir des données nécessite une collecte fiable de ces dernières. Nous nous basons sur des approches diverses pour réaliser analyse exploratoire ou de modélisation. Ils existent beaucoup de données qui sont en général atypiques, déséquilibrées et coûteuses. D'où le besoin de bien les exploiter afin de prendre la meilleure décision ou d'établir le meilleur diagnostic.

Le traitement des données passe par une phase de grande importance qui est le filtrage. Cette phase est d'une grande d'intérêt surtout lorsque les données sont bruitées, rares où aussi déséquilibrées. L'objectif principal de filtrage consiste à laisser passer les informations utiles et éliminer les informations indésirables (bruitées). Plusieurs types de filtrage utilisés lors de traitement des données, mais dans notre cadre de travail, nous nous sommes beaucoup plus intéressés au filtrage adaptatif et plus précisément à l'algorithme du gradient stochastique ou aussi appelé l'algorithme des moindres carrés moyens (en Anglais *Least Mean Square* - LMS) pour pondérer les données minoritaires. Le filtrage adaptatif est basé sur la recherche de paramètres optimaux par minimisation d'un critère de performance. En général cette minimisation se fait en recherchant les moindres carrés.

L'algorithme LMS est beaucoup plus utilisé dans le domaine du traitement de signal (annulation de bruits, identification des systèmes, …etc.) [146, 147, 148] et l'égalisation adaptative du canal de communication [147]. Cette approche est aussi utilisée comme une technique d'estimation des signaux (par exemple estimation du signal Electrocardiogramme (ECG) [149, 150]) et comme une règle d'apprentissage pour la correction des erreurs utilisé par le classifieur MLP (Multi Layer Perceptron) [147]….

Nous commençons par quelques définitions de filtrage adaptatif où nous citons quelques caractéristiques. Nous présentons tout d'abord le principe du filtrage adaptatif et aussi nous présenterons brièvement les différents critères de choix de l'algorithme. Ensuite, nous définissons la fonction de minimisation qui est l'erreur quadratique moyenne et les différentes méthodes d'ajustage des paramètres du filtre adaptatif. Par la suite nous détaillons l'algorithme le plus utilisé dans le filtrage adaptatif qui est l'algorithme du gradient stochastique (*Least Mean Square –LMS-algorithm*), où nous parlons du principe de l'algorithme LMS et nous citons aussi quelques

définitions de base et quelques caractéristiques de LMS. Nous terminons ce chapitre par une conclusion.

II. Principe du filtrage adaptatif

Un filtrage est dite adaptatif si ses paramètres (les coefficients) sont modifiés selon un critère donné dès qu'une nouvelle valeur du vecteur devient disponible. Ces modifications doivent suivre l'évolution des systèmes dans leur environnement aussi rapidement que possible [151].

Pour définir un filtre adaptatif, il est nécessaire de [148] :

- Déterminer l'architecture du filtre (FIR, IIR, etc.) ;
- Définir une fonction de minimisation ;
- Choisir un algorithme de correction des paramètres du filtre.

II.1 Conception architecturale du filtre

- Représentation d'un filtre : On peut représenter un filtre par une boite qui prend une information en entrée notée x, la traite et donne une réponse en sortie notée y. Le traitement est modélisé par une fonction de transfert H (voir Figure 3.1) [148].

Figure 3. 1. Diagramme de principe d'un filtre.

- Nous notons plusieurs types de filtres : filtres FIR, IIR, ... Pour cette étude, nous avons choisi deux types de filtres linéaires, les filtres FIR et les filtres IIR.

II.1.1. Filtres non récursifs : FIR

Un filtre FIR (Figure 3.2) est un filtre à réponse impulsionnelle finie (Finite-duration Impulse Response). C'est à dire que, lorsqu'on lui applique une impulsion en entrée, sa réponse contient un

nombre fini de termes non-nuls. De plus, un filtre FIR est nécessairement causal, i.e. il ne dépend pas des échantillons à venir mais uniquement des échantillons passés [148].

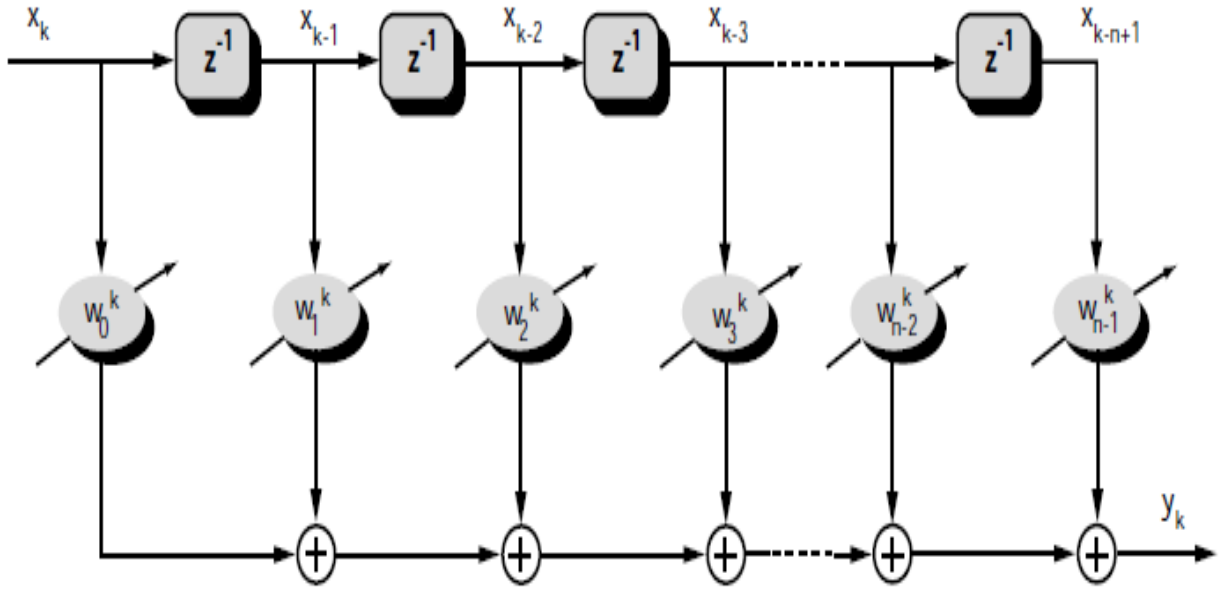


Figure 3. 2. Diagramme de principe d'un filtre à réponse impulsionnelle finie (FIR), z^{-1} représente un retard d'un pas.

Soit $\vec{x}$ le vecteur d'entrée du filtre et $\vec{y}$ sa sortie tels que :

$$\vec{x} = \{x_k, x_{k-1}, \ldots, x_{k-N+1}\} \quad (3.1)$$

$$\vec{y} = \{y_k, y_{k-1}, \ldots, y_{k-N+1}\} \quad (3.2)$$

Où

- x_k représente la valeur de x à l'instant k ;
- et N la taille du filtre concerné.

Soit $\overrightarrow{W_k}$ le vecteur poids à l'instant k du filtre FIR considéré :

$$\overrightarrow{W_k} = \{w_0^k, w_1^k, \ldots, w_{N-1}^k\} \quad (3.3)$$

La sortie du filtre est le scalaire y_k définie par la relation :

$$y_k = \sum_{i=0}^{N-1} w_i^k \cdot x_{k-i} = \vec{W}_k^T . \vec{x} \quad (3.4)$$

II.1.2. Filtres récursifs : IIR

Un filtre IIR est un filtre à réponse impulsionnelle infinie (*Infinite-duration Impulse Response*). La sortie d'un filtre IIR est constituée d'une combinaison linéaire des entrées et des sorties précédentes du filtre (Figure 3.3) [148].

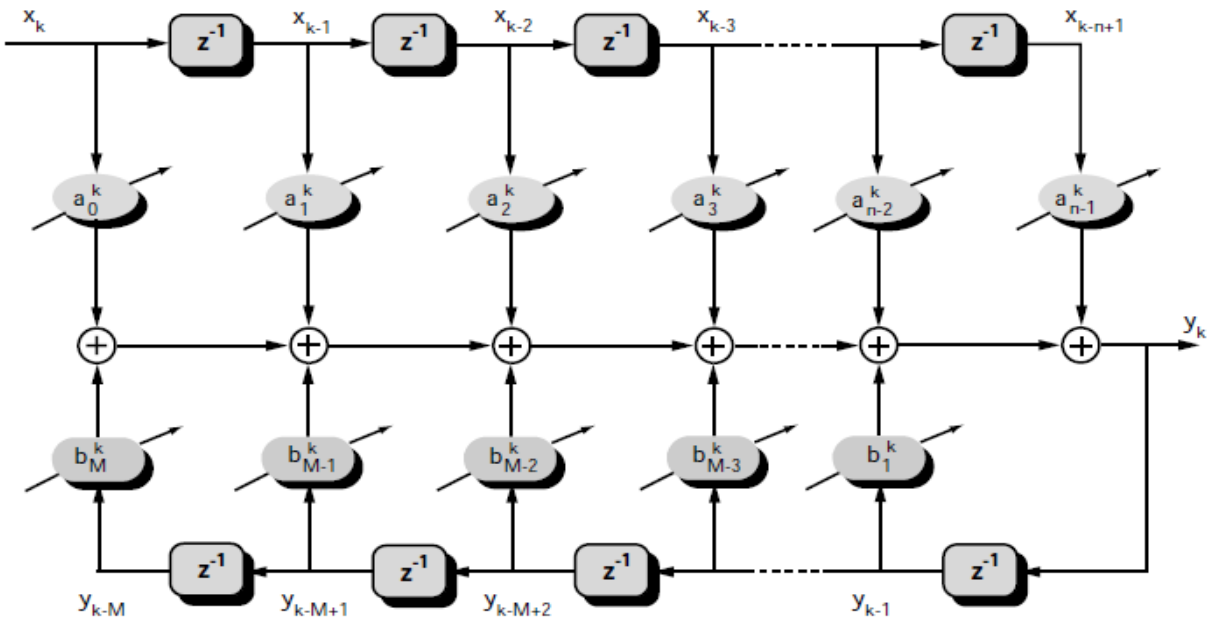


Figure 3. 3. Diagramme de principe d'un filtre a réponse impulsionnelle infinie (IIR).

Un tel filtre n'est pas nécessairement causal. De plus, la nomination IIR fait référence à la capacité du filtre à avoir une réponse impulsionnelle infinie, et n'implique pas nécessairement que c'est le cas. On peut considérer qu'il s'agit d'une mise en garde sur la prédisposition de ce type de filtre à la rétroaction et l'instabilité. Soit A_k et B_k, les vecteurs poids du filtre IIR considéré, à l'instant k [148] :

$$A_K = \{a_0^k, a_1^k, \dots, a_{N-1}^k\} \quad (3.5)$$

Et

$$B_K = \{b_1^k, b_2^k, \dots, b_M^k\} \quad (3.6)$$

La sortie du filtre est le scalaire y_k défini par la relation [148] :

$$y_k = \sum_{i=0}^{N-1} a_i^k . x_{k-i} + \sum_{j=1}^{M} b_j^k . y_{k-j} = A_k^T X_k + B_k^T . Y_{k-1} \quad (3.7)$$

Le filtre IIR est donc défini par une équation de récurrence.

II.1.3. Filtres adaptatifs

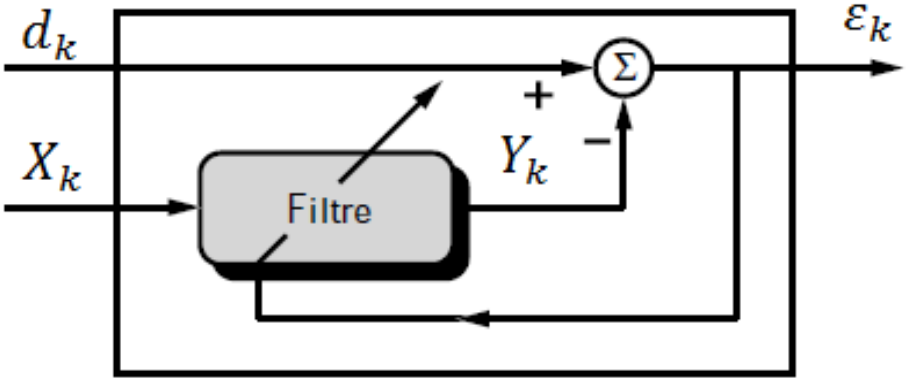


Figure 3. 4. Diagramme de principe d'un filtre adaptatif.

En règle générale, le but d'un système adaptatif est de filtrer le vecteur d'entrée X_k de façon à obtenir en sortie un vecteur désiré d_k. La sortie réelle du filtre est notée Y_k. Le vecteur d'erreur ε_k est l'écart entre le vecteur désiré d_k et le vecteur de sortie du filtre Y_k (Figure 3.4) [148] :

$$\varepsilon_k = d_k - Y_k \qquad (3.8)$$

Cette information est retournée au processus adaptatif qui évalue la similarité des vecteurs au regard d'un critère de performance donné et il modifie la réponse du filtre afin d'améliorer cette similarité (la valeur des poids détermine la réponse du filtre).

Dans la plus part des cas pratiques, le processus tend à minimiser le carré d'une valeur moyenne qui mesure l'écart entre la sortie du système Y_k et une réponse désiré d_k.

Dans un certain nombre de cas, le vecteur en entrée sera la mesure d'une grandeur physique évoluant au cours du temps. On confondra alors l'indice k de l'échantillon courant et l'instant k de la mesure.

- Les entrées du filtre FIR de taille N seront alors les X_{k-i} avec $i = 0, \dots, N-1$.
- Le vecteur d'entrée est défini par $X_k = \left(X_0^k, X_1^k, \dots, X_{N-1}^k\right)^T$, ou par $X_k = (X_k, X_{k-1}, \dots, X_{k-N+1})^T$.
- Le vecteur poids qui définit les paramètres du filtre est donné par $W_k = \left(W_0^k, W_1^k, \dots, W_{N-1}^k\right)^T$.

III.1.3. Principe du filtrage adaptatif

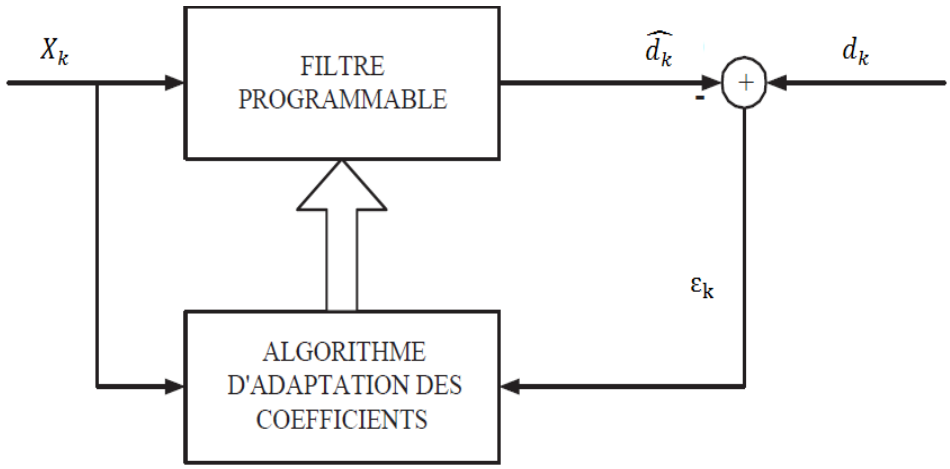


Figure 3. 5. Schéma d'un système de filtrage adaptatif.

Le filtrage adaptatif consiste à approcher le filtre optimal de Wiener par l'utilisation d'une boucle de retour et d'un algorithme de minimisation (voir la Figure 3.5) lorsque les fonctions d'auto et d'inter-corrélation sont inconnues [147].

Afin de formuler le principe du filtrage adaptatif, il est utile de revenir au filtrage de Wiener, mais en se plaçant dans un cadre explicitement discret. Pour ce faire on considère l'observation donnée par une séquence de valeurs $\{X_k, k \in \mathbb{Z}\}$ que l'on filtre par un filtre de réponse impulsionnelle finie de longueur N, décrite par le vecteur $W^T = [W_0, W_1, \dots W_{N-1}]$ pour obtenir en sortie les approximations $\widehat{d_k}$ de la grandeur désirée d_k. On a ainsi [152] :

$$\widehat{d_k} = \sum_{i=0}^{N-1} W_k \, X_{k-i} \qquad (3.9)$$

L'équation de l'erreur correspondante est :

$$\varepsilon_k = d_k - \widehat{d_k} \qquad (3.10)$$

En notant $X^T = [X_k, X_{k-1}, \dots, X_{k-N+1}]$ le vecteur d'état des observations au temps n, l'équation de l'erreur sera donc sous la forme :

$$\varepsilon_k = d_k - W^T X_k \qquad (3.11)$$

Par suite, la puissance de l'erreur $P : = \mathbb{E}\{{\varepsilon_k}^2\}$ a pour valeur :

$$P = \mathbb{E}\left\{ {d_k}^2 - 2 d_k \, W^T X_k + w^T \left(X_k \, {X_k}^T\right) W \right\}$$

$$= \mathbb{E}\left\{{d_k}^2\right\} - 2 \, W^T C_{dx} + W^T \, R_{xx} \, W \qquad (3.12)$$

En introduisant le vcecteur d'inter-corrélation $C_{dx} \coloneqq \mathbb{E}\{d_k\, X_k\}$ entre la sortie désirée et l'observation, et la matrice d'auto-corrélation $\mathrm{R_{xx}} \coloneqq \mathbb{E}\{\mathrm{X_k\, X_k}^{\mathrm{T}}\}$ de cette même observation.

Le filtre optimal et celui qui minimise la puissance de l'erreur. Il est donc comme suit :

$\nabla P = 0$

Ce qui conduit à la solution suivante :

$$W_* = R_{xx}^{-1}\, C_{dx} \qquad (3.13)$$

La puissance minimale de l'erreur associée ayant pour valeur

$$P_* = \mathbb{E}\{\, {d_k}^2\} - W_*^T R_{xx} \mathrm{W}_* \qquad (3.14)$$

II.1.3.2. Choix de l'algorithme approprié

Le choix de l'algorithme se fera selon les critères suivants [147, 153] :

1. La rapidité de convergence qui concerne le nombre d'itérations nécessaires pour arriver à un point de convergence très proche de la solution optimale de Wiener.
2. La mesure de la distance entre cette solution optimale et la solution obtenue.
3. La capacité de poursuite (tracking) des variations du processus. nous examinons quels sont les algorithmes qui sont vraiment adaptatifs.
4. La robustesse au bruit.
5. La complexité d'architecture.
6. La structure (modularité, parallélisme, …).
7. Les meilleures propriétés numériques (stabilité et précision). Il doit être stable avec une précision satisfaisante.

Nous nous intéresserons dans le cadre de notre travail qu'aux trois premiers critères de choix, ces derniers sont vérifiés par l'algorithme le plus utilisé dans le domaine du filtrage adaptatif, i.e. algorithme LMS (Least Mean Squares) [147].

II.2 Erreur quadratique moyenne

Une fois la structure du filtre choisi, il est nécessaire de déterminer une fonction de minimisation ou fonction de coût. Celle-ci sera notamment la fonction de l'erreur en sortie, et éventuellement du module des poids (régularisation), des corrections antérieurs, etc. Prenons le cas

le plus fréquent où l'on va chercher à minimiser le carré de l'erreur quadratique moyenne. Considérons un filtre FIR dont le vecteur d'erreur est défini par [148] :

$$\varepsilon_k = d_k - Y_k = d_k - W^T X_k \quad (3.15)$$

En élevant au carré :

$$\varepsilon_k{}^2 = d_k{}^2 - 2 d_k W^T X_k + W^T X_k X_k^T W \quad (3.16)$$

On prend la moyenne sur un ensemble C des entrées :

$$E[\varepsilon_k{}^2] = E\left[d_k{}^2\right] - 2E[d_k W^T X_k] + E[W^T X_k X_k^T W] \quad (3.17)$$

En supposant la matrice des poids W fixe sur C :

$$E[\varepsilon_k{}^2] = E\left[d_k{}^2\right] - 2W^T . E[d_k X_k] + W^T E[X_k X_k^T] W \quad (3.18)$$

Soit R la matrice de covariance définie par

$$R = E[X_k X_k^T] \quad (3.19)$$

P la matrice de corrélations croisées définie par (la matrice des corrélations croisées entre la sortie de référence et l'entrée du filtre) :

$$P = E[d_k X_k] \quad (3.20)$$

Soit ξ_k la moyenne du carré de l'erreur :

$$\xi_k = E[\varepsilon_k{}^2] = E\left[d_k{}^2\right] - 2W^T . P + W^T . R . W \quad (3.21)$$

Cette équation définie une surface, appelée surface de performance, dont les deux caractéristiques principales sont [148] :

- Seuls les termes de premier et deuxième ordre en W apparaissent. Il s'agit donc d'une fonction quadratique des poids du filtre FIR ;
- Comme c'est l'expression de la moyenne du carré d'une valeur réelle, elle n'est jamais négative.

Étant donné ces caractéristiques, il s'avère que cette surface se réduit à une parabole dans le cas 1-D (seulement un poids), à un paraboloïde dans les cas 2-D (2 poids) et à une hyper paraboloïde qui n'est jamais négative dans le cas N-D.

II.3 D'ajustage des paramètres du filtre

II.3.1. Méthode du gradient déterministe - Cas 2-D :

Si on considère un filtre du second ordre ($W_k = [W_0^k ; W_1^k]^T$), d'après (3.21) la courbe $\xi_k = f(W_k)$ est un paraboloïde concave de section elliptique (contours d'iso-moyenne du carré de l'erreur), et parce qu'elle représente la valeur moyenne du carré d'un réel, elle est toujours positive et orientée vers le haut [148].

Une telle surface aura un minimum global (point le plus bas du paraboloïde) qui correspondra aux paramètres optimaux des poids. Cette valeur peut être déterminée en calculant la valeur pour laquelle le gradient de la surface de performance est nul (plan tangent parallèle au plan $[W_0; W_1]$). Soit dans le cas général [148] :

$$\nabla_{W_k} = \frac{\partial \xi_k}{\partial W_k} = [\frac{\partial \xi_k}{\partial w_1} \frac{\partial \xi_k}{\partial w_2} \dots . \frac{\partial \xi_k}{\partial w_N}]^T = 2\,R.W - 2P \qquad (3.22)$$

Le vecteur $W^\bullet$ qui minimise la moyenne du carré de l'erreur est donc celui qui annule ∇_{W_k} [148] :

$$2\,R.W^\bullet - 2P = 0 \;\Rightarrow\; W^\bullet = R^{-1}\,P \qquad (3.23)$$

C'est la solution de Wiener-Hopf (quel que soit le nombre de dimensions). Lorsque le système minimise la moyenne du carré de l'erreur, le vecteur poids tend vers la valeur optimale $W^\bullet$**.** D'après l'équation (3.23), la matrice optimale des poids du filtre peut toujours être calculée et ne nécessite donc pas l'intervention d'un algorithme qui fasse converger la valeur des poids vers une valeur optimale. Les poids pourraient prendre leur valeur optimale dès le départ. Le filtre serait alors conventionnel (non adaptatif) avec les poids fixés à leur valeur optimale [148].

Pourquoi nous utilisons un système adaptatif ? Les poids du filtre fixés à leurs valeurs calculées ne resteront pas optimaux. Ils devront être réajustés avec la nouvelle valeur de la solution de Wiener-Hopf, ce qui nécessite le calcul de deux matrices, inverser l'une d'entre elle, et les multiplier. Cette charge de calcul ne peut être assumée en temps réel. La méthode de plus profonde descente permet de réduire ce temps de calcul [148].

II.3.2. Méthode de plus profonde descente

II.3.2.1. Principe de la méthode

On sait que le gradient d'une surface (il en va de même pour une hyper surface) est un vecteur du domaine qui a pour sens l'accroissement maximal sur la surface. C'est-à-dire que le vecteur gradient va pointer dans la direction des poids qui vont accroître le plus la valeur de l'erreur. Par conséquent, évoluer le long de la surface dans le sens contraire du gradient (∇) permet de se déplacer vers le minimum de cette surface. Faire évoluer le vecteur poids dans le sens opposé au gradient à chaque itération constitue le principe de la méthode de plus profonde descente. On suit la ligne de plus grande pente de la fonction de coût pour descendre vers le minimum [148] :

$$W_{k+1} = W_k - \mu \nabla_{W_k} = W_k - 2\mu(RW - P) \qquad (3.24)$$

Le pas μ détermine la vitesse de convergence et la stabilité de l'algorithme. Le paramètre μ est appelé facteur de convergence ou d'apprentissage.

Cette méthode présente quelques inconvénients notables [148] :
– Le choix de μ est empirique ;
– Si μ est trop petit, le nombre d'itérations peut être excessivement important ;
– Si μ est trop grand, l'algorithme peut osciller autour du minimum sans converger ;
– Rien ne garantit que le minimum trouvé soit un minimum global.

II.3.2.2. Coût en calcul de l'algorithme

Variables à calculer	Y_k	ε_k	W_{k+1}	***Total***
Multiplications	N	0	N+1	2N+1
Additions	N-1	1	N	2N

Tableau 3.1. Nombre d'opérations de calcul pour chacun des paramètres de l'algorithme de descente de gradient

La méthode de plus profonde descente nécessite donc de l'ordre de $2N$ +1 multiplications et $2N$ additions (Tableau 3.1.) [148].

II.3.2.3. Stabilité de la méthode

Soit $W^{\bullet}$ la solution de Wiener-Hopf, on définit l'erreur sur les poids par $C_k = W_k - W^{\bullet}$ [148] :

$$W_{k+1} = W_k - 2\mu(RW_k - P)$$
$$W^{\bullet} = W^{\bullet} - 2\mu(RW^{\bullet} - P)$$
$$\mathrm{C_{k+1}} = \mathrm{C_k} - 2\mu(\mathrm{RC_k})$$

On a donc la relation suivante :

$$\mathrm{C_{k+1}} = (1 - 2\mu\mathrm{R})\mathrm{C_k} = \cdots = (1 - 2\mu\mathrm{R})^{\mathrm{k+1}}\mathrm{C_0} \qquad (3.25)$$

Cette série est stable si l'expression $(1 - 2\mu R)^k$ converge vers 0.

On introduit la décomposition aux valeurs propres de R :

$$R = L\Lambda L^{-1} \qquad (3.26)$$

Où Λ est la matrice diagonale de R et L la matrice de passage correspondante.

On peut donc écrire :

$$1 - 2\mu R = L.L^{-1} - 2\mu\, L\Lambda\, L^{-1}$$
$$= L(1 - 2\mu\,\Lambda)L^{-1}$$

On a donc que $\Lambda_\mu = [1 - 2\mu\,\Lambda]$ est la matrice des valeurs propres de $(1 - 2\mu R)$. Donc :

$$(1 - 2\mu R)^k = L(\,\Lambda_\mu\,)^k L^{-1} \qquad (3.27)$$

On procède au changement de variable $V_k = L^{-1}c_k$. D'après les équations (3.25) et (3.27) on peut donc écrire :

$$V_k = (\,\Lambda_\mu\,)^k.V_0 \qquad (3.28)$$

La condition de stabilité s'écrit donc :

$$\left|\lambda_{\mu,i}\right| < 1 \text{ , pour } i = 1 \ldots N \qquad (3.29)$$

Or nous avons vu que $\Lambda_\mu = [1 - 2\mu\Lambda]$. On peut donc réécrire la condition de stabilité de la manière suivante :

$|1 - 2\mu\lambda_i| < 1$, pour $i = 1 \ldots N$

$\Rightarrow 0 < \mu\lambda_i < 1$

L'algorithme de plus profonde descente est stable si :

$$0 < \mu < \frac{1}{\lambda_{max}} \qquad (3.30)$$

II.3.2.4. Convergence de l'algorithme

On revient à l'erreur c_k, entre les paramètres courants du filtre W_k et l'optimum $W^\bullet$, définie par [148] :

$c_k = W_k - W^\bullet$ Où $W^\bullet = R^{-1}P$

On a donc d'après (3.24) :

$$c_{k+1} = c_k - 2\mu(RW_k - P)$$
$$= c_k - 2\mu(Rc_k + RW^\bullet - P)$$
$$= (1 - 2\mu R)c_k$$

On a donc la formule de récurrence suivante :

$$c_{k+1} = (1 - 2\mu R)^{k+1}\, c_0 \qquad (3.31)$$

Le carré de la norme de l'erreur c_k est donc défini par :

$$\|c_k\|^2 = c_k^T c_k$$
$$= c_0^T\, (1 - 2\mu R)^{2k}\, c_0$$

Par définition, R est une matrice réelle symétrique. R est donc diagonalisable (3.26) dans une base orthonormale. Il existe donc une matrice de passage orthogonale telle que :

$L^{-1} = L^T$

De plus, la puissance entière *n* d'une matrice diagonalisable est donnée par :

$R^n = \mathrm{L}\,\Lambda^n \mathrm{L}^{-1}$

Enfin, lorsque nous élevons une matrice diagonale à une puissance revient à élever chacun des coefficients qui la compose à cette même puissance.

On introduit ces résultats dans l'équation (3.31) ce qui nous permet d'écrire :

$$(1 - 2\mu\mathrm{R})^{2\mathrm{k}} = \mathrm{L}(1 - 2\mu\Lambda)^{2\mathrm{k}}\mathrm{L}^{\mathrm{T}} \qquad (3.32)$$

On a donc

$$\|\mathrm{c_k}\|^2 = \mathrm{c_0^T\, L}(1 - 2\mu\Lambda)^{2\mathrm{k}}\mathrm{L^T\, c_0}$$
$$= \sum_{i=0}^{N-1}(1 - 2\mu\lambda_i)^{2k}\, {\zeta_i}^2 \qquad (3.33)$$

Où $\boldsymbol{\zeta}_i$ est la i-ème composante du vecteur ζ définie par :

$\zeta = L^T\, c_0$

Si l'on fait l'approximation $(1 - 2\mu\lambda_i) \approx e^{-2\mu\lambda_i}$ alors l'équation (3.33) devient :

$\|\mathrm{c_k}\|^2 = \sum_{i=0}^{N-1} {\zeta_i}^2 . e^{-4k\mu\lambda_i}$

On peut donc comparer l'erreur à une somme de modes de constantes de temps $\tau_i = (\mu\lambda_i)^{-1}$ différentes :

$$\|c_k\|^2 = \sum_{i=0}^{N-1} \zeta_i{}^2 . e^{-4\frac{k}{\tau_i}} \qquad (3.34)$$

Si le paramètre μ est optimisé pour la convergence suivant la relation (3.30), on a :

$$\tau_i = \frac{\lambda_{max}}{\lambda_i} \qquad (3.35)$$

Le taux de convergence initial (k faible) est donc dominé par les modes les plus rapides (λ_{max}). En revanche, le taux de convergence en fin d'apprentissage (k important) est dominé par les modes les plus lents (λ_{min}). Pour des petits λ_{min} la convergence de l'algorithme peut être très lente. Globalement, la convergence sera d'autant plus lente que la matrice R de corrélation du vecteur d'entrée présentera des valeurs propres dispersées [148].

L'algorithme sera ralenti en début d'apprentissage par ses λ_{max} importants, et à nouveau ralenti en fin d'apprentissage par des λ_{min} trop faibles. On remarque donc que la qualité de la convergence va dépendre des données en entrée (au travers de R) [148].

La méthode de plus profonde descente constitue une méthode adaptative pour atteindre le minimum, mais conserve le problème de la charge de calcul. En effet, calculer le gradient $\nabla = 2R.W - 2P$ nécessite le calcul de deux matrices P et R. Ce qui rend l'algorithme inutilisable pour des applications temps réel. On notera qu'il existe d'autres méthodes basées sur ce principe de descente vers le minimum global que nous ne développerons pas ici. Les méthodes les plus utilisées sont [148] :

- la méthode de Gauss-Newton qui tient compte de la concavité de la surface d'erreur. Elle a besoin de calculer les dérivées secondes de la fonction coût ce qui permet d'augmenter le temps de calcul ;
- la méthode du gradient conjugué où les directions des corrections successives doivent vérifier des conditions d'orthogonalité.

III. Algorithme LMS - Méthode du gradient stochastique

III.1 Définitions de base

L'algorithme du gradient stochastique où aussi nommé l'algorithme des moindres carrés moyens (*Least Mean Squares* – LMS-) est relativement facile à implémenter et repose sur un principe assez simple a été proposé par Widrow et Hoff en 1960 [148, 154].

La version de base du LMS est un cas spécial du filtre adaptatif du gradient descendant (steepest descent) bien connu. Le but de cette technique est de réduire au minimum une fonction de coût quadratique en mettant à jour itérativement des poids de sorte qu'ils convergent à la solution optimale.

Dans l'algorithme LMS, les carrés des erreurs quadratiques moyennes sont minimisés par la résolution d'un système d'équations linéaires et les erreurs de tous les échantillons de données sont de même coût [147, 155, 156].

La clef de l'algorithme LMS est d'utiliser une approximation du gradient plutôt que de calculer sa valeur exacte. On ne tient compte que de l'échantillon courant et non plus de l'ensemble des échantillons d'entrée [148] :

$$\overline{\nabla}_{w_k} = \frac{\partial}{\partial w_k} (\varepsilon_k{}^2) \qquad (3.36)$$

On obtient alors la formule d'actualisation des poids suivants :

$$W_{k+1} = W_k + \mu\, \varepsilon_k\, X_k \qquad (3.37)$$

Cette équation constitue le cœur de l'algorithme LMS de Widrow-Hoff, utilisé à l'origine comme outil d'identification de systèmes linéaires. Cet algorithme est souvent appelé algorithme du gradient stochastique dans la mesure où l'on a substitué la valeur instantanée du gradient à sa valeur moyenne et que l'on calcule donc un gradient $\overline{\nabla}^k$ bruité. Il est possible que la diminution de l'erreur en un point soit compensée par une augmentation de l'erreur pour les autres points. On suppose donc que des ajustements locaux vont finir par converger vers une solution globale [148].

L'algorithme LMS est l'un des plus utilisés parmi les algorithmes de filtrage adaptatif pour de multiples raisons [148] :

- Historiquement, il est considéré comme un algorithme ancien.
- Il est très simple à utiliser ;
- Lors des expérimentations, il donne de bons résultats (même si sa convergence est parfois lente) ;
- Il requière un nombre limité de calculs ;
- Il permet d'actualiser les poids lors de la présentation de chaque nouvel échantillon en entrée, permettant ainsi une adaptation permanente du filtre ;
- Il s'adapte très bien à des changements progressifs des grandeurs statistiques du vecteur.
- La complexité de son implémentation est extrêmement réduite.

III.2 Processus de l'algorithme LMS

Nous avons un ensemble composé de n échantillons distribués d'une manière identique et indépendante [147, 153, 154, 155, 156] :

$$s = \{(x_1, y_1), \dots (x_n, y_n)\} \tag{3.38}$$

$X_k \in R^d$ et $y_k \in \{-1,1\}$, $k = 1,2 \dots, n$. La classification consiste à trouver un hyperplan $W.X + b = 0$ là où $W \in R^d$ et $b \in R$.

Lors d'une classification, la solution donnée par l'algorithme des moindres carrés moyens peut être trouvée par la résolution d'une minimisation quadratique :

$$\min_w \frac{1}{n} \sum_{i=1}^{n} (Y_k - W.X_k)^2 \tag{3.39}$$

L'algorithme LMS est certainement l'algorithme adaptatif le plus populaire qui existe en raison de sa simplicité.

Puisque $R = E\{X_k X_k{}^T\}$ et $P = E\{X_k Y_k\}$ sont inconnus, on approchera ces grandeurs déterministes par des valeurs estimées $\widehat{R_k}$ et $\widehat{P_k}$ à l'instant k. Dans le cas du LMS, on choisit les valeurs estimées les plus simples possibles, à savoir :

$$\begin{aligned} \widehat{R_k} &= X_k X_k{}^T \\ \widehat{P_k} &= X_k Y_k \end{aligned} \tag{3.40}$$

Ce sont simplement les valeurs estimées instantanées des corrélations.

En remplaçant $\widehat{R_k}$ et $\widehat{P_k}$ dans l'algorithme du gradient déterministe suivant :

$$W_{k+1} = W_k - \frac{1}{2} \mu \, g_k \tag{3.41}$$

Où

$$g_k = \partial J[W_k]/\partial W_k$$
$$= -2E\{X_k\,\varepsilon_k\}$$
$$= -2P + 2R\,W_k \qquad (3.42)$$

Est le gradient de la fonction coût $J[w_k] = E\{\varepsilon_k{}^2\}$. Cet algorithme peut encore s'écrire en utilisant l'erreur :

$$\varepsilon_k = y_k - X_k{}^T W_k \qquad (3.43)$$
$$W_{k+1} = W_k + \mu\,E\{X_k\,\varepsilon_k\} \qquad (3.44)$$

On obtient :

$$W_{k+1} = W_k + \mu\,[\,\widehat{P_k} - \widehat{R_k} W_k\,]$$
$$= W_k + \mu\,X_k\,[\,y_k - X_k{}^T W_k\,]$$
$$= W_k + \mu\,X_k\,\varepsilon_k \qquad (3.45)$$

Qui est l'algorithme LMS. On remarquera que W_k est maintenant une variable aléatoire [puisqu'à chaque nouvelle itération k, W_k dépend des processus aléatoires X_k et y_k].

A chaque itération de l'algorithme LMS, les coefficients du filtre adaptatif sont mis à jour selon une technique de descente de gradient afin de tendre vers la solution optimal.

III.3 Principe de l'algorithme LMS

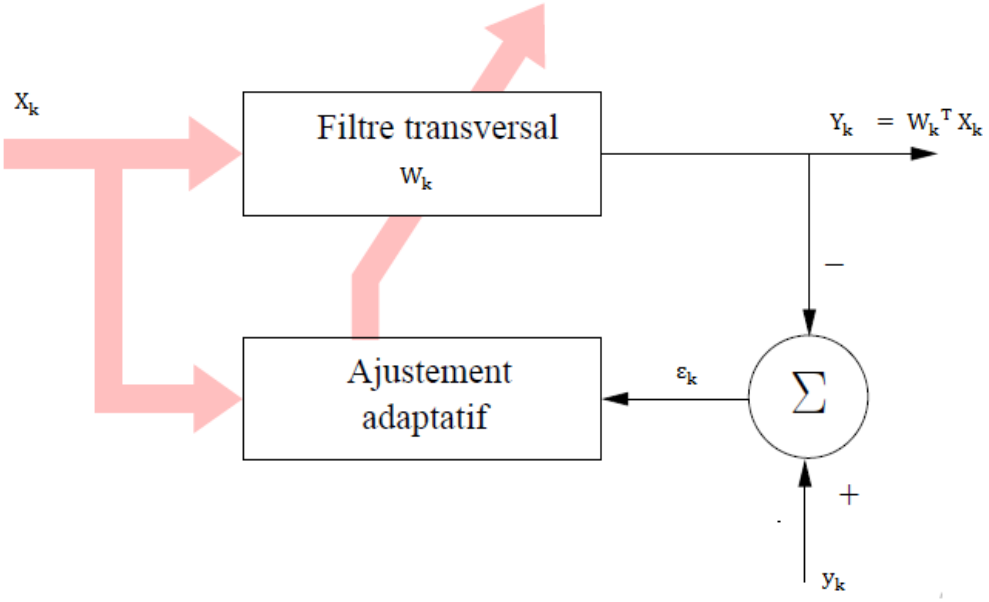


Figure 3. 6. Filtrage adaptatif basé sur l'algorithme LMS.

L'algorithme du gradient stochastique (GS) ou *Least Mean Square* (LMS) est l'un des premiers algorithmes d'estimation utilisé en identification des systèmes. Il est très proche du concept d'approximation stochastique en statistique développé par Robbins et Monro [157]. C'est aussi l'algorithme le plus utilisé dans les applications de filtrage adaptatif grâce à sa simplicité de mise en œuvre et à son coût de calcul extrêmement réduit. L'algorithme LMS est une méthode très simple pour former un système adaptatif linéaire pour réduire au minimum les carrés des erreurs quadratiques moyennes [151].

Cet algorithme est composé de deux parties de base : une partie de filtrage linéaire, qui permet de générer la sortie du filtre linéaire en présence d'une observation d'entrée, et une partie adaptative permettant d'ajuster automatiquement les paramètres du filtre, en se basant sur un critère lié à l'erreur d'estimation. En d'autres termes, à chaque instant n, un nouveau vecteur d'entrée X_K est traité par l'algorithme LMS pour actualiser le vecteur W_k, réponse du filtre estimateur ou vecteur d'état qui peut être considéré aussi comme étant la réponse impulsionnelle du vecteur qu'on cherche à estimer. Les itérations de cet algorithme sont résumées comme suit [147, 151] :

Algorithme 3.1. Algorithme de LMS

Initialisation des coefficients du filtre : W_0.

Pour toute séquence : k=1 : N

Appliquer le vecteur X_k en entrée de l'algorithme

Calcul de la sortie du filtre : $Y_k = W_k^T X_k$

Calcul de l'erreur : $\varepsilon_k = y_k - X_k^T W_k$

Mise à jour des poids du filtre : $W_{k+1} = W_k + \mu X_k \varepsilon_k$

Fin pour

Où W_k est le vecteur de coefficients du filtre estimateur qui représente la réponse du vecteur à l'instant n. W_0 prend une valeur arbitraire, généralement égale à 0 et μ est le pas d'adaptation de l'algorithme dit aussi le gradient de l'algorithme LMS.

III.4 Coût de l'algorithme en calcul

Variables à calculer	Y_k	ε_k	W^{k+1}	***Total***
Multiplications	N	0	1	N+1
Additions	N-1	1	1	N+1

Tableau 3. 2. Nombre d'opérations de calcul pour chacun des paramètres de l'algorithme LMS

L'adaptation des poids du filtre requière donc le même temps de calcul que celui du calcul de la sortie du filtre (Tableau 3.2). La quantité de calcul nécessaire a été réduite à 2N additions-produits (N additions et N multiplications) ce qui fait de cet algorithme un algorithme particulièrement approprié aux applications temps réel. Où N est le nombre de coefficients du filtre [148].

III.5 Facteur de convergence

Dans l'algorithme LMS, le facteur de convergence μ joue un rôle prépondérant dans sa stabilité et la rapidité avec laquelle la matrice des poids du filtre va converger vers l'optimum du système. Il peut également varier avec le temps, même si généralement on le garde constant à une valeur déterminée par l'expérimentation pour une application donnée [148].

Une valeur importante de μ va accélérer la convergence et l'adaptativité de l'algorithme, mais peut-être moins stable, autour de la valeur minimale. En revanche, les valeurs plus faibles de μ vont favoriser la stabilité du filtre autour de l'optimum [148].

Les algorithmes les plus utilisés ont généralement fait l'objet d'études analytiques de stabilité, on dispose donc des méthodes de calcul pour borner le facteur de convergence et éviter une divergence de l'algorithme. Lorsqu'il n'est pas possible d'effectuer à priori ce calcul, une solution pratique consiste à augmenter progressivement μ à partir de 0 jusqu'à obtenir la divergence de l'algorithme, on diminue ensuite la valeur de μ jusqu'à une valeur typiquement égale à la moitié de sa valeur de divergence [148].

III.6 Convergence en moyenne

On mesure l'écart l_k entre le vecteur poids W^k et la solution de Wiener-Hopf $W^\bullet$[148, 152,153] :

$$l_k = W^k - W^\bullet \tag{3.46}$$

$$\begin{aligned} W^{k+1} &= W^k - \mu\,(X_k.X_k^T.W^k - X_k.d_k) \\ W^\bullet &= W^\bullet - \mu\,(\mathrm{E}[X_k.X_k^T].W^\bullet - E[X_k.d_k]) \\ \hline l^{k+1} &= l^k - \mu(X_k.X_k^T.W^k - E[X_k.X_k^T].W^\bullet - (X_k.d_k - E[X_k.d_k])) \end{aligned}$$

On a donc :

$$\begin{aligned} E\left[l^{k+1}\right] &= E\left[l^k\right] - \mu.\left(E[X_k.X_k^T]E[l^k] - 0\right) \\ &= E\left[l^k\right] - \mu.R.E[l^k] \\ &= (1 - \mu R)\,.E[l^k] \end{aligned} \tag{3.47}$$

On a donc la relation suivante :

$$E\left[l^{k+1}\right] = (1 - \mu R)\,.E[l^k] = \cdots = (1 - \mu\mathrm{R})^{k+1}E[l^0] \tag{3.48}$$

Cette série est stable si l'expression $(1 - \mu R)^k$ converge vers 0.

On introduit la décomposition aux valeurs propres de R :

$$R = L\Lambda\, L^{-1} \tag{3.49}$$

Où Λ est la matrice diagonale de R et L la matrice de passage correspondante.

On peut donc écrire :

$$\begin{aligned} 1 - \mu R &= L.L^{-1} - \mu\, L\Lambda\, L^{-1} \\ &= L(1 - \mu\,\Lambda\,)L^{-1} \end{aligned}$$

On a donc que $\Lambda_\mu = [1 - \mu\,\Lambda]$ est la matrice des valeurs propres de $(1 - \mu R)$. Donc :

$$(1 - \mu R)^k = L(\,\Lambda_\mu\,)^k L^{-1} \tag{3.50}$$

On procède au changement de variable $V_k = L^{-1}c_k$. D'après les équations (3.48) et (3.50) on peut donc écrire :

$$V_k = (\,\Lambda_\mu\,)^k.V_0 \tag{3.51}$$

La condition de stabilité s'écrit donc :

$$|\lambda_{\mu,i}| < 1 \text{ , pour } i = 1 \ldots N \tag{3.52}$$

Or nous avons vu que $\Lambda_\mu = [1 - \mu\Lambda]$. On peut donc réécrire la condition de stabilité de la manière suivante :

$|1 - \mu\lambda_i| < 1$, pour $i = 1 \dots N$

$\Rightarrow 0 < \mu\lambda_i < 2$

L'algorithme LMS est stable si :

$$0 < \mu < \frac{2}{\lambda_{max}} \qquad (3.53)$$

Où $\lambda_{\max}$ représente la valeur propre maximale de la matrice d'auto corrélation (la matrice de covariance) $\widehat{R_k}$.

La convergence de l'algorithme est inversement proportionnelle à la propagation des valeurs propres de la matrice d'auto corrélation $\widehat{R_k}$. Pour des valeurs propres de $\widehat{R_k}$ qui sont très répandues, la convergence peut être lente.

IV. Conclusion

Parmi les nombreux algorithmes d'actualisation des poids conçus pour les filtres adaptatifs, nous nous sommes intéressés aux algorithmes de minimisation de l'erreur quadratique en sortie, de type LMS, qui constitue l'approche la plus simple et la plus robuste pour faire varier les coefficients.

Dans ce chapitre, nous avons présenté le principe de filtrage adaptatif et quelques définitions de base. Puis nous avons décrit la structure de l'algorithme du gradient stochastique LMS ainsi que leur principe, propriétés et performances.

Dans le chapitre suivant, nous allons proposer une méthode de pondération des données basée sur l'algorithme des moindres carrés moyens (LMS), qui pénalise les erreurs des différents échantillons par des poids différents, où les données minoritaires sont pondérées. Pour surmonter le problème de l'apprentissage de l'ensemble de données déséquilibrées. Pour la suite de ce travail, nous allons étudier l'application de cette méthode sur différentes ensembles de données médicales déséquilibrées avec différents classifieurs et nous discutons par la suite les résultats obtenus.

Chapitre 4 : Résultats et discussion

I. Introduction

Le déséquilibre dans la répartition des données, et en particulier dans les données médicales, cause un problème majeur lors de la classification. En effet, nous rencontrons souvent des classes de pathologies minoritaires qui sont mal représentées lors de la phase d'apprentissage. Afin de remédier à ce problème, nous proposons dans le cadre de notre thèse de doctorat une méthode de pondération LMS (Least Mean Square), qui pénalise les erreurs des différents échantillons par des poids différents.

Dans ce chapitre nous présentons nos différentes contributions proposées pour traiter le problème de la classification des données médicales déséquilibrées, pour cela nous divisons ce chapitre en 4 parties où chacune d'elles traite des résultats d'une contribution proposée :

- Partie 1 : Etude de l'intérêt de la pondération des données minoritaires (Contribution 1).
- Partie 2 : Etude de l'intérêt de la pondération des données médicales en utilisant d'autres classifieurs (Contribution 2).
- Partie 3 : Etude de l'intérêt de la méthode de pondération LMS par rapport à d'autres méthodes existantes (Contribution 3).
- Partie 4 : Comparaison de nos résultats avec l'état de l'art

Pour l'implémentation de ces étapes, nous avons fait recourt au logiciel : Matlab 2014a.

II. Description des ensembles de données

Dans cette thèse de doctorat, nous utilisons sept ensembles de données médicales extraites du dépôt d'UCI (A. Frank and A. Asuncion) [158]. Ces ensembles de données sont souvent utilisées par les méthodes de fouille de données pour analyser et valider ces approches. Parmi ces sept ensembles de données nous avons cinq ensembles binaires (Pima Indian Diabetes (PIMA), Wisconsin Breast Cancer (WBC), Wisconsin Diagnostic Breast Cancer (WDBC), Liver Disorder (LD) et Appendicitis) et deux multi- classes (Breast Tissue avec 4 Classes (BT- 4 Classes) et Breast Tissue avec 6 Classes (BT- 6 Classes)). Les principales caractéristiques de ces ensembles de données sont représentées dans Tableau 4.1 :

Ensembles de données	*Classes*	*Attributs*	*Instances*	*Classe Minoritaire*	*Classes Majoritaires*
Pima	2	8	768	268	500
WBC	2	9	683	137	546
WDBC	2	31	569	212	357
LD	2	6	345	145	200
Appendicitis	2	7	106	21	85
BT- 4 Classes	4	9	106	Car (21)	(Fad , Mas, Gla)+ Con + Adi (85)
BT- 6 Classes	6	9	106	Car (21)	Fad + Mas + Gla+ Con + Adi (85)

Tableau 4. 1. Caractéristiques des ensembles de données utilisés

Ces ensembles de données utilisés ont été largement employées dans le domaine de la classification, nous présentons ci-dessous une description succincte de ces ensembles :

II.1 PIMA

L'ensemble de données médicale Indians Pima du Diabetes a été choisi du dépôt d'UCI (A. Frank and A. Asuncion) qui réalise une étude sur 768 femmes Indiennes. Le diagnostic est une valeur binaire variable «classe» qui permet de savoir si le patient montre des signes de diabète selon les critères de l'Organisation Mondiale de la Santé. Ces exemples sont répartie en deux classe (500 exemples appartiennent à la classe 0 (Classe non diabétique), 268 exemples appartiennent à la classe 1 (Classe diabétique)). Les huit descripteurs cliniques sont :

1. Npreg : nombre de grossesses (Ngross).
2. Glu : concentration du glucose plasmatique (mg/dl).
3. BP : tension artérielle diastolique (mm Hg) (PAD).
4. SKIN : épaisseur de pli de peau du triceps (mm).
5. Insuline : dose d'insuline (mu U/ml).
6. BMI : index de masse corporelle (poids en kg/ m^2).
7. PED : fonction de pedigree de diabète (l'hérédité).
8. Age : âge (Année).

II.2 Wisconsin Breast Cancer (WBC)

L'ensemble de données du cancer du sein dénommé « Wisconsin Breast Cancer Dataset » a été collecté à l'Université du Wisconsin. Il contient 699 échantillons avec nef descripteurs qui sont détaillés dans le Tableau 4.2. Les mesures sont assignées à une valeur entière comprise entre 1 et 10 (1 étant le plus proche de bénigne et 10 le plus anaplasique (proche de maligne)).

Malheureusement, il existe 16 cas parmi les 699 cas avec des données manquantes d'attributs. Après élimination de ces cas, nous obtenons un ensemble de 683 patientes. Ces exemples sont répartie en deux classe (546 exemples appartiennent à la classe 2 (Classe bénigne), 137 exemples appartiennent à la classe 4 (Classe maligne)).

Attributs	***Domaine***
1. Clump Thickness	1 – 10
2. Uniformity of Cell Size	1 - 10
3. Uniformity of Cell Shape	1 – 10
4. Marginal Adhesion	1 - 10
5. Single Epithelial Cell Size	1 - 10
6. Bare Nuclei	1 - 10
7. Bland Chromatin	1 - 10
8. Normal Nucleoli	1 - 10
9. Mitoses	1 – 10

Tableau 4. 2. Ensemble de données Wisconsin breast cancer (description des attributs).

II.3 Wisconsin Diagnostic Breast Cancer (WDBC)

L'ensemble de données du cancer du sein dénommé « Wisconsin Diagnostic Breast Cancer Dataset » a été choisi du dépôt d'UCI. Il a été créé par William H. Wolberg, W.Nick Street et Olvi. L. Mangasarian. Les caractéristiques des noyaux de cellules dans le sein sont extraites à partir d'une image scannée. L'ensemble de données WDBC contient 569 cas avec 31 descripteurs. Ces exemples sont répartis sur deux classes (357 exemples appartiennent à la classe 0 (Classe bénigne), 212 exemples appartiennent à la classe 1 (Classe maligne)).

II.4 Liver disorder

L'ensemble de données du trouble hépatique du foie dénommé « Liver Disorder » a été choisi du dépôt d'UCI. Il est utilisé dans cette étude pour évaluer et tester les performances de l'algorithme proposé. Cet ensemble de données contient 345 cas avec 6 attributs. Ces exemples sont répartis sur deux classes (1 pour le diagnostic positif (145 cas), 2 pour le diagnostic négatif (200 cas)). Les principales caractéristiques de cet ensemble de données sont présentées dans le Tableau 4.3 :

Attributs	***Signification***
MCV	mean corpuscular volume
Alkphos	alkaline phosphotase
SGPT	alamine aminotransferase
SGOT	aspartate aminotransferase
Gammagt	gamma-glutamyl transpeptidase
Drinks	number of half-pint equivalents of alcoholic beverages drunk per day (consummation alcoolique)

Tableau 4. 3. Ensemble de données Liver disorder (description des attributs).

II.5 Appendicitis

Cet ensemble de données a été créé par Shalom Weiss [159]. Il est souvent utilisé pour valider et tester l'approche proposée. Cet ensemble de données contient 106 échantillons avec sept descripteurs numériques. Ces exemples sont répartis sur deux classes (1 pour le diagnostic positif (80,2%), 0 pour le diagnostic négatif (19,8%)). Les 7 caractéristiques sont décrites dans le Tableau 4.4 :

Attributs	***Description***
1	WBC1
2	MNEP
3	MNEA
4	MBAP
5	MBAA
6	HNEP
7	HANEA

Tableau 4. 4. Ensemble de données Appendicitis (description des attributs).

II.6 BT- 6 Classes

Cet ensemble de données recueille des mesures de spectroscopie d'impédance électrique effectuée sur des échantillons de tissu du sein. Chacun de ces échantillons appartient à l'une des 6 classes possibles (carcinome, fibro-adénome, mastopathie, glandulaire, conjonctif et adipeux). Cet ensemble contient 106 instances et 9 attributs. Les caractéristiques sont décrites dans le Tableau 4.5 :

Attributs	***Information***
I0	Impedivity (ohm) at zero frequency
PA500	phase angle at 500 KHz
HFS	high-frequency slope of phase angle
DA	impedance distance between spectral ends
AREA	area under spectrum
A/DA	area normalized by DA
MAX IP	maximum of the spectrum
DR	distance between I0 and real part of the maximum frequency point
P	length of the spectral curve

Tableau 4. 5. Ensemble de données BT- 6 Classes (description des attributs).

Les classes de tissu sont représentées comme suit :

1. Carcinoma (Car) : 21 cas.
2. Fibro-adenoma (Fad) :15 cas.
3. Mastopathy (Mas) :18 cas.
4. Glandular (Gla) :16 cas.
5. Connective (Con) : 14 cas.
6. Adipose (Adi) : 22 cas.

II.7 BT- 4 Classes

Cet ensemble de données recueille des mesures de spectroscopie d'impédance électrique effectuée sur des échantillons de tissu du sein. Chacun de ces échantillons appartiennent à l'un des 4 classes possibles (carcinoma (Car), fibro-adénome (Fad) + mastopathie (Mas) + glandulaire

(Gla), conjonctif (Con) et adipeux (Adi)). Cet ensemble de données contient 106 instances et 9 attributs. Il comporte les mêmes caractéristiques que L'ensemble de BT- 6 Classes (ensemble originale) (voir le Tableau 4.5). Les classes de tissue sont représentées comme suit :

1. Car : 21 cas
2. Fad + Mas + Gla : 49 cas (*les classes sont rassemblés dans la même classe parce que ne peuvent pas être distingués de façon précise*).
3. Con : 14 cas
4. Adi : 22 cas

III. Critères d'évaluation

Pour évaluer la capacité de prédiction des modèles construits, nous utilisons quatre critères d'évaluation et ils sont définis comme suit :

1) **Le taux de classification correcte (TC%)** : est le taux de reconnaissance ;

$$TC = (VP + VN) / (VP + VN + FP + FN) * 100 \quad (4.1)$$

Où vrais positifs (VP) désignent la classification correcte des échantillons positifs, vrais négatifs (VN) désignent la classification correcte des échantillons négatifs. Faux Positifs (FP) désignent la classification erronée des échantillons négatifs dans les échantillons positifs. Faux Négatifs (FN) désignent La classification erronée des échantillons positifs dans les échantillons négatifs.

2) **Sensibilité (SE%) :** Le pourcentage d'échantillons positifs qui sont correctement classés ;

$$SE = VP / (VP + FN) * 100 \quad (4.2)$$

3) **Spécificité (SP%) :** le pourcentage d'échantillons négatifs qui sont correctement classés ;

$$SP = VN / (VN + FP) * 100 \quad (4.3)$$

4) **Gmean (Geometric mean) :** Il fournit un moyen simple d'évaluer la capacité du modèle de classer correctement la classe minoritaire et majoritaire par la combinaison de sensibilité et spécificité dans un seul métrique. Gmean est considéré comme une mesure de la précision équilibrée et il est défini comme suit :

$$\text{Gmean} = \sqrt{\text{Sensibilité} * \text{Spécificité}} \quad (4.4)$$

IV. Contribution 1 : Intérêt de la pondération des données minoritaires

Cette section consiste à créer un déséquilibre entre les classes dans l'ensemble d'apprentissage par la diminution du nombre des cas malades et augmentation du nombre des cas non malades. Nous appliquons un algorithme de classification neuronal multicouche avec et sans la méthode de pondération proposée LMS sur les différents basses de données crées. Sa phase d'apprentissage est réalisée par l'algorithme de Levenberg-Marquardt.

Nous nous étudions ici, l'intérêt de l'application de la méthode de pondération LMS et si elle apporte une amélioration à la classification sur les différents degrés de déséquilibre. Pour évaluer les performances des modèles construits nous avons utilisé les quatre critères d'évaluation citées précédemment (voir la section III Critères d'évaluation).

La procédure de classification des données médicales déséquilibrées utilisée dans cette section est représentée par les étapes suivantes :

- Première étape : Classification neuronale des données médicales déséquilibrées.
- Deuxième étape : Classification neuronale des données médicales équilibrées par l'algorithme de pondération LMS ; Cette étape est scindée en deux parties :
  - Application de l'algorithme de pondération LMS pour surmonter le problème de déséquilibre des données. Dans ce travail, nous choisissons le pas d'adaptation μ de l'algorithme LMS égal 0.5 via expérimentation.
  - Classification neuronale des données équilibrées.
- Troisième étape : Comparaison entre les deux approches c.-à-d. étude comparative entre les résultats obtenus avant et après l'équilibrage.

Remarque : *les tests sont réalisés sur l'ensemble de données PIMA.*

IV.1 Classification neuronale des données médicales déséquilibrées

Dans cette étape, nous avons utilisé l'ensemble de données PIMA et le Classifieur Neuronal Multicouche (CNMC) en prenant en considération les expérimentations suivantes *:*

1. Expérimentation N°1 : ensemble de données équilibrée avec 50% de cas diabétiques et 50% de cas non diabétiques.

2. Expérimentation N°2 : ensemble de données non équilibrée avec 40% de cas diabétiques et 60% de cas non diabétiques.
3. Expérimentation N°3 : ensemble de données non équilibrée avec 30% de cas diabétiques et 70% de cas non diabétiques.
4. Expérimentation N°4 : ensemble de données non équilibrée avec 20% de cas diabétiques et 80% de cas non diabétiques.
5. Expérimentation N°5 : ensemble de données non équilibrée avec 10% de cas diabétiques et 90% de cas non diabétiques.
6. Expérimentation N°6 : ensemble de données non équilibrée avec 5% de cas diabétiques et 95% de cas non diabétiques.

Et nous avons obtenu les performances présentées dans le Tableau 4.6 :

Ensembles de données	***TC (%)***	***SE (%)***	***SP (%)***	***Gmean (%)***
50% cas diabétiques / 50% cas non diabétiques	58	55.56	58.90	57.21
40% cas diabétiques / 60% cas non diabétiques	59	40.74	65.75	51.76
30% cas diabétiques / 70% cas non diabétiques	63	37.04	72.60	51.86
20% cas diabétiques / 80% cas non diabétiques	72	33.33	86.30	53.63
10% cas diabétiques / 90% cas non diabétiques	74	18.52	94.52	41.83
5% cas diabétiques / 95% cas non diabétiques	71	0	97.26	0

Tableau 4. 6. Les performances du Classifieur Neuronal Multicouche selon différentes répartitions de données.

Avec ces performances obtenues, nous pouvons dire que pour notre modèle la spécificité augmente lorsque le nombre des patients non diabétiques augmente, la sensibilité diminue lorsque le nombre des patients diabétiques diminue ; c.à.d. le patient non diabétique est classé non diabétique (VN) avec beaucoup de succès par contre dans quelques cas le patient diabétique est classé non diabétique (FN). Ce qui veut dire que le classifieur a réalisé une mauvaise identification des données positives (classe minoritaire c.à.d. La classe qui contient les cas diabétiques). Ce qui peut générer un risque majeur pour la santé du patient. Ce phénomène constitue un sérieux

problème dans le domaine de la classification des données médicales déséquilibrées. Pour remédier à ce problème nous avons appliqué une approche de pondération basée sur l'algorithme des moindres carrés moyens (LMS) pour équilibrer ces ensembles de données.

Nous avons obtenu des valeurs faibles de Gmean lorsque les données minoritaires sont mal classées, même si les échantillons majoritaires sont classés avec une grande précision. Nous avons obtenu ce résultat dans l'expérimentation 6. Nous avons obtenu une valeur de Gmean égal à 0 lorsque les cas diabétiques ne sont pas reconnus par le classifieur (SE=0%).

IV.2 Classification neuronale des données médicales équilibrées par LMS

Dans cette étape nous avons appliqué l'algorithme de pondération des moindres carrés moyens (LMS) sur les différents ensembles de données utilisés dans les expérimentations précédentes pour pénaliser les erreurs des différents échantillons par des poids différents. Nous testons par la suite, cette approche d'équilibrage par les réseaux de neurones multicouches. Nous ciblons l'amélioration des performances de la classification après l'équilibrage. Et nous avons obtenu les performances présentées dans le Tableau 4.7.

Ensembles de données	***TC (%)***	***SE (%)***	***SP (%)***	***Gmean (%)***
50% cas diabétiques / 50% cas non diabétiques	100	100	100	100
40% cas diabétiques / 60% cas non diabétiques	100	100	100	100
30% cas diabétiques / 70% cas non diabétiques	100	100	100	100
20% cas diabétiques / 80% cas non diabétiques	100	100	100	100
10% cas diabétiques / 90% cas non diabétiques	100	100	100	100
5% cas diabétiques / 95% cas non diabétiques	99.00	96.30	100	98.13

Tableau 4. 7. Les performances du Classifieur Neuronal Multicouche avec utilisation de l'algorithme de pondération LMS selon différentes répartitions de données.

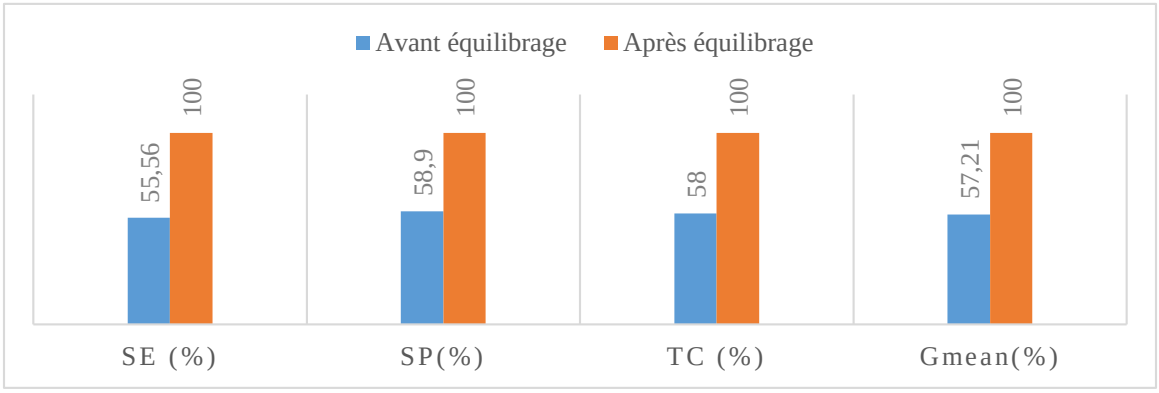
Nous remarquons que les performances de classification (SE, SP, TC et Gmean) ont augmenté après l'équilibrage des ensembles de données déséquilibrées par l'algorithme de pondération LMS.

La sensibilité obtenue par le classifieur neuronal est très grande ce qui veut dire que le classifieur a réalisé une bonne reconnaissance des données positives (classes minoritaires).

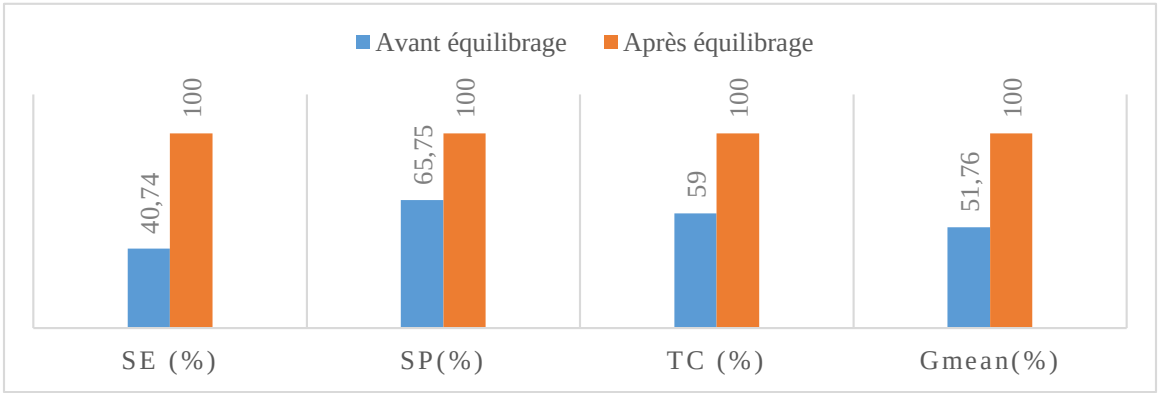
Avec ces performances, nous pouvons dire que l'approche proposée a donné une bonne spécificité, une bonne sensibilité, un bon taux de classification et un bon Gmean.

IV.3 Comparaison entre les deux approches

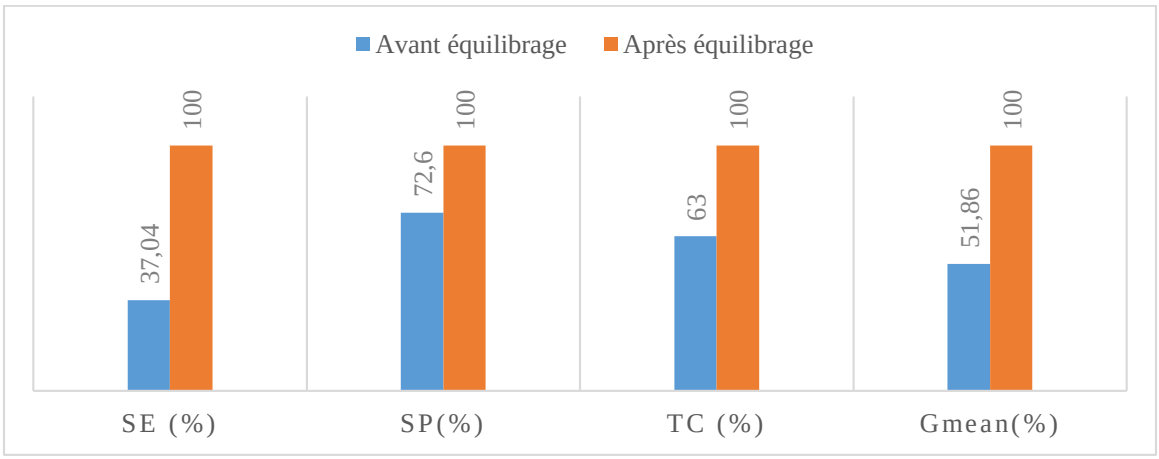
Dans cette section, nous avons réalisé une étude comparative entre les résultats obtenus avons et après l'équilibrage des différents ensembles de données déséquilibrées. Les résultats obtenus sont présentés dans la figure 4.1 (a, b, c, d, e, f).



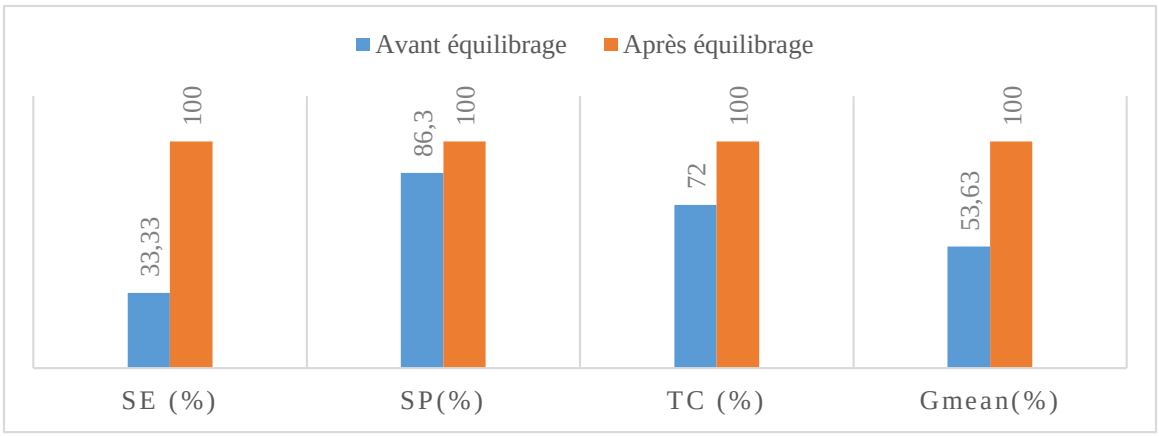
(a) 50% cas diabétiques /50% cas non diabétiques



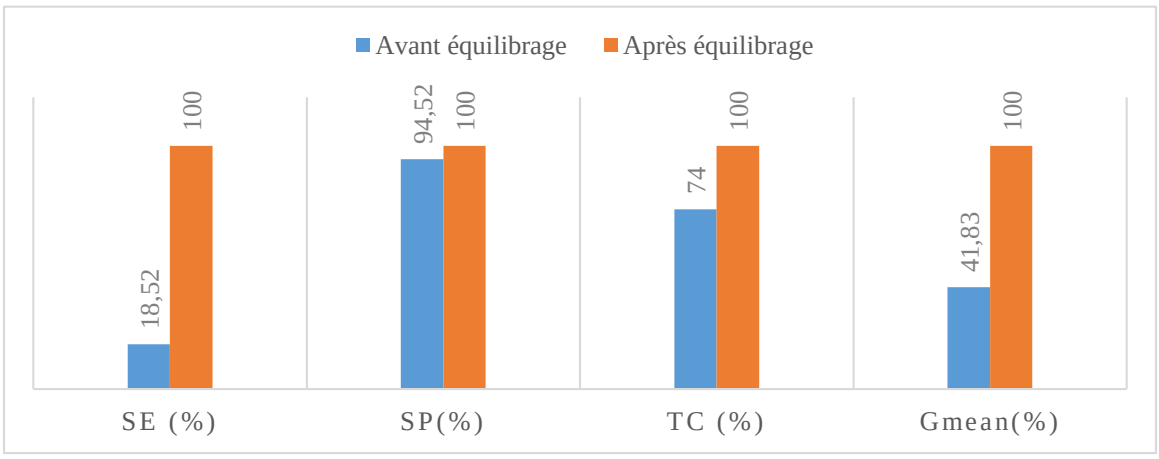
(b) 40% cas diabétiques / 60% cas non diabétiques



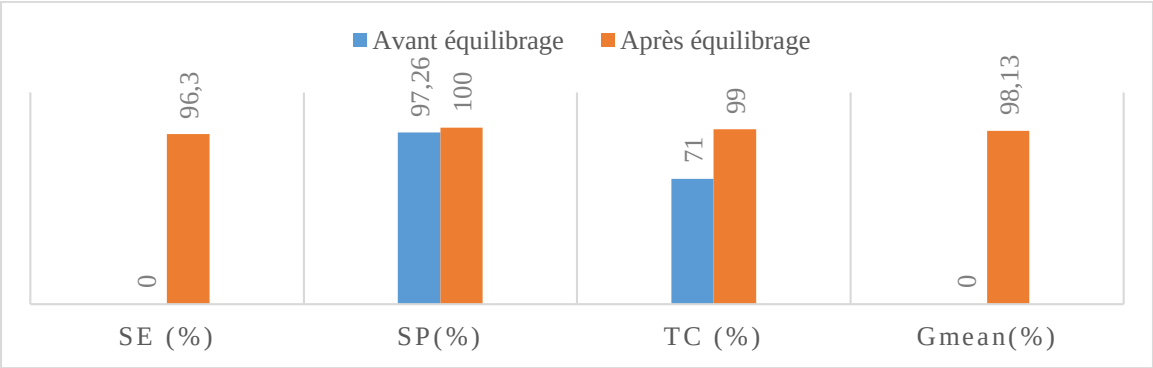
(c) 30% cas diabétiques / 70% cas non diabétiques



(d) 20% cas diabétiques / 80% cas non diabétiques



(e) 10% cas diabétiques / 90% cas non diabétiques



(f) 5% cas diabétiques / 95% cas non diabétiques

Figure 4. 1. Les résultats de classification obtenus par les deux approches.

Nous remarquons que l'utilisation de la méthode de pondération LMS pour l'équilibrage des ensembles de données médicales avant l'apprentissage a donné des meilleures performances de classification (SE, SP, TC et Gmean).

Donc après l'équilibrage, les performances de classification sont améliorées de manière significative.

Nous avons obtenu une très grande valeur de sensibilité (SE), le classifieur a réalisé une bonne reconnaissance des cas diabétiques.

IV.3.1. Performances de classification sur l'ensemble d'apprentissage

Nous avons réalisé une étude comparative entre les résultats des deux méthodes appliquées sur un ensemble de données PIMA qui contient 30 patients diabétiques et 70 patients non diabétiques (ensemble d'apprentissage), et les résultats obtenus avant et après l'équilibrage sont présentés dans le Tableau 4.8 :

Performances de classification	***TC (%)***	***SE (%)***	***SP (%)***	***Gmean (%)***	***VP***	***VN***	***FP***	***FN***
Avant équilibrage	79	53.33	90	69,28	16	63	7	14
Après équilibrage	100	100	100	100	30	70	0	0

Tableau 4. 8. Les performances obtenues avant et après l'équilibrage de l'ensemble de données non équilibré PIMA (30_70)

Nous remarquons qu'avant l'équilibrage ils y a 16 patients diabétiques par rapport les 30 patients diabétiques sont classés correctement (VP), 14 patients diabétiques par rapport les 30 patients diabétiques sont classés non diabétiques (FN), 63 patients non diabétiques par rapport les 70 patients non diabétiques sont classés correctement (VN) et 7 patients non diabétiques par rapport les 70 patients non diabétiques sont classés diabétiques (FP). Mais après l'équilibrage de cet ensemble de données par l'algorithme de pondération LMS où chaque échantillon est pondéré par un poids, nous avons obtenu des meilleures performances de classification. Le classifieur a réalisé une bonne reconnaissance des données positives (classe minoritaire) et des donnés négatives (classe majoritaire) car dans notre exemple les 30 patients diabétiques et les 70 patients non diabétiques sont bien classés.

Donc notre approche d'équilibrage des données a donné des meilleures performances de classifications (TC=100, SE=100, SP=100 et Gmean=100).

IV.3.2. Exemple d'application de la méthode de pondération LMS sur le descripteur Glu

Nous avons appliqué l'algorithme de pondération LMS sur un ensemble de données non équilibré (PIMA) contenant 30 patients diabétiques et 70 patients non diabétiques, les 8 descripteurs sont pondérés.

Nous avons présenté dans les deux figures suivantes seulement les résultats obtenus avant et après l'équilibrage du descripteur Glu (concentration du glucose plasmatique) :

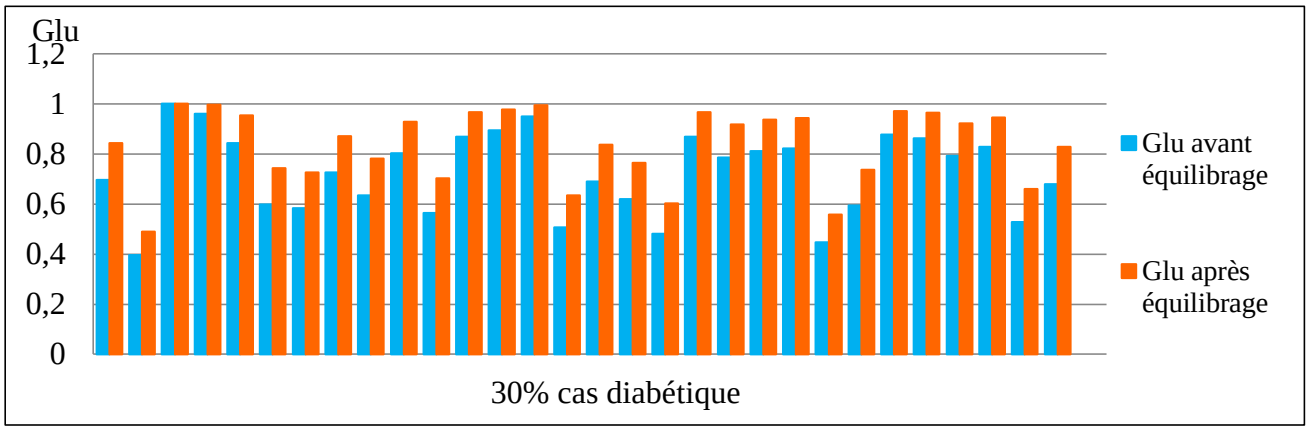


Figure 4. 2. Les valeurs de descripteur Glu obtenues avant et après l'équilibrage pour les cas diabétiques.

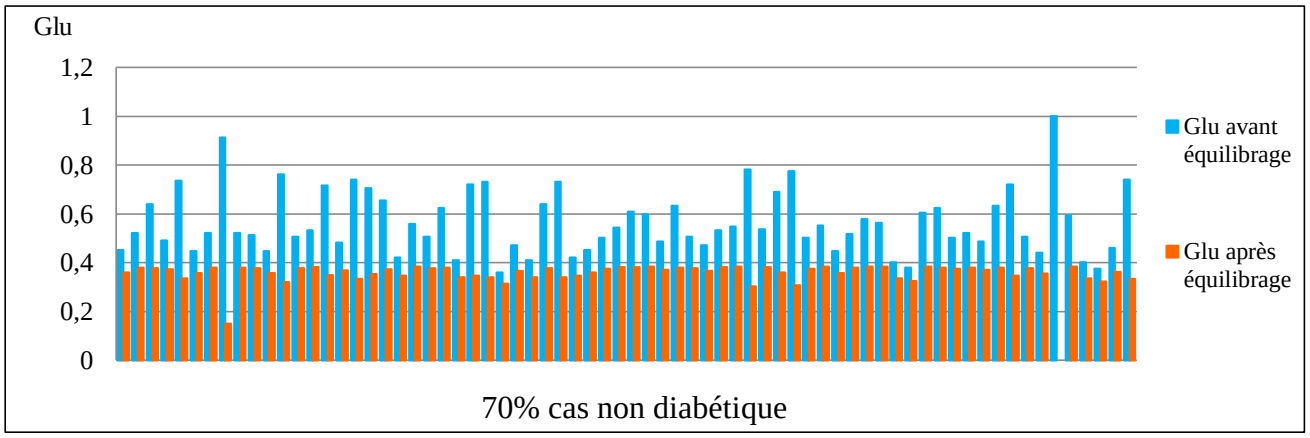


Figure 4. 3. Les valeurs de descripteur Glu obtenues avant et après l'équilibrage pour les cas non diabétiques.

Dans la figure 4.2, nous remarquons que les valeurs de descripteur Glu obtenues après l'équilibrage sont plus élevées par rapport aux valeurs avant l'équilibrage. Par contre dans la figure 4.3, nous remarquons que les valeurs de descripteur Glu obtenues après l'équilibrage sont plus faibles par rapport aux valeurs avant l'équilibrage. Donc notre algorithme d'équilibrage affecte des poids importants ($W > 1$) aux différents échantillons de la classe minoritaire (cas diabétiques) et des poids faibles ($W < 1$) aux différents échantillons de la classe majoritaire (cas non diabétiques) (voir Tableau 4.9 et Tableau 4.10).

Npreg	***Glu***	***BP***	***SKIN***	***Insuline***	***BMI***	***PED***	***Age***
1,073	1,120	1,124	1,125	1,047	1,124	1,046	1,109

Tableau 4. 9. Les Valeurs de poids W pour un cas de la classe minoritaire

Npreg	***Glu***	***BP***	***SKIN***	***Insuline***	***BMI***	***PED***	***Age***
0,998	0,863	0,963	0,818	0,995	0,792	0,997	0,917

Tableau 4. 10. Les Valeurs de poids W pour un cas de la classe majoritaire

IV.4 Cas d'un patient diabétique mal classé

Nous prenons un exemple d'un patient diabétique (patient N°: 3 dans l'ensemble PIMA) classé non diabétique (FN) avant l'équilibrage. Mais après l'application de l'algorithme de

pondération LMS, où chaque paramètre est pondéré par un poids w_i, le même patient a été classé diabétique (VP). Les résultats sont présentés dans la Figure 4.4 et le Tableau 4.11 :

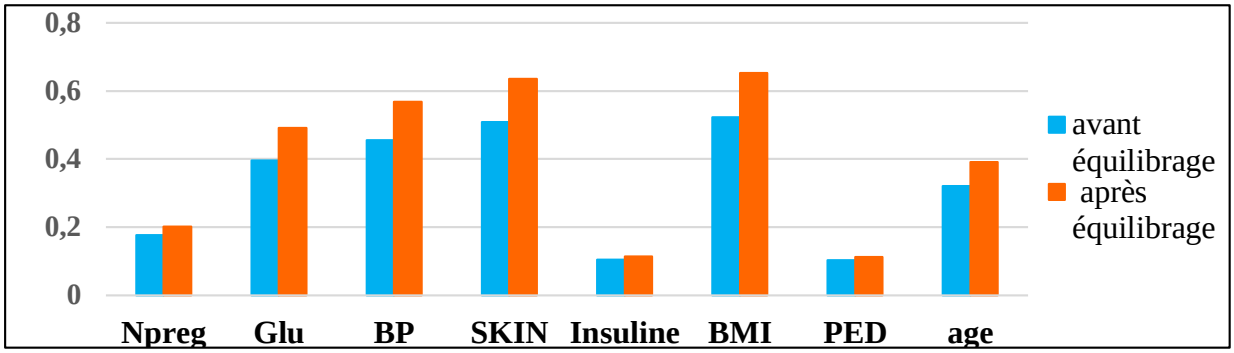


Figure 4. 4. Les résultats avant et après l'équilibrage des différents descripteurs d'un patient diabétique mal classé.

	Npreg	***Glu***	***BP***	***SKIN***	***Insuline***	***BMI***	***PED***	***Age***
Avant équilibrage	0,176	0,396	0,454	0,508	0,104	0,529	0,102	0,321
Valeur de W	**1,145**	**1,239**	**1,248**	**1,250**	**1,093**	**1,249**	**1,092**	**1,218**
Après équilibrage	0,202	0,491	0,567	0,635	0,114	0,652	0,112	0,391

Tableau 4. 11. Les résultats avant et après l'équilibrage des différents descripteurs d'un patient diabétique mal classé.

Nous remarquons dans le Tableau 4.11, que les valeurs de descripteurs d'un patient diabétique ont augmenté après l'application de l'algorithme de pondération LMS. Donc notre algorithme affecte un poids important ($W > 1$) à chaque descripteur. Nous remarquons clairement que l'algorithme de pondération LMS renforce l'importance des attributs.

A partir de ces résultats obtenus nous avons tiré les idées suivantes :

- L'application de l'algorithme de pondération LMS sur des différents degrés de déséquilibre a augmenté les performances des classifieurs neuronaux.
- Le classifieur neuronal multicouche avec l'algorithme de pondération LMS a réalisé une bonne reconnaissance des données minoritaires et des donnés majoritaires.

Dans l'étape suivante, nous étudions l'intérêt de l'application de la méthode de pondération LMS avec d'autres classifieurs sur plusieurs ensembles de données médicales déséquilibrées.

V. Contribution 2 : Intérêt de la pondération des données médicales en utilisant d'autres classifieurs

Pour voir l'intérêt de l'approche de pondération LMS sur les différents techniques de classification (CNMC, SVM, K-PPV), il est intéressent de comparer les performances obtenus par les classifieurs CNMC, SVM et K-PPV pour les deux cas (avec et sans LMS). L'objectif de choisir les critères d'évaluation utilisés dans le cadre de ce travail est pour évaluer les performances des différents classifieurs. En effet, pour le problème de classification des données déséquilibrées, la précision globale de classification ou d'une autre façon le taux de classification correcte n'est pas toujours une mesure appropriée étant donné que la performance d'un classifieur standard qui prédit chaque échantillon dans la classe majoritaire pourrait atteindre une très grande précision dans des domaines très asymétriques.

Dans notre travail, nous avons utilisé quatre critères d'évaluation intuitive (taux de classification correcte, sensibilité, spécificité et Gmean) pour évaluer et estimer la capacité de prédiction des classifieurs actuels.

Notons que la sensibilité et la spécificité fournissent une estimation de performance classe par classe, réalisant facilement une enquête sur la capacité prédictive d'une méthode de classification pour chaque exemple de la classe, en particulier la capacité prédictive pour les classes minoritaires sera intéressante dans notre cas d'étude.

Aussi, Gmean est plus utilisé dans la littérature pour évaluer la classification des données déséquilibrées. Ce critère permet de combiner la sensibilité et la spécificité, ce qui indique l'équilibre entre la performance de classification sur les données majoritaires et minoritaires. Une mauvaise performance de classification dans la prédiction des échantillons positifs (intéressants) conduit toujours à une valeur faible de Gmean, même si les échantillons négatifs sont classés avec une grande précision, ce qui est un cas commun pour les ensembles de données déséquilibrées.

La procédure de classification des données médicales déséquilibrées utilisée dans cette section comporte les trois étapes suivantes :

- Première étape : Classification des données médicales déséquilibrées.
- Deuxième étape : Classification des données médicales équilibrées par l'algorithme de pondération LMS ; Cette étape est scindée en deux parties :

- Application de l'algorithme de pondération LMS pour régler le problème de déséquilibre des données. Dans ce travail, nous choisissons μ=0.5 via expérimentation.
- Classification des données médicales équilibrées.

- Troisième étape : Comparaison entre les deux approches c.-à-d. étude comparative entre les résultats obtenus avant et après l'équilibrage.

> ***Remarque :*** *Dans cette partie les tests sont réalisés sur trois techniques de classification (CNMC, SVM et K-PPV) et sept ensembles de données médicales (PIMA, WBC, WDBC, Liver disorder, Appendicitis, BT- 4 Classes et BT- 6 Classes), dans le but d'augmenter la notion d'interprétabilité des résultats.*

Les résultats de l'étude comparative sont résumés dans le Tableau 4.12.

Ensembles de données	***Classifieurs***		***TC (%)***	***SE (%)***	***SP (%)***	***Gmean (%)***
PIMA	Classifieur CNMC	CNMC	73.85	51.28	83.52	65.44
		LMS	**99.24**	**97.44**	**100**	**98.71**
	Classifieur SVM	SVM	83.85	87.18	82.42	84.77
		LMS	**99.23**	**100**	**98.90**	**99.45**
	Classifieur K-PPV	K-PPV	80	61.54	87.91	73.55
		LMS	**100**	**100**	**100**	**100**
WBC	Classifieur CNMC	CNMC	90.79	65.38	98.30	80.17
		LMS	**99.56**	**98.08**	**100**	**99.03**
	Classifieur SVM	SVM	97.81	98.08	97.73	97.90
		LMS	**99.12**	**100**	**98.86**	**99.43**
	Classifieur K-PPV	K-PPV	98.68	96.15	99.43	97.78
		LMS	**100**	**100**	**100**	**100**
WDBC	Classifieur CNMC	CNMC	96.32	97.73	95.89	96.81
		LMS	**100**	**100**	**100**	**100**
	Classifieur SVM	SVM	97.37	97.73	97.26	97.49
		LMS	**100**	**100**	**100**	**100**
	Classifieur K-PPV	K-PPV	96.32	97.73	95.89	96.81
		LMS	**100**	**100**	**100**	**100**
Liver disorder	Classifieur CNMC	CNMC	77.39	65.85	83.78	74.28
		LMS	**100**	**100**	**100**	**100**

	Classifieur SVM	SVM	66.96	65.85	67.57	66.70
		LMS	**100**	**100**	**100**	**100**
	Classifieur K-PPV	K-PPV	61.74	41.46	72.97	55.00
		LMS	**99.13**	**97.56**	**100**	**98.77**
Appendicitis	Classifieur CNMC	CNMC	71.43	77.78	50.00	62.36
		LMS	**100**	**100**	**100**	**100**
	Classifieur SVM	SVM	74.29	74.07	75.00	74.53
		LMS	**100**	**100**	**100**	**100**
	Classifieur K-PPV	K-PPV	82.86	88.89	62.50	74.54
		LMS	**94.29**	**96.30**	**87.50**	**91.79**
BT- 4 Classes	Classifieur CNMC	CNMC	34.29	40.00	32.00	35.78
		LMS	**62.86**	**90.00**	**52.00**	**68.41**
	Classifieur SVM	SVM	40.00	60.00	32.00	43.82
		LMS	**54.29**	**100**	**36.00**	**60.00**
	Classifieur K-PPV	K-PPV	28.57	30.00	28.00	28.98
		LMS	**54.29**	**100**	**36.00**	**60.00**
BT- 6 Classes	Classifieur CNMC	CNMC	57.14	57.14	57.14	57.14
		LMS	**74.29**	**85.71**	**71.43**	**78.24**
	Classifieur SVM	SVM	45.71	85.71	35.71	55.32
		LMS	**65.71**	**85.71**	**60.71**	**72.13**
	Classifieur K-PPV	K-PPV	68.57	85.71	64.29	74.23
		LMS	**71.43**	**85.71**	**67.86**	**76.26**

Tableau 4. 12. Les résultats obtenus avant et après l'équilibrage de différents ensembles de données déséquilibrées

Dans le Tableau 4.12, Nous remarquons qu'avant l'équilibrage de différents ensembles de données médicales, la classe minoritaire n'est pas bien reconnue par les différentes techniques de classification (CNMC, SVM et K-PPV). Mais après l'équilibrage de ces ensembles de données non équilibrées, les performances de classification sont nettement améliorées de manière significative après l'application de l'algorithme de pondération LMS, où chaque échantillon a été pondéré par un poids. Comme nous avons dit dans la section précédente, l'algorithme de pondération LMS permet d'affecter des poids importants aux différents échantillons de la classe minoritaire et des poids faibles aux différents échantillons des classes majoritaires.

Cela confirme que le classifieur a réalisé une bonne reconnaissance des données positives (classe minoritaire) et des donnés négatives (classes majoritaires) car dans nos expérimentations,

les échantillons de la classe minoritaire et des classes majoritaires sont bien classés (VP et VN ont augmenté / FN et FP ont diminué).

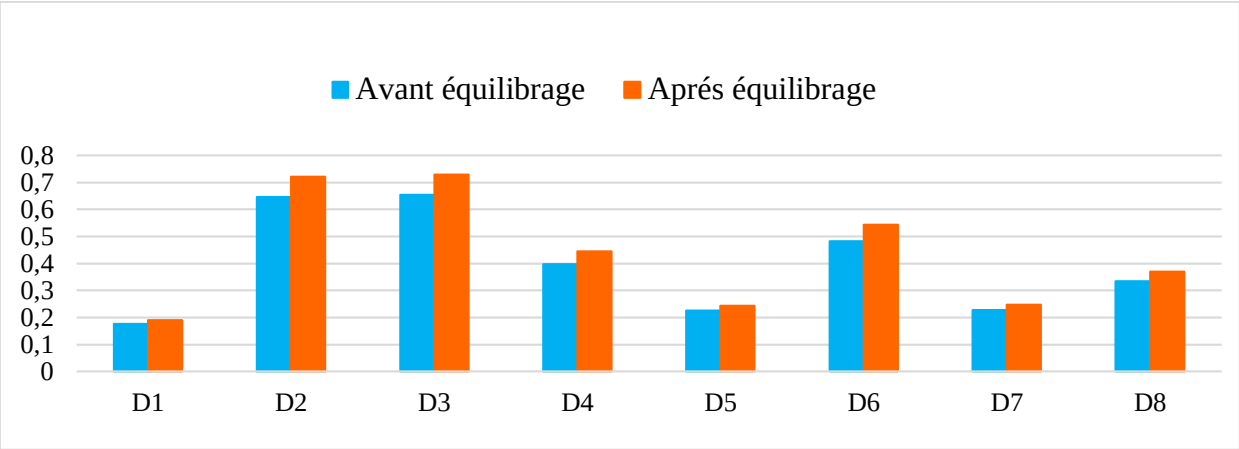
L'analyse des résultats obtenus, nous a permis de faire les remarques suivantes :

- Les résultats obtenus montrent que les performances de classification (SE, SP, TC et Gmean) ont augmenté après l'équilibrage de différents ensembles de données déséquilibrées par l'algorithme de pondération LMS.
- L'algorithme de pondération LMS a donné des meilleures performances de classification quel que soit le type de technique de classification utilisé (CNMC, SVM, K-PPV).
- Cette technique d'équilibrage est plus efficace sur les ensembles de données binaires et multi-classes. Même elle est plus performante. Mais pour les ensembles binaires, elle donne des très bons résultats par rapport au multi classes. Par exemple CNMC avec LMS donne les valeurs de Gmean suivantes : 98.71, 99.03, 100, 100, 100, 68.41 et 78.24, en appliquant respectivement les ensembles de données PIMA, WBC, WDBC, Liver disorder, Appendicitis, BT- 4 Classes et BT- 6 Classes.
- La classe minoritaire est bien identifiée par les différentes techniques de classification après l'application de l'algorithme de pondération LMS sur les différents ensembles de données médicales.

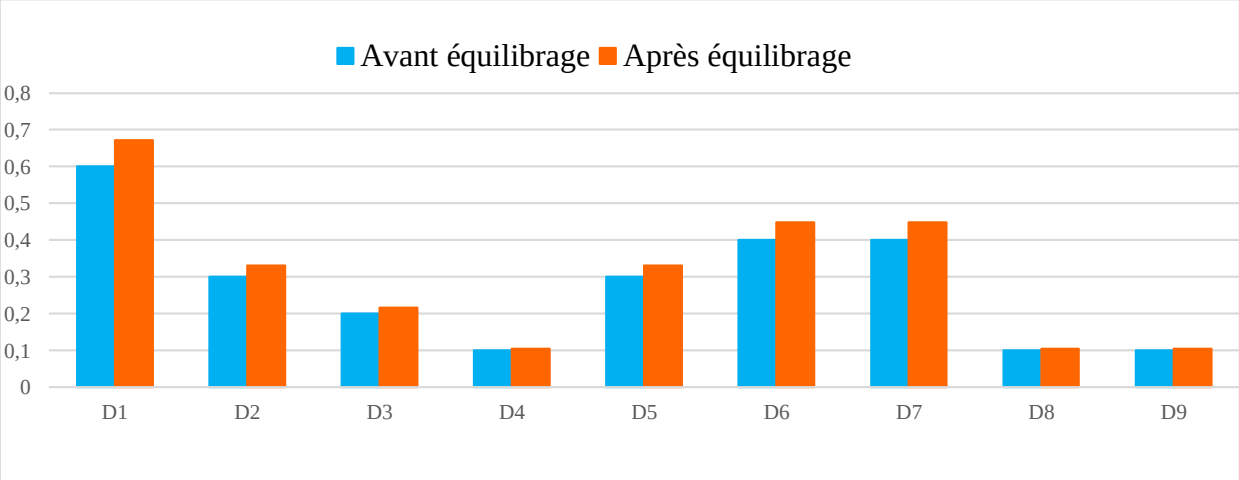
V.1 Comportement de descripteurs avant et après l'approche d'équilibrage

Dans cette section, pour valider l'influence de l'algorithme de pondération LMS sur les différentes techniques de classification (CNMC, SVM et K-PPV), nous avons comparé les valeurs de descripteurs avant et après l'équilibrage.

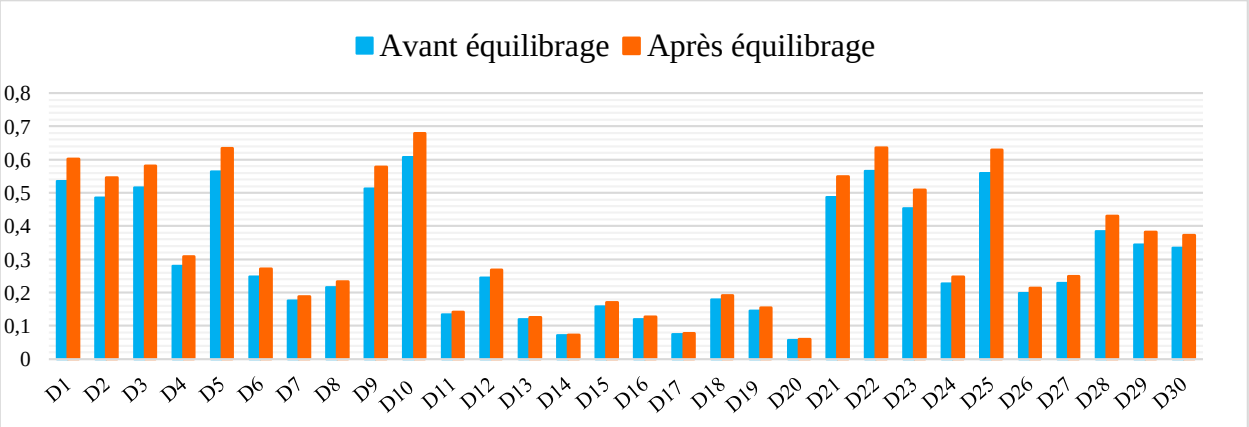
Nous prenons un exemple d'un patient de la classe minoritaire mal classé avant l'équilibrage de différents ensembles de données (PIMA, WBC, WDBC, Liver disorder, Appendicitis, BT- 4 Classes et BT- 6 Classes). Mais après l'application de l'algorithme de pondération LMS, où chaque descripteur est pondéré par un poids w_i, le même patient a été bien classé. Les résultats sont présentés dans la figure 4.5 (a, b, c, d, e, f, g).



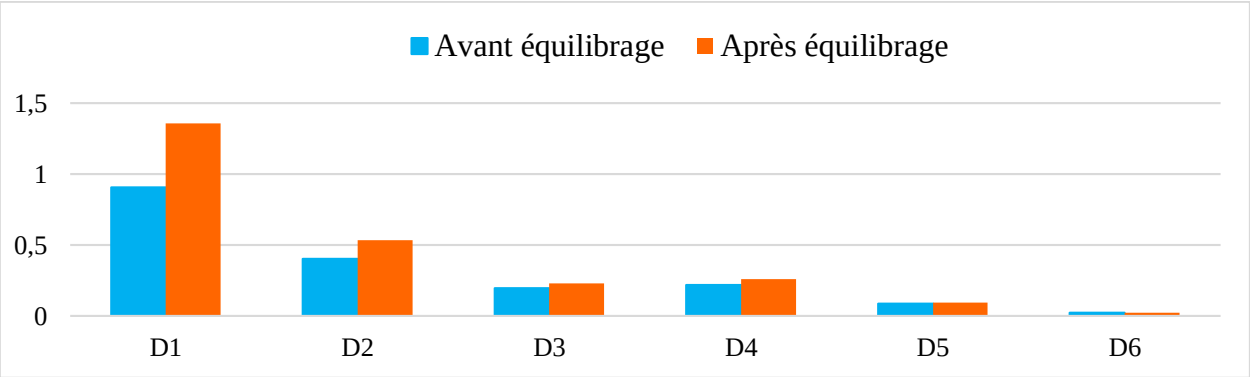
(a) PIMA (FN)



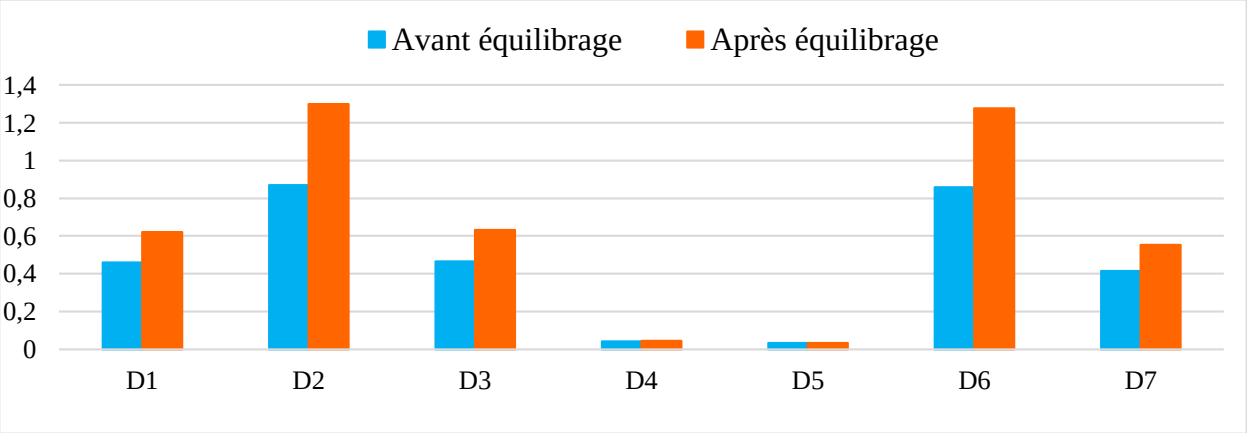
(b) WBC (FN)



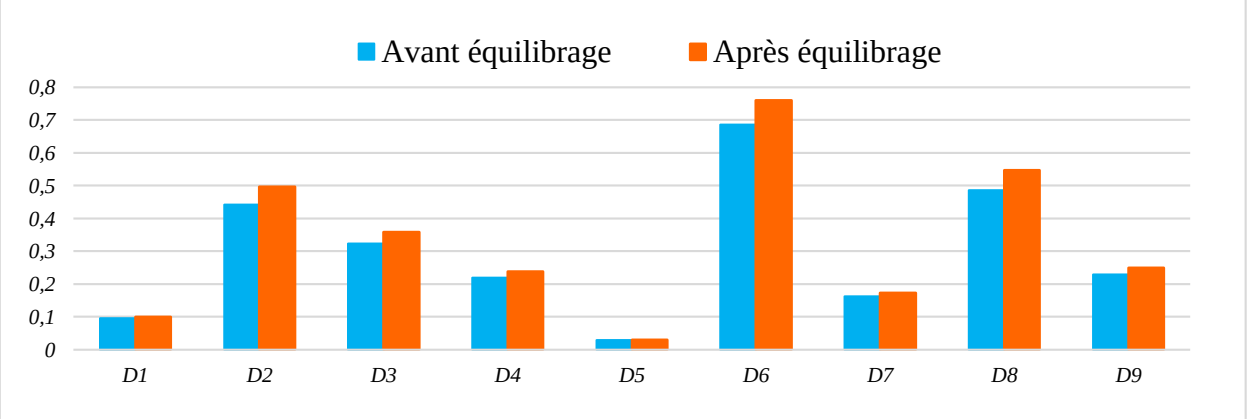
(c) WDBC (FN)



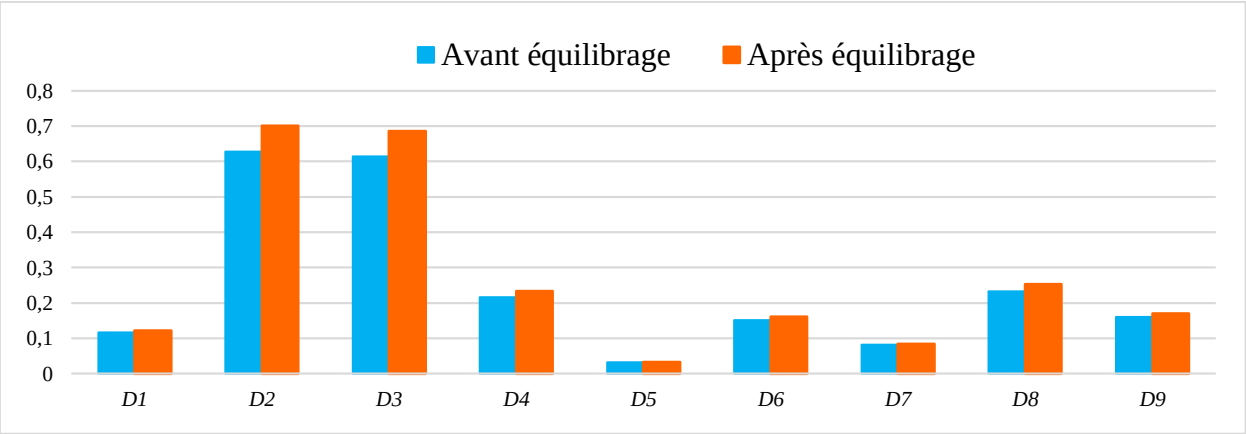
(d) Liver disorder (FN)



(e) Appendicitis (FP)



(f) BT-4 Classes



(g) BT- 6 Classes

Figure 4. 5. Les résultats obtenus avant et après équilibrage de différents ensembles de données d'un cas mal classé de la classe minoritaire.

Dans la figure 4.5 (a), nous remarquons que certains descripteurs dans l'ensemble données PIMA (D1, D5, D7) restent presque inchangées mais le reste changent avec un certain pourcentage qui renforce l'importance des attributs. Aussi dans les autres ensembles de données utilisées (WBC, WDB, Liver disorder, Appendicitis, BT- 4 Classes et BT- 6 Classes), nous avons obtenu quelques changements dans les différents descripteurs (figure 4.5 (b), (c), (d), (e), (f), (g)).

Dans la section suivante, nous réalisons une étude comparative entre l'algorithme de pondération LMS et les méthodes d'échantillonnage citées dans la littérature.

VI. Contribution 3 : Intérêt de la méthode de pondération LMS par rapport à d'autres méthodes existantes

L'objective principale de cette étape est de réaliser une étude comparative entre notre méthode de pondération proposé (LMS) et les autres méthodes d'équilibrage les plus répondues dans la littérature c.-à-d. les méthodes d'échantillonnage : sous-échantillonnage (*Under-Sampling* -US), sur-échantillonnage (*Over-Sampling* -OS) et SMOTE (*Synthetic Minority Oversampling Technique*). Le but est de voir l'intérêt de la méthode de pondération LMS par rapport aux autres méthodes citées dans la littérature.

Dans le présent travail, il est intéressent de comparer les performances obtenues par les classifieurs (CNMC, SVM et K-PPV) avant et après l'équilibrage de différents ensembles de données médicales par les différentes techniques d'équilibrage (LMS, US, OS et SMOTE). Et

aussi, nous avons réalisé une étude comparative entre les résultats obtenus après l'utilisation de la méthode de pondération LMS et les résultats obtenus après l'utilisation des méthodes d'échantillonnage (US, OS et SMOTE). Nous avons utilisé quatre critères d'évaluation (taux de classification correcte, sensibilité, spécificité et Gmean) pour évaluer et estimer la capacité de prédiction des classifieurs actuels [168].

La procédure de classification des données médicales déséquilibrées proposée dans cette partie comporte les quatre étapes suivantes :

- Première étape : Application de différents classifieurs (CNMC, SVM et K-PPV) sur des ensembles de données médicales déséquilibrées.
- Deuxième étape : Application de l'algorithme de pondération LMS et les méthodes d'échantillonnage (sous-échantillonnage, sur-échantillonnage et SMOTE) pour surmonter le déséquilibre des données.
- Troisième étape : Application de différents classifieurs (CNMC, SVM et K-PPV) sur les ensembles de données médicales équilibrées obtenues.
- Quatrième étape : Comparaison entre les résultats obtenus par la méthode de pondération LMS et les résultats obtenus par les méthodes d'échantillonnage.

> ***Remarque :*** *Dans cette partie les tests sont réalisés sur sept ensembles de données médicales (PIMA, WDBC, WBC, Liver disorder, Appendicitis, BT- 4 Classes et BT- 6 Classes).*

Les résultats de l'étude comparative sont résumés dans le Tableau 4.13.

Ensembles de données	***Approches***		***TC (%)***	***SE (%)***	***SP (%)***	***Gmean (%)***
PIMA	Classifieur CNMC	CNMC	73.85	51.28	83.52	65.44
		US	67.94	74.36	65.22	69.64
		OS	70.23	64.10	72.83	68.33
		SMOTE	73.28	43.59	85.87	61.18
		LMS	**99.24**	**97.44**	**100**	**98.71**
	Classifieur SVM	SVM	83.85	87.18	82.42	84.77
		US	77.86	74.36	79.35	76.81
		OS	76.34	33.33	94.56	56.14
		SMOTE	69.47	71.79	68.48	70.16

		LMS	**99.23**	**100**	**98.90**	**99.45**
	Classifieur K-PPV	K-PPV	80	61.54	87.91	73.55
		US	69.47	94.87	58.70	76.63
		OS	70.23	87.18	63.04	74.13
		SMOTE	77.10	41.03	92.39	61.57
		LMS	**100**	**100**	**100**	**100**
WDBC	Classifieur CNMC	CNMC	96.32	97.73	95.89	96.80
		US	82.63	77.40	100	87.98
		OS	98.42	98.63	97.73	98.18
		SMOTE	98.95	98.63	100	99.31
		LMS	**100**	**100**	**100**	**100**
	Classifieur SVM	SVM	97.37	97.73	97.26	97.49
		US	98.42	100	93.62	96.76
		OS	96.32	96.03	97.44	96.73
		SMOTE	97.89	100	91.67	95.74
		LMS	**100**	**100**	**100**	**100**
	Classifieur K-PPV	K-PPV	96.32	97.73	95.89	96.81
		US	89.47	86.30	100	92.90
		OS	96.32	97.95	90.91	94.36
		SMOTE	98.42	99.32	95.45	97.37
		LMS	**100**	**100**	**100**	**100**
WBC	Classifieur CNMC	CNMC	90.79	65.38	98.30	80.17
		US	92.51	80.77	95.45	87.80
		OS	96.05	96.15	96.02	96.08
		SMOTE	95.61	88.46	97.73	92.98
		LMS	**99.56**	**98.08**	**100**	**99.03**
	Classifieur SVM	SVM	97.81	98.08	97.73	97.90
		US	98.86	98.08	98.86	98.47
		OS	88.60	53.85	98.86	72.96
		SMOTE	98.25	98.08	98.30	98.19
		LMS	**99.12**	**100**	**98.86**	**99.43**
	Classifieur K-PPV	K-PPV	98.68	96.15	99.43	97.78
		US	98.25	94.23	99.43	96.79

		OS	98.25	98.08	98.30	98.19
		SMOTE	99.12	96.15	100	98.06
		LMS	**100**	**100**	**100**	**100**
Liver Disorder	Classifieur CNMC	CNMC	77.39	65.85	83.78	74.28
		US	45.81	70.73	56.76	63.36
		OS	37.39	60.98	24.32	38.50
		SMOTE	62.61	65.85	60.81	63.28
		LMS	**100**	**100**	**100**	**100**
	Classifieur SVM	SVM	66.96	65.85	67.57	66.70
		US	66.96	65.85	67.57	66.70
		OS	61.74	65.85	59.46	62.57
		SMOTE	68.70	65.85	70.27	68.02
		LMS	**100**	**100**	**100**	**100**
	Classifieur K-PPV	K-PPV	61.74	41.46	72.97	55.00
		US	53.04	87.81	33.78	54.46
		OS	66.96	21.95	91.89	44.91
		SMOTE	52.17	80.49	36.49	54.19
		LMS	**99.13**	**97.56**	**100**	**98.77**
Appendicitis	Classifieur CNMC	CNMC	71.43	77.78	50.00	62.36
		US	60.00	62.96	50.00	56.11
		OS	60.00	62.96	50.00	56.11
		SMOTE	68.57	70.37	62.50	66.32
		LMS	**100**	**100**	**100**	**100**
	Classifieur SVM	SVM	74.29	74.07	75.00	74.53
		US	77.14	77.78	75.00	76.38
		OS	77.14	77.78	75.00	76.38
		SMOTE	80.00	81.48	75.00	78.17
		LMS	**100**	**100**	**100**	**100**
	Classifieur K-PPV	K-PPV	82.86	88.89	62.50	74.54
		US	82.86	88.89	62.50	74.54
		OS	77.14	77.78	75.00	76.38
		SMOTE	77.14	74.07	87.5	80.50
		LMS	**94.29**	**96.30**	**87.50**	**91.79**

BT- 4 Classes	Classifieur CNMC	CNMC	34.29	40.00	32.00	35.78
		US	34.29	20.00	40.00	28.28
		OS	40.00	40.00	40.00	40
		SMOTE	37.14	30.00	40.00	34.64
		LMS	**62.86**	**90.00**	**52.00**	**68.41**
	Classifieur SVM	SVM	40.00	60.00	32.00	43.82
		US	20.00	40.00	12.00	21.91
		OS	40.00	30.00	44.00	36.33
		SMOTE	31.43	50.00	24.00	34.64
		LMS	**54.29**	**100**	**36.00**	**60.00**
	Classifieur K-PPV	K-PPV	28.57	30.00	28.00	28.98
		US	28.57	70.00	12.00	28.98
		OS	40.00	30.00	44.00	36.33
		SMOTE	51.43	20.00	64.00	35.78
		LMS	**54.29**	**100**	**36.00**	**60.00**
BT- 6 Classes	Classifieur CNMC	CNMC	57.14	57.14	57.14	57.14
		US	51.43	57.14	50.00	53.45
		OS	48.57	71.43	42.86	55.33
		SMOTE	45.71	71.43	39.29	52.98
		LMS	**74.29**	**85.71**	**71.43**	**78.24**
	Classifieur SVM	SVM	45.71	85.71	35.71	55.32
		US	40.00	85.71	28.57	49.48
		OS	45.71	42.86	46.43	44.61
		SMOTE	48.57	100	35.71	59.76
		LMS	**65.71**	**85.71**	**60.71**	**72.13**
	Classifieur K-PPV	K-PPV	68.57	85.71	64.29	74.23
		US	65.71	85.71	60.71	72.13
		OS	65.71	71.43	64.29	67.77
		SMOTE	65.71	71.43	64.29	67.77
		LMS	**71.43**	**85.71**	**67.86**	**76.26**

Tableau 4. 13. Les résultats obtenus avant et après l'équilibrage de différents ensembles de données déséquilibrées en utilisant des différentes techniques d'équilibrage.

Dans ce travail, nous étudions d'une façon empirique l'effet de l'application des algorithmes d'équilibrage (moindres carrés moyens –LMS-, sur-échantillonnage –US-, sous-échantillonnage –OS- et SMOTE) sur les performances des trois classifieurs choisis (SVM, K-PPV et CNMC).

Nous remarquons que les résultats de ces expérimentations montrent que les performances de classification (SE, SP, TC et Gmean) ont augmenté après l'équilibrage des ensembles de données déséquilibrées par l'algorithme de pondération LMS. Parce que dans l'algorithme LMS, les carrés des erreurs quadratiques moyennes sont minimisés par la résolution d'un système d'équations linéaires, où chaque échantillon est pondéré par un poids.

Nous remarquons aussi qu'avant l'équilibrage de différents ensembles de données médicales, la classe minoritaire n'est pas bien reconnue par les différentes techniques de classification (CNMC, SVM et K-PPV). Mais après l'équilibrage de ces ensembles de données non équilibrées par l'algorithme de pondération LMS, les performances de classification sont améliorées de manière significative.

Le classifieur a réalisé une bonne reconnaissance de classe minoritaire et des classes majoritaires. Comme nous l'avons mentionné dans la section précédente, l'algorithme de pondération LMS permet d'affecter des poids forts aux différents échantillons de la classe minoritaire et des poids faibles aux différents échantillons des classes majoritaires.

L'étude empirique montre que toutes les techniques d'équilibrage sont efficaces sur les ensembles de données binaires, alors que la plupart sont inefficaces sur les ensembles multi-classes [7]. Pratiquement, la méthode de pondération LMS a été efficace sur les ensembles de données binaires et multi classes et pour tous les classifieurs (CNMC, SVM et K-PPV), mais avec deux classes, elle donne des meilleurs performances de classification; par exemple CNMC avec LMS a donné les valeurs de Gmean suivants : 98.71, 99.03, 100, 100, 100, 68.41 et 78.24 en utilisant respectivement les ensembles de données PIMA, WBC, WDBC, Liver disorder, Appendicitis, BT-4 Classes et BT- 6 Classes. Par contre les méthodes d'échantillonnage sont utiles seulement sur les ensembles de données binaires, tandis qu'elles donnent des faibles performances sur les ensembles de données multi-classes [7]. Mais cette remarque n'est pas toujours valide pour tous les classifieurs et tous les ensembles de données. Par exemple pour l'ensemble de donnée binaire Appendicitis, le classifieur K-PPV a donné des meilleurs valeurs de Gmean : 74.54%, 76.38% et

80.50**%** pour US, OS et SMOTE respectivement. Par contre dans l'ensemble de donnés Liver disorder, le même classifieur a donné des faibles valeurs de Gmean : 54.46%, 44.91% et 54.19%. Dans l'ensemble de données multi-classe « BT- 4 Classes », nous avons obtenu des meilleurs valeurs de Gmean pour le classifieur K-PPV avec US, OS et SMOTE (28.98%, 36.33% et 35.78% respectivement) et des faibles valeurs de Gmean pour le classifieur SVM avec US, OS et SMOTE (21.91%, 36.33%, 34.64%).

D'après le Tableau 4.13, nous constatons que :

- La méthode SMOTE a donné des fables valeurs de Gmean dans quelques ensembles de données binaires et multi-classes, parce que cette méthode permet d'ajouter des individus synthétiques dans la classe minoritaire, elle peut provoquer un risque de sur-généralisation.
- La méthode US a donné des fables valeurs de Gmean sur quelques ensembles de données binaires et multi-classes, parce que la méthode US supprime des instances appartenant aux classes majoritaires et garde toutes les instances de la classe minoritaire. Elle provoque un risque de perte d'informations (risque de supprimer des individus importants dans l'ensemble d'apprentissage).
- Dans la technique OS, nous avons un problème de sur-apprentissage qui apparait. Ce problème est remarqué à partir des résultats du G-mean dans le Tableau 4.13.

Nous pouvons dire que les classifieurs donnent des meilleures performances de classification avec les ensembles de données équilibrées par l'utilisation de l'algorithme de pondération LMS.

Dans cette étape, nous avons déduit que l'application de la méthode de pondération LMS sur les différents ensembles de données médicales a donné des meilleures performances de classification par rapport aux autres techniques d'équilibrage (méthodes d'échantillonnage) appliquées dans la littérature.

Dans l'étape suivante, nous comparons notre méthode avec des autres méthodes citées dans la littérature et appliquées sur le même ensemble de données.

VII. Comparaison de nos résultats avec l'état de l'art

Dans cette section, nous avons comparé le taux de classification de notre méthode avec d'autres méthodes appliquées sur le même ensemble de données :

VII.1 Travaux testés sur l'ensemble de données PIMA

Le tableau suivant présente le taux de classification obtenu par notre méthode et les autres méthodes appliquées sur l'ensemble de données PIMA.

Auteurs	***Méthodes***	***Taux de classification (%)***
L. Gonzalez-Abril [160]	GSVM	74.15
Y.Shao and al. [161]	WLTSVM	76.78 ±0.35
Notre méthode	CNMC avec LMS	**99.24**
Notre méthode	SVM avec LMS	**99.23**
Notre méthode	K-PPV avec LMS	**100**

Tableau 4. 14. Taux de classification obtenu par notre méthode et les autres méthodes dans la littérature (PIMA).

L. Gonzalez-Abril et al. [160] ont proposé une nouvelle méthode de Support Vector Machine appelée GSVM (*Geometric (mean) Support Vector Machine*), qui est spécialement conçu pour traiter les problèmes de bi-classification, son objectif était la précision équilibrée entre les classes. Ils ont utilisé plusieurs ensembles de données (23) pour évaluer leurs résultats, ils ont obtenu un taux de classification de 74,15% pour l'ensemble de données Pima. Y.Shao and al. dans leurs articles [161], ils ont proposé une méthode efficace appelée « *Weighted Lagrangian Twin Support Vector Machine* ». Nous l'acronyme WLTSVM. Cette approche a été utilisée pour traiter le problème de classification des données déséquilibrées. Ils ont utilisé différents points d'apprentissage pour la construction des deux hyperplans proximaux, ils ont obtenu un taux de classification de 76,78 ± 0,35%. Dans le Tableau 4.14, Nous remarquons que le taux de classification obtenu par notre méthode (CNMC avec LMS, SVM avec LMS et K-PPV avec LMS) est le meilleur parmi les résultats obtenus par les autres méthodes.

VII.2 Travaux testés sur l'ensemble de données WBC

Le tableau suivant présente le taux de classification obtenu par notre méthode et les autres méthodes appliquées sur l'ensemble de données WBC :

Auteurs	*Méthodes*	*Taux de classification (%)*
Wang et Adrian [162]	S- AIRS	96.91
Y. Shao et al [161]	WLTSVM	96.30±0.31
Notre méthode	CNMC avec LMS	**99.56**
Notre méthode	SVM avec LMS	**99.12**
Notre méthode	K-PPV avec LMS	**100**

Tableau 4. 15. Taux de classification obtenu par notre méthode et les autres méthodes dans la littérature (WBC).

Wang et Adrian [162] ont proposé une méthode hybride, qui combine la méthode SMOTE (*Synthetic Minority Over-Sampling Technique*) avec le Système de reconnaissance Immunitaires Artificiels (*Artificial Immune Recognition System* -AIRS-) pour traiter le problème de données déséquilibrées. Cette approche est notée S- AIRS. Ils ont obtenu un taux de classification de 96,91%. Y. Shao et al. [161] ont proposé l'approche WLTSVM et ils ont obtenu un taux de classification de 96,30 ± 0,31%. D'après le Tableau 4.15, Nous remarquons que notre approche a donné un bon taux de classification avec les différents classifieurs (CNMC, SVM et K-PPV).

VII.3 Travaux testés sur l'ensemble de données WDBC

Le tableau suivant présente le taux de classification obtenu par notre méthode et les autres méthodes appliquées sur l'ensemble de données WDBC :

Auteurs	*Méthodes*	*Taux de classification (%)*
Wang et Adrian [162]	S- AIRS	96.52
G. NAGA RAMADEVI et al. [163]	K-PPV avec ré-échantillonnage	98.42
Notre méthode	CNMC avec LMS	**100**
Notre méthode	SVM avec LMS	**100**
Notre méthode	K-PPV avec LMS	**100**

Tableau 4. 16. Taux de classification obtenu par notre méthode et les autres méthodes dans la littérature (WDBC).

Wang et Adrian [162] ont proposé une méthode hybride S-AIRS. Cette approche a donné un taux de classification de 96,52%. G. NAGA RAMADEVI et al. [163] ont appliqué cinq classifieurs (K-PPV, SVM, régression logistique, C 4.5 et Forêts Aléatoires) avec et sans la technique de ré-échantillonnage sur quatre ensembles de données du cancer du sein, ils ont comparé les performances obtenues avant et après ré-échantillonnage des jeux de données. Cette approche a

donné un bon taux de classification de 98,42% par l'utilisation de K-PPV avec la méthode de ré-échantillonnage. Les résultats présentés dans le Tableau 4.16 ont montré que notre approche a donné toujours les meilleures performances.

VII.4 Travaux testés sur l'ensemble de données Liver disorder

Le tableau suivant présente le taux de classification obtenu par notre méthode et les autres méthodes appliquées sur l'ensemble de données Liver disorder.

Auteurs	***Méthodes***	***Taux de classification (%)***
Alberto Cano et al. [164]	DGC+	67.44
L. Gonzalez-Abril et al. [160]	GSVM	71.07
Notre méthode	CNMC avec LMS	**100**
Notre méthode	SVM avec LMS	**100**
Notre méthode	K-PPV avec LMS	**99.13**

Tableau 4. 17. Taux de classification obtenu par notre méthode et les autres méthodes dans la littérature (Liver disorder).

Alberto Cano et al. [164] ont proposé un algorithme de classification de la gravitation de données pondérées appelé *weighted Data Gravitation Classification* (DGC+) qui compare le champ gravitationnel pour les différentes classes de données pour prédire la classe avec la plus haute importance. La proposition améliore l'algorithme de gravitation de données précédentes par l'apprentissage des poids optimaux des attributs pour chaque classe et elle résout certains de leurs problèmes tels que les attributs nominaux maniabilité, performance de données déséquilibrées, et le filtrage de données bruitées. Ils ont obtenu un taux de classification de 67,44%. L. Gonzalez-Abril et al. [160] ont proposé dans leurs travaux la méthode GSVM. Ils ont obtenu un taux de classification de 71,07%. Dans le Tableau 4.17, nous remarquons aussi que notre approche a donné un bon taux de classification en utilisant les différents classifieurs (CNMC, SVM et KNN).

VII.5 Travaux testés sur l'ensemble de données Appendicitis

Le tableau suivant présente le taux de classification obtenu par notre méthode et les autres méthodes appliquées sur l'ensemble de données Appendicitis.

Auteurs	*Méthodes*	*Taux de classification (%)*
Alberto Cano et al. [164]	DGC+	84.09
Kung-Jeng et al. [165]	BSMAIRS	92.5926
Notre méthode	CNMC avec LMS	**100**
Notre méthode	SVM avec LMS	**100**
Notre méthode	K-PPV avec LMS	**94.29**

Tableau 4. 18. Taux de classification obtenu par notre méthode et les autres méthodes dans la littérature (Appendicitis).

Alberto Cano et al. [164] ont proposé une méthode de classification de la gravitation de données pondérées appelée *weighted Data Gravitation Classification* (DGC+). Cette approche a obtenu un taux de classification de 84,09%. Kung-Jeng et al. [165] ont développé une approche hybride de classification, ils ont combiné la technique de sur échantillonnage BSM (*Borderline Synthetic Minority oversampling*) et un Système de Reconnaissance Immunitaires Artificiels (AIRS) comme moyen d'optimisation globale avec l'algorithme du plus proche voisin utilisé comme un classifieur locale. Cette approche est notée BSMAIRS. Pour l'évaluation de leurs résultats, Kung-Jeng et al. ont utilisé la stratégie « *fivefold cross validation* » et ils ont obtenu cinq taux de classification ; le meilleur taux obtenu avec 92,5926%. Dans le Tableau 4.18, nous remarquons que notre approche a obtenu des résultats meilleurs.

VII.6 Travaux testés sur l'ensemble de données BT- 4 Classes

Le Tableau 4.19 présente le taux de classification obtenu par notre méthode et les autres méthodes appliquées sur l'ensemble de données BT- 4 Classes.

Auteurs	*Méthodes*	*Taux de classification (%)*
Rafael and al. [166]	SVM	53
Notre méthode	CNMC avec LMS	**62.86**
Notre méthode	SVM avec LMS	**54.29**
Notre méthode	K-PPV avec LMS	**54.29**

Tableau 4. 19. Taux de classification obtenu par notre méthode et les autres méthodes dans la littérature (BT- 4 Classes).

Rafael et al. [166] ont confirmé qu'une méthode simple de recherche aléatoire est suffisante pour ajuster les hyper-paramètres de SVM. Un ensemble des expérimentations pour comparer la performance de cinq techniques d'optimisation a été utilisé : Trois méta-heuristiques (*Genetic Algorithm* (GA), *Particle Swarm Optimization* (PSO) et *Estimation of Distribution Algorithms* (EDA)) couramment utilisé, recherche aléatoire (Random Search) et recherche Grille (Grid Search). Ils ont obtenu un taux de classification de 53% par l'utilisation de SVM avec GA pour les différentes évaluations. Dans le Tableau 4.19, Nous remarquons que le taux de classification obtenu par notre méthode est meilleur en comparant à d'autres méthodes utilisées.

VII.7 Travaux testés sur l'ensemble de données BT- 6 Classes

Le tableau suivant présente le taux de classification obtenu par notre méthode et les autres méthodes appliquées sur l'ensemble de données BT- 6 Classes.

Auteurs	***Méthodes***	***Taux de classification (%)***
Antonio and al. [167]	TSEAFS	60.93
Rafael and al. [166]	SVM	42
Notre méthode	CNMC avec LMS	**74.29**
Notre méthode	SVM avec LMS	**65.71**
Notre méthode	K-PPV avec LMS	**71.43**

Tableau 4. 20. Taux de classification obtenu par notre méthode et les autres méthodes dans la littérature (BT- 6 Classes).

Antonio et al. [167] ont combiné les méthodes de sélection de fonction (*Feature Selection* - FS) avec un algorithme d'évolution en deux étapes (*Two-Stage Evolutionary Algorithm -TSEA*) basé sur les réseaux neuronaux de l'unité de produit (product unit neural networks). Cette approche est notée TSEAFS. La méthode améliorée a été essayé avec quatre filtres. Ils ont utilisé 18 ensembles de données. Tout d'abord, FS est appliqué sur les ensembles de données de façon à éliminer les variables redondantes et non pertinentes. La réduction du nombre d'entrées peut diminuer le nombre de nœuds dans la couche cachée et elle a aussi permet de simplifier le modèle associé. Plusieurs pistes de la TSEA ont été réalisées pour lisser le caractère stochastique par utilisation des valeurs moyennes afin de compléter une analyse statistique des résultats obtenus. Cette approche comprenait une comparaison empirique globale entre les modèles obtenus sans et

avec la méthode de sélection de fonction. Ils ont obtenu un bon taux de classification de 60,93% par l'utilisation de troisième filtre (FS3). Rafael et al. [166] ont étudié l'hypothèse selon laquelle une simple méthode de recherche aléatoire est suffisante pour ajuster les hyper-paramètres de SVM. Ils ont obtenu un taux de classification de 42% par l'utilisation de SVM avec GA pour les évaluations 200 et 10000. Dans le Tableau 4.20, nous remarquons que notre méthode obtient un meilleur taux de classification.

VIII. Conclusion

Pour créer une application performante utilisée pour la reconnaissance des données médicales déséquilibrées, nous avons implémenté une méthode de pondération des données minoritaires basée sur l'algorithme des moindres carrés moyens (LMS), dans le but de minimiser l'erreur pour atteindre une meilleure classification. L'objectif principal de cette méthode est d'affecter des poids forts aux différents échantillons de la classe minoritaire et des poids faibles aux différents échantillons des classes majoritaires afin de créer un équilibre entre les classes.

L'étude réalisée montre que l'utilisation de la méthode de pondération LMS est effectivement très pertinente pour l'équilibrage des ensembles de données médicales.

Les résultats obtenus après l'utilisation de cette méthode sont très prometteurs et sont bien situés parmi les travaux déjà réalisés dans la littérature.

Conclusion générale

Conclusion

Dans nos travaux de recherche, nous avons traité un problème majeur rencontré par les méthodes d'apprentissage supervisé dans beaucoup d'application du monde réel. Ce problème est le déséquilibre des classes, lorsqu'au moins une classe est sous représentée par rapport aux autres. C'est particulièrement le cas du domaine de diagnostic médical, lorsque nous cherchons à proposer des modèles permettant de prédire la présence d'une maladie chez un patient. Ce problème constitue une contrainte majeure aux algorithmes d'apprentissage standards qui présentent une grande précision sur les classes majoritaires et une mauvaise précision sur les classes minoritaires qui sont des fois d'un grand intérêt médical. Dans le monde réel, plusieurs travaux de recherche sont proposés pour traiter le problème de déséquilibre des classes, soit par une modification au niveau des données (les techniques de pondération), soit par une adaptation des algorithmes de classification, ou par une minimisation de coûts d'erreur du mauvaise classification (les techniques sensibles aux coûts).

Dans notre travail, nous avons proposé une méthode de pondération des données basée sur l'algorithme des moindres carrés moyens appelée LMS (*Least Mean Square*), qui permet d'améliorer les performances lors d'un processus d'apprentissage sur les données déséquilibrées. Cette méthode consiste à affecter des poids forts aux différents échantillons de la classe minoritaire et des poids faibles aux différents échantillons des classes majoritaires dans le but de minimiser le coût d'erreur pour atteindre une meilleure classification. L'avantage principal de la technique de pondération LMS est le fait de garder le même ensemble de données c.à.d. nous ne modifions pas l'ensemble (même taille et même types de données), elle consiste à affecter des poids aux différents échantillons de la classe minoritaire et les classes majoritaires, ainsi que le coût d'erreur de mauvaise classification diminue.

Dans nos expérimentations, nous avons présenté trois contributions principales pour améliorer les performances de la classification des données déséquilibrées. Nous avons testé plusieurs approches de classification (RNMC, SVM et K-PPV). Nous avons remarqué que les performances de classification ont augmenté après application de l'algorithme d'équilibrage LMS sur des ensembles de données avec différents degrés de déséquilibre.

L'étude empirique montre que la méthode de pondération LMS est plus efficace sur les ensembles de données binaires et multi classes quelques soit le type de classifieur, mais avec deux

classes elle donne des meilleurs performances de classification. Par contre les méthodes d'échantillonnage sont utiles seulement sur les ensembles de données binaires, tandis qu'elles donnent des faibles performances sur les ensembles de données multi-classes. Nous avons remarqué aussi que la méthode de pondération LMS a donné des meilleures performances de classification par rapport aux autres techniques d'échantillonnage appliquées dans la littérature quelques soit le type de technique de classification utilisé. Cela confirme que le classifieur a réalisé une bonne reconnaissance des classes minoritaires et majoritaires.

Les travaux et les résultats de ce sujet de thèse ont confirmé clairement que l'utilisation de la méthode de pondération LMS est effectivement très pertinente pour équilibrer les ensembles de données médicales, ainsi que les résultats obtenus après l'application de cette méthode sont très prometteurs et ils sont comparables aux travaux déjà réalisés dans la littérature.

Perspectives

Quelques pistes de recherches peuvent être ouvertes à partir des résultats obtenus dans cette thèse de doctorat, nous citons en particulier les points suivants :

- L'interprétabilité des résultats (application de la méthode de pondération LMS sur d'autres classifieurs dits intelligents à base de la logique floue et les méthodes d'ensemble comme boosting, bagging)
- Traitement des problèmes des petits disjoints, des données manquantes, des données bruitées, des données chevauchées et des données frontières pour les ensembles de données binaires et multi-classes.
- Application de notre méthode sur des ensembles de données réelles (binaire et multi classes) de très grande taille avec un degré de déséquilibre plus fort.
- Etude du problème de déséquilibre dans le cas des ensembles de données multi-labels.
- Généralisation de notre méthode sur d'autres domaines d'application (biochimique, biologique, bio-informatique,…).
- Traitement du problème de déséquilibre dans l'apprentissage non supervisé et semi-supervisé.

Production scientifiques

Article de revue international

1. **Belarouci, S.,** Bouchikhi, S. and Chikh, M.A. Comparative study of balancing methods: case of imbalanced medical data. Int. J. Biomedical Engineering and Technology, (Paper in press).

Articles de conférences internationales

1. **Sara Belarouci**, Sarra Bouchikhi and M. Amine. Chikh. Neuronal classification of imbalanced medical data. Biomedical Engineering International Conference (BIOMEIC'14), 15-16 October, Tlemcen, Algeria, 2014.
2. **Sara Belarouci,** Sarra Bouchikhi and M. Amine. Chikh. A study of imbalanced medical data using the least mean square algorithm. International Conference on Distributed Systems and Decision (ICDSD'14), 07-08 December, Oran, Algeria, 2014.
3. Sarra Bouchikhi, **Sara Belarouci** and M. Amine. Chikh. Classification of Cardiac Arrhythmias Using HMMs. La Troisième conférence International sur les Système Complexes (CISC'14), 09-10 Décembre, Jijel, Algérie, 2014.

Articles de conférences nationales

1. **Sara Belarouci**, M. Amine. Chikh. Classification neuronale des données médicales déséquilibrées. 4ème journée doctorale en Génie biologique et médical (JD-GBM '2014), 15 mai, Tlemcen, Algérie, 2014.
2. **Sara Belarouci**, M. Amine. Chikh. A Comparison study of balancing methods Case of imbalanced medical data. 5ème journée doctorale en Génie biologique et médical (JD-GBM'2015), 11 Juin, Tlemcen, Algérie, 2015.
3. **Sara Belarouci**, M. Amine. Chikh. Classification of imbalanced medical data using CART Classifier with LMS Algorithm. 6ème Journée doctorale en Génie Biologique et Médicale (JD-GBM'2016), 5 mai, Tlemcen, Algérie, 2016.

Bibliographie

[1] Emna Bahri. Amélioration des méthodes adaptatives pour l'apprentissage supervise des données réelles. Thèse de doctorat, Université de Lumière Lyon 2, France, Décembre 2010.

[2] PATTERSON, G., AND ZHANG, M. Fitness functions in genetic programming for classification with unbalanced data. In *Proceedings of the 20th Australasian Joint Conference on Artificial Intelligence*, Vol. 4830 of *LNCS*, pp. 769–775, 2007.

[3] DOUCETTE, J., AND HEYWOOD, M. I. GP classification under imbalanced data sets: Active sub-sampling and AUC approximation. In *Proceedings of 11th European Conference in Genetic Programming (EuroGP 08)*, pp. 266– 277, 2008.

[4] Urvesh Bhowan. Genetic Programming for Classification with Unbalanced Data. Ph.D. thesis, Victoria University of Wellington, 2012.

[5] Y. Sun, A. K. C. Wong, and M. S. Kamel. Classification of imbalanced data: A review. *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 23, pp. 687-719, 2009.

[6] Piyasak Jeatrakul. Enhancing Classification Performance over Noise and Imbalanced Data Problems. Ph.D. thesis, Murdoch University, March 2012.

[7] Z.-H. Zhou and X.-Y. Liu. Training cost-sensitive neural networks with methods addressing the class imbalance problem. *IEEE Transactions on Knowledge and Data Engineering*, Vol.18, No.1, pp.63-77, 2006.

[8] HOLMES, J. H. Differential negative reinforcement improves classifier system learning rate in two-class problems with unequal base rates. In *Proceedings of the Third Annual Conference on Genetic Programming*, pp. 635–644, 1998.

[9] PAZZANI, M., MERZ, C., MURPHY, P., ALI, K., HUME, T., AND BRUNK, C. Reducing misclassification costs. In *Proceedings of the 11th International Conference of Machine Learning*, Morgan Kaufmann, pp. 217–225, 1994.

[10] YAN, L., DODIER, R., MOZER, M. C., AND WOLNIEWICZ, R. Optimizing classifier performance via the Wilcoxon-Mann-Whitney statistic. In *Proceedings of the Twentieth International Conference on Machine Learning (ICML 03)*, pp. 848–855, 2003.

[11] CARUANA, R. Data mining in metric space: An empirical analysis of supervised learning performance criteria. In *Proceedings of ROC Analysis in AI Workshop (ECAI)*, ACM Press, pp. 69–78, 2004.

[12] S. B. Kotsiantis. Supervised machine learning: A review of classification techniques. *Informatica*, Vol. 31, pp. 249-268, 2007.

[13] Japkowicz, N. Aaai' workshop on learning from imbalanced data sets. Technical Report WS-00-05, AAAI, 2000.

[14] Chawla, N., N. J. and Kolcz, A. ICML' Workshop on learning from imbalanced data sets. 2003.

[15] Chawla, N., Japkowicz, N., & Kolcz, A. Special Issue on Class Imbalances. *SIGKDD Explorations*, Vol.6, No.1, pp.1-6, 2004.

[16] Provost, F. Machine learning from imbalanced data sets 101. *Invited paper for the AAAI' 2000 Workshop on Imbalanced Data Sets*, 2000.

[17] Yanmin Sun. Cost-Sensitive Boosting for Classification of Imbalanced Data. Ph.D. thesis, University of Waterloo, Waterloo, Ontario, Canada, 2007.

[18] Victoria Lopez, Alberto Fernandez, Salvador Garcia, Vasile Palade, Francisco Herreraa. An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information Sciences*, Vol. 250, pp.113–141, 2013.

[19] A. Orriols-Puig, E. Bernado-Mansilla, D.E. Goldberg, K. Sastry, P.L. Lanzi. Facetwise analysis of XCS for problems with class imbalances. *IEEE Transactions on Evolutionary Computation*, Vol. 13, No.5, pp. 1093-1119, 2009.

[20] G.M. Weiss, F.J. Provost. Learning when training data are costly: the effect of class distribution on tree induction. *Journal of Artificial Intelligence Research*, pp. 315–354, 2003.

[21] T. Jo, N. Japkowicz. Class imbalances versus small disjuncts. *ACM SIGKDD Explorations Newsletter*, Vol. 6, No. 1, pp. 40-49, 2004.

[22] K. Napierala, J. Stefanowski, S. Wilk. Learning from imbalanced data in presence of noisy and borderline examples. In: *Proceedings of the 7th International Conference on Rough Sets and Current Trends in Computing (RSCTC'10), Springer Berlin Heidelberg*, pp. 158–167, 2010.

[23] G.M. Weiss, Mining with rarity: a unifying framework, *ACM SIGKDD Explorations Newsletter*, Vol. 6, No 1, pp. 7-19, 2004.

[24] J.H. Friedman, R. Kohavi, Y. Yun. Lazy decision trees. In: *Proceedings of the AAAI/IAAI*, Vol. 1, pp. 717–724, 1996.

[25] G.M. Weiss. Mining with rare cases. In: O. Maimon, L. Rokach (Eds.). Data Mining and Knowledge Discovery Handbook. Springer, pp. 765– 776, 2005.

[26] G.M. Weiss. The impact of small disjuncts on classifier learning. In*: R. Stahlbock, S.F. Crone, S. Lessmann (Eds.), Data Mining: Annals of Information Systems*, Vol. 8, Springer, pp. 193–226, 2010.

[27] N. Japkowicz. Concept-learning in the presence of between-class and within-class imbalances. *In: E. Stroulia, S. Matwin (Eds.), Proceedings of the 14th Canadian Conference on Advances in Artificial Intelligence (CCAI'08),* Lecture Notes in Computer Science, Vol. 2056, Springer, pp. 67–77, 2001.

[28] R.C. Holte, L. Acker, B.W. Porter. Concept learning and the problem of small disjuncts. In: *Proceedings of the International Joint Conferences on Artificial Intelligence,* IJCAI'89, pp. 813–818, 1989.

[29] D.R. Carvalho, A.A. Freitas. A hybrid decision tree/genetic algorithm method for data mining. *Information Sciences*, Vol. 163, No. 1, pp. 13-35, 2004.

[30] K.M. Ting. The problem of small disjuncts: its remedy in decision trees. In: Proceedings of the 10th Canadian Conference on Artificial Intelligence (CCAI'94), pp. 91–97, 1994.

[31] P. Ducange, B. Lazzerini, F. Marcelloni. Multi-objective genetic fuzzy classifiers for imbalanced and cost-sensitive datasets. *Soft Computing*, Vol. 14, No. 7, pp.713–728, 2010.

[32] R.E. Schapire. A brief introduction to boosting. In: *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'99)*, pp. 1401–1406, 1999.

[33] S.J. Raudys, A.K. Jain. Small sample size effects in statistical pattern recognition: recommendations for practitioners. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 13, No.3, pp. 252–264, 1991.

[34] M. Wasikowski, X.-W. Chen. Combating the small sample class imbalance problem using feature selection. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 22, No. 10, pp. 1388–1400, 2010.

[35] R.C. Prati, G.E.A.P.A., Batista. Class imbalances versus class overlapping: an analysis of a learning system behavior. In: *Proceedings of the 2004 Mexican International Conference on Artificial Intelligence (MICAI'04)*, pp. 312–321, 2004.

[36] V. Garcia, R.A. Mollineda, J.S. Sanchez. On the k-NN performance in a challenging scenario of imbalance and overlapping. *Pattern Analysis Applications*, Vol. 11, No. 3–4, pp. 269–280, 2008.

[37] M. Denil, T. Trappenberg. Overlap versus imbalance. *In: Proceedings of the 23rd Canadian Conference on Advances in Artificial Intelligence (CCAI'10), Lecture Notes on Artificial Intelligence*, Vol. 6085, pp. 220–231, 2010.

[38] J. Luengo, A. Fernandez, S. Garcia, and F. Herrera. Addressing data complexity for imbalanced data sets: analysis of SMOTE-based oversampling and evolutionary undersampling. *Soft Computing*, Vol. 15, N.10, pp.1909–1936, 2011.

[39] R. Martin-Felez, R.A., Mollineda. On the suitability of combining feature selection and resampling to manage data complexity. In *Current Topics in Artificial Intelligence*, Springer Berlin Heidelberg, Vol. 5988, pp. 141–150, 2009.

[40] J.A. Saez, J. Luengo, F. Herrera. A first study on the noise impact in classes for fuzzy rule based classification systems. In: *Proceedings of the 2010 IEEE International Conference on Intelligent Systems and Knowledge Engineering (ISKE'10),* IEEE Press, pp. 153–158, 2010.

[41] X. Zhu, X. Wu. Class noise vs. attribute noise : a quantitative study. *Artificial Intelligence Review*, Vol. 22, No. 3, pp. 177–210, 2004.

[42] R. Batuwita, V. Palade. FSVM-CIL: fuzzy support vector machines for class imbalance learning. *IEEE Transactions on Fuzzy Systems*, Vol. 18, No. 3, pp. 558–571, 2010.

[43] C. Seiffert, T.M. Khoshgoftaar, J. Van Hulse, A. Folleco. An empirical study of the classification performance of learners on imbalanced and noisy software quality data. *Information Sciences*, Vol. 259, pp. 571-595, 2014.

[44] T.M. Khoshgoftaar, J. Van Hulse, A. Napolitano. Comparing boosting and bagging techniques with noisy and imbalanced data. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, Vol.41, No. 3, pp.552–568, 2011.

[45] M. Kubat, S. Matwin. Addressing the curse of imbalanced training sets: one-sided selection. In: *Proceedings of the 14th International Conference on Machine Learning (ICML'97),* 1997, pp. 179–186.

[46] D.J. Drown, T.M. Khoshgoftaar, N. Seliya. Evolutionary sampling and software quality modeling of high-assurance systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part A, Vol.* 39, No.5, pp. 1097–1107, 2009.

[47] H. Han, W.Y. Wang, B.H. Mao. Borderline–SMOTE: a new over–sampling method in imbalanced data sets learning. In: *Proceedings of the 2005 International Conference on Intelligent Computing (ICIC'05), Lecture Notes in Computer Science,* Vol. 3644, pp. 878–887, 2005.

[48] C. Bunkhumpornpat, K. Sinapiromsaran, C. Lursinsap. Safe–level–SMOTE: Safe–level–synthetic minority over–sampling TEchnique for handling the class imbalanced problem. In: *Advances in Knowledge Discovery and Data Mining, Springer Berlin Heidelberg*, pp. 475–482, 2009.

[49] H. He, Y. Bai, E.A. Garcia, S. Li. ADASYN: adaptive synthetic sampling approach for imbalanced learning. In: *Proceedings of the 2008 IEEE International Joint Conference on Neural Networks (IJCNN'08),* pp. 1322–1328, 2008.

[50] V. Lopez, A. Fernandez, M.J. del Jesus, F. Herrera. A hierarchical genetic fuzzy system based on genetic programming for addressing classification with highly imbalanced and borderline data-sets. *Knowledge-Based Systems,* Vol. 38, pp.85–104, 2013.

[51] R. Alaiz-Rodriguez, N. Japkowicz. Assessing the impact of changing environments on classifier performance. In: *Proceedings of the 21st Canadian Conference on Advances in Artificial Intelligence (CCAI'08),* Springer-Verlag, Berlin, Heidelberg, pp. 13–24, 2008.

[52] J.Q. Candela, M. Sugiyama, A. Schwaighofer, N.D. Lawrence. Dataset Shift in Machine Learning. *The MIT Press*, 2009.

[53] H. Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, Vol. 90, No. 2, pp. 227–244, 2000.

[54] J.G. Moreno-Torres, F. Herrera. A preliminary study on overlapping and data fracture in imbalanced domains by means of genetic programming-based feature extraction. In: *Proceedings of the 10th International Conference on Intelligent Systems Design and Applications (ISDA'10)*, pp. 501–506, 2010.

[55] D.A. Cieslak, N.V. Chawla. Analyzing pets on imbalanced datasets when training and testing class distributions differ. In: *Proceedings of the Pacific- Asia Conference on Knowledge Discovery and Data Mining (PAKDD08)*, Osaka, Japan, pp. 519–526, 2008.

[56] J.G. Moreno-Torres, X. Llora, D.E. Goldberg, and R. Bhargava. Repairing fractures between data using genetic programming-based feature extraction: a case study in cancer diagnosis. *Information Sciences*, Vol. 222, pp. 805–823, 2013.

[57] S. Bickel, M. Bruckner, and T. Scheffer. Discriminative learning under covariate shift. *Journal of Machine Learning Research*, Vol. 10, pp. 2137–2155, 2009.

[58] A. Globerson, C.H. Teo, A. Smola, and S. Roweis. An adversarial view of covariate shift and a minimax approach. In: *J. Quinonero Candela, M. Sugiyama, A. Schwaighofer, N.D. Lawrence (Eds.), Dataset Shift in Machine Learning, The MIT Press,* pp. 179–198, 2009.

[59] Simon Marcellin. Arbres de décision en situation d'asymétrie. Thèse de doctorat, Université de Lumière Lyon II, France, Septembre 2008.

[60] M. Maloof. Learning when data sets are imbalanced and when costs are unequal and unknown. In *ICML Workshop on Learning from Imbalanced Data Sets II*, Vol. 2, 2003.

[61] Peter D. Turney. Types of cost in inductive concept learning. In *Workshop on Cost-Sensitive Learning at ICML2000*, pp. 15–21, Stanford University, California, 2002.

[62] Ming Tan. Cost-sensitive learning of classification knowledge and its applications in robotics. *Mach. Learn*., Vol. 13, No. 1, pp.7–33, 1993.

[63] Nathalie Japkowicz. Concept learning in the presence of between-class and within-class imbalances. In: *Advances in artificial intelligence*. Springer Berlin Heidelberg, London, UK, pp. 67–77, 2001.

[64] S. Visa and A. Ralescu. Learning imbalanced and overlapping classes using fuzzy sets. In *Workshop on Learning from Imbalanced Datasets (ICML'03),* Vol. 3, 2003.

[65] Kolcz, A. Chowdhury, and J. Alspector. Data duplication: An imbalance problem?. *In ICML'2003 Workshop on Learning from Imbalanced Datasets*, 2003.

[66] S. Kotsiantis, D. Kanellopoulos, and P. Pintelas. Handling imbalanced datasets: A review. *GESTS International Transactions on Computer Science and Engineering*, Vol. 30, No.1, pp.25–36, 2006.

[67] N. Japkowicz. Learning from imbalanced data sets: A comparison of various strategies. In *AAAI Workshop on Learning from Imbalanced Data Sets*, Menlo Park, CA, Vol.68, pp. 10–15, 2000.

[68] Nathalie Japkowicz. The class imbalance problem: Significance and strategies. In *Proceedings of the 2000 International Conference on Artificial Intelligence (IC-AI'2000),* Vol. 1, pp. 111–117, 2000.

[69] Nathalie Japkowicz and Shaju Stephen. The class imbalance problem: A systematic study. *Intell. Data Anal*., Vol. 6, No. 5, pp. 429–449, 2002.

[70] Cheikh Ndour. Modélisation statistique de la mortalité maternelle et néonatale pour l'aide à la planification et à la gestion des services de santé en Afrique Sub-Saharienne. Thèse de doctorat, Université de Pau et des Pays de L'ADOUR & université Gaston Berger de SAINT-LOUIS, août 2006.

[71] Aymen CHERIF. Réseaux de neurones, SVM et approches locales pour la prévision des séries temporelles. Thèse de doctorat, Université François - Rabelais de Tours, juillet 2013.

[72] G. H. Nguyen, A. Bouzerdoum, and S. L. Phung. Learning pattern classification tasks with imbalanced data set. In Pattern Recognition, P.-Y. Yin, Ed.: InTech, 2009.

[73] T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano. Supervised neural network modeling: An empirical investigation into learning from imbalanced data with labeling errors. IEEE Transactions on Neural Networks, Vol. 21, pp. 813-830, 2010.

[74] S. C. Chen, S. W. Lin, T. Y. Tseng, and H. C. Lin. Optimization of backpropagation network using simulated annealing approach. In IEEE International Conference on Systems, Man and Cybernetics (SMC '06), pp. 2819-2824, 2006.

[75] D. Anderson and G. McNeill. Artificial neural networks technology: A DACS state-of-the-art report. Kaman Sciences Corporation, 1992.

[76] J. Nazari and O. K. Ersoy. Implementation of back-propagation neural networks with MatLab. *School of Electrical Engineering*, Purdue University, 1992.

[77] R. Ball and P. Tissot. Demonstration of artificial neural network in Matlab. *Division of Nearshore Research*, Texas A&M University, 2006.

[78] Settouti Nesma. Renforcement de l'Apprentissage Structurel pour la Reconnaissance du Diabète. Thèse de Magister, Université de Tlemcen, Algérie, Juin 2011.

[79] V. Vapnik. The nature of Statistical Learning Theory. *Springer-Verlag*, New York, USA, 1995.

[80] A. Sun, E.-P. Lim and Y. Liu. On strategies for imbalanced text classification using SVM: A comparative study. *Decision Support Systems*, Vol. 48, No. 1, pp. 191-201, 2009.

[81] Y. Tang, Y.-Q. Zhang, N. V. Chawla, and S. Krasser. SVMs modeling for highly imbalanced classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, Vol. 39, pp. 281-288, 2009.

[82] HASAN, Mohamadally et BORIS, Fomani. SVM : Machines à vecteurs de support ou séparateurs à vastes marges. *Rapport technique, Versailles St Quentin,* France. Cité, 2006, pp. 64.

[83] WASSILA, Boukhari et MOHAMED, Benyettou. Identification Biométrique des Individus par leurs Empreintes Palmaires «Palmprints» : Classification par la Méthode des Séparateurs à Vaste Marge (SVM). LAMOSI- Université des Sciences et de la Technologie d'Oran-USTO- Algérie.

[84] Grâce CAPO-CHICHI. Apprentissage Automatique pour la détection de relations d'affaire. Thèse de Maîtrise, Université de Montréal, Avril 2012.

[85] E. Fix, J.L. Hodges. Discriminatory analysis, nonparametric discrimination consistency properties. Technical Report 4, United States Air Force, Randolph Field, TX.

[86] J. Laurikkala. Improving identification of difficult small classes by balancing class distribution. In *Proceedings of the 8th Conference on AI in Medicine in Europe: Artificial Intelligence Medicine*, London: Springer-Verlag, pp. 63-66, 2001.

[87] R. Barandela, J. S. Sanchez, V. Garcia, and E. Rangel. Strategies for learning in class imbalance problems. *Pattern Recognition*, Vol. 36, No.3, pp. 849-851, 2003.

[88] Antoine Cornuéjols, Laurent Miclet et Yves Kodratoff. Apprentissage artificiel : Concepts et algorithmes. Eyrolles, 2002 (première édition), broché, ISBN: 2-212-11020-0.

[89] Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., & Herrera, F. A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid based approaches. *IEEE Transactions on Systems, Man, and Cybernetics Part C: Applications and Reviews*, Vol. 42, No. 4, pp.463–484, 2012.

[90] T. Joachims. A support vector method for multivariate performance measures. In *Proceedings of the 22nd international conference on Machine learning*, ACM, pp. 377–384, 2005.

[91] G. Wu and E. Y. Chang. Class-boundary alignment for imbalanced dataset learning. In *ICML 2003 workshop on learning from imbalanced data sets II*, Washington, DC, pp. 49–56, 2003.

[92] G. Wu and E. Y. Chang. Kba: Kernel boundary alignment considering imbalanced data distribution. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 17, No. 6, pp.786–795, 2005.

[93] Roberto D'Ambrosio. Handling Imbalanced Datasets by Reconstruction Rules in Decomposition Schemes. Ph.D. thesis, Campus Bio-Medico University, Roma, Italy, May 2014.

[94] K. Huang, H. Yang, I. King, and M. R. Lyu. Learning classifiers from imbalanced data based on biased minimax probability machine. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2004)*, IEEE, Vol. 2, pp. II-558-II-563, 2004.

[95] TAO Xiao-yan, JI Hong-bing. A Modified PSVM and its Application to Unbalanced Data Classification. *In: Third International Conference on Natural Computation (ICNC 2007)*, IEEE, pp. 488-490, 2007.

[96] WANG, Benjamin X. et JAPKOWICZ, Nathalie. Boosting support vector machines for imbalanced data sets. *Knowledge and Information Systems*, Vol. 25, No. 1, pp. 1-20, 2010.

[97] Lei Zhu, Shaoning Pang, Gang Chen, and Abdolhossein Sarrafzadeh. Class imbalance robust incremental LPSVM for data streams learning. In: *The 2012 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 10-15 June, Brisbane, QLD, pp. 1-8, ISSN: 2161-4393, 2012.

[98] Z. Liu, H. Cao, X. Chen, et al. Multi-fault classification based on wavelet SVM with PSO algorithm to analyse vibration signals from rolling element bearings. *Neuro-computing*, Vol. 99, No. 1, pp. 399 – 410, 2013.

[99] S. Datta, S. Das. Near- Bayesian support vector machines for imbalanced data classification with equal or unequal misclassification costs. *Neural Netw*. Vol. 70, pp. 39–52, 2015.

[100] X. Wu et al. Top 10 algorithms in data mining. *Knowledge and Information Systems*, Vol. 14, No.1, pp. 1-37, 2008.

[101] Y. Song, J. Huang, D. Zhou, H. Zha, and C. L. Giles. Iknn: Informative k-nearest neighbor pattern classification. *In Proc. PKDD*, Vol. 4702, pp. 248-264, 2007.

[102] L. Yuxuan and X. Zhang. Improving k nearest neighbor with exemplar generalization for imbalanced classification. *Advances in Knowledge Discovery and Data Mining*, Springer, pp. 321–332, 2011.

[103] W. Liu and S. Chawla. Class confidence weighted knn algorithms for imbalanced data sets. *Advances in Knowledge Discovery and Data Mining*, pp. 345–356, 2011.

[104] X. Zhang, Y. Li. A positive-biased nearest neighbour algorithm for imbalanced classification. *In: Advances in Knowledge Discovery and Data Mining*, Springer, pp. 293–304, 2013.

[105]C. Liu, L. Cao, P. Yu. Coupled fuzzy k-nearest neighbors classification of imbalanced non-IID categorical data. In: *Proceedings of the International Joint Conference on Neural Networks*, pp. 1122–1129, 2014 a.

[106] C. Liu, L. Cao, P. Yu. A hybrid coupled k-nearest neighbour algorithm on imbalance data. In: *Proceedings of the International Joint Conference on Neural Networks*, pp.2011–2018, 2014b.

[107] Guansong Pang, Huidong Jin, Shengyi Jiang. CenKNN: a scalable and effective text classifier. *Data Mining and Knowledge Discovery*, Vol. 29, No 3, pp. 593-625, 2015.

[108] Y. L. Murphey, H. Wang, G. Ou and L. A. Feldkamp. OAHO: An effective algorithm for multi class learning from imbalanced data. *Proc. of IEEE International Joint Conference on Neural Networks*, Orlando, FL, USA, pp. 406-411, 2007.

[109] R. Anand, K. G. Mehrotra, C. K. Mohan and S. Ranka. An improved algorithm for neural network classification of imbalanced training sets. I*EEE Transactions on Neural Networks*, Vol.4, No.6, pp.962- 969, 1993.

[110] Y. Lu, H. Guo and L. Feldkamp. Robust neural learning from unbalanced data samples. *Proc. of the IEEE World Congress on Computational Intelligence*, Anchorage, AK, USA, pp.1816-1821, 1998.

[111] X. Fu, L. Wang, K. S. Chua and F. Chu. Training RBF neural networks on unbalanced data. *Proc. of the International Conference on Neural Information Processing, Singapore*, Vol. 2, pp. 1016-1020, 2002.

[112] R. Alejo, V. Garcia, J. M. Sotoca, R. A. Mollineda and J. S. Snchez. Improving the classification accuracy of RBF and MLP neural networks trained with imbalanced samples. *Proc. of the 7th International Conference on Intelligent Data Engineering and Automated Learning*, Burgos, Spain, pp. 464-471, 2006.

[113] Z. Q. Zhao. A novel modular neural network for imbalanced classification problems. *Patter Recognition Letters*, Vol.30, pp.783-788, 2008.

[114] G. H. Nguyen, A. Bouzerdoum and S. L. Phung. A supervised learning approach for imbalanced data sets. *Proc. of the 19th International Conference on Pattern Recognition*, pp.1-4, 2008.

[115] A. Adam, I. Shapiai, Z. Ibrahim, M. Khalid, L. C. Chew, L. W. Jau and J. Watada. A Modified Artificial Neural Network Learning Algorithm for Imbalanced Data Set Problem. *Second International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN),* IEEE, 28-30 July, Liverpool, pp.44 – 48, 2010.

[116] Asrul Adam, Zuwairie Ibrahim, Mohd Ibrahim Shapiai, Lim Chun Chew, Lee Wen Jau, Marzuki Khalid and Junzo Watada. A two-step supervised learning artificial neural network for imbalanced dataset problems. *International Journal of Innovative Computing, Information and Control (ICIC),* Vol. 8, pp. 3163-3172, 2012.

[117] M. D. Pérez-Godoy, A. J Rivera, C. J. Carmona, et al. Training algorithms for radial basis function networks to tackle learning processes with imbalanced data-sets. *Appl. Soft Comput.* Vol. 25, pp. 26–39, 2014.

[118] Umi Mahdiyaha, M. Isa Irawana, Elly Matul Imahb. Integrating Data Selection and Extreme Learning Machine for Imbalanced Data. *International Conference on Computer Science and Computational Intelligence (ICCSCI 2015),* Procedia Computer Science, Vol. 59, pp. 221 – 229, 2015.

[119] Chawla, N.V., Bowyer, K.W., Hall, L.O., & Kegelmeyer, W.P. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence and Research*, Vol.16, pp.321-357, 2002.

[120] Jong Myong Choi. A Selective Sampling Method for Imbalanced Data Learning on Support Vector Machines. *Ph.D. Dissertation*, Iowa State University, Paper 11529, 2010.

[121] N.V. Chawla, A. Lazarevic, L.O. Hall, K.W. Bowyer. SMOTEBoost: Improving prediction of the minority class in boosting. In: *Proceedings of 7th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'03)*, September 22-26, Cavtat Dubrovnik, pp. 107–119, 2003.

[122] M. Gao, X. Hong, S. Chen, C.J. Harris. A combined SMOTE and PSO based RBF classifier for two-class imbalanced problems. *Neurocomputing*, Vol. 74, pp. 3456–3466, 2011.

[123] Kung-Jeng Wanga, Bunjira Makond, Kun-Huang Chen and Kung-Min Wang. A hybrid classifier combining SMOTE with PSO to estimate 5-year survivability of breast cancer patients. *Applied Soft Computing*, Vol. 20, pp. 15–24. , 2014,

[124] José A. Sáez, Julián Luengo, Jerzy Stefanowski and Francisco Herrera. SMOTE–IPF: Addressing the noisy and borderline examples problem in imbalanced classification by a re-sampling method with filtering. *Information Sciences*, Vol. 291, pp. 184–203, 2015.

[125] I. Tomek. Two modifications of cnn. *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 6, pp.769–772, 1976.

[126] Silvia Cateni, Valentina Colla and Marco Vannucci. A method for resampling imbalanced datasets in binary classification tasks for real-world problems. *Neurocomputing*, Vol.135, pp.32–41, 2014.

[127] Zhang, J. and Mani, I. KNN approach to unbalanced data distributions: A case study involving information extraction. In *proceeding of the ICML Workshop on Learning from Imbalanced Datasets, 2003.*

[128] R. Barandela , R. M. Valdovinos, J. S. Sánchez, and F. J. Ferri. The imbalanced training sample problem: under or over sampling?. In *Structural, Syntactic, and Statistical Pattern Recognition*, Springer Berlin Heidelberg, Vol. 3138, pp. 806-814, 2004.

[129] THOMAS, Julien. Apprentissage supervisé de données déséquilibrées par forêt aléatoire. 2009. Thèse de doctorat. Université de Lumière Lyon 2, France, Février 2009.

[130] S.-J. Yen and Y.-S. Lee. Cluster-based under-sampling approaches for imbalanced data distributions. *Expert Systems with Applications*, Vol. 36, pp. 5718–5727, 2009.

[131] M. Mostafizur Rahman and D. N. Davis. Cluster Based Under-Sampling for Unbalanced Cardiovascular Data. In *the Proceedings of the World Congress on Engineering (WCE 2013),* July 3 - 5, London, U.K., 2013.

[132] Annarita D'Addabbo and Rosalia Maglietta. Parallel selective sampling method for imbalanced and large data classification. *Pattern Recognition Letters*, Vol. 62, pp. 61–67, 2015.

[133] Y. X. Peng and J. Yao. AdaOUBoost: adaptive over-sampling and under-sampling to boost the concept learning in large scale imbalanced data sets. In *Proceedings of the ACM SIGMM International Conference on Multimedia Information Retrieval (MIR '10)*, Philadelphia, Pa, USA, pp. 111–118, March 2010.

[134] Yun Qian, Yanchun Liang, Mu Li , Guoxiang Feng and Xiaohu Shi. A resampling ensemble algorithm for classification of imbalance problems. *Neurocomputing*, Vol. 143, pp. 57–67, 2014.

[135] Qiang Wang. A Hybrid Sampling SVM Approach to Imbalanced Data Classification. *Abstract and Applied Analysis, Hindawi Publishing Corporation*, Vol. 2014, Article ID 972786, 7 pages, http://dx.doi.org/10.1155/2014/972786.

[136] P. Domingos. Metacost: a general method for making classifiers cost-sensitive. In KDD '99: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 155–164. ACM, 1999.

[137] W. Fan, S.J. Stolfo, J. Zhang, P.K. Chan. Adacost: misclassification cost-sensitive boosting. In: *Proceedings of the 16th International Conference on Machine Learning (ICML'99),* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, Vol. 99, pp. 97–105, 1999.

[138] M.V. Joshi, V. Kumar, R.C. Agarwal. Evaluating boosting algorithms to classify rare classes: comparison and improvements. In: *Proceedings of the 2001 IEEE International Conference on Data Mining (ICDM'01), IEEE Computer Society,* Washington, DC, USA, pp. 257–264, 2001.

[139] Y. Sun, M.S. Kamel, A.K.C. Wong, Y. Wang. Cost-sensitive boosting for classification of imbalanced data. *Pattern Recognition*, Vol. 40, No. 12, pp. 3358–3378, 2007.

[140] Haibo He, Edwardo A. Garcia. Learning from Imbalanced Data‖. *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*, VOL. 21, NO. 9, 2009.

[141] Zhi-Hua Zhou and Xu-Ying Liu. On multi-class cost-sensitive learning. *Computational Intelligence*, Vol. 26, No. 3, pp. 232–257, 2010.

[142] Adel Ghazikhani, Reza Monsefi, Hadi Sadoghi Yazdi. Ensemble of online neural networks for nonstationary and imbalanced data streams. *Neurocomputing*, Vol. 122, pp. 535–544, 2013.

[143] Yong Zhang and Dapeng Wang. A Cost-Sensitive Ensemble Method for Class-Imbalanced Datasets. In *Abstract and applied analysis, Hindawi Publishing Corporation*, 2013).

[144] Cao, P., Zhao, D., & Zaiane, O. An optimized cost-sensitive SVM for imbalanced data learning. In: *Advances in Knowledge Discovery and Data Mining*. Springer Berlin Heidelberg, pp. 280-292, 2013.

[145] Duan, W., Jing, L., & Lu, X. Y. Imbalanced data classification using cost sensitive support vector machine based on information entropy. In: *Advanced Materials Research*, Vol. 989, pp. 1756–176, 2014.

[146] Komal R. Borisagar and Dr. G.R.Kulkarni. Simulation and Comparative Analysis of LMS and RLS Algorithms Using Real Time Speech Input Signal. *Global Journal of Researches in Engineering,* Vol.10, No. 5, pp. 44, 2010.

[147] Fateh Bouguerra. Contribution à l'optimisation des télécommunications dans les réseaux mobiles. Thèse de Magister, Université de Batna, Batna, Algérie, 2011.

[148] Eric Vitale. Analyse et contrôle des écoulements instationnaires décollés. Thèse de doctorat, Institut National Polytechnique de Toulouse, Toulouse, France, 2005.

[149] Salvador Olmos and Pablo Laguna. Steady-State MSE Convergence of LMS Adaptive Filters with Deterministic Reference Inputs with Applications to Biomedical Signals. *IEEE TRANSACTIONS ON SIGNAL PROCESSING*, VOL. 48, NO. 8, pp. 2229-2241, 2000.

[150] Sk. Nore Johny Basha, Mohammad Zia-Ur-Rahman and Dr B V Rama Mohana Rao. Noise Removal from Electrocardiogram Signals using Leaky and Normalized version of Adaptive Noise Canceller. International Journal of Computer Science & Communication Networks, Vol. 1, No. 1, 2011.

[151] Abderrazek ABDAOUI. Chapitre, I. V. ESTIMATIONS ADAPTATIVES DANS UN CONTEXTE NON STATIONNAIRE : CANAL IONOSPHÉRIQUE. Chapitre de thèse de doctorat, pp. 81, 2006.

[152] Patrick Flandrin. "Data first", ou comment piloter l'analyse par les données. CNRS & Ecole Normale Supérieure de Lyon, 2012-2013.

[153] J.-F. Bercher & P. Jardin. Introduction au filtrage adaptatif. ESIEE Paris, 2003.

[154] Jutten, C. Filtrage lineaire optimal. *During the Fifth Year of the Department 3i Options Images and Signals and Automatic to Joseph Fourier*, Polytech Grenoble University, Grenoble, France, 2010.

[155] Chen, Y., Gu, Y. and Hero, A.O. Regularized least mean square algorithms. 2010, ArXiv: 1012.5066.

[156] Fillon, T. Traitement Numérique du Signal Acoustique pour une Aide aux Malentendants. Thèse de doctorat, Université de Paris, Paris, France, 2004.

[157] H. Robbins et S. Monro. A stochastic approximation method. *Annal Math Stat*, Vol. 22, pp. 400–407, 1951.

[158] UCI MACHINE LEARNING REPOSITORY, http://archive.ics.uci.edu/ml/data / (last update 01/03/2015).

[159] S. M. Weiss, and C. A. Kulikowski. Computer Systems That Learn. 1991.

[160] L. Gonzalez-Abril, H. Nuñez, C. Angulo and F. Velasco. GSVM: An SVM for handling imbalanced accuracy between classes inbi-classification problems. *Applied Soft Computing*, Vol.17, pp. 23–31, 2014.

[161] Yuan-Hai Shao, Wei-Jie Chen, Jing-Jing Zhang, Zhen Wang and Nai-Yang Deng. An efficient weighted Lagrangian twin support vector machine for imbalanced data classification. *Pattern Recognition*, Vol. 47, No 9, pp. 3158-3167, 2014.

[162] Kung Jeng Wang and Angelia Melani Adrian. Breast Cancer Classification Using Hybrid Synthetic Minority Over-Sampling Technique and Artificial Immune Recognition System Algorithm. *International Journal of Computer Science and Electronics Engineering (IJCSEE)*, Vol. 1, No. 3, 2013.

[163] G. NAGA RAMADEVI, Dr. K. USHA RANI and Dr. D. LAVANYA. Evaluation of Classifiers Performance using Resampling on Breast cancer Data. *International Journal of Scientific & Engineering Research*, Vol. 6, No. 2, 2015.

[164] Alberto Cano, Amelia Zafra, and Sebastián Ventura. Weighted Data Gravitation Classification for Standard and Imbalanced Data. *IEEE Transactions on cybernetics*, Vol. 43, No. 6, 2013.

[165] Kung-Jeng Wang, Angelia Melani Adrian, Kun-Huang Chen and Kung-Min Wang. A hybrid classifier combining Borderline-SMOTE with AIRS algorithm for estimating brain metastasis from lung cancer: A case study in Taiwan. *Computer methods and programs in biomedicine*, Vol.119, pp. 63–76, 2015.

[166] Rafael G. Mantovani, André Luis Debiaso Rossi, Joaquin Vanschoren, Bernd Bischl and André C. P. L. F. de Carvalho. Effectiveness of Random Search in SVM hyper-parameter tuning. In *IEEE proceedings of the International Joint Conference on Neural Networks (IJCNN 2015)*, July12-17, Killarney, Ireland, pp. 1-8, 2015.

[167] Antonio J. Tallón-Ballesteros, César Hervás-Martínez, José C. Riquelme and Roberto Ruiz. Feature selection to enhance a two-stage evolutionary algorithm in product unit neural

networks for complex classification problems. *Neurocomputing,* Vol. 114, pp. 107–117, 2013.

[168] Belarouci, S., Bouchikhi, S. and Chikh, M.A. Comparative study of balancing methods: case of imbalanced medical data. Int. J. Biomedical Engineering and Technology, (Paper in press).

Résumé

Dans cette thèse nous traitons un problème majeur rencontré par les méthodes d'apprentissage supervisé dans beaucoup d'application du monde réelle telles que la détection de fraudes ou d'intrusions, le diagnostic médical, la classification de textes, etc. Ce problème concerne le déséquilibre des classes, lorsqu'au moins une classe est sous représentée par rapport aux autres. C'est particulièrement le cas du domaine de diagnostic médical, en effet nous rencontrons souvent des classes de pathologies minoritaires qui sont mal représentées lors de la phase d'apprentissage. Ce problème perturbe les algorithmes d'apprentissage, qui ils présentent une grande précision sur les classes majoritaires et une faible précision sur les classes minoritaires. Afin de remédier à ce problème, nous proposons dans le cadre de cette thèse de doctorat une méthode de pondération basée sur l'algorithme des moindres carrés moyens (LMS), qui pénalise les erreurs des différents échantillons par des poids différents. Cette méthode affecte des poids forts aux différents échantillons de la classe minoritaire et des poids faibles aux différents échantillons des classes majoritaires. Après cette phase d'équilibrage, nous testons plusieurs approches de classification (RNMC, SVM et K-PPV) sur les nouveaux ensembles de données équilibrés. Dans ce travail plusieurs contributions ont été réalisées dans le but d'améliorer les performances de la classification des données déséquilibrées et de focaliser la classification sur les classes minoritaires qui sont d'un grand intérêt lors du traitement des ensembles de données médicales déséquilibrées.

Les travaux réalisés durant cette thèse ont confirmé clairement que l'utilisation de la méthode de pondération LMS est effectivement très pertinente pour équilibrer les ensembles de données médicales, ainsi que les résultats obtenus sont très prometteurs et ils sont comparables aux travaux existants dans la littérature.

Mots clés : *Données déséquilibrées, Réseaux de Neurones, Séparateurs à Vastes Marges, K- plus proche voisin, Méthodes d'échantillonnage, Algorithme LMS.*

Abstract

In this thesis, we treat a major problem related to supervised learning methods in many real world applications such as fraud or intrusion detection, medical diagnosis, the text classification, etc. This problem concerns the imbalance of classes, when at least one class is underrepresented compared to other classes. This is particularly the case of the medical diagnostic field, indeed we often encounter classes of minority diseases that are poorly represented in the learning phase. This problem disrupts learning algorithms that can present high accuracy on the majority class and poor accuracy on the minority classes. To remedy this problem, we propose in this thesis a weighting method based on a Least Mean Square (LMS) algorithm, which penalizes errors of different samples with different weights. This method affects strong weights for the different samples of the minority class and low weights for the different samples of the majority classes. After this balancing phase, we test several classification approaches (NN, SVM and k-NN) on new balanced datasets. In this work, several contributions were made in order to improve the performances of the classification of imbalanced data and focus the classification on minority classes that are most important in the treatment of imbalanced medical datasets.

The work realized during this thesis has clearly confirmed that the use of the LMS weighting method is actually very relevant to balance medical datasets. The obtained results are very promising and they are comparable to the existing works in the literature.

Keywords: *Imbalanced data, Neural Networks, Support Vector Machine, K-Nearest Neighbors, Sampling methods, LMS algorithm.*

الملخص

في هذه الأطروحة نحن نتعامل مع مشكلة كبيرة تتعلق بأساليب التعلم تحت المراقبة التي واجهتها العديد من التطبيقات في العالم حاليا، مثل الكشف عن الغش أو التسلل، التشخيص الطبي، تصنيف النص، الخ. هذه المشكلة تتمثل في عدم التوازن بين الفئات، عندما تكون فئة واحدة على الأقل ممثلة تمثيلا ضعيفا بالمقارنة مع الفئات الأخرى. هذا هو الحال بصفة خاصة في مجال التشخيص الطبي حيث ان المعطيات المتعلقة بالأمراض تمثل دائما فئة الاقلية في معظم المعطيات الطبية وتظهر بشكل ضعيف في مرحلة التعلم. هذه المشكلة تؤثر في طريقة التعلم التي يمكن أن تقدم دقة عالية على معطيات الأغلبية ودقة رديئة على معطيات الأقلية. لمعالجة هذه المشكلة، نقترح في هذه الأطروحة طريقة الترجيح على أساس خوارزمية (LMS)، التي تقوم بتخفيض الاخطاء لعينات مختلفة بأوزان مختلفة. هذه الطريقة تحدد اوزان مرتفعة لعينات مختلفة من معطيات الأقلية واوزان منخفضة لعينات مختلفة من معطيات الأكثرية. بعد هذه المرحلة التي تتمثل في تحقيق التوازن، نحن نقوم باختبار ثلاث مصنفات (RNMC، SVM، K-PPV) على مجموعات المعطيات المتوازنة الجديدة. في هذا العمل قدمت عدة مساهمات من أجل تحسين أداء عملية تصنيف المعطيات الغير متوازنة وتركيز التصنيف على معطيات الأقلية التي هي الأكثر أهمية في معالجة مجموعات المعطيات الطبية الغير متوازنة.

العمل المنجز خلال هذه الأطروحة أكد بوضوح أن استخدام طريقة الترجيح LMS هو في الواقع مهم جدا لتحقيق التوازن في مجموعات المعطيات الطبية، والنتائج التي تم الحصول عليها واعدة جدا وقابلة للمقارنة مع الاعمال المنجزة حاليا في هدا المجال.

الكلمات المفتاحية: المعطيات الغير متوازنة، الشبكات العصبية، آلة دعم المتجه، الجار الأقرب، خوارزمية LMS.