UNIVERSITE SAAD DAHLAB DE BLIDA

Faculté des Sciences de l'ingénieur

Département d'Électronique

THESE DE DOCTORAT

Spécialité : ELECTRONIQUE

LES METHODES AVANCEES POUR LA CLASSIFICATION SEMI-SUPERVISEE DE DONNEES PARTIELLEMENT CONNUES

Par

Mohamed Abdelkader BENCHERIF

Devant le jury composé de :

NAAMANE Abderrahmane	Professeur, U. de Blida	Président
GUESSOUM Abderezak	Professeur, U. de Blida	Rapporteur
BENBLIDIA Nadjia	Professeur, U. de Blida	Membre
GUERTI Mhania	Professeur, Ecole Polytechnique, Alger	Membre
SAYOUD Halim	Professeur, USTHB	Membre
BOUCHAFFRA Djamel	Professeur, CDTA, Alger	Membre
BAZI Yakoub	Docteur, U. King Saud, Saudi Arabia	Invité

Blida, Avril 2015

RESUME

Vue l’importance incommensurable de la télédétection dans les différent domaines telles que l’agriculture, la gestion des inondations, la détection des lieu d’incendie, l’exploration des minerais,…etc., et vue la complexité de la classification de toutes ces données manuellement ; en termes de temps et de coût associé à l'expertise de l'étiquetage. Nous proposons une nouvelle méthode semi-supervisée de classification automatique des images satellitaires, appelée ELMRW, qui intègre : les informations spectrales des images, la classification, à priori, par la machine d’apprentissage extrême (ELM), ainsi que l'algorithme de la marche aléatoire (RW) ; dans un contexte d'apprentissage actif (AL).

Notre méthode a surpassé différents algorithmes de l'état de l'art, notamment pour l'image satellitaire, "**référence**", de l'Université de Pavie en Italie, ou notre nouvel algorithme, appelé "**ELMRW**" a obtenu un taux de global de classification des pixels de 99.85%±0.032, avec un facteur de kappa de 0.998.

ABSTRACT

Due to the immeasurable importance of remote sensing in various fields such as agriculture, flood management, detection of fire places, exploration of minerals, ... etc., and the inherent complexity of manually classifying all these data in terms of time and associated expertise cost. We propose a new automatic semi-supervised classification of images, called ELMRW that integrates: the spectral information of the remote sensing images, an a priori classification using the extreme learning machines (ELM) and the random walker (RW) algorithm through an active learning (AL) approach.

Our new algorithm, called "**ELMRW**" outperformed different algorithms of the state of the art, through the classification of the **reference** satellite image of the University of Pavia in Italy, with an overall classification pixels' rate of 99.85%±0.032, and a kappa factor of 0.998.

ملخص

نظرا للأهمية البالغة للاستشعار عن بعد في مختلف المجالات مثل الزراعة، إدارة الفيضانات، الكشف عن أماكن الحريق، والتنقيب عن المعادن، **...**الخ، و لصعوبة التصنيف اليدوي لكل هذه البيانات من ناحية الوقت و الكلفة المرتبطة بالمختصين **.**نقترح طريقة جديدة تسمى **ELMRW** لتصنيف أوتوماتيكي في إطار شبه- إشراف لصور الأقمار الصناعية. وتستعمل هذه الخوارزمية المعلومات الطيفية للصورة، و التصنيف باستعمال خوارزمية التعليم القصوى **(ELM)**، إضافة الى خوارزمية المشي العشوائي و التعلم النشط.

أظهرت الدراسة تفوق طريقتنا على عدة خوارزميات مختلفة معاصرة، بإستعمال صورة **"**مرجعية**"**، لجامعة بافيا في إيطاليا، و كان معدل التصنيف العام يساوي 99.85٪ ± 0.032، مع عامل كابا بقيمة 0.998.

REMERCIEMENTS

Je tiens en premier lieu à remercier Allah, pour sa miséricorde, pour toutes les chances qu'il m'a offert dans la vie, pour tous les sens dont il m'a pourvu et pour tous les gens qui sont mes amis, pour des parents merveilleux que je ne cesserais d'oublier, qui m'ont permis d'être parent à mon tour, je ne pourrais dire que, dieu protège ceux qui vont lire ces mots, et ceux qui ne pourront les lire.

Je tiens à remercier du fond du cœur mes directeurs de thèse, le Pr Guessoum, et le Dr Yakoub Bazi, pour leur soutien, et leur franchise sur le choix de ce travail.

Je remercie le Professeur Naamane pour avoir accepté de présider ma soutenance, j'exprime toute ma gratitude et tout mon respect au Pr Guerti M'Hania et au Pr Benblidia Nadia pour toute l'attention qu'elles ont portées à mon travail, malgré toutes les charges pédagogiques qu'elles assurent et pour leur précieux avis sur mon travail.

Je remercie très vivement le Pr Sayoud et le Pr Boucheffra pour avoir accepté d'évaluer mon travail très minutieusement, avec tous les engagements les incombant.

Je tiens également à adresser spécialement mes très profonds remerciements au Dr Benselama Zoubir, Docteur à l'université de Blida pour toute l'aide qu'il m'a donné au fil des années…

Je tiens aussi à adresser mes très profonds remerciements au Dr Mansour Alsulaiman, Docteur à l'université de King Saud en Arabie Saoudite pour sa confiance et son soutien.

Je remercie vivement tous mes collègues du CDTA, pour les bons moments qu'on a passé ensemble.

Je tiens, aussi, à remercier mon épouse qui a sacrifié plein de nuits et de journées à s'occuper des enfants, dans l'attente que j'accomplisse cette thèse.

Je remercie particulièrement ma fille Nour-el-Houda pour avoir écrit les longues équations de cette thèse.

TABLE DES MATIERES

LISTE DES ILLUSTRATIONS, GRAPHIQUES ET TABLEAUX

INTRODUCTION

Au cours de ces dernières années, l'apprentissage actif (AL) est devenu très répandu dans la communauté scientifique en général [1-6] et dans la communauté de télédétection spécifiquement [7-9]. L'AL se présente comme une solution complémentaire pour l'amélioration de la classification des données, dont l'étiquetage complet, de toutes les classes existantes, reste une problématique. Vu le nombre incommensurable de ces données, l'expert ne peut étiqueter ou attribuer des classes à toutes ces images [10, 11], vue son temps limité ainsi que le cout associé à cette expertise.

Dans le cadre de la télédétection, tout particulièrement, l'AL contribue au processus de classification d'un ensemble de pixels ou de régions de l'image dans un processus itératif, tout en maintenant une interaction minimale avec l'expert. Le but de l'AL est d'utiliser seulement une partie minimale de l'ensemble des pixels tout en améliorant le taux de classification. Le processus itératif se base sur les pixels classés avec une forte incertitude, c.à.d. les pixels dont la classe prédite ne correspond pas uniquement à la classe réelle, mais correspond à une appartenance croisée entre deux ou plusieurs classes, avec des degrés d'appartenance complémentaires. Dans ce cas, ces pixels sont envoyés à l'expert pour étiquetage, puis le classificateur se base sur cette information pour en étiqueter d'autres pixels similaires, et ainsi de suite. Le contexte d'incertitude est un facteur dominant dans le processus de l'AL, car il accentue la fouille dans la direction des pixels les plus incertains et investi sur cette disparité. Toute la stratégie de recherche s'accentue alors à sélectionner les pixels les plus utiles au classificateur dits 'incertains' pour améliorer le modèle [12], minimisant ainsi le nombre de pixels nécessaires à l'apprentissage, en vue de maintenir la capacité de discrimination la plus haute possible.

D'autres travaux en télédétection investissent aussi sur l'amélioration des critères spatiaux-spectraux [13] [14] pour sélectionner les pixels incertains, qui

s’intègrent dans le paradigme de l’apprentissage semi-supervisé et qui visent à utiliser les pixels étiquetés conjointement avec les pixels non étiquetés, ce qui semble être une alternative très intéressante.

Habituellement, ce paradigme d'apprentissage actif fait référence à la théorie des graphes pour construire la relation entre les pixels étiquetés et non étiquetés par la matrice du Laplacien [7, 15, 16]. Ensuite, le classification est formulée par l’introduction d’une fonctionnelle régularisée, dont la solution finale est exprimée dans un espace noyau [17, 18]

Il convient de rappeler que ce concept a généralement été adopté pour la classification utilisant la machine à vecteurs de support (SVM) [16]. Cependant, dans le contexte de l’apprentissage actif, ce mode d'apprentissage surchargera le classificateur et ne permettra qu'une exploitation partielle de l'information spatiale et contextuelle, et ne permet pas une bonne exploitation de la puissance des méthodes d'optimisation à base de graphes.

L’intégration de nouveaux concepts révolutionnaires en classification des données tels les machines d’apprentissage extrêmes (ELM) [19-25], a changé l’approche traditionnelle des réseaux de neurones en intégrant une nouvelle stratégie de sélection des paramètres du réseau ; les rendant plus performants que les machines à vecteurs support (SVM) [26-29], ou les réseaux Deep Learning (DBN).

Dans diverses contributions récentes, l’ELM a montré des résultats intéressants pour la classification d'images hyper-spectrales [17, 30-33], ainsi que la simplicité que présente ce classificateur dans une formulation unifiée, pour la classification binaire, la classification multi-classes, ainsi que les problèmes de régression, à travers une forme analytique. Nous avons opté dans ce travail à l'intégration de l'ELM avec l'AL, à travers une fonction de mappage des données dans un espace noyau ou kernel, en générant les estimations initiales des pixels non étiquetés, qui seront des conditions initiales à l'algorithme de la marche aléatoire, dans l'intention de classer les pixels restants de l'image. Il est à noter que notre nouvelle méthode de classification appelée ELMRW [18] peut être considérée comme une approche similaire aux champs de Markov [34]. Cependant, la variable de lissage de l’image est introduite à l'aide d'un réseau et

la solution du problème d'optimisation est unique et elle est donnée sous une forme compacte.

Dans le premier chapitre nous allons définir la classification semi-supervisée, en positionnant le problème de classification avec des données étiquetées de taille très réduite comparées à des données non étiquetées largement disponibles avec des couts très réduits.

Dans le second chapitre, nous présenterons le concept de l'apprentissage actif, appelé communément apprentissage par requêtes, ainsi que le critère d'évaluation des différentes requêtes.

Dans le troisième chapitre, nous introduirons un outil essentiel à notre travail, qui est la théorie des graphes ; différents exemples pratiques seront illustrés sur l'utilisation des distances, de la matrice de similarité et la matrice du Laplacien.

Dans le quatrième chapitre, le classificateur ELM sera introduit, ainsi que la formulation de notre problème d'optimisation, en incluant la théorie de la marche aléatoire qui contribuera à l'étiquetage des pixels non-étiquetés par une formule incluant le concept de régularité.

Dans le cinquième chapitre seront décrits les différentes images satellitaires utilisées, ainsi que l'ensemble des résultats de la classification. L'impact de la variation des paramètres de régularisation sera aussi détaillé ; enfin nous compléterons notre travail par une conclusion et quelques recommandations dans une perspective d'améliorer ce travail.

CHAPITRE 1

1. LA CLASSIFCATION SEMI-SUPERVISEE

Do not estimate a density if you need to estimate a function.

Do not estimate a function if you need to estimate values at given points.

Do not estimate predictive values if your goal is to act well.

Vapnik 1995

1.1. Introduction

Durant les dernières décennies, la classification semi-supervisée a vu le jour, en vue de solutionner une nouvelle problématique de classification émanant du flux incommensurable de données. Toutes ces nouvelles données sont générées, par les nouveaux appareils d'acquisition, avec des résolutions nettement améliorées et en quantités passant du méga-octet au téraoctet dans l'espace de quelques années, en plus de tous les appareils déjà existant, tels les télescopes, les caméras de surveillance, les satellites, les sonars, etc.

Selon des estimations statistiques d'IBM [35], l'on considère que chaque jour il y'aurait la création de Téra-documents, entre bases de données biométriques et biologiques, documentaires, séquences vidéos, films, cours, articles scientifiques, journaux, rapport de stage, thèses, caricatures, fichiers de sauvegarde, jeux, images médicales, images prises par les touristes, ainsi que de nouvelles pages web et bases de données de la parole et ceci dans presque toutes les langues.

Selon le même rapport [35] "La production globale d'information en 2012 a atteint 2.8 zettabytes (ZB), soit 2.8 trillion GB, mais seulement 0,5% de ce volume a été utilisé à des fins d'analyse. Le volume de données devrait atteindre 40 ZB en 2020, soit 5,247 GB par personne, avec les économies émergentes en large partie responsable de cette croissance en volume"

Donc classer toutes ces bases de données n'est plus un exemple de démonstration, mais un procédé qui est sujet à une stratégie de classification soit supervisé, soit non-supervisée, soit une alternative médiane qui s'oriente vers classer seulement une partie infime des données et laisser les autres données non étiquetées à une machine, qui utilise un ou plusieurs algorithmes de classification, appelé communément classification semi-supervisée. Donc au résultat, quoi qu'augmentera la taille des données, on ne classera qu'une partie de celles-ci, et cette partie sera la plus réduite possible, par les approches telles que l'apprentissage actif ou l'apprentissage par requêtes.

Nous verrons que le domaine du semi-supervisé a un avenir très impressionnant en termes d'élargissement des classes existantes, selon des degrés de similarité basés sur des attributs finement sélectionnés.

1.2. Histoire du semi-supervisé

L'apprentissage semi-supervisé (SSL) est un mi-chemin entre l'apprentissage supervisé et l'apprentissage non supervisé. En plus des données non étiquetées (classe inconnue), l'algorithme est fourni avec des informations additionnelles de contrôle - mais pas nécessairement pour tous les exemples. Souvent, ces paramètres sont les étiquettes associées à quelques exemples, parfois en nombre très réduit. Dans ce cas, les données fournies au SSL sont les instances : $X := (x_i)_{i \in N}$, qui sont divisées en deux parties : les points $X_L := (x_1, \dots x_l)$, pour lesquels les étiquettes $Y_L := (y_1, \dots y_l)$, sont fournies, et les points $X_U := (x_{l+1}, \dots x_{l+u})$, dont les étiquettes sont inconnues. C'est l'apprentissage "standard" semi-supervisé comme défini dans la littérature [10].

D'autres approches considèrent le SSL comme de l'apprentissage non-supervisé guidé par des contraintes ; En revanche, la plupart des autres approches considèrent le SSL comme de l'apprentissage supervisé avec des informations supplémentaires sur la distribution des exemples fournis.

Cette dernière interprétation semble être la plus proche pour la plupart des applications, où le but est le même que dans l'apprentissage supervisé : c'est-à-dire : prédire une valeur cible pour une donnée x_i. Cependant, ce point de vue ne s'applique pas réellement si le nombre et la nature des classes ne sont pas

connus à l'avance, mais doivent être déduits des données. En conclusion, le SSL considéré comme du non-supervisé avec contraintes reste l'hypothèse la plus plausible dans de telles situations.

Le problème lié au SSL a été introduit par Vapnik [10]. Depuis plusieurs décennies, autrefois appelé apprentissage transductif. Dans ce contexte, un ensemble d'apprentissage étiqueté et un ensemble de test sans étiquettes sont fournis. L'idée de la transduction est d'effectuer des prédictions uniquement sur l'ensemble de test. Ceci est en contraste à l'apprentissage inductif, où le but est de définir une fonction de prédiction sur un ensemble plus large de données, généralement non disponibles lors de l'entrainement.

Probablement, la première idée sur l'utilisation de données non étiquetées, dans la classification est l'auto-apprentissage. Celle-ci est également connu sous le nom d'auto-formation, d'auto-étiquetage, d'auto-apprentissage ou aussi apprentissage piloté par décision, il s'agit de l'algorithme "wrapper", qui utilise à plusieurs itérations un apprentissage supervisé, il commence par l'étiquetage d'un ensemble réduit de données, à chaque étape, une partie des points non étiquetés est étiquetée conformément à la fonction de décision ; puis la méthode supervisée est réutilisée en utilisant ses propres prédictions. Cette idée est apparue dans la littérature depuis un certain temps par exemple, avec Scudder en 1965 et Fralick en 1967; ou avec Agrawala en 1970 [10].

Un aspect insatisfaisant de l'auto-apprentissage est que l'effet du wrapper dépend de la méthode supervisée. Si l'auto-apprentissage est utilisé avec le risque empirique de minimisation et le risque 1-0-perte, les données non étiquetées n'auront aucun effet sur la solution. Mais si, une méthode de maximisation de marge est utilisée, la limite de la décision est repoussée au loin des points non étiquetés, dans d'autres cas, il semblerait que l'auto-apprentissage ne correspond nullement à l'hypothèse de démarrage.

Le SSL est étroitement lié au concept d'inférence transductif, ou transduction, mis au point par Vapnik et Chervonenkis en 1974; et Vapnik et Sterin en 1977[10]. Contrairement à l'inférence inductive, aucune règle de décision générale n'est déduite, mais seulement les étiquettes des points non étiquetés (ou tests) sont générées.

L'apprentissage semi-supervisé a connu son réel envol dans les années 1970, lorsque le problème d'estimation de la règle linéaire discriminante de Fisher avec des données non étiquetées a été considéré par Hosmer en 1973, par McLachlan en 1977, par O'Neill en 1978 ainsi que McLachlan et Ganesalingam en 1982[10].

Plus précisément, ce cadre de travail considérait le cas où chaque classe était conditionnellement une densité Gaussienne avec la même matrice de covariance. La vraisemblance du modèle est alors maximisé en utilisant les données étiquetées et non-étiquetées, à l'aide d'un algorithme itératif tel que la maximisation de l'espérance (EM). D'autres approches avaient considérées l'utilisation d'un mélange de distributions multinomiales.

Le taux d'apprentissage dans un cadre probablement correcte (PAC) [10, 36-38] a réellement été adapté pour l'apprentissage semi-supervisé, lors de l'apprentissage d'un mélange de deux gaussiennes par Ratsaby et Venkatesh en 1995 [39]. Dans le cas d'un mélange indéfini, Castelli et Cover en 1995 ont montré qu'avec un nombre infini de points non étiquetés, la probabilité d'erreur a une convergence exponentielle vers le risque de Bayes (par rapport au nombre d'exemples étiquetés).

L'intérêt pour le SSL a augmenté dans les années 1990, principalement en raison de ses applications dans des problèmes de langage naturel et de classification de textes. Notons que le terme SSL a été utilisé pour la première fois [10], par Merz et al., en 1992 pour la classification avec des données étiquetées et non étiquetées.

1.3. Problématique de l'étiquetage des données

Les nouvelles méthodes de technologie d'acquisition automatique de données permettent de collecter de nombreuses variables mesurables sur divers éléments à très faible coût. Toutefois, la variable d'intérêt est souvent la plus difficile à obtenir que les autres (étiquetage de la classe ou degré d'appartenance). Ceci est particulièrement vrai dans les problèmes de prédiction.

Dans ce cas, il est souhaitable d'apprendre une règle qui permette de prédire la variable d'intérêt, étant donné un ensemble «d'autres» variables obtenues à coût réduit. Dans cet ordre d'idée, le praticien dispose souvent d'un grand nombre de

données non étiquetées et d'un plus petit nombre de données étiquetées. Notons que le cout de l'étiquetage reste potentiellement conditionné par la disponibilité de l'expert, son degré d'expertise et le cout relatif à cet étiquetage.

Nous illustrons trois exemples.

Le premier exemple est celui de la lecture du code postal dans les centres de tri postaux. De nombreux codes postaux sont numérisés de manière automatique à moindre frais. À partir de l'image numérisée, on souhaite classer les différentes nouvelles images, en fonction des chiffres possibles (taille du code postal en vigueur). Cette classification est longue et coûteuse si elle est effectuée par un opérateur humain ; de plus l'opérateur humain peut se fatiguer après des heures de travail, augmentant ainsi le risque d'erreur.

Le second exemple concerne l'indexation du contenu audiovisuel. Différents centres de transmission de données vidéos ou audio disposent de millions d'heures d'enregistrement, qui au fil des années deviennent plus un problème de gestion et d'archivage que de diffusion rapide. Il est alors impératif de classer ces informations pour les retrouver plus rapidement et/ou plus facilement par la suite. L'indexation des séquences vidéos/audio par les experts est très coûteuse, tandis qu'une indexation automatique est moins fastidieuse et moins coûteuse.

Le troisième exemple concerne la reconnaissance automatique de visages, dans de nombreux sites internet de partage de photos, il est maintenant possible de nommer les visages. L'objectif, parmi d'autres, plus intéressant, est de retrouver toutes les photos contenant une même personne. Ici le nombre de fois où la personne a été aperçue (étiquetée) est souvent plus petit devant le nombre de fois où la personne a réellement apparue.

1.4. Classification des données

Le concept de classification est une réalisation humaine, il faut classer les données pour les réutiliser, toutefois l'augmentation des dimensions des attributs et le nombre incommensurable de données rend le cerveau humain dans une incapacité de calcul, ce qui nous amène à l'apprentissage par machines, qui :

- avec des algorithmes à étapes bien structurées,

- avec des hypothèses de classification bien définies,
- ainsi qu'un besoin accrue de généralisation,

Tend la limite de la fouille de données vers un cadre plus général.

La classification des données se répertorie en trois axes principaux :

- La classification non supervisée,
- La classification supervisée,…
- La classification semi-supervisée,

Nous nous intéresserons dans notre contexte d'étude à la classification semi-supervisée, et les méthodes inhérentes à cette approche, toutefois nous présenterons dans la section suivante quelques différences subtiles avec la classification non-supervisée, en vue de motiver notre choix.

1.5. Classification non-supervisée

La classification non supervisée se situe dans un cadre exploratoire. Le nombre de classes ainsi que la signification de la variable qui explique l'hétérogénéité des données sont a priori inconnus. L'objectif de l'analyse est de déterminer des groupes, les plus homogènes entre eux et les plus différents les uns des autres. Nous développons ici quatre exemples, dans divers domaines :

Premièrement en reconnaissance du locuteur, ou les différents attributs des locuteurs sont généralement des attributs spectraux à dimension réduite (<<100), ces attributs sont classés, selon des centres multidimensionnels, qui n'ont pas une explication visuelle ou auditive bien déterminée, d'où un mélange de Gaussienne par locuteur ou par groupe de locuteurs avec des indentations d'attributs très complexes. Les centres ou centroides des locuteurs sont plus sujet a de l'expérimental qu' à une sélection logique de leur nombre [40].

Deuxièmement en marketing, ce type de méthode est utilisé pour faire de la typologie clients (classification selon les types). L'objectif est de partitionner les clients en un certain nombre de catégories, compte tenu des diverses données recueillies. Le but est d'élaborer des campagnes de publicité ciblées pour chaque catégorie de clients, comme le font les opérateurs de téléphonie, en essayant de

cibler les besoins par catégorie de personnes dans la société (étudiants, femmes de foyers, professionnels,….) [41, 42]

Troisièmement en biologie, ou la recherche s'oriente à structurer les êtres vivants en un nombre fini de classes ou d'espèces et obtenir une classification hiérarchique en règnes, embranchements et classes. Soit, uniquement en partition finale par exemple juste "en espèces". Dans ce dernier cas, il convient de définir ce qu'est une espèce, avec des individus homogènes au sein d'une même espèce, mais hétérogènes d'une espèce à l'autre [43, 44].

1.6. La classification semi-supervisée (SSL)

La classification semi-supervisée se divise en deux catégories principales :

Le cas transductif [45] : Cette catégorie se concentre sur la classification des données d'apprentissage dont l'étiquette ou la classe sont inconnues, les données non aperçues dans la phase d'apprentissage ne pourront pas être classées systématiquement.

Le cas inductif [46] : Cette catégorie utilise les données d'apprentissage pour déterminer les paramètres du classificateur ou prédicteur qui sera ensuite utilisé pour classer de nouvelles données.

1.7. Contexte et hypothèses du semi-supervisé

Une question naturelle se pose : Est-ce que le SSL apporte un plus significatif à l'exploration des données, plus précisément, en comparaison avec un algorithme supervisé qui utilise uniquement les données étiquetées, peut-on espérer avoir une prédiction plus précise en prenant en compte les données-non étiquetées ?

Dans une formulation plus mathématique, Ayant une connaissance sur la distribution des données $p(\boldsymbol{x})$, est ce que cette connaissance peut apporter une information additionnelle à l'inférence sur $p(y|x)$. Si ce n'est pas le cas, le SSL ne n'apporte aucune amélioration par rapport à l'apprentissage supervisé. Il serait aussi possible, naïvement, de dire que l'intégration de données non étiquetées dégrade la précision sur la prédiction et fausser l'inférence.

Ce qui est subtile dans le SSL, c'est qu'il est gouverné par des hypothèses définies comme suit [10]:

1.7.1. Hypothèse de régularité

Nous exposons maintenant une généralisation de l'hypothèse de la régularité qui est utile pour l'apprentissage semi-supervisé. Alors que dans le cas du supervisé, la sortie varie régulièrement avec la distance, nous prenons également en compte la densité des entrées. L'hypothèse est que la fonction d'étiquetage du SSL est plus lisse dans les régions denses que dans les régions à faible densité :

> Si deux points x_1, x_2 (pixels pour l'image) sont dans une région à haute densité et sont proches, alors il serait de même pour leurs étiquettes correspondantes y_1, y_2.

Notons que par transitivité, cette hypothèse implique que si deux points sont reliés par un chemin de haute densité (par exemple, si elles appartiennent au même cluster ou groupe), alors les étiquettes sont susceptibles d'être à proximité.

Si, d'autre part, les deux pixels sont séparés par une région de faible densité, alors les étiquettes ne sont pas nécessairement proches.

Notons aussi que l'hypothèse de régularité du SSL s'applique à la fois à la régression et à la classification.

1.7.2. Hypothèse de cluster ou grappe

Supposons que les données (pixels) de chaque classe ont tendance à former un cluster (propriété d'appartenance au même groupe). Alors, les données non étiquetés pourraient aider à trouver les « bordures » de chaque cluster avec plus de précision. L'idée est alors de recourir à un algorithme de groupement (clustering) et d'utiliser les pixels étiquetés pour affecter une classe à chaque cluster. C'est en effet l'une des premières formes du SSL.

L'hypothèse se reformule ainsi :

> Si des pixels sont dans le même cluster, alors ils sont susceptibles d'être dans la même classe.

Cette hypothèse peut être considérée comme raisonnable, sur la base de l'existence même des classes : s'il existe un continuum densément peuplé d'objets, il peut sembler peu probable qu'ils ne soient de classes différentes. Notez que l'hypothèse de cluster ne signifie pas que chaque classe constitue un seul groupe compact, mais signifie seulement que, généralement, on n'observe pas d'objets de deux classes distinctes dans une même «grappe». L'hypothèse de cluster peut facilement être considérée comme un cas particulier de l'hypothèse de régularité su SSL proposé précédemment, étant donné que les clusters sont souvent définis comme étant des ensembles de pixels qui peuvent être reliés par des chemins courts qui traversent seulement les régions à haute densité.

L'hypothèse de cluster peut être formulée de façon équivalente :

> Séparation de faible densité : La frontière de décision devrait se situer dans une région à faible densité.

Cette hypothèse est à relier aux SVM transductifs de Vapnik en 1998; et Joachims en 1999 [47], elle favorise les frontières de classement se situant dans des zones de faible densité. L'information apportée par les données non étiquetées, permet alors une approximation plus précise de cette densité [10].

1.7.3. Hypothèse de dimensionnalité

Comment cela peut-il être utile ? Un problème bien connu par les nombreuses méthodes statistiques et algorithmes d'apprentissage est la soi-disant calamité de la dimension des données ou (Curse of Dimensionality: COD) [48]. Ce problème est lié à la dimensionnalité des données qui augmente de façon exponentielle avec le nombre de dimensions, notamment lors du calcul des taches statistiques telle que l'estimation fiable des densités ou dans les inverses des matrices de covariances par exemple. C'est un problème qui affecte directement les approches génératives qui sont fondées sur l'estimation de densité de l'espace d'entrée.

> Un problème connexe pour les méthodes discriminatoires à hautes dimensions, est que les distances deux à deux ont tendance à devenir plus proches, et donc moins expressives.

Si les données s'encapsulent dans une petite région de faible dimension appelée aussi "collecteur", alors l'algorithme d'apprentissage peut essentiellement fonctionner dans un espace de dimension réduite, évitant ainsi la calamite de la haute dimensionnalité.

> Au fait, les algorithmes travaillant avec les collecteurs peuvent être vus comme une mise en œuvre de l'hypothèse de régularité du SSL : ces algorithmes utilisent la métrique du collecteur pour calculer les distances géodésiques.

Si nous considérons le collecteur comme une approximation d'une région à forte densité, il devient clair que dans ce cas, l'hypothèse de régularité du SSL se réduit à l'hypothèse de régularité au niveau de l'apprentissage supervisé appliqué sur un collecteur.

Notons que si le collecteur est intégré dans l'espace de grande dimension d'entrée dans un mode courbé «Curved Fashion» (c'est à dire qu'il n'est pas seulement un sous-espace), les distances géodésiques diffèrent de celles de l'espace d'entrée.

En utilisant des estimations de densité plus précises et des distances plus appropriées, l'hypothèse du collecteur peut être utile pour la classification ainsi que pour la régression.

1.7.4. Concept de Transduction

Comme mentionné précédemment, certains algorithmes fonctionnent naturellement dans un environnement transductif, selon la philosophie avancée par Vapnik, l'estimation de problèmes à grande dimension devrait tenter de suivre le principe suivant :

> Le principe de Vapnik : Lorsque vous essayez de résoudre un problème, il ne faut pas résoudre un plus problème difficile comme étape intermédiaire.

Prenons comme exemple l'apprentissage supervisé, où les prévisions d'étiquettes $\boldsymbol{Y}$ correspondant à certains objets $\boldsymbol{X}$ sont souhaitées. Les modèles

génératifs estiment la densité de $\boldsymbol{X}$ comme étape intermédiaire, alors que les méthodes discriminatoires estiment directement les étiquettes.

D'une façon plus claire, si les prédictions des étiquettes ne sont requises que pour un ensemble de test donné, la transduction peut être plus souhaitable que l'induction. Alors que la méthode inductive déduit une fonction $f: \boldsymbol{X} \rightarrow Y$ sur tout l'espace $\boldsymbol{X}$, et après évalue les $f(X_i)$ des points de test. La transduction consiste à estimer directement un ensemble fini d'étiquettes de test, c'est une fonction $f: X_u \rightarrow Y$ définie seulement sur les données de test.

Notons que la transduction [10] n'est pas le SSL, car certains algorithmes semi-supervisés sont transductifs, et d'autres sont inductifs.

Maintenant, supposons que nous utilisons un algorithme transductif qui produit une meilleure solution que l'algorithme inductif, tous deux entrainés sur les mêmes données étiquetées (sans prendre en compte les données non étiquetées). La différence de performance pourrait être due à l'une des considérations suivantes, sinon de leur combinaison :

- La transduction suit le principe de Vapnik plus étroitement que ne le fait l'induction.
- L'algorithme transductif tire parti des données non étiquetées d'une manière similaire au SSL.

1.8. Conclusion

Dans ce premier chapitre, nous avons introduit la classification semi-supervisée, et les différentes hypothèses qui la régissent, de telles considérations sont obligatoires, car non pas qu'elles ouvrent des axes de recherche dans le SSL, mais aussi tendent à régulariser les formulations mathématiques des problèmes d'optimisation reliés au SSL.

Dans le deuxième chapitre, nous allons introduire l'apprentissage actif, en expliquant les diverses notions complémentaires à ce type de stratégie, en vue de l'intégrer dans notre méthode proposée au quatrième chapitre.

CHAPITRE 2

2. L'APPRENTISSAGE ACTIF – ACTIVE LEARNING

«Lorsque tu fais quelque chose, sache que tu auras contre toi

Ceux qui voulaient faire la même chose,

Ceux qui voulaient faire le contraire et

L'immense majorité de ceux qui ne voulaient rien faire»

Confucius

2.1. Introduction

Dans ce chapitre, nous allons introduire le concept d'apprentissage actif en reconnaissance des motifs dans le cas général et la classification des données satellitaires ou en télédétection dans un cas plus particulier.

Dans un certain nombre de situations réelles, le praticien dispose d'un ensemble de pixels non étiquetés. Il a la possibilité d'en étiqueter quelques-uns. L'apprentissage actif consiste alors à choisir le plus "judicieusement" possible les pixels à étiqueter de cet ensemble non étiqueté. Ce cadre est appelé apprentissage actif par opposition à l'apprentissage passif, qui lui choisit les points à étiqueter au hasard.

L'apprentissage actif est une approche interactive qui requiert un algorithme d'apprentissage, un ensemble de pixels, un expert qui peut si nécessaire étiqueter un nombre minimal d'instances (pixels).

L'expertise d'étiquetage est un processus complexe en termes de temps et de coût. Donc le problème de classification se résout à utiliser l'algorithme d'apprentissage en n'étiquetant que des pixels bien sélectionnés par ce dernier.

2.2. Apprentissage par requêtes

L’apprentissage actif, (AL) de par sa conception de base est aussi appelé «apprentissage par requêtes», ou «conception expérimentale optimale». C’est une stratégie d’apprentissage qui repose sur le fait que "l’algorithme" choisit ou sélectionne les pixels qui augmentent le taux de reconnaissance, concept totalement diffèrent des méthodes traditionnelles ou passives de classification semi-supervisée. Cette approche pallie le défaut des autres méthodes qui utilisent des milliers voire des millions de pixels en vue d’entrainer le ou les classificateurs, avec comme contrainte majeure, que la plus grande partie de ces pixels ne participent pas à incrémenter la classification, voire dans beaucoup de cas décroissent le taux de classification ou de reconnaissance [49, 50].

2.3. Domaines d’application

En vue de clarifier le concept de l’AL et les domaines d’application, nous allons présenter quelques applications dans l’état de l’art.

2.3.1. La télédétection

Lors de la capture de milliers ou de millions d'images sous différents angles, différentes résolutions de différents capteurs spectrométriques, il s’avère que les experts ne peuvent étiqueter ce nombre incommensurable d’images, alors il est plus que nécessaire de sélectionner des images clés et d’en étiqueter que quelques pixels significatifs pour former une image qui sera nommée "Ground Truth" (GT). Donc une image modèle, parmi des millions d'images disponibles. L’AL tend à solutionner cette problématique et différents travaux dans ce contexte ont vu le jour [8, 9, 17, 18, 51-54].

2.3.2. La parole

Le nombre incommensurable de segments de parole enregistré chaque minute dans les chaines de TV, radio, conférences,…. tend à rendre la tache de segmentation manuelle très fastidieuse. Ainsi pour quelques minutes de parole, plusieurs heures d’attention et d’expertise, de différents intervenants, doivent être réalisées pour avoir une segmentation fiable. Donc la segmentation de la parole est l’un des domaines de prédilection de l’apprentissage actif [3, 55, 56].

Il serait judicieux de disposer d'un ou de plusieurs algorithmes qui sélectionnent les segments de **phonèmes les plus incertains**, en vue de les étiqueter par le/les experts, et introduire cet étiquetage aux algorithmes automatiques pour finir la segmentation phonémiques en entités élémentaires plus certaines.

2.3.3. L'extraction d'information de documents

Décider si un article appartient à telle catégorie ou à telle classe, est un autre domaine d'application des nouvelles approches de l'apprentissage actif. Car la sélection de documents types n'est pas toujours la stratégie idéale, certains documents peuvent appartenir à différentes classes, par une appartenance croisée. Donc une stratégie de sélection intelligente augmenterait l'indépendance du système de sélection de la présence continue de l'expert, d'où l'apprentissage sur seulement les documents à forte incertitude de classement [5, 9, 16, 57, 58].

2.3.4. Le domaine médical

Les médecins font de l'apprentissage actif depuis la nuit des temps. Ceci se démontre par le fait que les médecins de moindre expérience peuvent intervenir sur les cas simples de fatigue, de petite fièvre, etc. Toutefois si les symptômes se compliquent, l'avis du médecin de rang supérieur se voit plus que stratégique. Donc les cas les plus délicats sont traités par un expert, selon un système de requête, qui est l'essence même de l'apprentissage actif, avec comme exemple remarquable dans le domaine de la segmentation d'images médicales, les travaux réalisés par Grady [59] ou d' autres chercheurs [60, 61].

2.4. Notions complémentaires

Il y'a différents scenarios à émettre les requêtes [4, 12, 62], et différentes stratégies à sélectionner les meilleures ou potentielles données (pixels) nécessaires à formuler ces requêtes. Dans ce qui suit, nous adopterons l'approche de sélection d'un ensemble de pixels bien défini, contenant une partie étiqueté «$\boldsymbol{L}$ » et une autre partie non-étiqueté «$\boldsymbol{U}$», les pixels non-étiquetés sont en nombre nettement inférieur aux pixels étiquetés.

La figure 2.1, présente un modèle graphique des phases de l'apprentissage actif [7], ou l'intervention de l'expert tend à être minimale par le fait que l'algorithme d'apprentissage AL sélectionne seulement les pixels utiles à la

fonction d'apprentissage et ayant un degré d'incertitude élevé, c.à.d. les pixels que l'algorithme croit ne pas pourvoir bien classer.

Au fait, lorsque l'expression «l'algorithme sélectionne » est utilisée, il est sujet d'une décision prise sur l'appartenance des données, selon la contrainte de minimisation d'une fonction objective ou fonctionnelle sujette à un critère de classement.

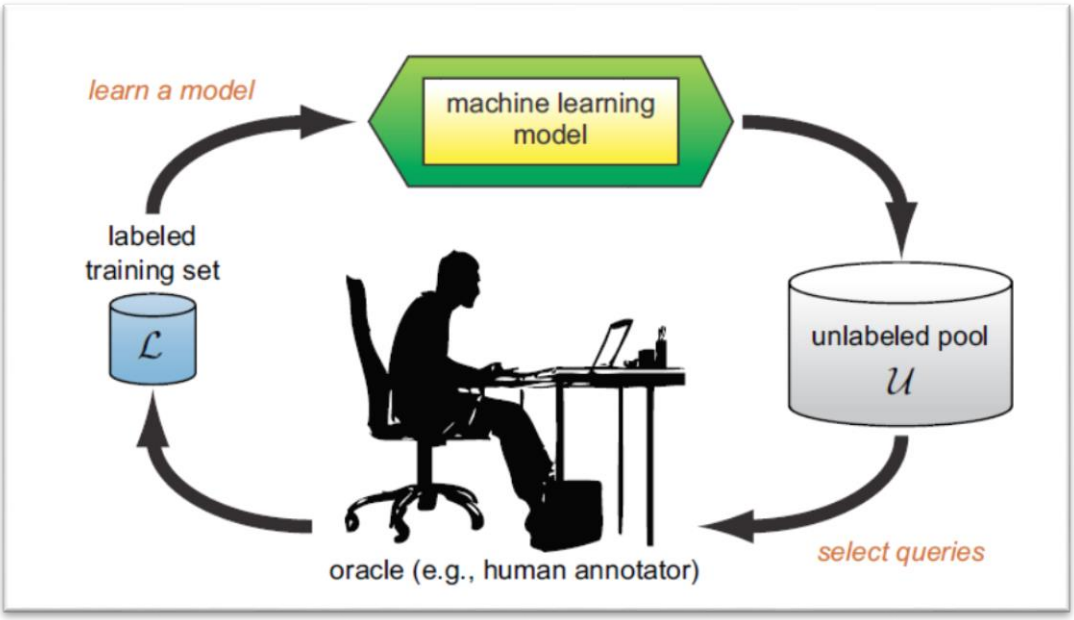


Figure 2.1 : Modèle d'apprentissage actif

La figure 2.2 représente deux modèles d'apprentissage (passif et actif), et l'impact de l'actif sur l'augmentation de la classification [9].

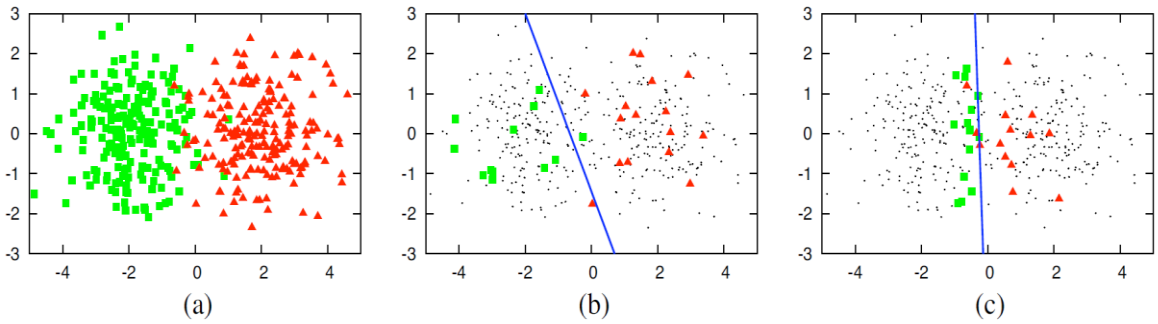


Figure 2.2 : Un exemple illustratif sur le jeu de données «Toy data», issu équitablement de deux distributions gaussiennes

La figure 2.2(a) représente l'ensemble des pixels, le figure 2.2(b), représente la classification par un modèle de régression logistique entrainé avec 30 instances, la ligne inclinée représente la bordure de décision avec un taux de classification de 70%, la figure 2.2(c) illustre les résultats de la classification active avec 30 instances "dument sélectionnées" atteignant un taux de classification de 90%, la ligne de décision est plus pointue.

2.5. Cœur de l'apprentissage actif (AL)

Dans cette section, nous allons donner un aperçu sur les principaux éléments de l'AL à usage généralisé, en décrivant trois concepts fondamentaux relatifs à l'AL. Les notions d'optimisation de l'espace de sortie et les concepts heuristiques actives d'apprentissage, ainsi que l'incertitude et la diversité. Le but de cette description est de décrire les problèmes spécifiques à la télédétection relatifs à l'AL, plutôt que de présenter et comparer les bases spécifiques de ces heuristiques [7].

2.6. Conception du réseau optimal dans l'espace de sortie

Trouver un ensemble d'apprentissage déterminant pour la classification d''images (ou l'extraction de paramètres biophysiques) peut être considéré comme la tâche stratégique de conception d'un réseau de surveillance optimal [7]. Étant donné un réseau, (par exemple : les pixels d'apprentissage), ajouter de nouvelles mesures, afin d'améliorer la performance actuelle de l'algorithme, dans le cadre de la télédétection se réduit à la fastidieuse tâche de trouver de nouveaux paramètres, où la sortie peut être mesurée soit par un utilisateur ou par un dispositif de détection.

En géostatistique, un grand nombre de travaux traite des méthodes de remplissage de l'espace (Space Filling Methods) [63], visant à remplir l'espace d'entrée, souvent caractérisé par la localisation spatiale des pixels d'apprentissage.

En télédétection, l'accentuation a souvent concernée les méthodes systématiques, par lesquelles les pixels sont acquis sur une grille régulière ou par des méthodes stratifiées. Le nombre de pixels est équilibré selon une estimation de l'abondance des classes présentes dans l'image, ou tout autre paramètre

pertinent ayant une plus grande variabilité, et qui correspond à un plus grand nombre de pixels requis. Cette dernière stratégie, en particulier, a facilité l'amélioration des résultats lorsque les données ont été obtenues par échantillonnage aléatoire "Random Sampling" (RS)[64], mais une connaissance préalable du paramètre pertinent sur lequel fonder la stratification, est nécessaire.

Le problème de l'autocorrélation spatiale, entre les pixels, est souvent ignoré [7]. Cette stratégie correspond à une phase d'exploration, où il n'y a aucune tentative de contrôler le pouvoir prédictif du modèle directement, ce qui correspondrait à une phase d'exploitation. L'apprentissage actif cherche à combler cette lacune. Au lieu d'optimiser la couverture de l'espace d'entrée, l'AL considère l'espace de sortie, c'est à dire, les prédictions du modèle, et classe les pixels potentiels d'apprentissage en fonction de la confidence de prédiction du modèle actuel.

L'AL répond à la question "Quels pixels doivent être ajoutés à l'ensemble d'apprentissage pour améliorer la généralisation d'un modèle donné ?

L'ensemble d'apprentissage défini par l'AL est donc spécifique au modèle et ne doit pas être désigné pour une exploration générale.

Les pixels qui devraient avoir le plus d'impact sur le modèle actuel sont sélectionnés, et les pixels pour lesquels le modèle fournit une prédiction avec **une grande certitude sont ignorés**.

2.7. Fonctionnement de l'AL

En suivant la terminologie de Li et Sethi [65], un algorithme d'AL se résume au quintuple (C, L, S, U, Q), avec :

- C : Classificateur.
- L : Ensemble de données d'apprentissage étiquetées.
- S : Utilisateur cherchant à étiqueter les pixels de l'ensemble non étiqueté U.
- Q : Critère de classement des pixels de U, (parfois heuristique).

Dans le cadre de la télédétection, les ensembles L et U sont composées de pixels d-dimensionnel, d étant le nombre total de bandes spectrales.

Pour les pixels de l'ensemble L, les étiquettes sont connues :

$$L = \{x_i, y_j\}_{i=1}^{l} \quad 2.1$$

Tandis que pour les pixels de U, seul le vecteur d'entrée est connu :

$$U = \{x_i, ?\}_{i=1}^{u} \quad 2.2$$

Les deux ensembles couvrent les pixels de l'image $n = u + l$.

Contrairement aux méthodes systématiques ou de stratification, C et l'utilisateur S interagissent en permanence lors de la construction du modèle de l'AL :

- C prédit les classes de sortie de U.
- S approvisionne C par les pixels selon le critère de classement Q.

Pour cette raison, le processus d'apprentissage actif est naturellement itératif, dans ce sens, nous pouvons formuler qu'a un certain moment ou état du système ϵ, et pour une certain ensemble L^{ϵ}, la réponse de C^{ϵ}, est différente et produit un classement des pixels candidats. En se basant sur ce classement, S approvisionne C avec les pixels d'apprentissage et les nouveaux pixels passent de U^{ϵ} à L^{ϵ}, créant ainsi les nouveaux $U^{\epsilon+1}$ à $L^{\epsilon+1}$, la Figure 2.3 illustre ce concept de classification. 7

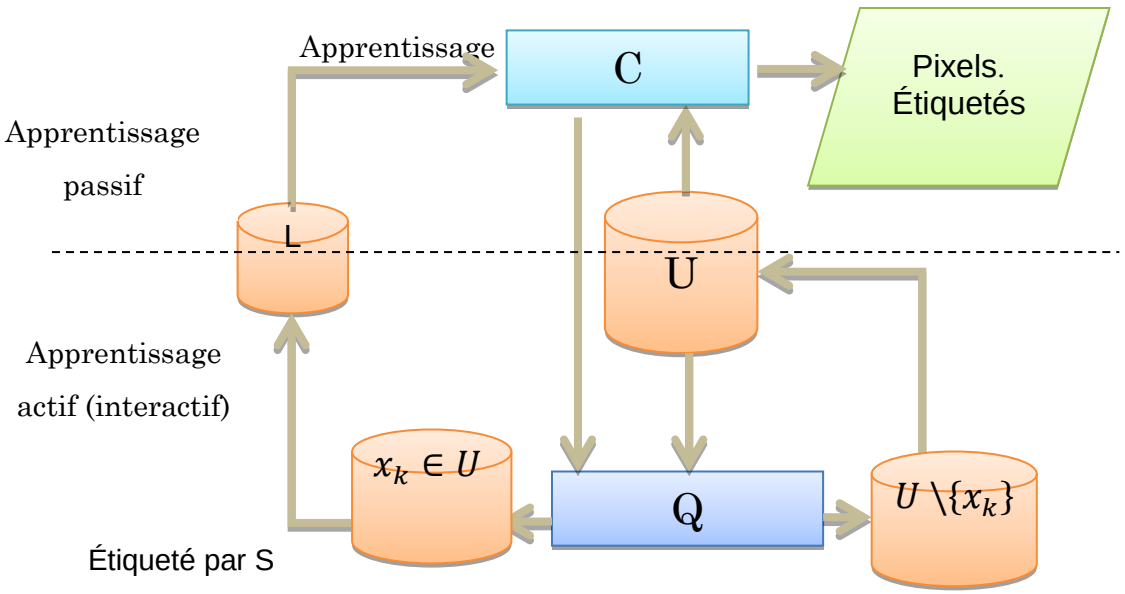


Figure 2.3 : Concept de l'apprentissage actif

2.8. Le critère d'évaluation Q

Le critère d'évaluation Q heuristique différencie l'AL des stratégies d'échantillonnage classiques [65]:

- Ce critère Q est basé sur la sortie du classificateur courant C^{ϵ} et ne concerne qu'indirectement l'espace d'entrée (sauf si l'heuristique est conçu pour renforcer cette relation, comme dans les stratégies multi-vues et spectrales/spatiales).
- Ce critère est destiné à fournir des informations pour le classement des pixels candidats par rapport à leur contribution potentielle, sur la base du classificateur courant C^{ϵ}. Cette valeur peut être évaluée en fonction de l'incertitude/la confidence d'un pixel et de sa diversité.

2.8.1. Incertitude des étiquettes des pixels

Les pixels non étiquetés ne sont pas égaux dans un contexte "informatif" pour le classificateur courant C^{ϵ}.c'est à dire qu'ils ne portent pas la même contribution dans un espace discriminatoire.

Considérons le cas d'un classificateur SVM à titre d'exemple [16, 65], seuls les pixels qui ont une chance de devenir des vecteurs de support sont instructifs, parce que les autres pixels seraient écartés, même s'ils sont ajoutés à l'ensemble d'apprentissage $L^{\epsilon+1}$. Dans ce sens, un bon critère Q doit être capable d'affecter des rangs élevés aux pixels qui ont une grande chance de devenir des vecteurs supports. Ceci est fortement lié à la notion d'incertitude du pixel :

> Un pixel qui peut être manipulé/classé correctement par le modèle actuel, n'a pratiquement aucune chance de devenir un vecteur de support, tandis qu'un pixel situé dans la marge, c'est-à-dire à proximité des vecteurs de support, est incertain **et donc très instructif**.

C'est une différence majeure par rapport aux approches stratifiées, où le choix des pixels est équilibré par une mesure de variabilité, et donc d'une manière sous-optimale pour le modèle.

Plusieurs heuristiques sont basées sur ce concept :

L'échantillonnage marginale (Marginal Sampling) (MS) [62]: qui minimise la distance à l'hyperplan le plus proche. Les méthodes d'échantillonnage de rupture des liens (Breaking Ties)(BT) [12] et le niveau multi-classe d'incertitude (multiclass-level uncertainty) [66] considèrent la confidence des deux classes les plus probables. Cette famille d'heuristiques est le plus étudiée dans la communauté de l'apprentissage actif (voir, par exemple, les travaux de [8] et [67]). Les approches basées sur un comité de modèles sont aussi considérées, ou les modèles sont entrainés à partir de sous-ensembles de L^{ϵ}, selon le critère de l'entropie sur les requêtes du "Bagging", (Entropy Query-by-Bagging-EQB) [7]) ou sur des sous-ensembles "d-dimensionnels" de l'espace des attributs, (multi-vues). Les figures 2.4 et 2.5 illustrent la différence entre ces deux derniers concepts.

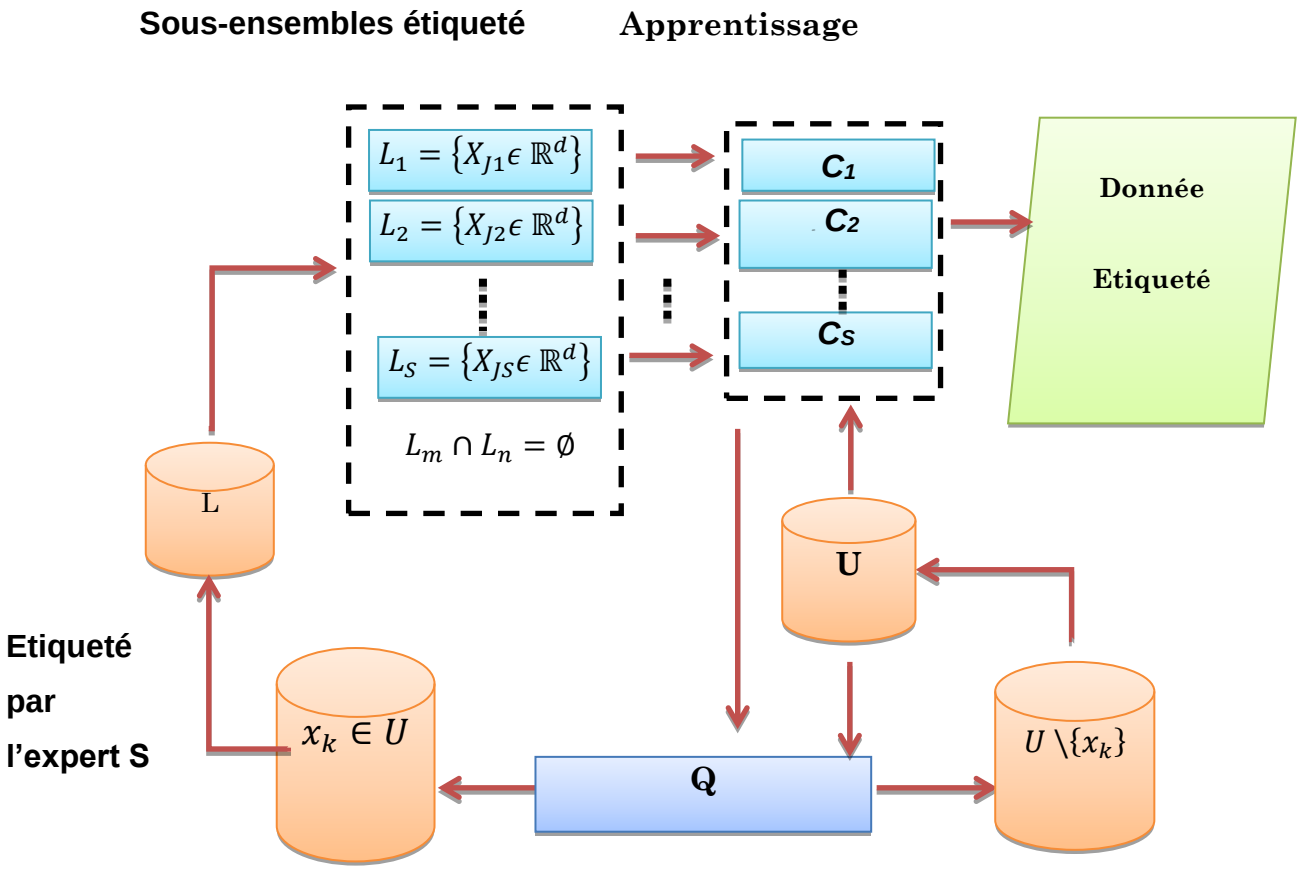


Figure 2.4 : Requêtes par comité dans l'apprentissage actif

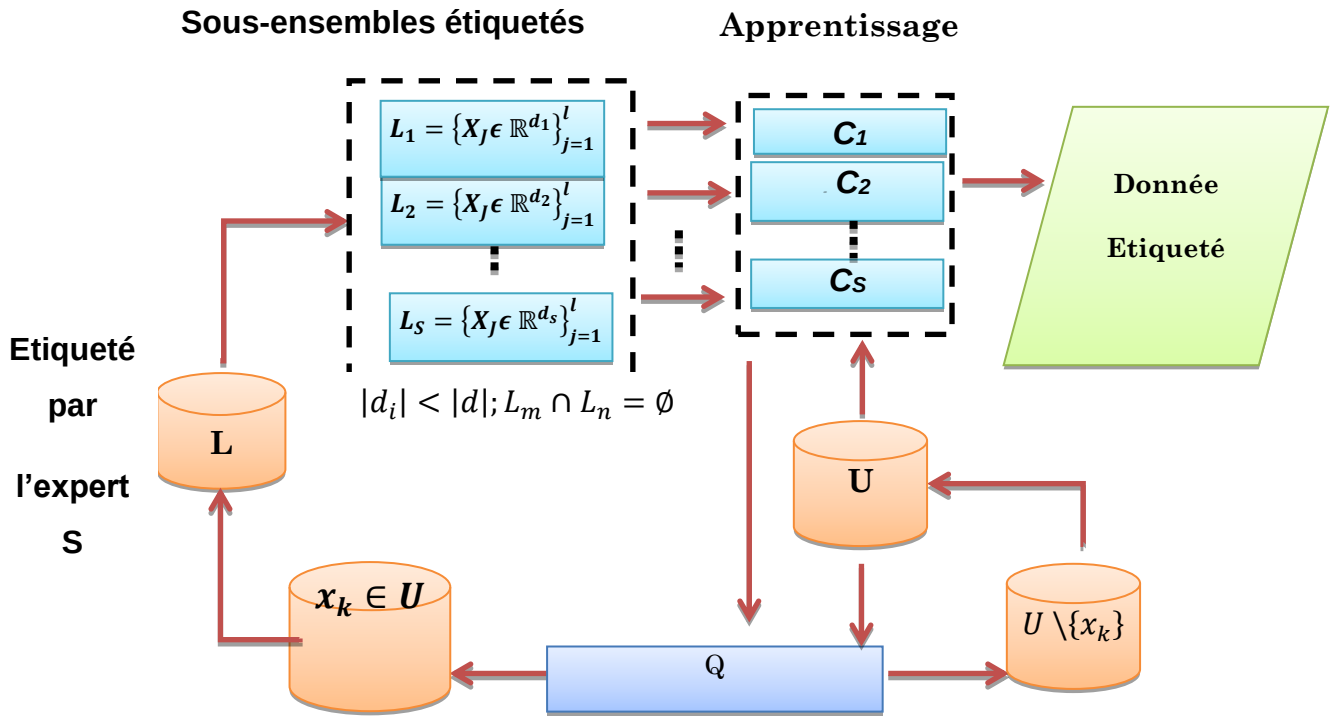


Figure 2.5 : Application de la méthode des Multi-vues en apprentissage actif

Lors de la sélection d'un seul pixel par itération, ces méthodes heuristiques renvoient le pixel le plus incertain (ou le moins informatif) dans U^{ϵ}.

Les méthodes heuristiques basées sur les comités ont l'avantage d'être indépendantes du système de classification [7]. La stratégie BT peut également être appliqué à tous les modèles délivrant les probabilités à posteriori, où l'heuristique est utilisée avec l'analyse discriminante linéaire [66], (les auteurs l'ont utilisée avec une régression logistique multinomiale), de même que l'approche proposée dans [8], où la divergence sur les probabilités à posteriori est utilisée, soit avec le maximum de vraisemblance, soit avec le classificateur hiérarchique binaire.

2.8.2. Diversité dans l'apprentissage actif

Sélectionner un seul pixel par itération n'est pas un objectif ambitieux pour l'AL, car le C^{ϵ} doit être ré-entrainer avec l'ensemble $L^{\epsilon+1}$, En outre, les pixels requis par seulement leur incertitude peuvent être redondant par rapport à l'autre. Par conséquent, de nombreuses études ont été consacrées à la question de diversité de la donnée ou du pixel [67].

Si plusieurs pixels sont sélectionnés en une seule fois, cette sélection est appelé lot de sélection ou "Batch Selection", l'ensemble des pixels doit être aussi diversifié que possible, pour éviter la redondance. Dans ce contexte, le lot le plus efficace de pixels est inclus dans $L^{\epsilon+1}$. Le lot résultant contient alors divers pixels incertains pour le C actuel.

La diversité est aussi un facteur important pour les stratégies de l'AL qui impliquent les multi-entrées ou multi-vues, voir Figures 2.4 et 2.5, dont le rôle est d'explorer les différences en matière d'information dans l'espace des entrées, par rapport au critère choisi et pour exploiter tout potentiel de calcul parallèle.

2.9. Conclusion

Les données issues de la télédétection sont intrinsèquement variées en termes de leur origine multi source, de leur contenu spatio-temporel, ainsi que des nouveaux attributs spectraux additionnels [13, 17]. Donc, l'exploration de ces données se diverge sur plusieurs fronts et nécessite une stratégie de classification efficace, point que l'on verra dans le chapitre 5.

Dans le prochain chapitre 4, nous allons développer un autre outil indispensable à notre contexte de travail, qui est la théorie des graphes et son application aux images, sous la considération qu'une image peut être représentée par une matrice multidimensionnelle.

CHAPITRE 3

3. THEORIE DES GRAPHES

Ce qui s'apprend sans peine ne vaut rien et ne demeure pas.
René Barjavel

3.1. Introduction

L'idée intuitive de l'analyse de données s'appuie sur le principe : " ceux qui se ressemblent s'assemblent". En faisant une analogie entre similarité des données et proximité des points dans l'espace des attributs, on peut chercher dans la structure des données, telles qu'elles se présentent, des groupements naturels selon l'idée : il est très probable que, dans l'espace d'attributs, des points proches représentent des données d'un même groupe et que des points lointains représentent des données qui appartiennent à des groupes différents [68].

Dans ce chapitre, nous allons présenter un des outils fondamentaux utilisé en classification semi-supervisée, qui est le mappage d'une image en graphe ou réseau, et présenter l'intérêt que suscite cette application dans le développement d'applications de classification, soit dans le domaine médical [69-72] ou de la télédétection[73-75] ou en diagnostic [76, 77],…

Nous commencerons par définir les notions de base telles que : la notion de distance, de similarité, de structure de graphes et quelques propriétés mathématiques, puis nous aborderons différents algorithmes de l'état de l'art sur les méthodes de segmentation de graphes et la classification inhérente dans le domaine du semi-supervisé, et nous illustrerons quelques exemples d'applications.

3.2. Notions de distance

La similarité a pour objet de quantifier la ressemblance entre deux données. En faisant l'analogie avec la proximité, elle est basée dans la plupart des cas sur la notion mathématique de distance. En effet, il est admis que deux points séparés, dans l'espace des attributs, par une grande distance correspondent à deux données non similaires, tandis que deux points proches (au sens de cette distance) correspondent à deux données qui sont similaires. Pour cette raison, nous allons tout d'abord introduire la notion de distance pour ensuite définir la fonction de similarité [68].

La fonction de distance utilisée pour définir la distance d_{ij} entre les points représentant les données (x_i, x_j) est une application de $\mathbb{R}^{\boldsymbol{D}} \times \mathbb{R}^{\boldsymbol{D}}$ dans $\mathbb{R}^{+}$.

- $\boldsymbol{D}$ étant la dimension des attributs x_i, x_j.

La distance doit respecter les propriétés suivantes :

– Non négativité : $d_{ij} \geq 0$

– Symétrie : $d_{ij} = d_{ji}$

– Séparation : $d_{ij} = 0 \Rightarrow i = j$

– Minimalité : $d_{ii} = 0$

– Inégalité triangulaire : $d_{ij} \leq d_{ik} + d_{kj}$

Plusieurs fonctions de distance ont été définies dans la littérature, nous nous intéressons dans notre contexte aux distances symétriques. Ces fonctions ont une valeur proche de zéro pour un couple de points proches dans l'espace des attributs et une valeur qui tend vers l'infini pour un couple de points éloignés dans l'espace des attributs. Les distances les plus courantes sont :

3.2.1. Distance Euclidienne

La distance Euclidienne, qui est la distance la plus utilisée, est définie comme suit :

$$\delta_{ij} = \left[\sum_{r=1}^{d}\left(x_{ir} - x_{jr}\right)^2\right]^{1/2} \quad 3.1$$

3.2.2. Distance de Manhattan

La distance de Manhattan est définie par :

$$\delta_{ij} = \sum_{r=1}^{d}\left|x_{ij} - x_{jr}\right| \quad 3.2$$

3.2.3. Distance de Minkowski

La distance de Minkowski est une généralisation de la distance Euclidienne et de la distance de Manhattan. Elle est définie par :

$$\delta_{ij} = \left[\sum_{r=1}^{d}\left(x_{ir} - x_{jr}\right)^q\right]^{1/q} \quad 3.3$$

- où q est un entier positif non nul.

Une forme plus générale est la distance pondérée :

$$\delta_{ij} = \left[\sum_{r=1}^{d} a_r\left(x_{ir} - x_{jr}\right)^q\right]^{1/q} \quad 3.4$$

- a_r étant un coefficient de pondération associé à chaque attribut.

A partir de la matrice des données $\boldsymbol{X}$, on construit la matrice des distances $\Delta(n \times n)$ de terme général δ_{ij} caractérisant la distance séparant chaque paire de points (x_i, x_j):

$$\Delta = \begin{bmatrix} 0 & \dots & \delta_{1i} & \dots & \delta_{1n} \\ \dots & \dots & \dots & \dots & \dots \\ \delta_{i1} & \dots & 0 & \dots & \delta_{in} \\ \dots & \dots & \dots & \dots & \dots \\ \delta_{n1} & \dots & \delta_{ni} & \dots & 0 \end{bmatrix} \quad 3.5$$

3.3. Notion de similarité

Il est nécessaire d'évaluer les ressemblances ou les dissemblances qui existent au sein des données d'une image par exemple. Le terme de "fonction de similarité" ou plus simplement celui de similarité est alors utilisé.

La similarité notée w_{ij} exprime la ressemblance entre les données (x_i, x_j). C'est une application de $\mathbb{R}^{\boldsymbol{D}} \times \mathbb{R}^{\boldsymbol{D}}$dans $[0,1]$ telle que :

– Propriété de Symétrie : w_{ij}=w_{ji}

– Normalisation : $w_{ij} \in [0,1]$, avec $w_{ij} \geq w_{ji}$

Une similarité proche de 1 indique que les données sont similaires, tandis qu'une valeur proche de 0 indiquent qu'elles sont différentes.

Les fonctions de similarité peuvent être exprimées sous des formes multiples (cosinus, coefficient de corrélation de Pearson, Gaussienne, voire floue....) [68].

Les fonctions de similarité les plus courantes sont, la fonction cosinus et la fonction Gaussienne.

3.3.1. La fonction cosinus

La fonction cosinus est surtout utilisée dans l'analyse des documents[78, 79]. Elle est définie comme suit :

$$w_{ij} = \left|cos\left(x_i, x_j\right)\right| = \frac{x_i^T x_j}{\|x_i\|\|x_j\|} \qquad 3.6$$

Deux données sont d'autant plus similaires que leurs points associés sont placés sur une même droite passant par l'origine de l'espace des attributs. Cette fonction de similarité est donc sensible à la direction des données projetées dans l'espace des attributs. Son principal inconvénient réside dans l'impossibilité de différencier les données qui ont des formes ou des directions similaires et qui sont très éloignées les unes des autres. Cette mesure n'est pas couramment utilisée en analyse de données, où les différences entre données sont plus liées à leur amplitude qu'à leur direction dans l'espace des attributs.

3.3.2. La fonction Gaussienne

La fonction Gaussienne est basée sur la distance Euclidienne[80, 81]. En général, la fonction de similarité doit prendre en compte les relations de voisinage entre les données. C'est pour cette raison que la fonction Gaussienne basée sur la distance Euclidienne entre points est souvent utilisée. Elle est définie par :

$$w_{ij} = exp\left(-\frac{1}{2\sigma^2}\delta_{ij}^2\right) \quad 3.7$$

- où δ_{ij} est la distance Euclidienne entre les points associés aux données x_i et x_j définies dans l'équation 3.1.

Le paramètre de dispersion σ doit être choisi de telle sorte qu'il soit adapté à la dispersion locale des données disponibles [68]. En effet, quand la distance Euclidienne séparant x_i et x_i est inférieure à $\sqrt{2}\sigma$, le terme au sein de l'exponentielle est inférieur à -1. La mesure de similarité entre ces deux points s'approche alors de la valeur 1. Par contre, quand la distance séparant x_i et x_i est nettement supérieure à $\sqrt{2}\sigma$, le terme au sein de l'exponentielle est supérieur à -1. La mesure de similarité est alors proche de 0.

Cette fonction de similarité prend ses valeurs dans l'intervalle continu $[0,1]$. La valeur 0 signifie une similarité nulle entre données (x_i, x_j) associées à des points éloignés dans l'espace des attributs (δ_{ij} tend vers $+\infty$), tandis que la valeur 1 correspond à une grande similarité entre données associées à des points proches dans l'espace des attributs ($\delta_{ij} = 0$).

Plus la distance, séparant deux points dans l'espace des attributs, est grande, plus la similarité entre les données associées est petite. A partir de la matrice de distance Δ, on construit alors la matrice de similarité $W(n \times n)$ de terme général w_{ij} caractérisant la similarité entre chaque paire de données(x_i, x_j) à partir de leurs représentations dans l'espace des attributs :

$$W = \begin{bmatrix} 1 & \dots & w_{1i} & \dots & w_{1n} \\ \dots & \dots & \dots & \dots & \dots \\ w_{i1} & \dots & 1 & \dots & w_{in} \\ \dots & \dots & \dots & \dots & \dots \\ w_{n1} & \dots & w_{ni} & \dots & 1 \end{bmatrix} \quad 3.8$$

3.4. Exemples pratiques

Voici quelques exemples pratiques qui illustrent les notions de données, attributs ainsi que le calcul de distance et de similarité entre ces données.

3.4.1. Matrice de données [68]

Soit un ensemble de 4 données représenté par $X = \{x1, x2, x3, x4\}$, caractérisé par 2 attributs f_1 et f_2 (*nombre d'instances* = 4, *dimension* = 2). Ces données sont définies par la matrice $\boldsymbol{X}$ comme suit :

$$\boldsymbol{X} = \begin{bmatrix} 4 & 4 \\ -0.5 & -0.5 \\ 4 & 2.5 \\ -0.5 & 0.5 \end{bmatrix} \qquad 3.9$$

La figure 3.1 montre la représentation de ces données dans l'espace $\mathbb{R}^2$.

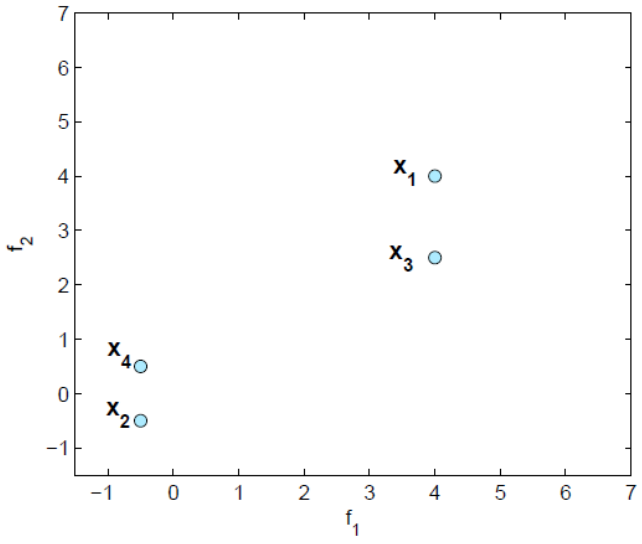


Figure 3.1 : Représentation des données dans l'espace $\mathbb{R}^2$.

3.4.2. Matrice de distances

La matrice de distance $\Delta\ (n \times n)$ calculée entre ces données dans l'espace $\mathbb{R}^2$en utilisant la distance Euclidienne de l'équation (3.1) est :

$$\Delta = \begin{bmatrix} 0 & 6.364 & 0.7548 & 0.0172 \\ 6.364 & 0 & 0.5802 & 0.8825 \\ 1.5 & 5.4083 & 0 & 4.9244 \\ 5.7009 & 1 & 4.9244 & 0 \end{bmatrix} \qquad 3.10$$

Nous pouvons ainsi remarquer que les points x_2 et x_4 les plus proches dans l'espace des attributs sont séparés par la distance minimale ($\delta_{24} = \delta_{42} = 1$), tandis que les points x_1 et x_2, les plus éloignés dans l'espace des attributs, sont séparés par la distance maximale ($\delta_{12} = \delta_{21} = 6.364$).

Pour illustrer l'influence du choix de la fonction de similarité, nous allons calculer la similarité de deux façons différentes : tout d'abord, en utilisant la fonction cosinus, puis en utilisant la fonction Gaussienne basée sur la distance Euclidienne.

3.4.3. Matrice de similarité cosinus

En utilisant la fonction cosinus de l'équation (3.6), la matrice de similarité W est alors :

$$w = \begin{bmatrix} 1 & 1 & 0.9744 & 0 \\ 1 & 1 & 0.9744 & 0 \\ 0.9744 & 0.9744 & 1 & 0.2249 \\ 0 & 0 & 0.2249 & 1 \end{bmatrix} \qquad 3.11$$

Nous pouvons remarquer que les points x_1 et x_2 séparés par la distance, la plus grande dans l'espace des attributs ($\delta_{12} = \delta_{21} = 6.364$), correspondent à la paire de données la plus similaire ($w_{12} = w_{21} = 1$), puisque ces deux points appartiennent à une même droite passant par l'origine.

L'angle x_1Ox_2 étant égal à 0 degré, son cosinus est donc égal à 1. Par contre, les points $x2$ et $x4$ séparés par la distance la plus faible dans l'espace des attributs ($\delta_{24} = \delta_{42} = 1$) correspondent à la paire de données la moins similaire ($w_{24} = w_{42} = 0$).

Cet exemple illustre que la fonction de similarité basée sur le cosinus n'est pas sensible aux distances séparant les points dans l'espace des attributs.

3.4.4. Matrice de similarité Gaussienne

En utilisant la fonction de similarité Gaussienne de l'équation (3.7) et en fixant σ à 2, la matrice de similarité W est alors :

$$W = \begin{bmatrix} 1 & 0.0063 & 0.7548 & 0.0172 \\ 0.0063 & 1 & 0.5802 & 0.8825 \\ 0.7548 & 0.0258 & 1 & 0.0483 \\ 0.0172 & 0.8825 & 0.0483 & 1 \end{bmatrix} \quad 3.12$$

Nous pouvons remarquer que les points x_1 et x_2 séparés par la distance la plus grande dans l'espace des attributs, correspondent à la paire de données la moins similaire ($w_{12} = w_{21} = 0.0063$), tandis que les points x_2 et x_4 séparés par la distance la plus faible dans l'espace des attributs, correspondent à la paire de données la plus similaire ($w_{24} = w_{42} = 0.8825$).

Cet exemple met en évidence que[68] :

- La similarité basée sur la fonction Gaussienne dépend de la distance séparant les points associés aux données dans l'espace des attributs.
- La similarité basée sur la distance entre les points dans l'espace des attributs semble donc être mieux adaptée au regroupement des données.

Après avoir illustré, sur un exemple de données, la notion de similarité, nous allons montrer comment représenter ces données sous la forme d'un graphe.

3.5. Notion de Graphe

La théorie des graphes est une théorie qui s'est vue développée depuis des décennies[82], elle touche les domaines les plus complexes, tels que la téléphonie, les réseaux de communication, les réseaux électriques, les systèmes de contrôle distribué et récemment sur les réseaux sociaux [83] tels que Facebook et Twitter [84].

Cette théorie a l'avantage de présenter une topologie globale des cas étudiés, et de pourvoir faire des propagations à travers ce réseau, selon des méthodes mathématiques, en vue de classifier ou de faire ressembler des types à attributs incomplets dans des sphères à priori définies,…

Le concept de graphe est utilisé comme un modèle de représentation des données, dès que celles-ci sont «intrinsèquement» liées entre elles. Il permet d'exprimer les relations et de révéler les dépendances entre ces données [68].

L'analyse de ces graphes a pour objectif de concevoir des représentations synthétiques qui puissent exprimer l'interaction entre les différentes données représentées.

Un graphe (G) est défini comme suit : [85]

- Un ensemble $V = V(G)$ tels que ses éléments sont appelés points, vertex ou nœuds de G.
- Un ensemble $E = E(G)$ de paires de nœuds distinctes non-ordonnées appelées arêtes de G.

Le graphe est noté $G(V, E)$, lorsque l'on relie toutes les parties constructives du graphe, comme illustré dans les figures 3.2 et 3.3.

Les nœuds A et B sont dits adjacents ou voisins s'il y'a un arc $e_1 = \{A, B\}$, qui les relie immédiatement. Les nœuds 7A et B sont dits nœuds terminaux de e_1. L'arc e_1 est dit incident sur chaque nœud A et B s'il a pour nœud terminaux A et B.

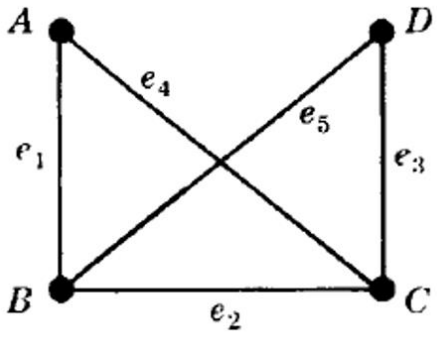


Figure 3.2 : Exemple d'un graphe de 4 nœuds (A, B, C, D) et 5 arcs (e_1 … e_5))

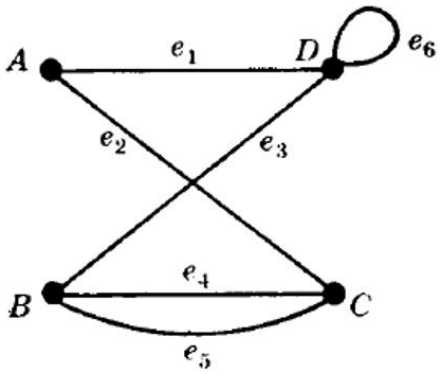


Figure 3.3 : Exemple d'un multi-graphe de 4 nœuds (A, B, C, D) et 6 arcs (e_1 … e_5))

3.5.1. Degré d'un nœud ou vertex

Le degré d'un sommet d'un graphe G, écrit deg(sommet), est égal au nombre d'arêtes de G qui se terminent en ce sommet, c'est-à-dire, qui sont incidents sur ce sommet. Etant donné que chaque arc est compté deux fois dans le comptage des degrés des sommets de G.

Théorème : Le somme des degrés des arcs de G est égal au nombre d'arcs de G.

Dans le cas de la Figure 3.1, $deg(A) = 2$; $deg(B) = 3$; $deg(C) = 3$; $deg(D) = 2$

La somme des degrés est égale à dix, ce qui exprime le double des arcs de G.

3.5.2. Sous-Graphes

Etant donné un graphe $G(V, E)$ et un graphe $H(V', E')$, H est appelé un sous-graphe de G, si les sommets et les arêtes de H sont contenus dans les sommets et les arêtes de G, c'est-à-dire, si $V' \subseteq V$ et $E' \subseteq E$. En particulier :

(i) un sous-graphe $H(V', E')$ de $G\ (V, E)$ est appelé sous-graphe induit par les sommets V' si l'ensemble des arêtes de E' contient toutes les arêtes de G dont les extrémités appartiennent aux nœuds de H.

(ii) Si V est un sommet de G, $G - V$ est le sous-graphe de G obtenu par suppression de G du sommet V et la suppression de toutes les arêtes de G qui contiennent V.

(iii) Si e est une arête dans G, alors $G - e$ est le sous-graphe de G obtenu en supprimant simplement l'arête e de G.

3.5.3. Graphes iso-morphiques

Les graphes $G(V, E)$ et $G^*(V*, E*)$ sont dits isomorphes, s'il existe une correspondance « un à un » → $f: V \to V*$, tel que $\{u, v\}$ est une arête de G si et seulement si $\{f(u), f(v)\}$ est une arête de G^*.

La Figure 3-4 donne dix graphiques représentant quelques lettres de l'alphabet. Nous pouvons notre que A et R sont des graphes isomorphes. De plus,

F et T sont des graphes isomorphes, K et X sont des graphes isomorphes et M, S, V et Z sont des graphes isomorphes.

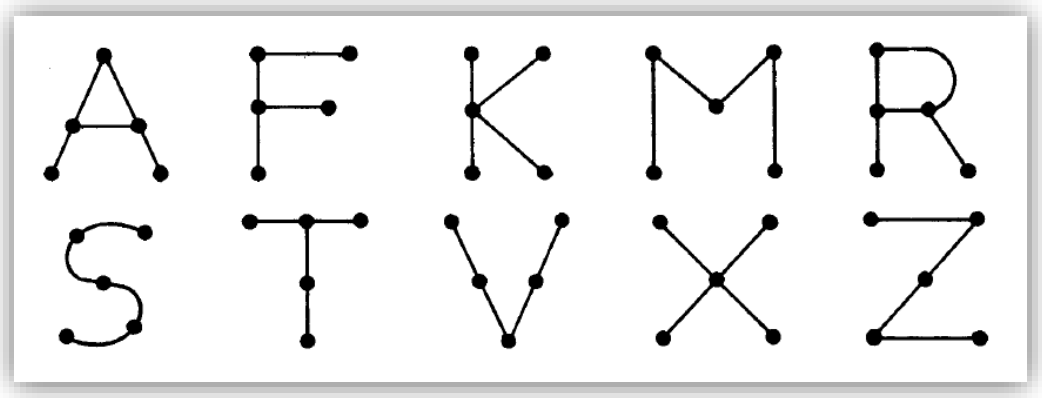
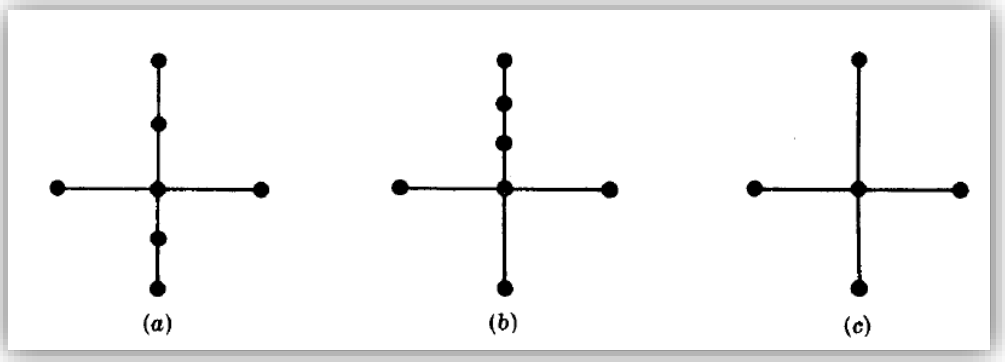


Figure 3.4. : Graphes isomorphes

3.5.4. Graphe homéo-morphiques

Etant donné un graphe G, nous pouvons obtenir un nouveau graphe en divisant G avec des sommets supplémentaires. Les deux graphes G et G^* sont dits homéo-morphique, s'ils sont obtenus à partir du même graphe ou d'un graphe isomorphe. Les graphes (a) et (b) de la figure 3.5 ne sont pas isomorphes, mais ils sont homéomorphes, car ils peuvent être obtenus à partir du graphe de (c) en ajoutant des sommets additionnels.



\

Figure 3.5. Graphes homomorphes

3.5.5. Graphe et connectivite

Un chemin dans un multi-graphe G se compose d'une séquence alternée de sommets et d'arêtes de la forme : $\boldsymbol{v_0, e_1, v_1, e_2, v_2, \ldots, e_{n-1}, v_{n-1}, e_n, v_n}$

- Où chaque arête e_i contient les sommets v_{i-1}et v_i (qui apparaissent sur les côtés de e_i dans la séquence).

Le nombre « n » d'arêtes est appelé la longueur du chemin. Quand il n'y a pas d'ambiguïté, le chemin est notée par sa séquence de sommets $(v_0, v_1, \ldots, v_n)$. Le chemin est dit fermé si $v_0 = v_n$.

Un chemin simple est un chemin dans lequel tous les nœuds sont distincts.

Un cycle est un chemin fermé de longueur 3 ou plus dans lequel tous les sommets sont distincts sauf $v_0 = v_n$. Un cycle de longueur k est appelé un cycle-k.

Considérons le graphe de la figure 3.6, avec les séquences suivantes :

$\alpha = (P_4, P_1, P_2, P_5, P_1, P_2, P_3, P_6)$

$\beta = (P_4, P_1, P_5, P_2, P_6)$

$\gamma = (P_4, P_1, P_5, P_2, P_3, P_5, P_6)$

$\delta = (P_4, P_1, P_5, P_3, P_6)$

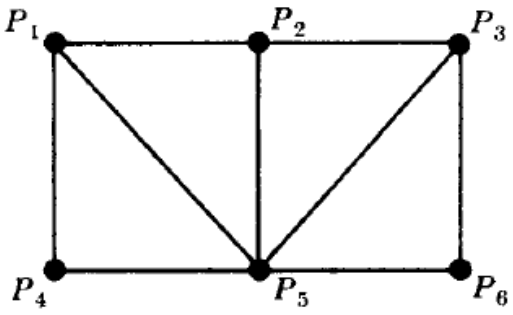


Figure 3.6 : Graphe connecté avec différents chemins

La séquence α est un chemin de P_4 à P_6 ; mais ce n'est pas une piste car l'arc $(P_1\ P_2)$ est utilisé deux fois.

La séquence β n'est pas un chemin, car il n'y a pas d'arc entre (P_2, P_6).

La séquence γ est une piste, car aucun nœud n'est est utilisé deux fois ; mais c'est un simple chemin car le sommet P_5 est utilisé deux fois.

La séquence δ est un simple chemin de P_4 à P_6; mais il n'est pas le chemin le plus court (par rapport à la longueur) de P_4 à P_6. Le plus court chemin de P_4 à P_6 est le chemin simple (P_4, P_5, P_6) qui a une longueur de 2.

La figure 3.7 illustre un graphe non-connecté, avec ses différents chemins.

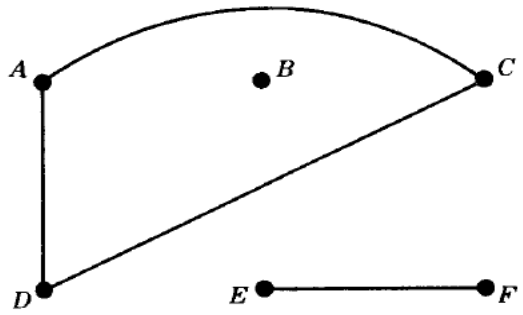


Figure 3.7 : Graphe non-connecté, avec différents chemins

3.5.6. Connectivite et graphs connexes

Un graphe G est relié ou connecté, s'il existe un chemin entre deux, **quelconques**, de ses sommets. Le graphe de la figure 3.6 est connecté, mais le graphe de la figure 3.7 n'est pas connecté, car, par exemple, il n'y a pas de chemin entre les sommets D et E.

Etant donné G un graphe. Un sous-graphe H de G est dit composant connexe de G si H n'est pas contenu dans un sous-graphe connexe plus grand que G. Il est intuitivement évident que tout graphe G peut être partitionné en ses composantes connexes. Par exemple, le graphe G de la figure 3.7 a trois composantes connexes, les sous-graphes induits par les ensembles de sommets {A, C, D}, {E, F} et {B}.

Le sommet B de figure 3.7 est appelé sommet isolé puisque B n'appartient à aucun arc, en d'autres termes, $deg(B) = 0$ donc le sommet B constitue lui-même une composante connexe du graphe.

Remarque : Formellement parlé, en supposant que tout sommet u est relié à lui-même, la relation «u est relié à v» est une relation d'équivalence sur l'ensemble des sommets d'un graphe G et les classes d'équivalence de la relation forment les composantes connexes de G.

3.5.7. Distance et diamètre dans un graphe

Considérons un graphe connexe G. La distance entre les sommets u et v dans G, notée $d(u,v)$, est la longueur du plus court chemin entre u et v. Le diamètre de G noté $diam(G)$ est la distance maximale entre deux points quelconques de G. Par exemple, la figure 3.8(a), $d(A,F) = 2$ et de $diam(G) = 3$, alors que dans la figure 3.8 (b), $d(A,F) = 3$ et $diam(G) = 4$.

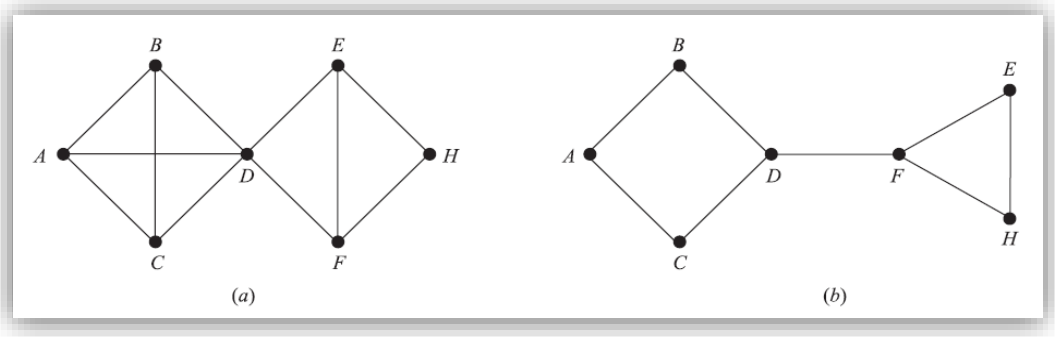


Figure 3.8. : Graphes connexes avec différents diamètres

Soit G un graphe, un nœud v de G est appelé- un point de coupure si $G - v$ est débranché. (Rappelons que $G - v$ est le graphe obtenu de G en supprimant v et tous les nœuds contenant v.) Une arête e de G est appelée un pont si $G - e$ est débranché. (Rappelons que $G - e$ est le graphe obtenu à partir de G en supprimant simplement l'arc e).

Dans la figure 3.8(a), le sommet D est un point de coupure et il n'y a pas de pont, alors que dans la figure 3.8(b), l'arête= [D, F] est un pont. (Ces points terminaux D et F sont nécessairement des points de césure).

3.6. Représentation d'une matrice par un graphe

Dans notre contexte d'étude, les données «matricielles », sont représentées sous forme d'un graphe de similarité non orienté et pondéré de façon à modéliser la relation de voisinage de ces différentes données ou pixels.

Ce graphe est défini comme suit : $G = (V,E)$ où :

- V est l'ensemble des nœuds : à chaque point x_i on associe un nœud s_i .

- E est l'ensemble des arcs entre les différents nœuds. Il correspond au produit cartésien $V \times V$.

A chaque arc reliant deux nœuds v_i et v_j (i diffèrent de j) est attribué un poids. Ce poids n'est autre qu'une fonction de similarité w_{ij} $(0 \leq w_{ij} \leq 1)$ calculée entre les données x_i et x_j.

Afin de mieux comprendre la représentation de données par un graphe de similarité des données, issues de la matrice de l'équation 3.9. Le graphe représentatif de ces données, est construit comme suit :

Nous associons à chacun des points x_i dans l'espace $\mathbb{R}^2$ un nœud noté s_i. Les 4 points seront ainsi représentés par 4 nœuds. Ensuite, comme un arc relie chaque paire de nœuds deux à deux, nous aurons un total de 6 arcs. Nous pondérons chaque arc reliant deux nœuds s_i et s_j par la similarité w_{ij} entre leurs données correspondantes x_i et x_j en utilisant la matrice de similarité de l'équation 3.12, basée sur la distance Euclidienne.

Par exemple, l'arc reliant les nœuds s_3 et s_4 est pondéré par la valeur 0.0483 puisque la similarité entre les points x_3 et x_4 est $w_{34} = w_{43} = 0.0483$.

Il est important de signaler que la similarité au sein d'un même nœud n'est pas représentée dans ce graphe, car elle est toujours égale à 1.

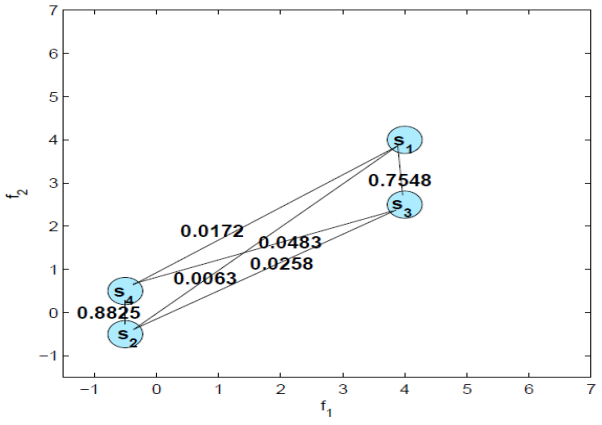


Figure 3.9 : Représentation des données et du graphe correspondant

3.7. Matrice d'adjacence

Une autre manière de maintenir un graphe dans une matrice, est d'utiliser la matrice d'adjacence qui se base sur **le voisinage des nœuds**, les arêtes deviennent presque invisibles et un graphe complexe devient une pure matrice $A = [a_{ij}]$, tel que :

$$a_{ij} = \begin{cases} 1 \; si \; v_i \; est \; adjacent \; \text{à} \; v_j \\ 0 \quad sinon \end{cases} \qquad 3.13$$

Si le graphe contient n nœuds, la matrice a une dimension de n lignes par n colonnes, donc l'ordre de la matrice est de O (n^2). Parmi les propriétés intéressantes de cette matrice est sa diagonale nulle et sa symétrie.

La matrice d'adjacence de la figure 3.10 (b) représente le graphe de la figure 3.10(a).

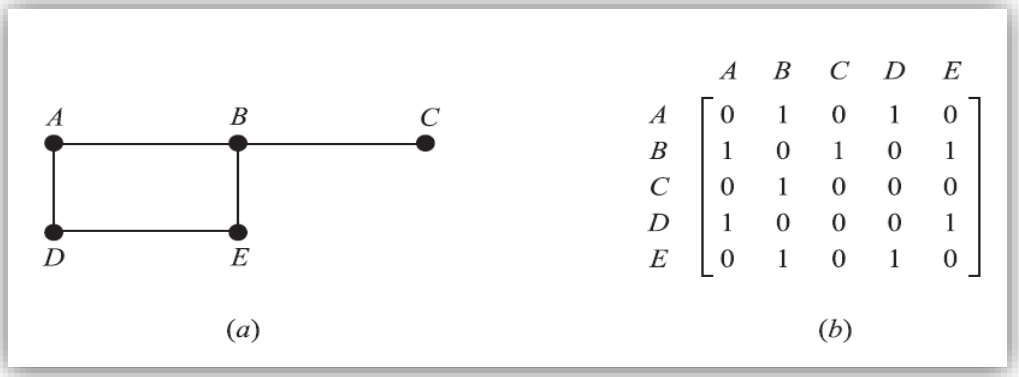


Figure 3.10 : Graphe connecté (a) et sa matrice d'adjacence (b)

3.8. Types de graphes

Il existe principalement deux types de graphes qui se distinguent par le nombre d'arcs reliant les nœuds entre eux.

– Graphe complètement connecté : Tous les nœuds du graphe sont connectés entre eux. Cela induit alors un très grand nombre d'arcs, même pour un petit nombre de nœuds. En effet, pour *n* données représentées par *n* nœuds du graphe correspondent $n(n-1)/2$ arcs.

Connecter tous les nœuds du graphe entre eux n'est pas toujours utile. Dans ce cas, il est préférable d'utiliser un graphe partiellement connecté.

– Le Graphe partiellement connecté se caractérise par des connexions partielles entre les nœuds liés à la notion de voisinage. On en distingue 2 catégories :

- ϵ −voisinage : seuls les nœuds v_i, v_j associés aux points x_i et x_j dont la distance δ_{ij} dans l'espace des attributs est inférieure à un certain seuil ε sont connectés, ϵ étant un réel fixé par l'utilisateur.
- k−voisinage : le nœud v_i associé au point x_i est connecté au nœud v_j associé au point x_i si x_j est parmi les k plus proches voisins de x_i au sens d'une distance dans l'espace des attributs, k étant un nombre entier fixé par l'utilisateur.

Il faut noter que la construction du graphe partiellement connecté est uniquement basée sur la distance séparant les points dans l'espace des attributs.

Il est intéressant d'étudier les coûts de calcul de ces 3 types de graphes en distinguant le coût de construction du graphe ; le coût de stockage du graphe ainsi que le coût de parcours du graphe. Il est vrai que le coût de stockage ainsi que le coût de parcours du graphe partiellement connecté ($\in$−voisinage et k− voisinage) sont inférieurs respectivement au coût de stockage et au coût de parcours du graphe complètement connecté, puisque seule une partie des arcs de connexion est représentée dans ce type de graphe.

Cependant, la construction du graphe complètement connecté nécessite $n(n-1)/2$ opérations pour le calcul des différentes valeurs de similarité. Pour un graphe ε−voisinage, à ce coût seront ajoutées $n(n-1)/2$ opérations afin de faire un seuillage des différentes valeurs de similarité et d'éliminer les valeurs inférieures au seuil. Tandis que, pour un graphe de k−voisinage, à ce coût sera ajouté le coût du classement des $(n-1)$ similarités, à savoir, $(n(n-1)/2) \times log(n(n-1)/2)$ opérations et le coût d'extraction des k−plus proches voisins de chacune des *n* données qui atteint $k \times n$ opérations.

Afin de mieux comprendre la différence entre le graphe complètement connecté, le graphe ϵ- voisinage et le graphe k-voisinage, nous allons reprendre

l'exemple précédent de la matrice de l'équation 3.9, et nous illustrerons les différents types de graphe en prenant en considération le nœud s_3 du graphe représentatif du point x_3, en nous appuyant sur la matrice de distance Δ et sur la matrice de similarité W. (En représentant uniquement les arcs du nœud s_3).

La figure 3.11 (a) illustre le cas du graphe complètement connecté où le nœud *s*3 est connecté à tous les autres nœuds du graphe.

La figure 3.11(b) illustre le cas du graphe de $\in-$ voisinage($\epsilon = 2$). Puisque seule δ_{31} = 1.5 est inférieure à ε d'après la matrice Δ, le nœud v_3 est connecté au nœud v_1 situé dans un disque centré en v_3 et de rayon ε .

La figure 3.11(c) illustre le cas du graphe de *k*-voisinage pour *k*=2. D'après la matrice des distances.

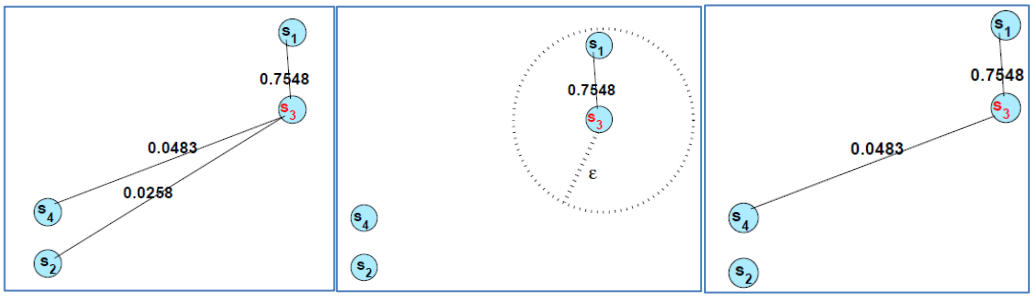


Figure 3.11 : Différents graphes de similarité-(a) Graphe totalement connecté, (b) Graphe de ε voisinage (ε=2), (c)Graphe des *k* plus proches voisins (k=2)

Δ, δ_{31} = 1.5, δ_{32} = 5.4083, δ_{34} = 4.9244. Les points x_1 et x_4 sont alors les 2 points les plus proches de x_3 dans l'espace des attributs. Le nœud v_3 est connecté aux nœuds v_1 et v_4 qui sont associés aux 2 plus proches voisins de x_3 dans l'espace des attributs.

Il est important de noter que les valeurs de *k* et ε ont une influence directe sur le nombre d'arcs du graphe. Plus *k* et ε sont grands, plus le nombre d'arcs reliant chaque nœud à ses voisins augmente.

3.9. Contexte des graphes dans la classification semi-supervisée

Dans cette partie, nous allons reprendre les définitions de la classification semi-supervisée, et les intégrer dans l'apprentissage des données en utilisant les graphes.

Etant donné des données d'apprentissage $(x_i, y_i), i = 1..L$ et $(x_j = L + 1..L + U)$, avec L : nombre de données étiquetées et U : nombre de données non-étiquetées.

Dans notre cas d'étude $U >> L$, et c'est ce contexte qui a permis l'émergence des nouveaux algorithmes pour solutionner cette problématique de données labellisées avec un coût largement élevé, comparé aux données gratuites non-labellisées en nombre infini. Il apparait aussi que lorsque U est large, le graphe devient très complexe.

Nous allons lister maintenant quelques algorithmes d'apprentissage utilisant les graphes, selon les considérations suivantes :

- Les nœuds du graphe sont directement associés aux données $(x_i, i = 1..L) \cup (x_j, j = L + 1..\ L + U)$,
- Un ou plusieurs nœuds appartiennent **à** une et une seule classe.
- Les classes sont en nombre fini,
- Les nœuds peuvent appartenir à une classe (Labélisé) ou peuvent être sans classe (non-labélisé).

3.9.1. Algorithme MINCUT

Le premier algorithme d'apprentissage semi-supervisé à base de graphe, est formulé sous forme d'un problème de graphe coupé [11, 86].Les cas étiquetés positifs sont des nœuds "source", desquels un liquide s'écoule. De même, les cas étiquetés négatifs sont des nœuds ou "puits", où le fluide disparaît.

L'objectif est de trouver un ensemble minimal d'arêtes qui après enlèvement de quelques "chemins", l'écoulement du liquide des sources aux puits est stoppé. Ceci définit une "coupure", ou une partition du graphe en deux ensembles de sommets. La "taille de la coupe" est mesurée par la somme des poids sur les nœuds définissant la coupe. Une fois que le graphe est divisé, les nœuds «sources» sont étiquetés classes positives, et les nœuds « puits » sont étiquetés classes négatives.

Mathématiquement, nous voulons trouver une fonction $f(x) \in [-1, 1]$ sur les nœuds, telle que $f(x_i) = y_i$ pour les instances labellisées, et la taille de la coupure est réduite au minimum :

$$\sum_{i,j:f(x_i)\neq f(x_j)} w_{ij} \qquad 3.14$$

La quantité définie par l'équation 3.14 définie la taille de la coupure, car si la quantité w_{ij} entre deux nœuds est enlevé, il doit être vrai que $f(\boldsymbol{x}_i) = f(\boldsymbol{x}_j)$

Le Mincut est formulé comme un problème de minimisation de risque régularisé, avec une fonction appropriée de perte et de régularisation.

Pour chaque nœud x_i labellisé, $f(x_i)$ est attaché à l'étiquette donnée par : $f(x_i) = y_i$

Cela peut être réalisé par une fonction de perte de la forme :[11]

$$c(\boldsymbol{x}, y, f(\boldsymbol{x})) = \infty.(y - f(\boldsymbol{x}))^2 \qquad 3.15$$

- Où $\infty \cdot 0 = 0$.

Cette fonction de perte est nulle si $f(x_i) = y_i$, et infinie autrement. Pour minimiser le risque régularisé, $f(x_i)$ devra être égale à y_i, sur les sommets étiquetés. La régularisation correspond à la taille de la coupure.

Rappel : nous exigeons $f(x) \in$ [-1, 1] pour tous les sommets x non labélisés. Par conséquent, la taille de la coupure peut être réécrite sous la forme

$$\Omega(f) = \sum_{i,j=1}^{l+u} w_{ij} \left(f(x_i) - f\left(x_j\right)\right)^2 /4 \qquad 3.16$$

Notons que la somme est maintenant sur toutes les paires de sommets. Si x_i et x_j ne sont pas connectés, alors $w_{ij} = 0$; si le nœud existe et n'est pas coupé, alors $f(x_i) - f(x_j) = 0$. Alors, la taille de la coupe est bien définie, même si on somme sur toutes les paires de sommets.

Dans l'équation 3.16, une forme équivalente aurait pour être utilisée : $|f(xi) - f(xj)|/2$, mais le terme quadratique est compatible avec d'autres approches discutées dans [11]. Le problème de risque du Mincut est alors régularisé par :

$$\min_{f:f(x)\in\{-1,1\}} \infty.\sum_{i=1}^{l}\left(y_i - f(x_i)\right)^2 + \sum_{i.j=1}^{l+u} w_{ij}\left(f(x_i) - f\left(x_j\right)\right)^2 \quad 3.17$$

Il s'agit d'un problème de programmation linéaire, car f est contrainte de produire des valeurs discrètes, soit -1 ou 1. Des algorithmes polynomiaux efficaces existent pour résoudre le problème du Mincut qui est un algorithme transductif d'apprentissage, car la solution f est définie uniquement sur les sommets et non pas sur tout l'espace des attributs.

La formulation du Mincut a une faille, qui peut être illustrée sur la figure 3.12, ou diverses coupures donneront 6 configurations ou solutions possibles !!!, ce qui donne lieu à utiliser le degré de confidence sur la labélisation.

Sur le graphe de la figure 3.11, la chaîne non pondérée dont un sommet est marqué à chaque extrémité, est une multi-solution pour le Mincut.

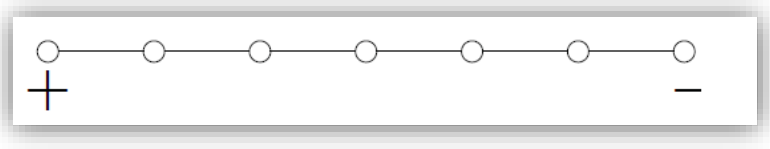


Figure 3.12 : Graphe à 7 sommets dont 2 extrémités sont étiquetés.

3.9.2. Les fonctions harmoniques

Le second algorithme d'apprentissage semi-supervisé utilise les fonctions harmoniques.

Une fonction harmonique est une fonction qui a les mêmes valeurs que les étiquettes fournies sur les données étiquetées, et satisfait la propriété de **moyenne pondérée** sur les données étiquetées ou non étiquetées.

$$f(x_i) = y_i \ , i = 1 \ldots l \quad 3.18$$

$$f\left(x_j\right) \leftarrow \frac{\sum_{k=1}^{l+u} w_{jk} f(x_k)}{\sum_{k=1}^{l+u} w_{jk}} \quad j = l+1 \ldots . l+u \quad 3.19$$

En d'autres termes, la valeur attribuée à chaque sommet non étiqueté est la moyenne pondérée des valeurs voisines. La fonction harmonique est la solution au même problème que l'équation 3.17, sauf que nous nous relaxons f pour prendre des valeurs réelles.

$$\min_{f:f(x)\in\mathfrak{N}} \infty.\sum_{i=1}^{l}\left(y_i - f(x_i)\right)^2 + \sum_{i,j=1}^{l+u} w_{ij}\left(f(x_i) - f\left(x_j\right)\right)^2 \quad 3.20$$

Ceci est équivalent à un problème d'optimisation générale :

$$\min_{f:f(x)\in\mathbb{R}} \sum_{i,j=1}^{l+u} w_{ij}\left(f(x_i) - f\left(x_j\right)\right)^2 \quad 3.21$$

$$\text{Sujet à : } f(x_i) = y_i \ , i = 1 \dots l \quad 3.22$$

La relaxation a un effet très distingué, car elle maintient une solution de forme compacte pour f, dont la solution est unique (sous certaines conditions) et elle est globalement optimale. L'inconvénient de la relaxation, c'est que la solution $f(x)$ est une valeur réelle dans [-1, 1] qui ne correspond pas vraiment à un label de classe.

Cela peut cependant être adressé par seuillage de $f(x)$ par rapport à zéro pour produire des étiquettes ou labels distincts (par exemple, si $f\ (x) >= \ 0, y\ = \ 1$ est prédit, et si $f(x) < 0, y\ = -1$).

La fonction harmonique f a de nombreuses interprétations intéressantes, par exemple :

Le graphe peut être vu comme un réseau électrique. Chaque sommet électrique est une résistance de valeur $\frac{1}{w_{ij}}$, ou de conductance équivalente w_{ij}.

Les sommets labélisés sont reliés à une batterie de 1 volt, de telle sorte que les sommets positifs se connectent au côté positif et les sommets négatifs se connectent à la terre, puis la tension établie à chaque nœud est la fonction harmonique, voir figure 3.11(a).

La fonction harmonique f peut aussi être interprétée par une marche aléatoire sur le graphe. Imaginez une particule au sommet i. A l'instant suivant, la particule se déplace de façon aléatoire à un autre sommet j, avec une probabilité proportionnelle à w_{ij} telle que :

$$P(j|i) = \frac{w_{ij}}{\sum_k w_{ik}} \quad 3.23$$

La marche aléatoire continue de cette façon jusqu'à ce que la particule atteigne l'un des sommets étiquetés. Ce qui est connu comme une marche aléatoire absorbante, où les sommets étiquetés sont les états absorbants. Ensuite, la valeur de la fonction harmonique $f(x_i)$ au sommet i, est la probabilité qu'une particule partant du sommet i atteigne finalement un sommet étiqueté positivement.

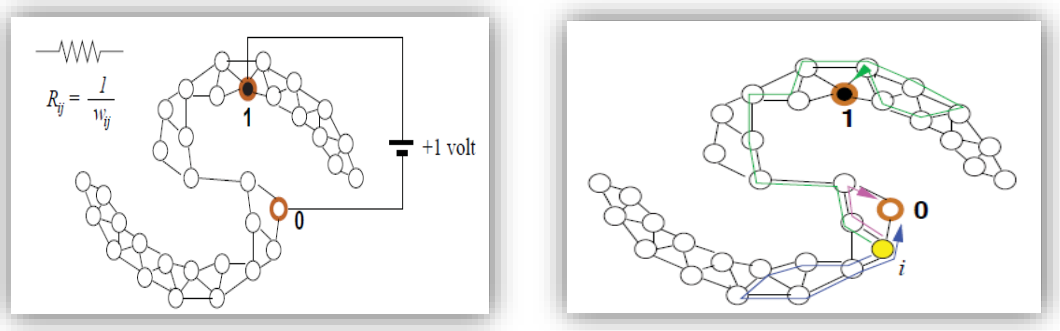


Figure 3.13: Equivalence entre la tension électrique et la fonction harmonique

La fonction harmonique peut être interprétée comme la tension dans un réseau électrique, ou la probabilité d'atteindre un sommet positif dans une marche aléatoire absorbant sur le graphe, comme illustre dans la figure 3.13.

Il s'agit d'une procédure itérative pour calculer la fonction harmonique de l'équation 3.20. Initialement, nous fixons $f(x_i) = y_i$ pour les sommets étiquetés $i = 1 \ldots l$, et une valeur arbitraire pour les sommets non étiquetés. Puis d'une manière Itérative nous mettons à jour la valeur de f de chaque sommet non étiqueté avec la moyenne pondérée de ses sommets voisins en utilisant la fonction suivante :

$$f(x_i) \leftarrow \frac{\sum_{j=1}^{l+u} w_{ij} f(x_j)}{\sum_{j=1}^{l+u} w_{ij}} \qquad 3.24$$

Cette procédure itérative est garantie de converger [11] vers la fonction harmonique, quelles que soient les valeurs initiales des sommets non étiquetés.

Cette procédure est parfois appelée propagation de l'étiquette, car elle propage les étiquettes des sommets étiquetés (qui sont fixés) progressivement à tous les sommets non étiquetés.

Nous allons maintenant présenter la forme compacte de la solution de la fonction harmonique. La solution est d'autant plus simplifiée par l'utilisation des matrices. Soit la matrice W de taille $(l+u)\times(l+u)$ la matrice des poids des arêtes dont les éléments sont les w_{ij}. Le graphe étant non orienté, W est une matrice symétrique. Ses éléments sont non négatifs.

Soit $D_{\mathrm{ii}} = \sum_{j=1}^{l+u} \mathrm{w}_{ij}$ la somme des degrés pondérés du sommet i, c'est à dire, la somme des poids des arêtes connectées au sommet i.

Soit D de dimension $(l+u)\times(l+u)$, la matrice diagonale en plaçant les D_{ii}, $i = 1\ldots l+u$ sur la diagonale.

Le graphe non normalisé de la matrice du Laplacien L est défini comme suit :

$$L = D - W \tag{3.25}$$

Soit $f = (f(\mathrm{x}_1),\ldots,f(\mathrm{x}_{l+u}))^T$ le vecteur des valeurs de f sur tous les sommets du graphe. La formule de régularisation de l'équation 3.17 peut être réécrite de la façon suivante :

$$\tfrac{1}{2}\sum_{i.j=1}^{l+u} w_{ij}\left(f(\boldsymbol{x}_i) - f\left(\boldsymbol{x}_j\right)\right)^2 = f^T L f \tag{3.26}$$

En supposant que les sommets sont ordonnés de telle sorte que les sommets étiquetés sont répertoriés en premier, on peut partitionner la matrice du Laplacien en quatre sous-matrices :

$$L = \begin{bmatrix} L_{ll} & L_{lu} \\ L_{ul} & L_{uu} \end{bmatrix} \tag{3.27}$$

Et partitionner en $f(f_l, f_u)$, et poser $\boldsymbol{y_l} = (y_1,\ldots,y_l)^{\boldsymbol{T}}$.

En solutionnant le problème d'optimisation sous contrainte, en utilisant des multiplicateurs de Lagrange, la solution harmonique est :

$$\boldsymbol{f_l} = \boldsymbol{y_l} \tag{3.28}$$

$$\boldsymbol{f_u} = -{L_{uu}}^{-1} L_{ul}\boldsymbol{y_l} \tag{3.29}$$

Exemple d'application : Fonction harmonique sur la chaîne graphe de la figure 3.11. avec lecture des sommets de gauche à droite des sommets [11].

Pour appliquer l'équation 3.29, nous avons besoin de permuter l'ordre des sommets comme suit (1, 7, 2, 3, 4, 5, 6), de sorte que les sommets étiquetés viennent en premier. A noter également $\boldsymbol{y}l = (1, -1)T$.

$$\text{Alors } f_u = (2/3, 1/3, 0, -1/3, -2/3)^T \qquad 3.30$$

Pour les étiquettes des sommets de gauche à droite. Un seuillage par rapport à zéro produira le résultat final des étiquettes.

$$y_{u\_final} = (1, 1, 1, -1, -1)^T \qquad 3.31$$

Cette solution correspond à l'intuition d'étiqueter le sommet du milieu comme classe positif, avec comme confidence la valeur de de la fonction f_u.

3.9.3. Régularisation par collecteur (Manifold Regularization)

Le Mincut ainsi que la fonction harmonique sont des algorithmes d'apprentissage transductifs. Ils apprennent à une fonction f, qui se restreint aux étiquettes du graphe. Il n'existe aucun moyen direct de prédire l'étiquette d'un exemple de test x^* [11] non présenté lors de l'entrainement, sauf si x^* est inclus dans le graphe comme nouveau sommet, et l'on refait les calculs.

Ceci n'est pas souhaitable, si l'on veut faire des prédictions sur un grand nombre d'instances de test. Ce qui serait judicieux, c'est de disposer d'un algorithme d'apprentissage semi-supervisé inductif.

En outre, le Mincut et la fonction harmonique fixent $f(x) = y$ pour les instances étiquetées, sans considérer le fait que quelques instances peuvent être fausses, Il n'est pas rare pour des jeux de données réelles d'avoir une étiquette bruitée. Il serait alors aussi intéressant que $\boldsymbol{f}$ soit en désaccord avec les étiquettes fournies.

La régularisation par collecteur répond à ces deux problèmes. Il s'agit d'un algorithme d'apprentissage inductif définissant *f* dans tout l'espace des attributs tels que la fonction $f: \boldsymbol{X} \rightarrow \boldsymbol{R}$, f est régularisée pour être lisse, par rapport au

graphe, par la matrice du Laplacien comme dans l'équation 3.26, Cependant, cette régularisation seule ne contrôle que *f*, sur les «$l + u$» instances.

Pour éviter que *f* soit non lisse, et donc d'avoir des performances de généralisation dégradées à l'extérieur des pixels d'apprentissage, il est nécessaire d'inclure un deuxième terme de régularisation, sur la norme de f, telle que :

$$\|f\|^2 = \int_{x \in X} f(x)^2 \qquad 3.32$$

En mettant les deux termes ensemble :

$$\Omega(f) = \lambda_1 \|f\|^2 + \lambda_2 \mathrm{f}^T L \mathrm{f} \qquad 3.33$$

Où λ_1, λ_2 ≥ 0 contrôlent l'équilibre relatif des deux termes. Pour permettre à f d'être en désaccord avec les étiquettes données, la fonction de perte $\mathrm{c}(\mathrm{x}, \mathrm{y}, \mathrm{f}(\mathrm{x})) = (\mathrm{y} - \mathrm{f}(\mathrm{x}))^2$, est introduite, toutefois cette fonction de perte ne pénalise pas considérablement les petits écarts. D'autres fonctions de perte existent dans l'état de l'art…, par exemple la fonction de perte charnière [11].

Le problème de régularisation collecteur devient alors

$$\min_{f:\mathrm{X} \to \boldsymbol{R}} \sum_{i=1}^{l} (\mathrm{y}_i - \mathrm{f}(\mathrm{x}_i))^2 + \lambda_1 \|f\|^2 + \lambda_2 \mathrm{f}^T L \mathrm{f} \qquad 3.34$$

L'équation 3.34 garantie que f soit optimale et admet une représentation dimensionnelle finie des $(l + u)$ instances. Il existe des algorithmes efficaces pour trouver le f optimal.

Au lieu de la matrice du graphe non normalisée du Laplacien L, la matrice du Laplacien normalisé $\mathcal{L}$ est trop souvent utilisé : [87]

$$\mathcal{L} = \mathrm{D}^{-1/2} \mathrm{L} \mathrm{D}^{-1/2} = \mathrm{I} - \mathrm{D}^{-1/2} \mathrm{W} \mathrm{D}^{-1/2} \qquad 3.35$$

Il en résulte un terme de régularisation légèrement différent

$$\mathrm{f}^T \mathcal{L} \mathrm{f} = \frac{1}{2} \sum_{i,j=1}^{l+u} \mathrm{w}_{ij} \left(\frac{\mathrm{f}(\mathrm{x}_i)}{\sqrt{\mathrm{D}_{ii}}} - \frac{\mathrm{f}(\mathrm{x}_j)}{\sqrt{\mathrm{D}_{jj}}} \right)^2 \qquad 3.36$$

D'autres variantes comme L^p ou $\mathcal{L}^{\mathrm{p}}$, avec p> 0, sont également possibles. Elles remplacent la matrice L dans l'équation 3.34, elles présentent globalement le

même lissage des étiquettes dans le graphe, avec quelques subtilités mathématiques [11].

3.10. L'hypothèse des méthodes basées sur les graphes

Les étiquettes sont "lisses", ce terme fortement utilisé en apprentissage semi-supervisé informe de la variation plus ou moins affine des étiquettes des sommets voisins dans un graphe, à savoir si elles varient lentement sur le graphe. autrement dit, si deux instances sont reliées par une arête de forte intensité, alors leur étiquettes ont tendance à être les mêmes, donc la notion de régularité peut être investie par l'étude spectrale du graphe.

Soit un vecteur φ le vecteur propre d'une matrice carrée A, alors $A\varphi = \lambda\varphi$, où λ est la valeur propre associée. Si φ est un vecteur propre, $c\varphi$ est aussi un vecteur propre, pour tout c non nul. La suite de cette thèse s'intéressera aux vecteurs propres de norme unitaire.

La théorie spectrale des graphes considère la matrice du Laplacien non-normalisée avec les propriétés suivantes :

- Il existe «$l+u$» valeurs propres (certaines peuvent être similaires) ayant pour vecteurs propres correspondant $(\lambda_i, \varphi_i)\ _{i=1}^{l+u}$. Ces paires sont appelées spectre de graphe.
- Les vecteurs propres sont orthogonaux : ${\varphi_i}^T\varphi_j = 0\ , i \neq j$ et donc forment une base.
- La matrice de Laplacien peut être décomposée en une somme pondérée de produits :
  - $L = \sum_{i=1}^{l+u} \lambda_i \varphi_i {\varphi_i}^T$ 3.37
- Les valeurs propres sont des nombres réelles non négatives, et peuvent être ordonnées telles que :

$$0 = \lambda_1 \leq \lambda_2 \leq \ldots + \leq \lambda_{l+u} \qquad 3.38$$

En particulier, le graphe a k composantes connexes si et seulement si $\lambda_1 = \lambda_2 = .. = \lambda_k = 0$. Les vecteurs propres correspondants sont constants sur chaque composante reliée, et zéro ailleurs.

3.11. Conclusion

Nous avons introduit dans ce chapitre un outil essentiel à notre travail qui se résume à l'utilisation des graphes dans la représentation des données des images satellitaires décrites dans le chapitre 5.

Parmi les points à retenir de ce chapitre est le calcul de la matrice de similarité et son utilisation dans le calcul de la matrice du Laplacien, qui sera intégré dans notre formulation mathématique dans le chapitre 4. L'hypothèse de régularité sera aussi considérée, ainsi que l'introduction de nouveaux paramètres de régularisation qui feront l'objet d'une étude particulière.

CHAPITRE 4

4. LA MACHINE D'APPRENTISSAGE EXTREME (ELM)

Nulle pierre ne peut être polie sans friction, nul homme ne peut parfaire son expérience sans épreuves.
Confucius

4.1. Introduction

Dans ce chapitre, nous allons décrire un nouvel outil de classification et de régression appelé : machine d'apprentissage extrême (ELM), en intégrant notre approche de sélection des pixels les plus informatifs par l'apprentissage actif du chapitre 3.

L'ELM est une stratégie qui a révolutionné les réseaux de neurones [19] en minimisant le temps d'apprentissage et en augmentant les taux de reconnaissance. L'idée de sélection à priori du nombre de nœuds est devenue obsolète, voire disparaitre avec l'intégration du noyau ou "kernel" dans la classification et la régression.

Le principe du réseau neuronal n'est pas modifié, mais le rôle de l'adaptation est reconsidéré, ainsi que le nombre de couches cachées qui se restreint à une seule couche. Ainsi, plutôt que d'ajuster tous les poids d'un réseau pour émuler une fonction, le réseau est constitué d'un grand nombre de neurones dans la couche interne. Les poids d'entrée sont initialisés aléatoirement une seule fois et restent avec cette valeur. L'adaptation, qui se fait en une seule fois aussi, porte donc uniquement sur les poids de la couche de sortie.

4.2. Généralité sur les réseaux de neurones (RN)

Le réseau de neurones est un réseau de cellules unitaires qui reçoivent la même information et décident différemment, comme illustré dans la figure 4.1., ce réseau est formé de plusieurs couches contenant chacune un nombre de neurones, chaque neurone possède des entrées et une sortie.

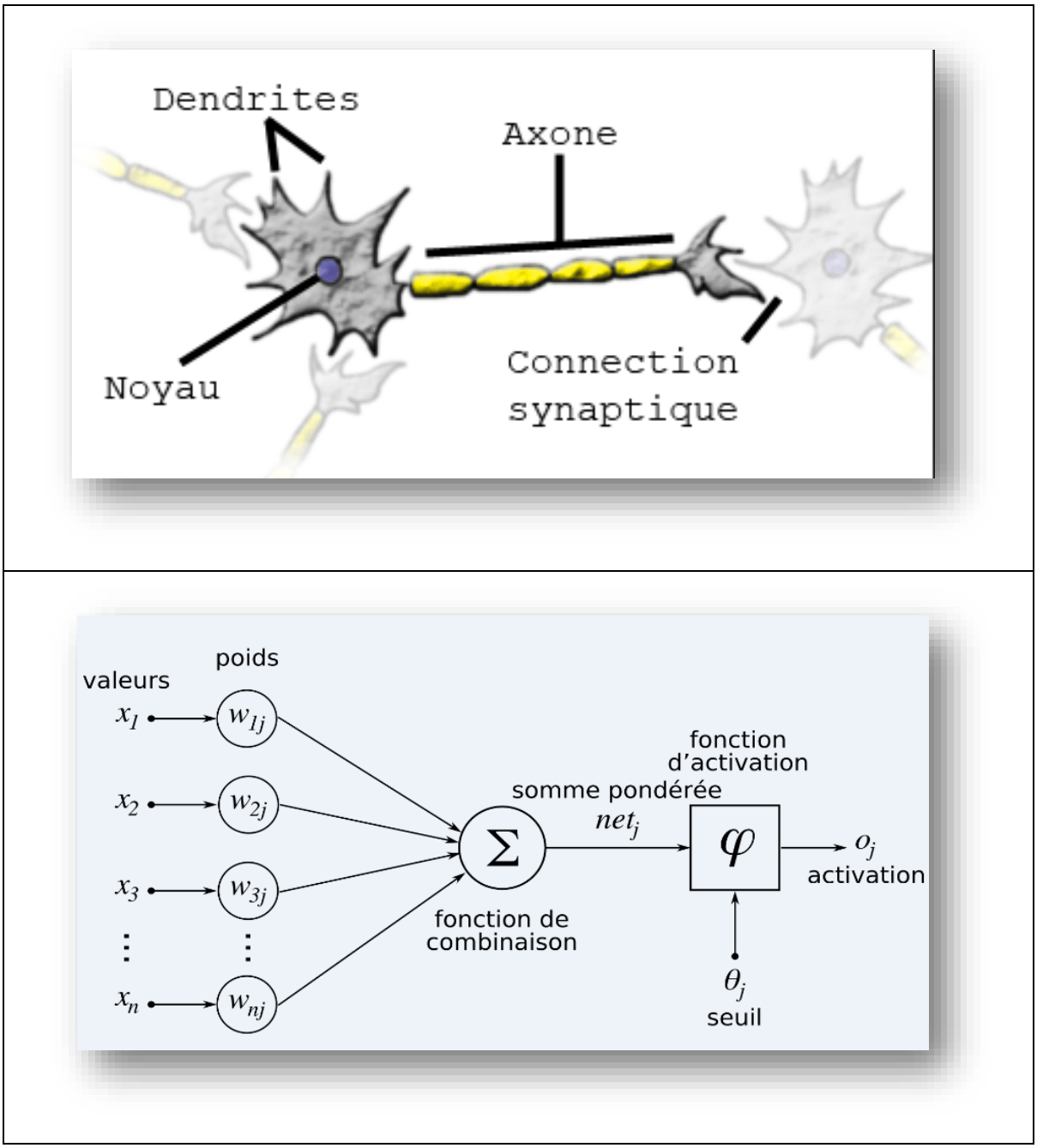


Figure 4.1. : Similarité entre réseau de neurone biologique (en bas) et sa version numérique (à haut)

Les RN ont deux caractéristiques essentielles :

- La première consiste en la décomposition d'une tâche complexe en multi taches, à décision plus ou moins basiques et la fusion de ces décisions permet une configuration plus complexe en classification, prédiction ou en régression.
- La seconde caractéristique, c'est que le réseau de neurone est adaptatif, chaque neurone qui est la brique de base, contient des paramètres qui peuvent être modifiés, mis à jour, en vue d'adapter le réseau à une tâche particulière. Ces modifications sont faites lors d'une phase appelée apprentissage du réseau.

Les réseaux de neurones sont utilisés dans des domaines très variés tels que la reconnaissance d'écriture manuscrite, la parole, l'imagerie, les télécommunications, les diagnostiques, les prédictions météo, etc.…

Les RN se divisent en deux classes principales, les réseaux non-bouclés et les réseaux récurrents, comme illustré dans la figure 4.2. Dans le contexte de cette thèse, nous nous intéressons aux réseaux monocouches et multicouches en vue d'expliquer les innovations de l'ELM.

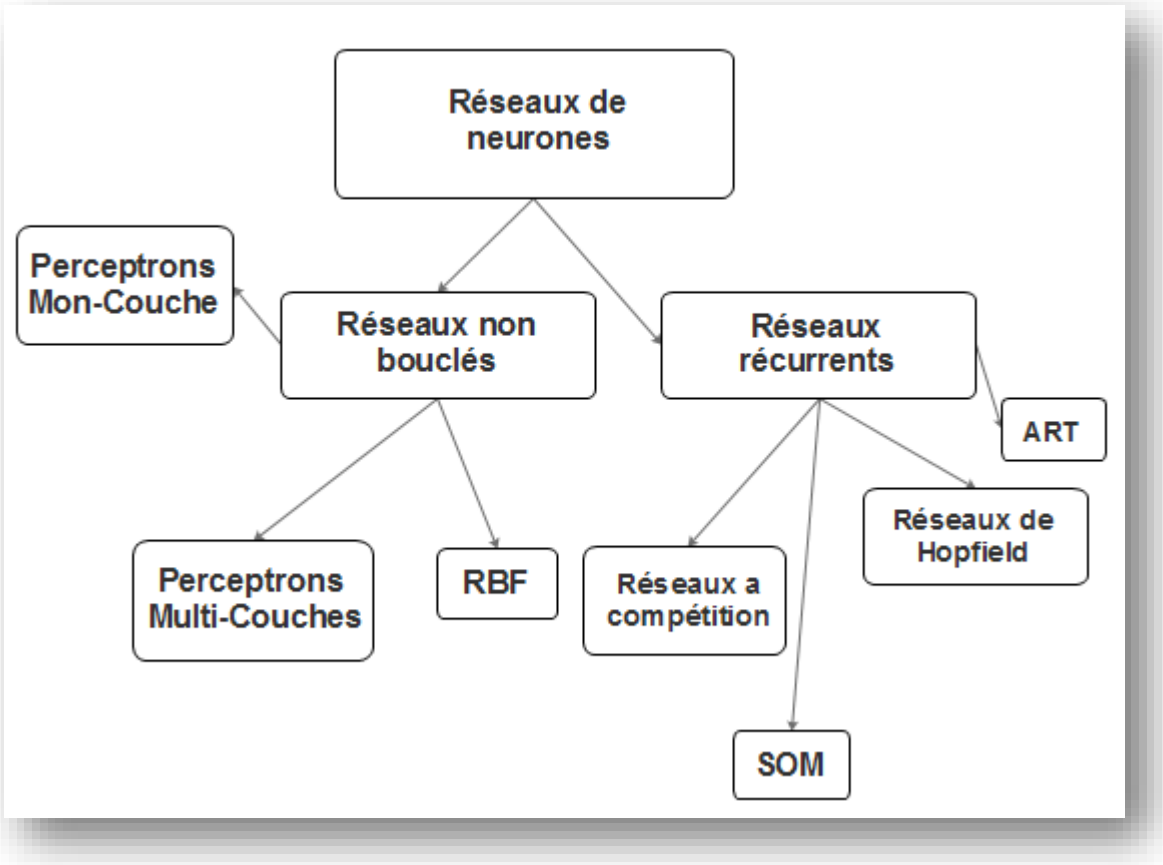


Figure 4.2 : Taxonomie des réseaux de neurones

4.2.1. Les réseaux monocouches (SLFN)

On dit qu'un RN est monocouche, si les neurones d'entrée sont entièrement connectés aux neurones de sorties, appelé aussi perceptron. Le réseau monocouche possède une structure comme celle représentée dans la figure 4.3.

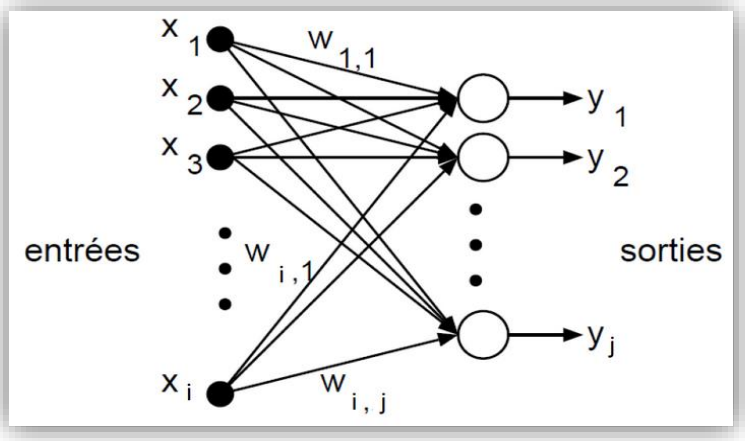


Figure 4.3 : Réseau monocouche

4.2.2. Les Réseaux multicouches (MLP)

Le model à multicouches, comme son nom l'indique contient différentes couches de décision, donc l'information passe à travers les neurones de la couche de gauche comme illustre dans la figure 4.4, vers la/les couches de droite permettant ainsi la propagation de la décision, et sa fusion dans chaque neurone avant de la transformer en décision à son tour vers la couche suivante, donc c'est une décision singulière basée sur l'expérience du groupe. Cette stratégie de décision s'affine lors de l'entrainement, ce qui permet à chaque neurone de se calibrer à la tache requise et participer ainsi à la décision finale.

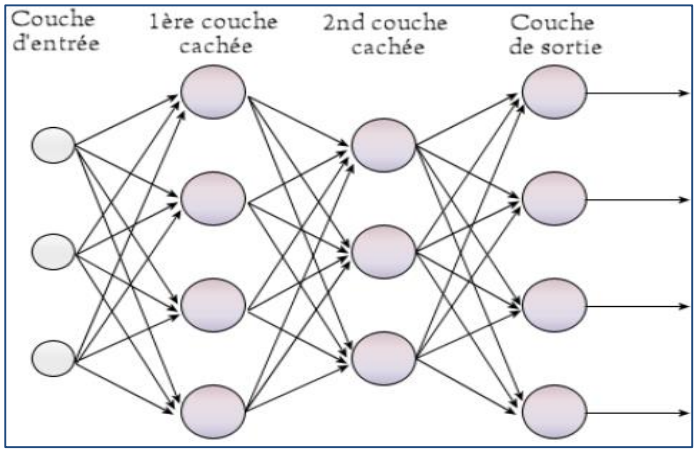


Figure 4.4 : Réseau multicouches à 2 couches cachées et 1 couche de sortie

Les caractéristiques d'un tel réseau sont les suivantes :

- La topologie est formée de plusieurs couches de neurones sans communication à l'intérieur d'une même couche.
- La couche d'entrée accepte les données à traiter en provenance d'une source extérieure au réseau.
- Une ou plusieurs couches cachées réalisent le traitement spécifique du réseau.
- Chaque neurone de chaque couche possède une liaison avec tous les neurones de la couche suivante, sa sortie est une entrée aux neurones de la couche suivante.
- La couche de sortie génère les prédictions.
- L'apprentissage des neurones se fait via des algorithmes tels que ceux basées sur la descente du gradient, les méthodes itératives,

etc… qui ont vus un développement accru ces dernières décennies. Le back-propagation (BP) et ses variantes sont les plus populaires.

4.3. Problématiques des Réseaux de Neurones

Les RN, comme d'autres algorithmes souffrent de trois problèmes définis comme suit :

- Le sur-apprentissage qui se produit lorsque le modèle commence à mémoriser les données d'entraînement, plutôt que d'apprendre à généraliser sur la tendance observée dans les données de test, d'où une mauvaise généralisation sur de nouvelles instances de test.
- Le risque de piège dans les minima locaux.
- Le temps d'exécution augmente exponentiellement avec l'augmentation de la dimension des données, le nombre de couches cachées, etc…

4.4. Motivation de la machine d'apprentissage extrême (ELM)

Vue l'importance des classificateurs dans le domaine de la segmentation d'images, et l'avancée des techniques de classification, une nouvelle approche a apparu ces dernières années [25], ce nouvel algorithme est une solution alternative aux RN et corrige tous les problèmes liés à ce classificateur, en plus de sa capacité à surpasser les SVM [5] sur différents problèmes de classification et de régression.

Une comparaison très intéressante a été réalisée par [19, 23] ou une panoplie d'expérience sur différentes catégories de problèmes et a été élaborée, et il été démontré que l'ELM concurrence et dépasse différentes variantes d'algorithmes dans l'état de l'art. Différents travaux ont apparus ensuite renforçant l'utilisation de l'ELM, notamment [21, 24, 88, 89].

Parmi les domaines de prédilection de l'ELM notons à titre d'exemple, la détection de la qualité des images codés JPEG [90], la détection automatique de l'épilepsie dans le domaine du biomédical [91], la détection de la vigilance basée sur l'électroencéphalogramme (EEG) [92], la prédiction des interactions protéines-protéines dans les séquences des acides aminés [93], la reconnaissance des visages [94], la prévision du cycle de vie d'une entreprise

dans la gestion financière [95], l'optimisation des ressources matérielles [96] etc.

Nous ne pourrons citer tous les travaux et domaines d'application, mais l'ELM prétend encore à quelques années de succès, et présente une formulation mathématique très intéressante, pour le cas binaire et multi-classes, ce qui sera décrit dans la section suivante.

4.5. Principe de l'ELM

Considérons, en premier lieu, un problème d'apprentissage supervisé avec un ensemble d'apprentissage de N pixels, $(X, Y) = (x_i, y_i)_N^{i=1}$ avec $x_i \in \mathbb{R}^d$ et y_i étant la classe correspondante de la donnée x_i

- Cas de la multi-classification : y_i est un vecteur binaire de taille n_0, et prend une valeur égale à 1 en une seule position.
- Cas de la régression : $y_i \in R^{\mathrm{d}}$

L'ELM [19, 22, 97] vise à apprendre une règle de décision ou une fonction d'approximation sur la base des données d'apprentissage.

En général, l'apprentissage se compose de deux étapes. La première étape est de construire la couche cachée en utilisant un nombre fixe de neurones dont les poids sont générés aléatoirement, et de fonctions non linéaires continues par morceaux, telle que la fonction sigmoïde $g(x;\theta) = \frac{1}{1+exp(-(\boldsymbol{a}^T\boldsymbol{x}+b))}$ ou la fonction gaussienne :$g\ (\boldsymbol{x},\theta) = exp(-b\|\boldsymbol{x}-\boldsymbol{a}\|)$

- avec $\theta = (\boldsymbol{a}, b)$ les paramètres de la fonction de mappage et $\|\cdot\|$ représente la norme euclidienne.

Une caractéristique intéressante de l'ELM est que les paramètres, de la fonction de la couche cachée, peuvent être générés de façon aléatoire selon une distribution de probabilité continue. Par exemple, la distribution uniforme sur l'intervalle $[-1, 1]$, Cela rend les ELM distincts des RN traditionnels et des SVM.

Les seuls paramètres libres, qui doivent être optimisés dans le processus d'apprentissage, sont les poids de sortie, c.à.d. les poids reliant les neurones de la couche cachée avec les nœuds de sortie.

Ceci revient à considérer l'apprentissage des ELM, comme résoudre un problème équivalent à la solution des moindres carrés régularisés, qui est plus efficace que l'apprentissage d'un SVM ou l'apprentissage avec rétro-propagation.

Dans cette étape, un nombre L fixe de neurones cachés (généralement définis par l'utilisateur) sont générés aléatoirement. Ces neurones sont responsables du mappage des données de l'espace des attributs de dimension d vers un espace de dimensions L, soit $d \times L$ valeurs réelles dans l'intervalle $[-1,1]$.

Soit $h(x_i) \in \mathbb{R}^{1\times L}$le vecteur de sortie de la couche cachée par rapport à x_i, et $\boldsymbol{W} \in \mathbb{R}^{L\times P}$, la matrice des poids de sortie.

La fonction de sortie de l'ELM est alors :

$$\boldsymbol{f}(x_i) = h(x_i).\mathbf{W}, \quad i = 1..N \tag{4.1}$$

Pour les cas binaires, la fonction de décision de l'ELM est définie par :

$$\boldsymbol{f}(x_i) = sign(h(x_i)\boldsymbol{W}) \tag{4.2}$$

La différence de l'ELM, par rapport aux algorithmes traditionnels, est sa tendance à minimiser l'erreur d'apprentissage, ainsi que la norme des poids de sortie. D'après la théorie de Bartlett [98], cette tendance dans les RN aide à la généralisation.

Donc l'ELM tend à minimiser les quantités suivantes :

$$\|\boldsymbol{HW} - \boldsymbol{\eta}\|^2 = \|\xi\|^2 \quad and \quad \|\boldsymbol{W}\| \tag{4.3}$$

$\boldsymbol{\eta}$: Vecteur cible de la classe prédite.

ξ : Erreur d'apprentissage.

$\boldsymbol{H}$: étant la matrice des poids de la couche cachée.

De l'équation 4.3, minimiser la norme des poids de sortie $\|\boldsymbol{W}\|$, est équivalent à maximiser la distance ou marge qui sépare deux différentes classes dans l'espace des attributs de l'ELM, soit : $\frac{2}{\|\boldsymbol{W}\|}$

La méthode minimale des moindres carrés a été utilisée dans la formulation de base de l'ELM dans [19, 97] en posant :

$$\boldsymbol{W} = \boldsymbol{H}^{**}\eta \qquad 4.4$$

Ou $\boldsymbol{H}^{**}$ est la matrice généralisée de Moore-Penrose (MP) [99], différentes méthodes existent dans la littérature pour calculer cet inverse, soit par :

- La méthode de projection orthogonale (MPO) [100]
- La méthode d'orthogonalité
- Les méthodes itératives
- La méthode de décomposition en valeurs singulières (SVD)

La MPO peut être utilisée dans les deux cas :

- $\boldsymbol{H^T H}$ est non-singulière et $\boldsymbol{H}^{**} = (\boldsymbol{H^T H})^{-1}\boldsymbol{H^T}$
- $\boldsymbol{HH^T}$ est non-singulière et $\boldsymbol{H}^{**} = \boldsymbol{H^T}(\boldsymbol{HH^T})^{-1}$

D'après la théorie de de la régression d'arête [101] , une quantité positive peut être ajoutée à la diagonale de la matrice $\boldsymbol{HH^T}$ ou $\boldsymbol{H^T H}$, ainsi la solution sera stable et aura une meilleure généralisation.

4.6. Formulation mathématique de l'ELM

D'après la théorie de l'ELM, le mappage des données par $h(x)$ peut prévaloir a l'ELM d'approximer n'importe quelle fonction continue [89, 102].

La j^{ième} sortie de l'ELM multi-classes avec P nœuds de sortie est donnée par :

$$f_j(x) = h(x)w_j \qquad 4.5$$

Où $w_j \in \mathbb{R}^{1\times P}$ est le vecteur des coefficients des poids entre la couche cachée et le $j^{éme}$ nœud de sortie.

$h(x)$ fait correspondre le vecteur d'entrée vers l'espace des attributs de l'ELM. Cette mise en correspondance des caractéristiques peut être faite dans un espace restreint, comme pour les classificateurs standards tels les RN ou dans un espace infini par l'application de l'astuce du noyau.

Le problème d'optimisation en norme L_2 associé à l'ELM [22] est défini comme suit :

$$\text{Minimiser : } \mathrm{L}_{\mathrm{Primal}_{\mathrm{ELM}}} = \frac{1}{2}\|W\|^2 + C\frac{1}{2}\sum_{\mathrm{i}=1}^{\mathrm{N}}\|\xi_i\|^2 \quad 4.6$$

$$\text{Sous la contrainte : } h(x_i).W = \eta_i{}^T - \xi_i{}^T \qquad i=1,\ldots N \quad 4.7$$

où C est un paramètre de régularisation.

$W = [w_1, \ldots, w_P]^T$ est une matrice de taille $P \times L$ formée en juxtaposant les poids des vecteurs de sortie w_j avec $j = 1, \ldots, P$.

$\eta_i = [\eta_{i1}, \ldots, \eta_{iP}]^T$ et $\xi i = [\xi_{i1}, \ldots, \xi_{iP}]^T$ sont les classes prédites et les erreurs d'apprentissage des P nœuds de sortie respectivement, par rapport au pixel d'apprentissage x_i.

Le vecteur cible η à toutes ses valeurs à 0, à l'exception de l'entrée qui correspond à l'étiquette y_i de la classe, qui est mise à 1.

En se basant sur le théorème de Karush-Kuhn-Tucker (KKT) [103], la détermination des poids d'apprentissage de l'ELM est équivalente à résoudre le problème dual d'optimisation :

$$\mathrm{L}_{\mathrm{Dual}_{\mathrm{ELM}}} = \frac{1}{2}\|W\|^2 + \mathrm{C}\frac{1}{2}\sum_{\mathrm{i}=1}^{\mathrm{N}}\|\xi_i\|^2 - \sum_{\mathrm{i}=1}^{\mathrm{N}}\sum_{\mathrm{j}=1}^{\mathrm{P}}\alpha_{i,j}\ (h\ (x_i)w_{\mathrm{j}} - \eta_{i,j} + \xi_{i,j}) \quad 4.8$$

En prenant les conditions d'optimalité de KKT,

$$\frac{\partial L_{Dual_{ELM}}}{\partial w_j} = 0 \rightarrow \mathrm{w_j} = \sum_{i=1}^{N}\alpha_{i,j}\ h\ (x_i)^T \rightarrow \mathrm{W} = \mathrm{H^T}\alpha \quad 4.9$$

$$\frac{\partial L_{Dual_{ELM}}}{\partial \xi_i} = 0 \rightarrow \alpha_i = \mathrm{C}\xi_i \quad , \ i = 1, \ldots, N \quad 4.10$$

$$\frac{\partial L_{Dual_{ELM}}}{\partial \alpha_i} = 0 \rightarrow h\ (x_i)W - \ \eta_i{}^{\mathrm{T}} + \xi_i{}^{\mathrm{T}} = 0\ , \ i = 1, \ldots, N \quad 4.11$$

$$\text{Avec } \alpha_i = [\alpha_{i,1}, \ldots, \alpha_{i,P}]^T \text{ et } \alpha = [\alpha_1, \ldots, \alpha_N]^T \quad 4.12$$

Le vecteur optimal des poids W^* peut être donné sous forme compacte [22] :

$$\mathrm{W}^* = \mathrm{H}^{\mathrm{T}} \left(\frac{\mathrm{I}}{C} + \mathrm{HH}^{\mathrm{T}}\right)^{-1} \eta \tag{4.13}$$

- Où H est la matrice de sortie de la couche cachée, définie comme suit :

$$\mathrm{H} = \begin{bmatrix} h(x_1) \\ \vdots \\ h(x_N) \end{bmatrix} = \begin{bmatrix} h_1(x_1) & \cdots & h_L(x_1) \\ \vdots & \ddots & \vdots \\ h_1(x_N) & \cdots & h_L(x_{\mathrm{N}}) \end{bmatrix} \tag{4.14}$$

- η est une matrice de taille $N \times P$ construite à partir des vecteurs de sortie cible η_i^T comme suit :

$$\eta = \begin{bmatrix} \eta_1^T \\ \vdots \\ \eta_N^T \end{bmatrix} = \begin{bmatrix} \eta_{11} & \cdots & \eta_{1P} \\ \vdots & \ddots & \vdots \\ \eta_{N1} & \cdots & \eta_{NP} \end{bmatrix} \tag{4.15}$$

- Et I est une matrice identité de taille $N \times N$.

La sortie de l'ELM est réécrite sous la forme finale suivante :

$$f(x) = h(\mathrm{x})\mathrm{W}^* = h(x)\mathrm{H}^{\mathrm{T}} \left(\frac{\mathrm{I}}{C} + \mathrm{HH}^{\mathrm{T}}\right)^{-1} \eta \tag{4.16}$$

Au cours de la phase de prédiction, le pixel de test $\mathbf{x}_\ell$ sera affecté de l'indice du nœud de sortie ayant la valeur la plus élevée. Autrement dit,

Si la sortie de l'ELM est de la forme :

$$\mathbf{f}^0(\mathbf{x}_\ell) = [f_1^0(\mathbf{x}_\ell), \ldots, f_P^0(\mathbf{x}_\ell)]^T \tag{4.17}$$

Alors la classe prédite du pixel $\mathbf{x}_\ell$ est donnée par :

$$y_\ell^* = \arg \max_{k \in (1,\ldots,P)} f_k^0(\mathbf{x}_\ell) \tag{4.18}$$

Dans l'espace du noyau ou l'espace kernel, la prédiction associée au pixel de test $\mathbf{x}_\ell$ est donné sous la forme compacte suivante :

$$\mathbf{f}^0(\mathbf{x}_\ell) = \begin{bmatrix} k\ (\mathbf{x}_\ell, \mathbf{x}_1) \\ \vdots \\ k\ (\mathbf{x}_\ell, \mathbf{x}_N) \end{bmatrix}^{\mathrm{T}} (\mathbf{I}/C + \mathbf{K})^{-1}\ \mathbf{Y} \tag{4.19}$$

Le premier terme de 4.19 est un vecteur de longueur N, et il représente les distances noyau entre un point de test $\mathbf{x}_\ell$ et les pixels d'apprentissage.

Nous pouvons remarquer que : l'ELM n'est pas sensitif au nombre de neurones cachés L par l'utilisation de l'astuce du kernel avec $\mathbf{K} = \boldsymbol{HH^T}$, mais il est seulement lié aux pixels d'apprentissage. Il ne dépend pas non plus du nombre de neurones de sortie.

De l'équation 4.19, on peut clairement remarquer que la classification et la prédiction se font d'une manière très similaire.

Pour des comparatifs avec d'autres classificateurs et des notions plus détaillés de l'ELM, nous référons le lecteur à lire [19-25]

4.7. Méthode proposée de classification avec l'ELM

De l'équation 4.19, la matrice des distances kernel, $\mathbf{K} \in \mathbb{R}^{N\times N}$ est calculée à partir des pixels d'apprentissage

Dans notre contexte d'étude, nous avons utilisé la fonction kernel Gaussienne définie par [22]:

$$k(\mathbf{x}_i, \mathbf{x}_j) = exp\left(-\gamma \left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2\right) \qquad 4.20$$

Ou γ est un paramètre inversement proportionnel à la largeur du kernel.

Pour des raisons de convenance de calcul, nous transformons la sortie de l'ELM en utilisant la fonction soft-max comme suit :

$$f_k^*(\mathbf{x}_\ell) = \frac{\exp\left(f_k^0(\mathbf{x}_\ell)\right)}{\sum_{j=1}^{P} \exp\left(f_j^0(\mathbf{x}_\ell)\right)},\ k = 1, \dots, P \qquad 4.21$$

Dans un contexte de perfectionnement et d'amélioration de la classification par l'ELM, nous avons développé une nouvelle approche qui concerne l'intégration de la marche aléatoire dans le graphe avec une fonctionnelle intégrant la matrice du Laplacien, en vue d'optimiser les classes prédites des pixels non étiquetés, avec comme classificateur de base l'ELM.

En vue d'avoir un classificateur robuste, les paramètres C et γ de l'ELM sont initialisés par l'algorithme de l'évolution différentielle (DE), qui est une approche méta-heuristique, qui se base sur l'évolution génétique, nous référons le lecteur à ces travaux pour plus de détails [104-106].

4.7.1. Classification avec l'ELMRW [18]

L'appellation ELMRW, est une combinaison de l'ELM avec l'algorithme de la marche aléatoire (RW). Ou l'ELM initialise les classes prédites des pixels non-étiquetés, et le RW utilise ces informations comme point de départ pour une nouvelle prédiction des pixels non-étiquetés.

Soit $G = (V, E)$ un graphe dont les sommets $v \in V$ et arêtes $e \in E$.

Chaque pixel de l'image I est associé à un sommet v_i et les sommets sont reliés par l'intermédiaire d'un réseau local de 8 connexions.

L'image contient l pixels étiquetés et u pixels non étiquetés.

Une arête s'étend sur deux sommets v_i et v_j est désigné par e_{ij} et le poids associé est noté par $\omega(e_{ij})$ ou tout simplement w_{ij}.

Un choix courant pour l'obtention de ces coefficients de pondération est la fonction de pondération gaussienne.

C'est à dire : $w_{ij} = exp(-\beta \|x_i - x_j\|^2)$ et β étant un paramètre libre.

Le degré d'un sommet v_i est $w_i = \Sigma \omega_{ij}$, pour tous les sommets incidents au sommet v_i.

La matrice du Laplacien indexé par les sommets (v_i, v_j) est donnée par :

$$L_{ij} = \begin{cases} d_{ij} & \text{if } i = j, \\ -\omega_{ij} & si\ i \text{ and } j \text{ connectés}, \\ 0 & \text{autre cas.} \end{cases} \quad 4.22$$

Dans le cas binaire, une définition possible du problème de minimisation de l'énergie peut être donnée sous la forme générale suivante [107] :

$$\min_f \sum_{e_{ij} \in E} \omega_{ij}{}^p \left\| f_i - f_j \right\|^q + \lambda \sum_{v_i \in V} \mu_i{}^p \| f_i - f_i^* \|^q \quad 4.23$$

Le premier terme est lié à la régularité de l'image. (Smoothness assumption)

Le second terme (terme de fidélité des données) encode une estimation initiale de la structure de l'image.

μ_i et λ sont respectivement, le poids local et global renforçant l'application de cette fidélité. Dans notre cas, c'est l'estimation initiale des probabilités à posteriori f_i^* de l'ELM.

En outre, les poids locaux μ_i pourraient être fixes à d_i, mais pour plus de simplicité, nous supposons μ_i= 1 pour tous les pixels de l'image.

Selon les différentes valeurs de p et q [107], différents modèles d'énergie peuvent être obtenus Dans notre cas, nous nous concentrons sur le cas ou p=1 et q=2, ce qui conduit à l'algorithme de la marche aléatoire [60]. Selon cette hypothèse, l'énergie correspondante devient alors :

$$\min_f \sum_{e_{ij}\in E} \omega_{ij} \left\| f_i - f_j \right\|^2 + \lambda \sum_{v_i \in V} \| f_i - f_i^* \|^2 \qquad 4.24$$

Dans le cas de la classification multi-classes, l'énergie peut être reformulée comme suit :

$$\min_f \sum_{e_{ij}\in E} \omega_{ij} \left\| \mathbf{f}_i - \mathbf{f}_j \right\|^2 + \lambda \sum_{v_i \in V} \mu_i \| \mathbf{f}_i - \mathbf{f}_i^* \|^2 \qquad 4.25$$

- Où f_i et f_j sont maintenant des vecteurs de probabilité (de dimension P) associés aux sommets v_i et v_j respectivement.
- $f_i^* \in \mathbb{R}^P$ est la probabilité à posteriori de l'ELM associé au sommet v_i.

En introduisant le Lagrangien $\mathcal{L}g$, l'équation 4.25 peut être écrite sous la forme matricielle suivante :

$$\mathcal{L}g = Trace(\mathbf{F}^T \mathbf{L} \mathbf{F}) + \lambda \| \mathbf{F} - \mathbf{F}^* \|_F^2 \qquad 4.26$$

- Où $\| \cdot \|_F$ est la norme de Frobenius[108].

$\mathbf{F} \in \mathbb{R}^{(u+l)\times P}$et $\mathbf{F}^* \in \mathbb{R}^{(u+l)\times P}$ sont construits à partir des vecteurs de probabilité $\mathbf{f}_k \in \mathbb{R}^P$ et $\mathbf{f}_k^* \in \mathbb{R}^P$ respectivement, avec $k = 1, \dots, u+l$.

$\mathbf{L} \in \mathbb{R}^{(u+l)\times(u+l)}$ Étant la matrice du Laplacien produite à partir de toutes les paires de sommets v_i and v_j possibles.

Notons que $\mathbf{L}$ est une matrice semi-définie positive et creuse [109].

Posons $V = V_l \cup V_u$, (l'ensemble des pixels étiquetés + l'ensemble des pixels non étiquetés), alors l'équation 4.26 peut être réécrite comme suit :

$$\mathcal{L}g = Trace\left(\begin{bmatrix}\mathbf{F}_l\\ \mathbf{F}_u\end{bmatrix}^T \begin{bmatrix}\mathbf{L}_l & \mathbf{B}\\ \mathbf{B}^T & \mathbf{L}_u\end{bmatrix}\begin{bmatrix}\mathbf{F}_l\\ \mathbf{F}_u\end{bmatrix}\right) + \lambda \left\|\begin{bmatrix}\mathbf{F}_l\\ \mathbf{F}_u\end{bmatrix} - \begin{bmatrix}\mathbf{F}_l^*\\ \mathbf{F}_u^*\end{bmatrix}\right\|_F^2 \qquad 4.27$$

Avec $\mathbf{F}_u \in \mathbb{R}^u$ inconnu,

$\mathbf{B} \in \mathbb{R}^{u\times l}$, $\mathbf{L}_l \in \mathbb{R}^{l\times l}$ et $\mathbf{L}_u \in \mathbb{R}^{u\times u}$ sont des sous-matrices de $\mathbf{L}$ issus de la décomposition de V en ensemble étiqueté et ensemble non étiqueté V_l et V_u, respectivement.

L'énergie fonctionnelle peut être réécrite comme suit :

$$\mathcal{L}g = Trace(\mathbf{F}_l^T\mathbf{L}_l\mathbf{F}_l + \mathbf{F}_u^T\mathbf{L}_u\mathbf{F}_u + 2\mathbf{F}_u^T\mathbf{B}^T\mathbf{F}_l) + \lambda\|\mathbf{F}_l - \mathbf{F}_l^*\|_F^2 + \lambda\|\mathbf{F}_u - \mathbf{F}_u^*\|_F^2 \qquad 4.28$$

Pour minimiser le Lagrangien de 4.28 par rapport à $\mathbf{F}_u$ on calcule :

$$\frac{d\mathcal{L}g}{d\mathbf{F}_u} = 2\lambda(\mathbf{F}_u - \mathbf{F}_u^*) + 2\mathbf{L}_u\mathbf{F}_u + 2\mathbf{B}^T\mathbf{F}_l = 0 \qquad 4.29$$

Ensuite, les probabilités associées aux pixels non étiquetés sont obtenus en résolvant le système linéaire d'équations suivant :

$$\mathbf{F}_u = (\mathbf{L}_u + \lambda\boldsymbol{I})^{-\mathbf{1}}(-\mathbf{B}^T\mathbf{F}_l + \lambda\mathbf{F}_u^*) \qquad 4.30$$

$\mathbf{L}_u + \lambda\mathbf{I}$ est une matrice semi-définie positive, et $\mathbf{L}$ une matrice creuse, générée à partir d'un diagramme en treillis de 8 pixels reliés, le système d'équations linéaires est alors résolu en un temps linéaire.

4.8. Etude de complexité

Différentes études ont été élaborées, en vue de comparer la complexité des ELM avec différents classificateurs de l'état de l'art, notamment les réseaux Deep Learning (DBN), les machines de Boltzman (DBM), etc …comme illustré dans le tableau 4.1 [22].

Nous remarquons que les temps d'entrainement sont de l'ordre des minutes, soit 7.5 minute pour l'ELM avec un taux de reconnaissance de 99.03%, tandis que les autres méthodes d'apprentissage accusent des temps d'entrainement en heures, avec des taux de même ordre, parfois nettement inférieurs.

Tableau 4.1 : Etude de complexité temporelle de l'ELM avec diverses méthodes d'apprentissage sur le jeu de données MNIST (OCR)

Méthodes d'Apprentissage	**Taux de Reconnaissance**	**Temps d'entrainement**
ELM	**99.03%**	**<7.5 minutes**
Deep Belief Networks (DBN)	98.87%	5.7 heures
Deep Boltzmann Machines (DBM)	99.05%	19 heures
SAE	98.6%	> 17 heures
SDAE	98.72%	> 17 heures

D'autres travaux tels [27] ont comparé les SVM, les ANN, les SANN avec l'ELM, en termes de temps d'exécution pour la classification de différents gènes, les auteurs ont expérimentalement trouvé qu'avec l'augmentation du nombre de gènes, les ELM présentent une meilleure convergence et des temps de reconnaissance nettement inférieurs. Tandis que les auteurs de [110], dans leur étude sur la classification des gestes, ont pu constater que les ELM présentent des temps d'entrainement nettement inférieurs aux SVM, avec des performances nettement supérieures, comme illustré dans le Tableau 4.2.

Tableau 4.2 : Reconnaissance des modèles gestuels [110]

	Entrainement		**Test**		**Entrainement incrémental**	
Algorithme	OA	Temps	OA	Temps	OA	Temps
ELM	96.93%	3.69s	90.33%	0.04s	69.67%	0.04s
SVM	88.59%	4.52s	88.00%	0.58s	63.67%	0.58s

4.9. Conclusion

Nous avons présenté, dans ce chapitre, la formulation mathématique de notre modèle d'apprentissage, qui correspond au calcul des étiquettes des pixels non- étiquetés, en intégrant le concept de régularité discuté au chapitre 1, ainsi que l'introduction d'un paramètre de fidélité qui encode la structure de l'image initiale, suivi d'un comparatif de l'ELM avec diverse méthodes de classification en termes de temps d'apprentissage et de taux de reconnaissance

Dans le prochain chapitre 5, nous allons exposer les images satellitaires utilisées dans notre travail, ainsi que les résultats de la classification des pixels non-étiquetés et l'impact des différents paramètres régularisant le processus de l'ELMRW.

CHAPITRE 5

5. DONNEES, RESULTATS ET INTERPRETATIONS

Dis-moi et j'oublierai, montre-moi et je me souviendrai,
Implique-moi et je comprendrai
Confucius

5.1. Introduction

Dans ce chapitre, nous allons présenter les différentes images satellitaires utilisées dans notre travail, ainsi que l'application de l'ELMRW sur ces images, en comparant les résultats avec l'état de l'art utilisant divers paramètres statistiques, notamment les taux de reconnaissance global (OA) et moyen (AA), avec comme paramètre d'accord la valeur du Kappa [111], ainsi que les déviations standards des paramètres précédents.

Différents paramètres de régularisation de notre fonctionnelle seront aussi variés, et nous verrons l'impact de chacun d'eux sur les taux de reconnaissance ou de classification.

5.2. Description des données d'apprentissage et de test

Nous avons opté pour trois images satellitaires, disponible au niveau de l'équipe de recherche, et qui ont déjà fait état de publication et de citation sur différents récents articles et revues scientifiques [54, 112-119]. Différents comparatifs avec l'état de l'art seront élaborés sur l'image référence de l'Université de Pavie.

5.2.1. Image satellitaire de Djeddah en Arabie Saoudite

Le premier ensemble de données représente une image multi-spectrale de très haute résolution (VHR) de taille 700 × 650 pixels acquise par le capteur IKONOS-2 en Juillet 2004, voir figure 5.1. Les pixels étiquetés par l'expert sont appelés «Ground Truth»(GT)

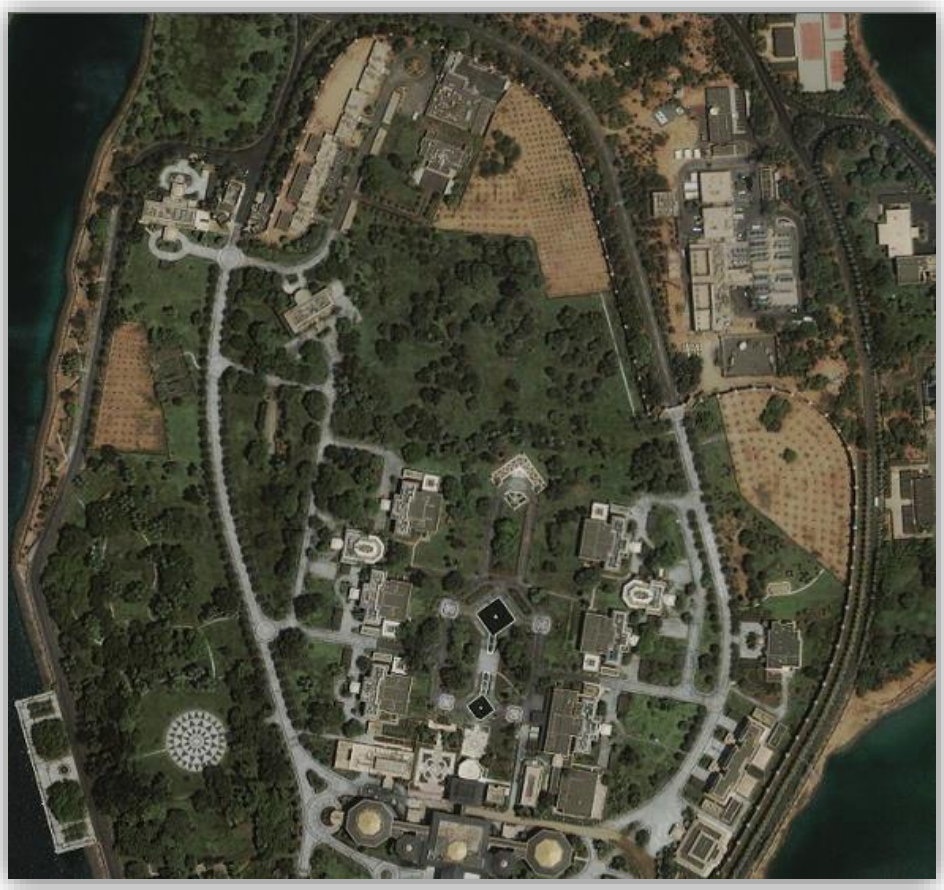


Figure 5.1. : Image satellitaire VHR de la ville de Djeddah en Arabie Saoudite

L'image a trois bandes spectrales avec une résolution spatiale de 1 m et se réfère à une partie de la ville de Djeddah, dans laquelle sept types de classes sont

dominantes: Deux types d'asphalte, du sol nu, de l'herbe, deux types de toits, des arbres et de l'eau, comme illustré dans la figure 5.2..

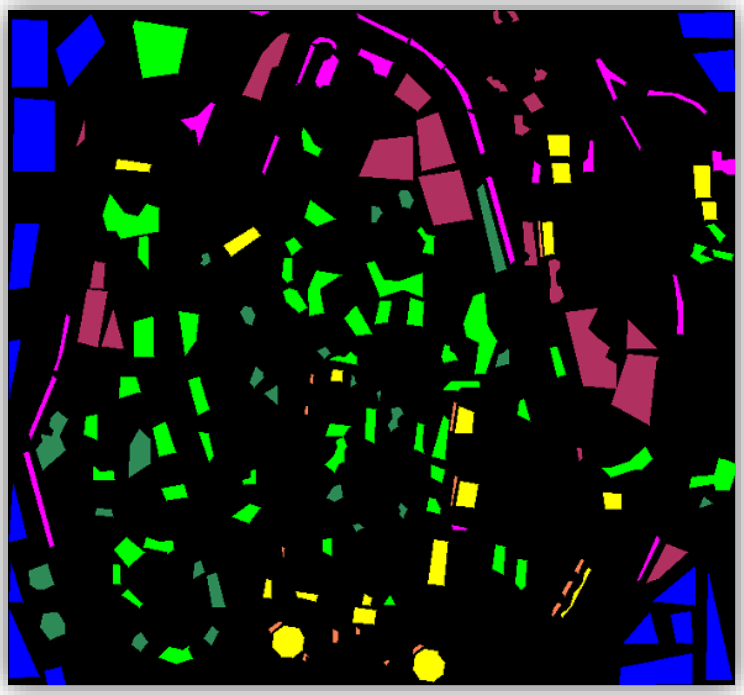


Figure 5.2. Pixels étiquetés par l'expert de l'image de l'image de Djeddah en Arabie Saoudite (GT)

Les statistiques de distribution des pixels étiquetés de Djeddah au nombre de 70339, du tableau 5.1 sont illustrés dans la figure 5.3.

Tableau 5.1 : Distribution des pixels étiquetés dans l'image de Djeddah (GT)

Classe	Type	Couleur	Pixels étiquetés
1	Bâti	(Jaune) 255-255-0	5940
2	Asphalte	(Magenta) 255-0-255	5089
3	Sol nu	(Marron) 176-48-96	14663
4	Végétation	(Vert) 0-255-0	17885
5	Arbre	(Vert de Mer) 46-139-87	5284
6	Ombre	(Corail) 255-127-80	462
7	Eau	(Bleu) 0-0-255	16617

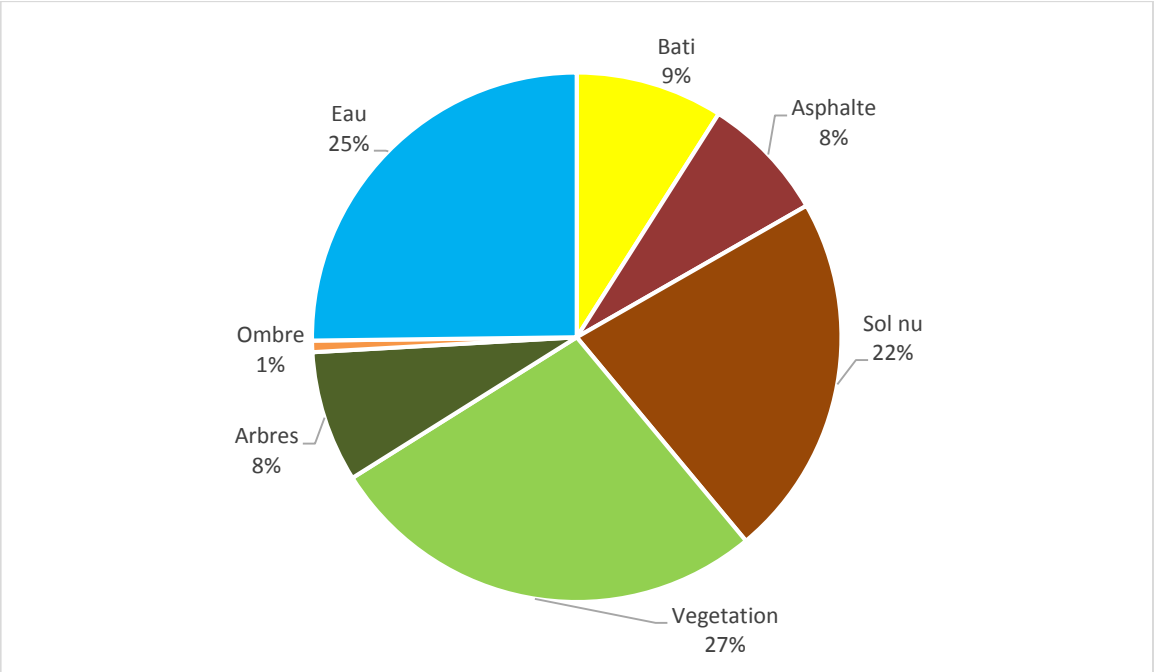


Figure 5.3. Statistiques sur l'image de Djeddah– Distribution des pixels étiquetés (GT)

5.2.2. Image satellitaire de Riyad en Arabie Saoudite

Le deuxième ensemble de données représente une image multi-spectrale de très haute résolution (VHR) de taille 800 × 800 pixels acquises par le capteur GeoEye-1 en Août 2010, voir figure 5.4.

Figure 5.4. : Image satellitaire de Riyad en Arabie Saoudite

L'image a trois bandes spectrales avec une résolution spatiale de 0,5 m et se réfère à une partie de la ville de Riyad (Arabie Saoudite). Huit classes sont considérées : deux types de toits, deux types d'asphalte, du sol nu, de l'herbe, des arbres et de l'eau, comme illustré dans la figure 5.5.

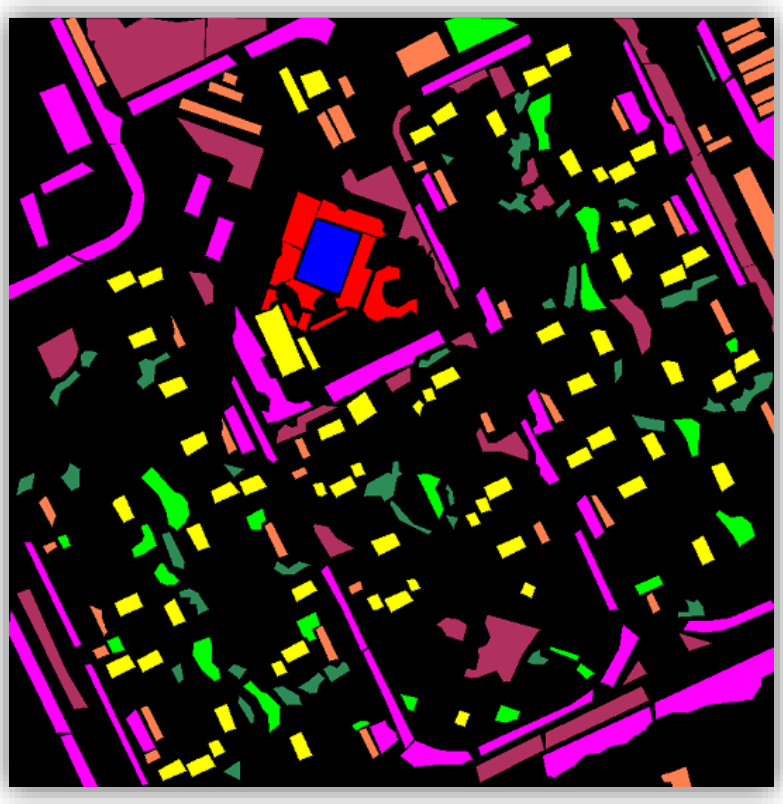


Figure 5.5. Pixels étiqueté par l'expert de l'image de Riyad en Arabie Saoudite (GT)

Les statistiques de distribution des pixels étiquetés au nombre de 151347, du tableau 5.2 sont illustrés dans la figure 5.5.

Tableau 5.2 : Nombre d'pixels étiquetés de l'image de Riyad (GT)

Classe	Type	Couleur	# Pixels étiquetés
1	Sol nu	(Marron) 176-48-96	32875
2	Asphalte	(Magenta) 255-0-255	45378
3	Arbres	(Vert de mer) 46-139-87	9238
4	Végétation	(Vert) 0-255-0	10591
5	Toit Blanc	(Corail) 255-127-80	14252
6	Toit sombre	(Jaune) 255-255-0	22064
7	Eau	(Bleu) 0-0-255	2998
8	Brique rouge	(Rouge) 255-0-0	7053

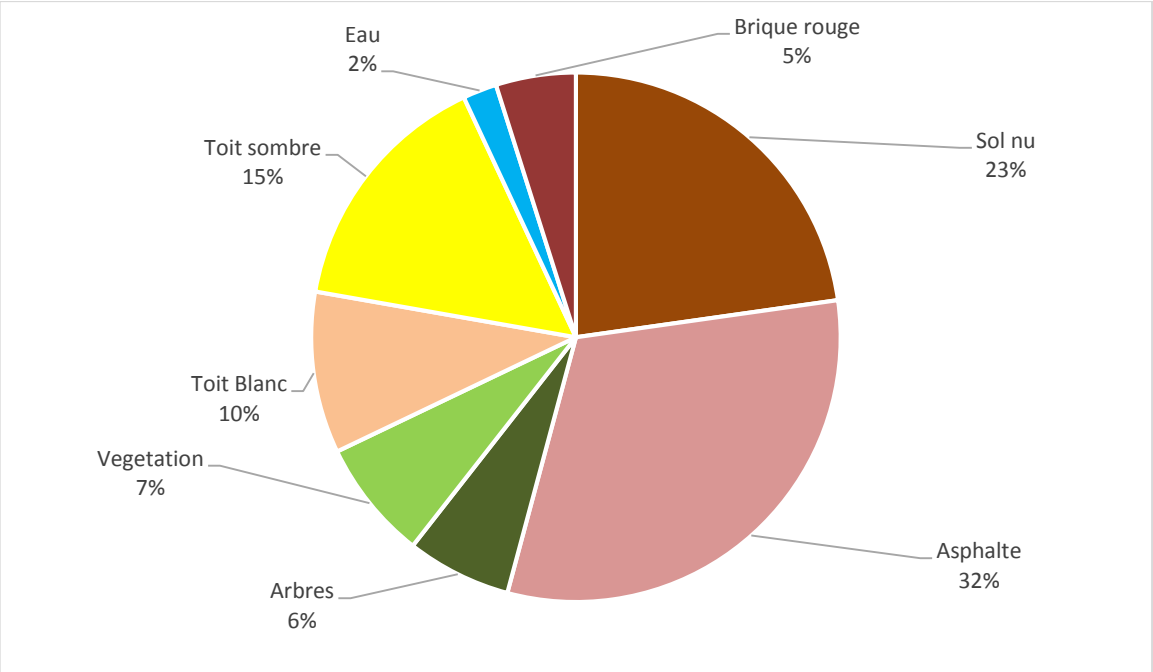
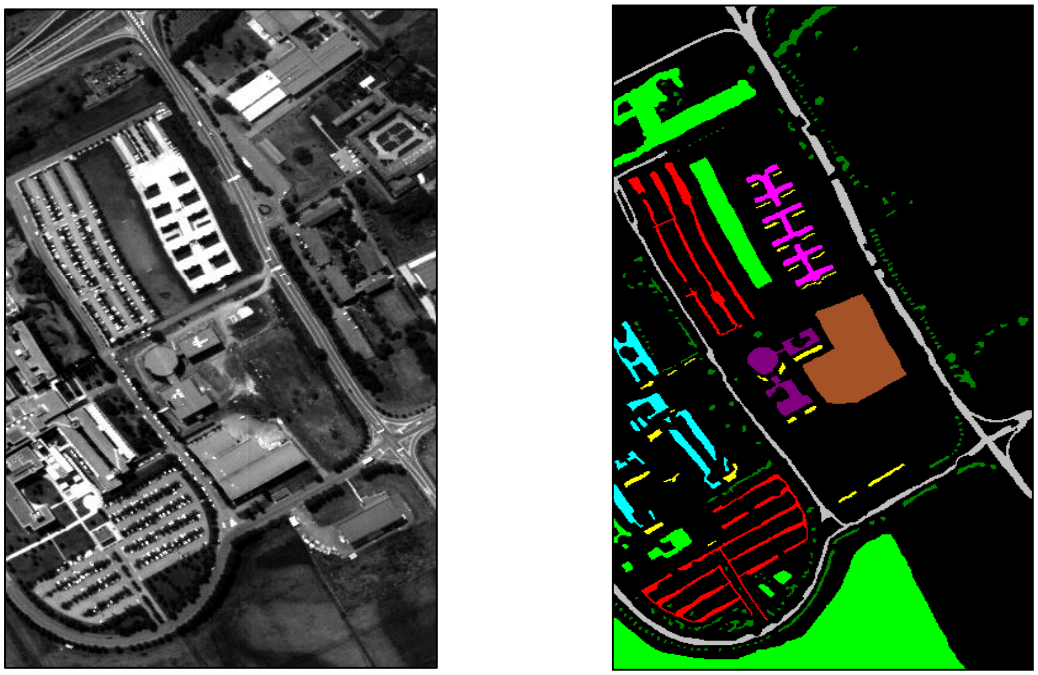


Figure 5.6. Statistiques sur l'image VHR de Riyad en Arabie Saoudite

5.2.3. Image satellitaire de l'Université de Pavie en Italie

Le troisième ensemble de données est l'image hyper-spectrale(HS) de l'Université de Pavie de taille 610 × 340 pixels, voir figure 5.7(a), c'est une image bien connue par la communauté de télédétection. Elle a été acquise par le capteur optique (ROSIS) sur une partie de l'Université de la ville de Pavie (Italie) en Juillet 2002.

L'image dispose de 103 bandes et se caractérise par une résolution spatiale de 1,3 m. Neuf classes sont à considérer, à savoir : l'asphalte, le sol nu, le bitume, la brique, les prairies, les ombres, les tuiles, les arbres et l'eau, comme illustré dans la figure 5.7(b).



(a) (b)

Figure 5.7 : Image satellitaire de l'Université de Pavie en Italie (a) et son GT (b)

Les statistiques de distribution des pixels étiquetés au nombre de 42776, du tableau 5.3 sont illustrés dans la figure 5.8.

Tableau 5.3 : Nombre de pixels étiquetés par un expert humain de l'image Hyper spectrale de Pavie (GT)

Classe	Type	Couleur	Pixels étiquetés
1	Asphalte	Gris	6631
2	Prairies	Vert	18649
3	Gravier	Cyan	2099
4	Arbre	Vert de Mer	3064
5	Tôle de Fer	Magenta	1345
6	Sol nu	Marron	5029
7	Bitume	Gris-brun	1330
8	Brique	Rouge	3682
9	Ombre	Jaune	947

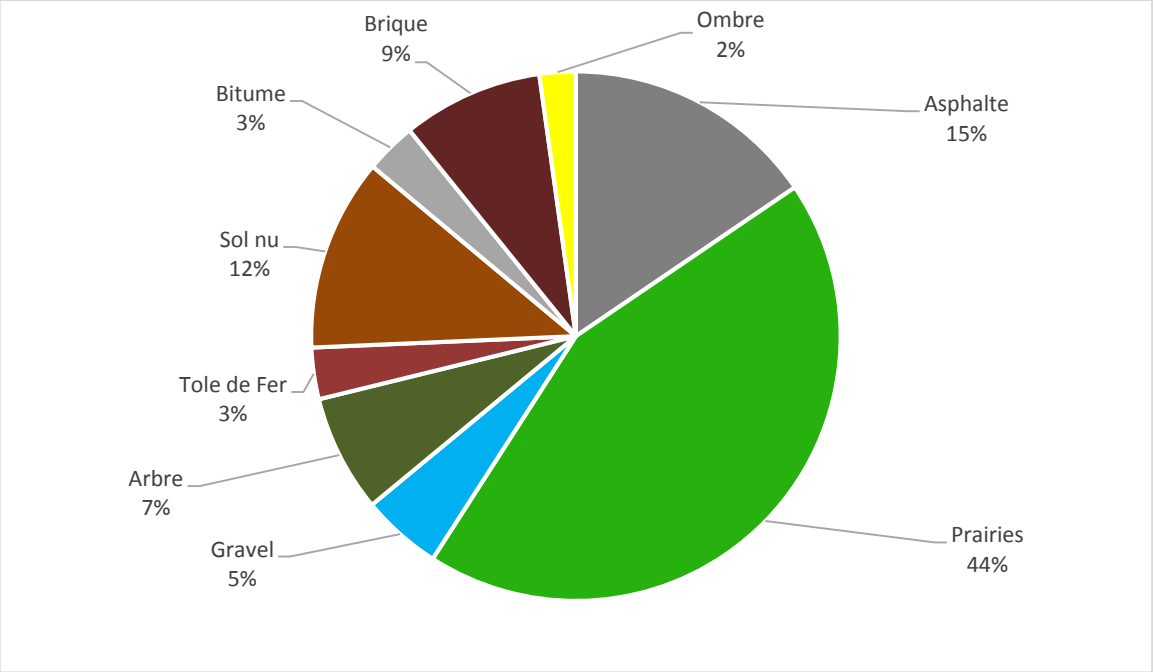


Figure 5.8. Statistiques sur l'image HS de l'université de Pavie en Italie (GT)

5.3. Comparatif des données d'apprentissage

Dans la perspective d'avoir une vue plus globale des données par images satellitaires, le tableau 5.4 illustre les différences relatives au nombre d'pixels par image.

Table 5.4. : Récapitulatif des 3 images satellitaires (Djeddah, Riyad, Université de Pavie) (GT)

Images	Capteurs Spectrométriques	Nombre de bandes	Nombre de classes	Nombre de pixels
Djeddah	IKONOS-2	3 (VHR)	7	70339
Riyad	GeoEye-1	3 (VHR)	8	151347
Université de Pavie	ROSIS	103 (HS)	9	42776

De plus amples détails sur les capteurs d'acquisition ROSIS, IKONOS et GeoEye-1 seront décrits dans l'appendice B.

5.4. Application du ELMRW

Dans la configuration de l'apprentissage actif, nous commençons d'abord par l'entrainement de l'ELMRW sur l'ensemble d'apprentissage S. Le modèle de classification qui en résulte est utilisé ensuite pour trier les pixels non étiquetés de l'ensemble d'apprentissage U. Chaque pixel est évalué selon le critère d'apprentissage actif (BT) [120], qui est basé sur les plus hautes probabilités postérieures de l'association d'un pixel à une classe donnée.

Dans le contexte multi-classes, la différence entre les deux plus fortes probabilités est indicative de la façon, dont un pixel est traité par le classificateur. Lorsque les deux probabilités les plus élevées se trouvent très proches, la confidence du classificateur est faible. C.à.d., que le pixel n'appartient pas vraiment à une classe unique spécifique, mais probablement peut appartenir à deux classes, avec des probabilités proches.

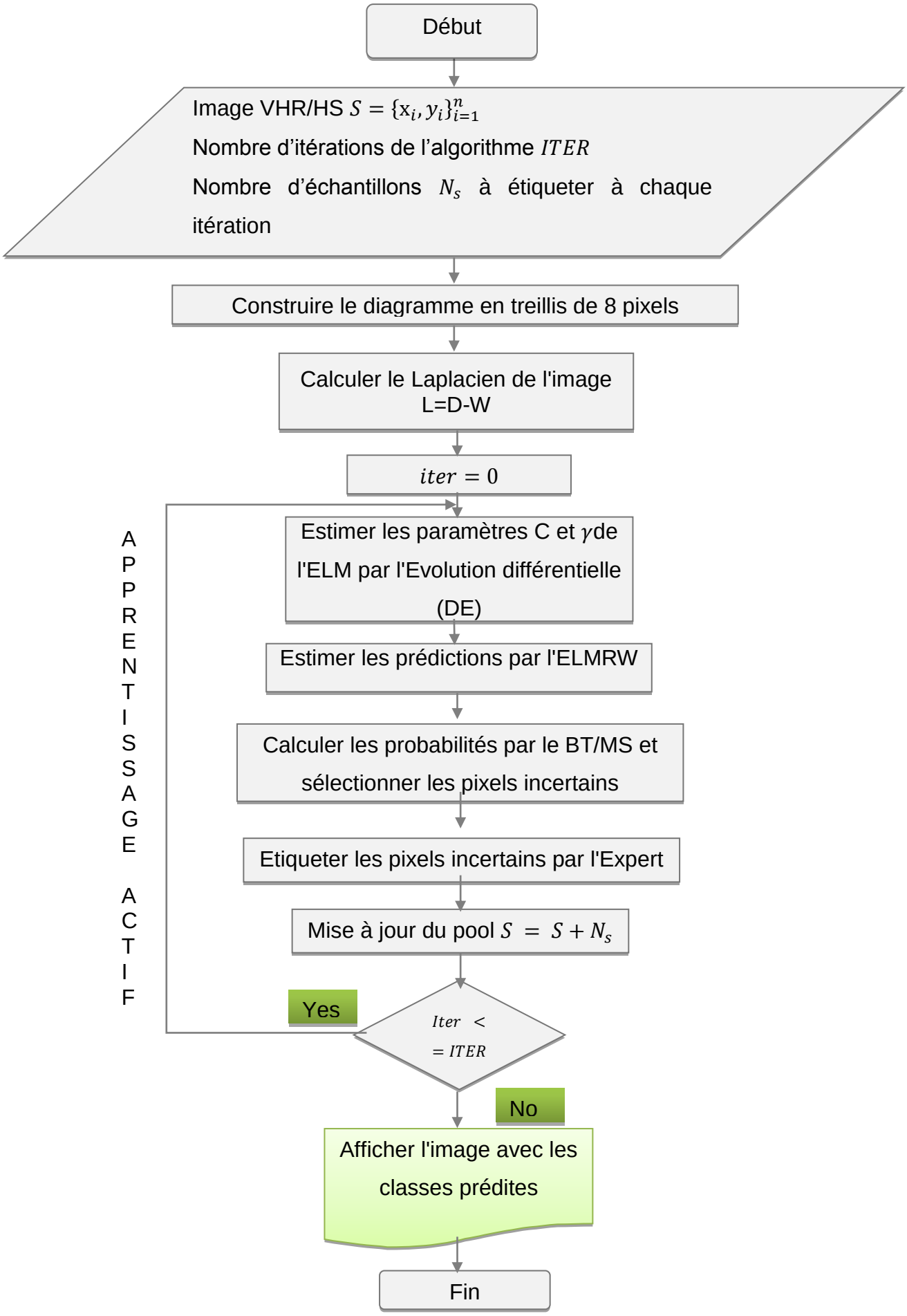
A partir des pixels triés, et selon leur degré d'appartenance par rapport à chaque classe, un nombre Ns de pixels incertains est dirigé vers l'expert humain pour les étiqueter, ces pixels sont alors ajoutés aux données d'apprentissage S.

N.B : Les pixels sont préalablement étiquetés et sont à notre disposition, mais ne sont pas présentés à l'algorithme qu'après requête, d'où le concept d'apprentissage par requêtes.

Les pixels qui portent confusion sont les plus incertains et doivent être utilisé comme données d'entrainement étiqueté dans la prochaine itération (initialement l'algorithme ne connait pas les données portant confusion), mais il les rencontre à chaque étape et forme les requêtes en vue d'un étiquetage par l'expert. C'est ce qui fait l'interaction entre algorithme (machine) et l'expert (humain), d'où le contexte d'interface homme-machine. Toutefois, nous essayons d'optimiser cette interaction en faisant varier les expériences, les ensembles de pixels initiaux d'apprentissage, etc.,…

L'ensemble du processus est réitéré jusqu'à ce qu'une condition de convergence prédéfinie soit satisfaite. L'algorithme suivant résume la stratégie d'apprentissage actif proposé : l'ELMRW-AL.

5.4.1. Organigramme de l'ELMRW utilisant l'AL



5.5. Configuration Expérimentale

Pour chaque image VHR ou HS satellitaire, nous divisons les données en deux ensembles de tailles égales, correspondant au jeu d'apprentissage U et un jeu de test. Le jeu d'apprentissage initial L (5 pixels par classe initialement) est généré aléatoirement à partir du jeu U, et sera incrémenté de $Ns = 10$ pixels à chaque itération. Puis nous exécutons le programme d'apprentissage actif sur 50 itérations en ajoutant à chaque fois 10 pixels par itération.

Pour obtenir des résultats statistiques fidèles, nous exécutons l'algorithme actif dix fois, à chaque fois avec différents jeu d'entrainement et de de test.

Les performances de classification sont présentées en fonction du taux de reconnaissance globale (OA), du taux de reconnaissance moyen (AA), ainsi que de la statistique Kappa[121] , ainsi que leurs déviations standards (σ) respectives, ces dernières mesures permettent d'évaluer la stabilité de l'algorithme d'apprentissage actif.

Dans les expériences qui suivent, et en vue de générer les attributs morphologiques (MP) des images satellitaires de Djeddah et Riyad [7], nous avons appliqué les opérateurs d'érosion et de dilatation, aux attributs spectraux originaux, l'operateur morphologique (SE) est de forme circulaire de dimensions [3, 6, 9] pixels. Ceci amène le vecteur d'attributs à 27 dimensions, comme illustré dans le tableau 6.1.

Pour l'image de l'université de Pavie, nous avons adopté la méthode proposé par [17], ou nous appliquons une analyse par composantes principales (PCA) en vue de diminuer le nombre de bandes de 103 à 5 bandes spectrales principales, qui sont renforcées par l'application du SE de (3,6 et 9 pixels), d'où un vecteur d'attributs de dimension 45, comme illustré ans le tableau 6.1.

Tableau 6.1 : Attributs des 3 images VHR/HS (Djedda, Riyad, Université De Pavie)

Images	Nombre de bandes spectrales originales	Nombre d'attributs associé à chaque pixel
Djedda (VHR)	3	$3 * (3 + 3 + 3)$ = **27**
Riyad (VHR)	3	
Université de Pavie (HS)	103 bandes réduites à 5 bandes (PCA)	$5 * (3 + 3 + 3)$ = **45**

Lors de l'entrainement de l'ELM, l'estimation des paramètres de régularisation C et γ est effectuée par une validation croisée sur trois ensembles de pixels ou validation à 3-plis [122].

Les limites de paramètres varient selon les segments $[10^{-3}..1000]$ et$[10^{-3}..10]$, respectivement, pour une recherche optimale, nous avons opté pour un modèle de sélection basé sur une optimisation utilisant l'évolution différentielle [17, 21, 123]

Pour le calcul des poids des graphes, le paramètre β est fixé à 1. Sachant que dans les expériences réalisées, ce paramètre ne s'avérait pas très critique.

Nous fixons aussi le paramètre λ à 0.01. La justification de ce choix sera discutée dans la partie des commentaires des expériences.

A des fins de comparaison, nous présentons les résultats de la méthode proposée versus le classificateur ELM de base, avec un échantillonnage aléatoire (RS) et un critère de classement de rupture des liens BT, toutefois tout autre critère d'apprentissage actif pourrait avoir été utilisée.

Toutes les expériences ont été réalisées sur un Core (TM) i7 avec une fréquence de 1.80GHz et 8.00 GB de RAM.

5.6. Résultats de la classification active avec l'ELM et l'ELMRW

Les figures 6.1 à 6.3 illustrent la reconnaissance globale versus le nombre de requêtes obtenues par les méthodes ELM-RS, ELMRW-RS, ELM-BT et ELMRW-BT sur les trois images satellitaires.

Les résultats de la classification obtenue à partir des données initiales d'apprentissage ainsi qu'après les 50 requêtes.

Pour l'image de Djeddah, L'expérience commence par 35 pixels initiaux jusqu'à atteindre les 535 pixels étiquetés.

Les résultats des différents algorithmes ELM (RS, BT), ELMRW (RS, BT) sont présentés dans le tableau 6.2 pour l'image de Djeddah, dans le tableau 6.3 pour l'image de Riyad, et dans le tableau 6.4 pour l'image de l'Université de Pavie.

Tableau 6.2 : Résultats des algorithmes de classification de l'image de Djeddah

Méthode→	ELM	ELMRW	ELM		ELMRW	
Sélection des pixels par →			RS	BT	RS	BT
Nombre de Pixels d'apprentissage (AL)	Données Initiales		Nombre de Pixels à 50 itérations			
	35		535			
OA (%)	83.81	86.75	95.75	98.12	97.45	**99.87**
σ_{OA}	2.61	2.84	0.36	0.17	0.30	**0.15**
AA (%)	81.12	83.92	91.36	96.73	93.92	**99.57**
σ_{AA}	3.01	3.59	1.88	1.06	2.10	**0.66**
Kappa	0.798	0.834	0.946	0.976	0.967	**0.998**
σ_{Kappa}	0.030	0.034	0.004	0.002	0.003	**0.001**

Les résultats des expériences intermédiaires, de l'OA par requêtes sont illustrés dans la figure 6.1 pour l'image satellitaire de Djeddah.

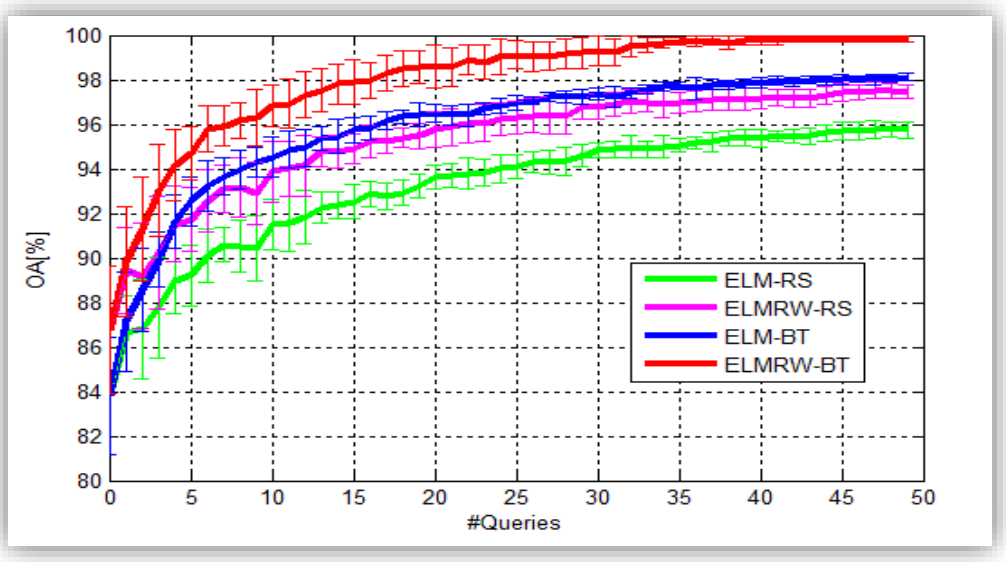


Figure 6.1 : Le taux de reconnaissance global (OA) versus le nombre de requêtes pour l'image de Djeddah.

Pour l'image de Riyad, L'expérience commence par 40 pixels initiaux jusqu'à atteindre les 540 pixels étiquetés.

Tableau 6.3 : Résultats des algorithmes de la classification de l'image de Riyad

Méthode→	ELM	ELMRW	ELM		ELMRW	
Sélection des pixels par →			RS	BT	RS	BT
Nombre de Pixels d'apprentissage (AL)	Données Initiales		Nombre de Pixels à 50 itérations			
	40		540			
OA (%)	78.23	81.26	93.55	96.22	96.88	**99.78**
σ_{OA}	3.96	4.62	0.38	0.13	0.56	**0.31**
AA (%)	82.84	86.04	93.23	96.49	96.60	**99.64**
σ_{AA}	2.14	2.42	0.63	0.28	0.67	**0.54**
Kappa	0.733	0.770	0.919	0.953	0.961	**0.997**
σ_{Kappa}	0.046	0.053	0.004	0.001	0.007	**0.003**

Les résultats des expériences intermédiaires, de l'OA par requêtes sont illustrés dans la figure 6.2 pour Riyad.

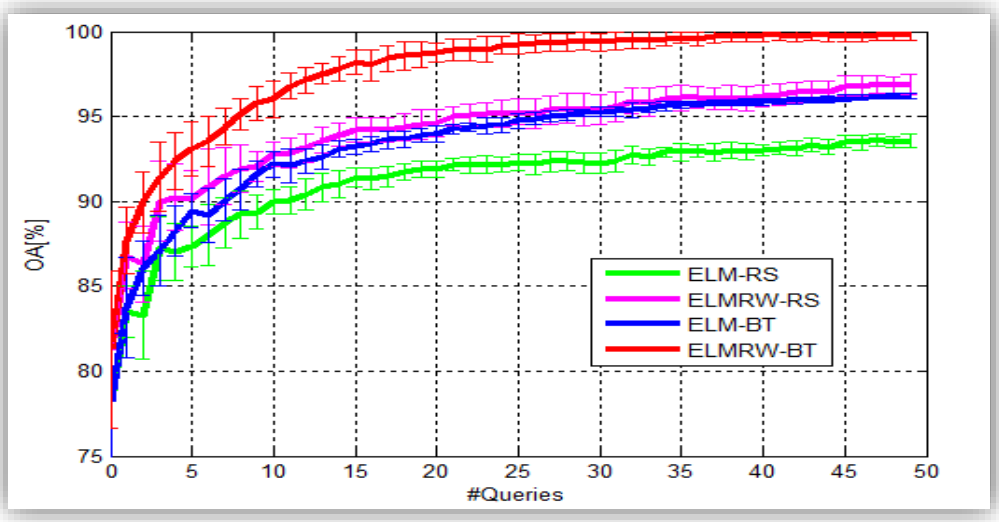


Figure 6.2 : Le taux de reconnaissance global (OA) versus le nombre de requêtes pour l'image satellitaire de Riyad.

Pour l'image de l'Université de Pavie, L'expérience commence par 45 pixels initiaux jusqu'à atteindre les 545 pixels étiquetés.

Tableau 6.4 : Résultats des algorithmes de la classification de l'image de l'Université de Pavie

Méthode→	ELM	ELMRW	ELM		ELMRW	
Sélection des pixels par →			RS	BT	RS	BT
Nombre de Pixels d'apprentissage (AL)	Données Initiales		Nombre de Pixels à 50 itérations			
	45		545			
OA (%)	75.30	81.44	97.47	99.77	98.17	**99.85**
σ_{OA}	6.55	7.20	0.43	0.032	0.19	**0.032**
AA (%)	80.79	84.15	96.94	99.68	97.18	**99.76**
σ_{AA}	4.97	5.50	0.58	0.068	0.26	**0.060**
Kappa	0.683	0.759	0.966	0.997	0.975	**0.998**
σ_{Kappa}	0.080	0.091	0.005	0	0.002	**0**

Les résultats des expériences intermédiaires, de l'OA par requêtes sont illustrés dans la figure 6.3 pour l'Université de Pavie.

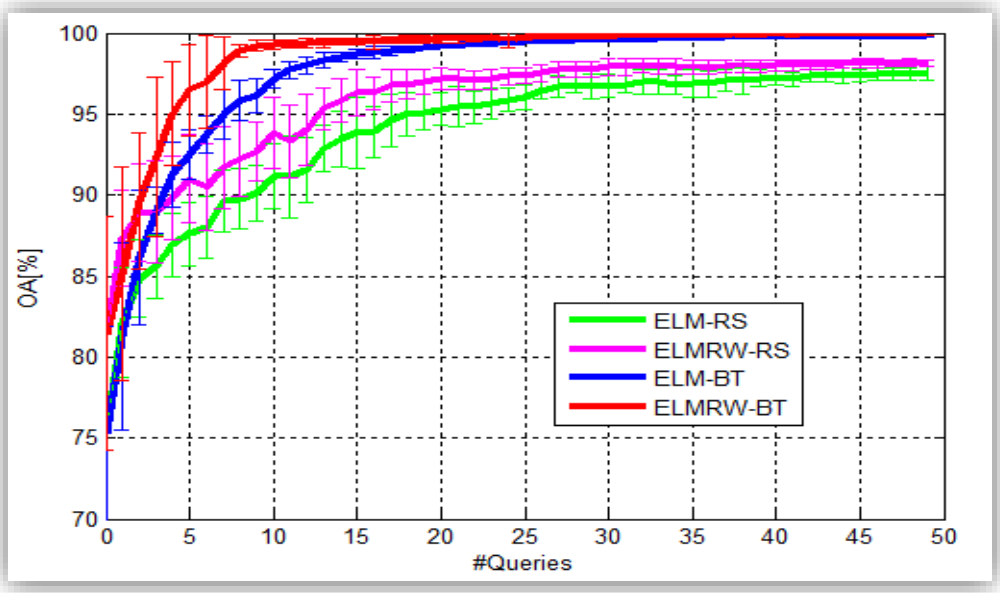


Figure 6.3 : Le taux de reconnaissance global (OA) versus le nombre de requêtes pour l'image satellitaire de l'Université de Pavie.

5.7. Discussion des résultats des requêtes de l'AL

En se basant sur les figures 6.1 à 6.3, nous remarquons l'avantage du critère de classement BT (Les deux plus proches probabilités) sur la sélection aléatoire des pixels RS. Dans les deux cas, les taux de reconnaissance du ELMRW sont clairement meilleures que l'ELM pour toutes les sélections du BT et du RS.

Parmi les scenarios investis, le ELMRW-BT est le meilleur scenario, le plus précis et le plus stable, ajouter à cela sa convergence rapide comparé aux autres scenarios.

Pour l'image de Djeddah, les valeurs obtenues, pour l'ELM et le ELMRW, de (OA, AA, Kappa) sont égales à (83.81%, 81.12%, 0.798) et (86.75%, 83.92%, 0.834), respectivement.

Pour l'image de Riyad, les valeurs obtenues pour l'ELM et le ELMRW, de (OA, AA, Kappa) sont égales à (78.23%, 82.84%, 0.733) et (81.26%, 86.04%, 0.770), respectivement.

Nous pouvons remarquer que les améliorations obtenues par l'ELMRW par rapport à l'ELM de base, sont de l'ordre de 4%, en termes de taux de reconnaissance, avec une augmentation du coefficient d'accord Kappa.

Après 50 requêtes, l'ELMRW-BT génère les valeurs de (OA, AA, Kappa) de (99.87%, 99.57%, 0.998), (99.78%, 99.64%, 0.997), et (99.85%, 99.76%, 0.998) pour les images de Djeddah, Riyad et l'Université de Pavie respectivement. Tandis que l' ELM-BT génère des valeurs des (OA, AA, Kappa) égales à (98.12%, 96.73%, 0.976), (96.22%, 96.49%, 0.953) et (99.77%, 99.68%, 0.997) pour les trois images respectivement.

De ces résultats, nous remarquons que l'ELMRW-BT donne de meilleurs résultats de classification pour les images de Djeddah et Riyad.

Toutefois, les résultats de l'image de l'Université de Pavie sont presque similaires pour l'ELMRW-BT et l'ELM-BT, toutefois nous remarquons la convergence rapide de l'ELMRW-BT par rapport à l'ELM-BT (approximativement 10 contre 25 requêtes)

En termes de vitesse de calcul, l'ELMRW ajoute un temps additionnel de 4, 6, et 2 secondes pour les images de Djeddah, Riyad, et l'Université de Pavie à chaque itération de l'algorithme du processus actif de l'ELM.

5.8. Analyse de sensibilité du paramètre de fidélité

Comme précédemment cité dans le chapitre 4, dans l'équation 4.23, le paramètre λ renforce la fidélité des données en encodant une estimation initiale de la structure de l'image. Toutefois, c'est un paramètre de régularisation, donc sa variation aura un impact sur les pixels étiquetés décrits par l'équation 4.30

Les figures 6.4, à 6.6 montrent les résultats de la classification obtenues par variation du paramètre λ sur l'image de Djeddah, Riyad et l'Université de Pavie, respectivement.

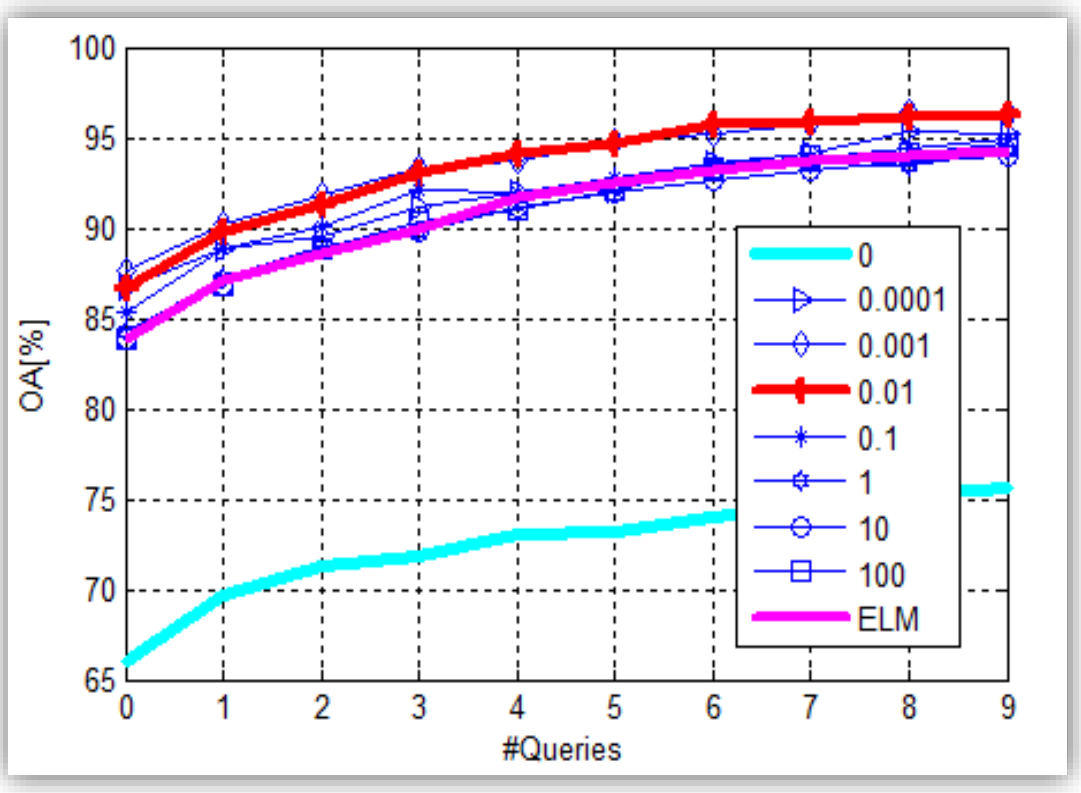


Figure 6.4 : Variation de l'OA en fonction de λ, (image de Djeddah)

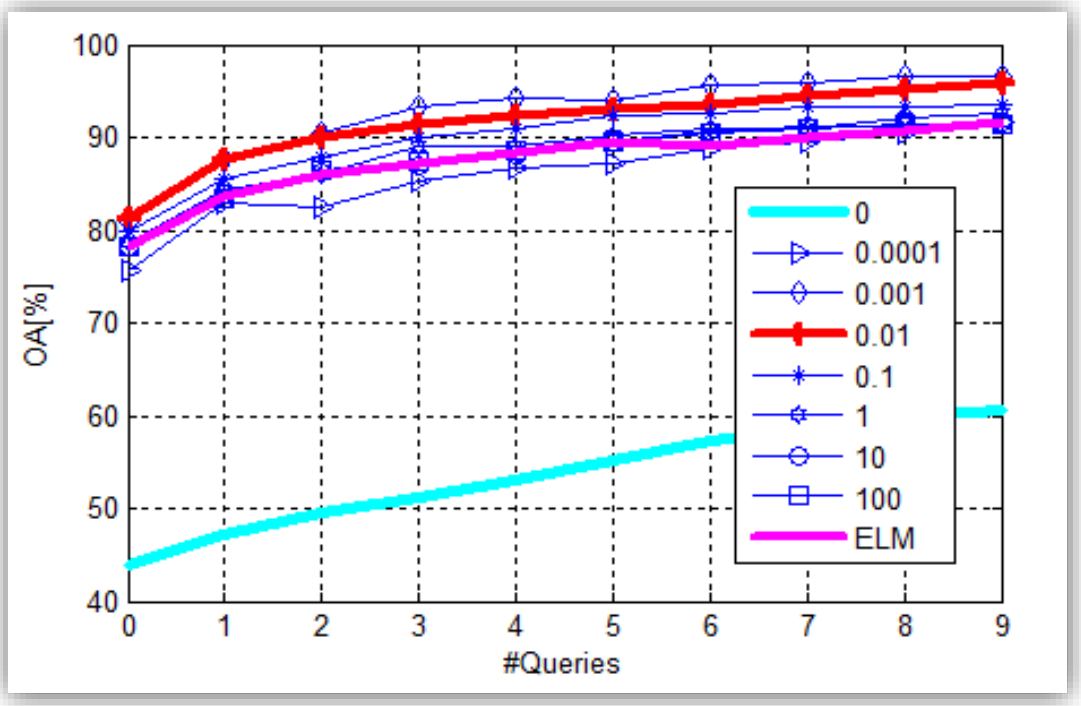


Figure 6.5 : Variation de l'OA en fonction de λ, (image de Riyadh)

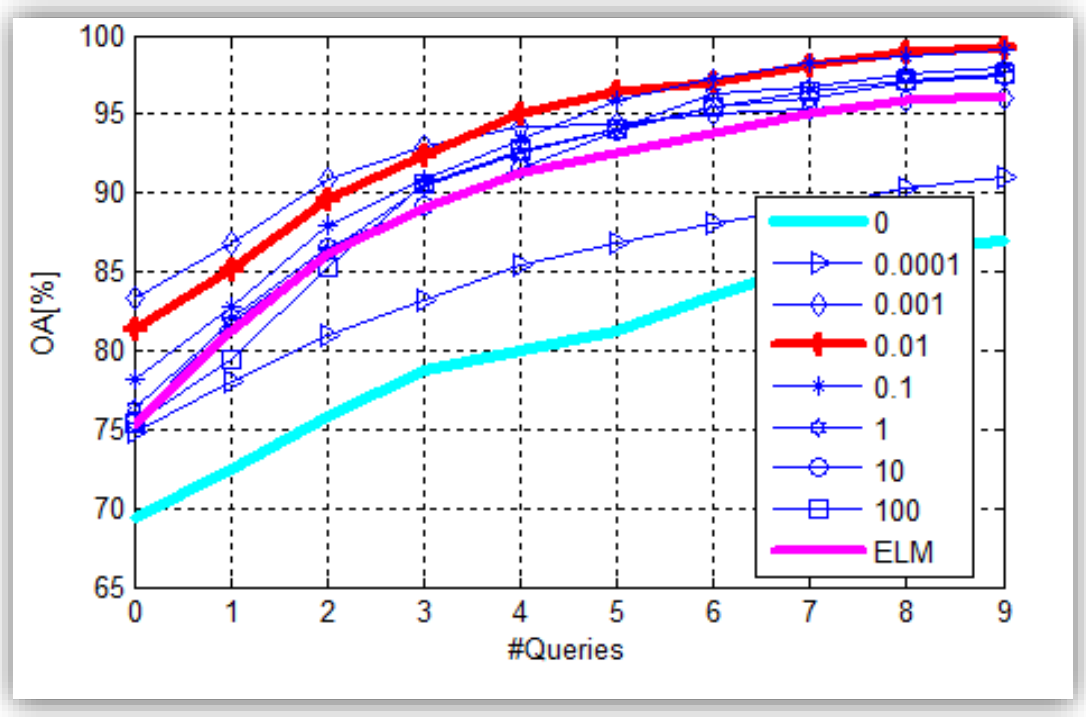


Figure 6.6 : Variation de l'OA en fonction de λ, (image de l'Université de Pavie)

Nous observons des figures 6.4 à 6.6 ; après rejet de l'information préalable apportée par l'ELM (c.à.d.$\lambda = 0$) ; Cas particulier ou l'ELM n'initialise plus l'ELMRW, que ce dernier fournit de très mauvais résultats. Dans ce cas, nous pouvons remarquer que l'ELMRW réagit ou se comporte comme l'algorithme standard de la marche aléatoire, et seulement les pixels d'apprentissage sont considérés comme information préalable pour estimer les étiquettes des pixels non-étiquetés.

En augmentant les valeurs de λ, l'ELMRW bénéficie des conditions initiales générées par l'ELM, pour donner de meilleurs taux de classification comparés à l'ELM de base. Nous observons que fixer la valeur de $\lambda = 0.01$, résulte en un comportement plus stable, alors qu'en augmentant les valeurs de λ, le ELMRW tend à générer des valeurs proches de l'ELM.

5.9. Comparaison de notre classification à l'état de l'art

En vue de positionner notre méthode par rapport aux travaux existants, une comparaison avec différents travaux de l'état de l'art relatifs à la classification de l'image de l'université de Pavie sont illustrés dans le Tableau 6.5.

Notons toutefois que différentes configurations expérimentales ont été élaborées dans chaque travail, ce qui complique l'unification d'une méthode de comparaison, toutefois, nous comparons la classification finale de l'image de l'université de Pavie, avec des réserves sur la méthodologie de chaque auteur.

Tableau 6.5 : Comparatif à l'état de l'art de la classification de l'image de l'Université de Pavie

Méthodes	OA (%)	AA (%)	Kappa
DBFE [114] en 2005	89.9	96.6	-
SVA [117] en 2011	94.47	96.58	0.9280
SVM [115] en 2012	92.97	93.97	-
MLR (BT) [113] en 2013	84.07±1.52	87.39±1.25	0.7694±0.0542
EBO [116] en 2013	90.17	-	-
WLC-GA [112] en 2013	97.99	96.42	-
SUnSAL$_{EMAP}$ [118] en 2014	90.87	93.60	0.8826
Méthode proposée : ELMRW [18]	**99.85±0.032**	**99.76±0.060**	**0.998**

Le tableau 6.5 montre que notre méthode proposée arrive à des taux de classifications supérieures à celles de l'état de l'art, par la nouvelle formulation de l'ELMRW.

5.10. Conclusion

Ce dernier chapitre est la synthèse de notre travail sur l'apprentissage actif et son intégration avec la machine d'apprentissage extrême et la marche aléatoire, ainsi que la présentation des résultats sur trois images satellitaires, notamment l'image HS de Djeddah et Riyad et enfin l'image VHR de l'Université de Pavie en Italie.

La méthode proposée a prouvé sa capacité à étiqueter les images en question après seulement 50 requêtes, en utilisant l'apprentissage actif basé sur le critère de sélection BT.

Les différents paramètres dont dépend notre fonctionnelle ont fait état d'une étude particulière, en vue de cibler les valeurs optimales à la classification, c'est ce qui fait l'essence même de trouver une solution selon des contraintes diverses.

CONCLUSION

Dans cette thèse, nous avons proposé une nouvelle méthode appelée "ELMRW" pour la classification d'images satellitaires. Cet algorithme est basé sur l'apprentissage actif, en formulant une fonctionnelle d'énergie intégrant à la fois : la matrice du Laplacien, les classes prédites par l'algorithme de la machine d'apprentissage extrême(ELM), ainsi que l'algorithme de la marche aléatoire(RW).

Cette nouvelle formulation présente deux propriétés intéressantes :

- Elle est intrinsèquement formulée comme un problème de classification, donc son extension vers des problèmes multi-classes, est initialement considérée.
- La solution de ce problème d'optimisation conduit à un système d'équations linéaires creux, qui peut être résolu efficacement dans un temps linéaire en utilisant des solveurs de matrice creuse.

Les résultats de la classification, de trois images satellitaires différentes (Djeddah, Riyad, et Université de Pavie), montrent que la solution proposée est stable, et peut réduire considérablement l'interaction avec l'expert ; Tout en maximisant le gain en précision par rapport aux résultats fournis par le classificateur ELM de base. Ce qui est l'essence même de l'intégration de l'algorithme d'apprentissage actif, qui tend à réduire le cout d'expertise tout en maximisant les taux de reconnaissance, par des méthodes mathématiquement solvables, en des temps minimaux.

En guise de comparaison, avec l'état de l'art, nous avons opté pour l'image satellitaire "**référence**" de l'Université de Pavie en Italie, ou notre nouvel algorithme ELMRW a montré sa supériorité, avec un taux de global de classification des pixels de 99.85%±0.032, avec un facteur de kappa de 99.8%.

Les développements futurs de ce travail peuvent être définis dans plusieurs directions, notamment investir sur d'autres modèles à base de graphes, en vue d'enrichir la formulation mathématique de la fonctionnelle énergétique, et définir de nouveaux critères de sélection active, particulièrement adapté à la nouvelle configuration.

APPENDICE

APPENDICE A

A. LISTE DES SYMBOLES ET ABREVIATIONS

AA	:	Average Accuracy	Taux de reconnaissance moyen
AL	:	Active Learning	Apprentissage actif
BS	:	Batch Selection	Sélection en lot
BT	:	Breaking Ties	Rupture des liens
DBM	:	Deep Belief networks	Réseaux deep belief
DBM	:	Deep Boltzman Machines	Machines Deep de Boltzman
ELM	:	Extreme learning Machine	Machine d’apprentissage extrême
GT	:	Ground Truth	Valeurs étiquetés
HS	:	HyperSpectral	Hyper spectrale
Mincut	:	Minimum cut	Coupure minimale
NN	:	Neural networks	Réseaux de Neurones (RN)
OA	:	Overall Accuracy	Taux de reconnaissance global
PCA	:	Principal Component Analysis	Analyse en composante principale
ROSIS	:	Reflective Optics System Imaging Spectrometer	Système Optique Réfléchissant pour imagerie Spectrométrique
RS	:	Random Sampling	Choix aléatoire
RW	:	Random Walk	Marche aléatoire
SAE	:	Stacked Auto Encodeurs	Auto-encodeurs empilés
SDAE	:	Stacked Denoising Auto-Encoders	Auto-encodeurs empilés debruitants
SE	:	Structural Element	Elément structurel
SLFN	:	Self-Learning Feed-forward Networks	Réseaux de neurones Feed-forward
SSL	:	Semi-Supervised Learning	Apprentissage semi-supervisé
SVD	:	Singular Value Décomposition	Décomposition en Valeurs singulières
SVM	:	Support Vector Machines	Machines à vecteurs supports
VHR	:	Very High Resolution	Trés haute resolution

APPENDICE B

B. CAPTEURS SPECTROMETRIQUES

Spectromètre ROSIS: Reflective Optics System Imaging Spectrometer

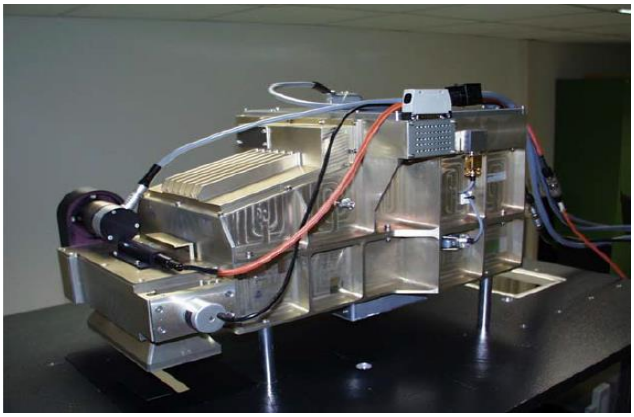


Figure A.1. Spectromètre ROSIS

Valeurs mesurées :

- Réflectance spectrale
- Images multi spectrales

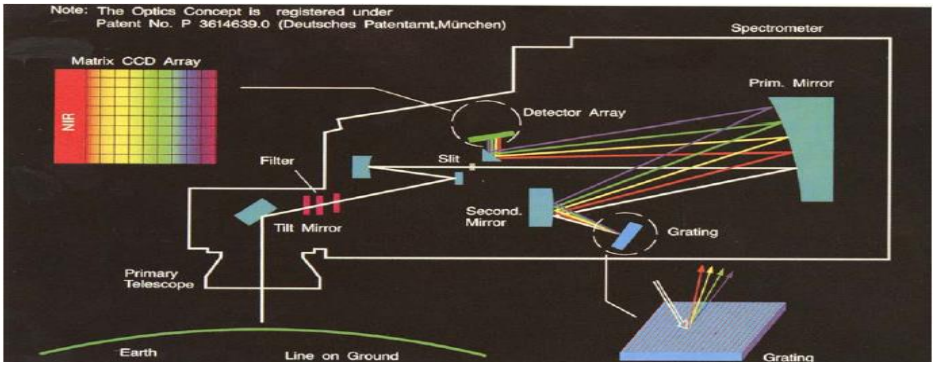


Figure A.2. Système optique de ROSIS aéroporté

Utilisation de ROSIS

L'utilisation de Rosis est réalisée dans les activités de recherche pertinentes de l'Institut de technologie de télédétection et le Centre des données de télédétection allemand au DLR Centre Oberpfaffenhofen. Il peut être utilisé pour surveiller la qualité de l'eau au moyen des substances contenues dans celui-ci, l'utilisation des terres, la couverture terrestre (indices de végétation), les dommages apportés aux forêts, la désertification, l'analyse des aérosols etc…

Spectromètre IKONOS :

Tableau A.1 : Tableau comparatif de différents spectromètres en télédétection

Satellite sensor; Provider	Spectral bands	Spatial resolution / swath width (at nadir)	Average revisiting time; off-track viewing angle	Price per km² (USD)
IKONOS Space Imaging	Panchromatic: 450-900 nm 445-516 nm 506-595 nm 632-698 nm 757-853 nm	0.81 m / 11 km 3.2 m 3.2 m 3.2 m 3.2 m	2-3 days ±30°	~ 18
QuickBird DigitalGlobe	Panchromatic: 450-900 nm 450-520 nm 520-600 nm 630-690 nm 760-900 nm	0.61 m / 16.5 km 2.44 m 2.44 m 2.44 m 2.44 m	1-3.5 days ±30°	~ 24
OrbView-3 ORBIMAGE	Panchromatic: 450-900 nm 450-520 nm 520-600 nm 625-695 nm 760-900 nm	1 m / 8 km 4 m 4 m 4 m 4 m	< 3 days ±50°	~ 20 (unconfirmed)
SPOT-5 Spot Image	Panchromatic: 480-710 nm 500-590 nm 610-680 nm 780-890 nm 1580-1750 nm	2.5, 5 m / 60 km 10 m 10 m 10 m 20 m	5 days ±27°	~ 1.2

Toutes informations additionnelles sur IKONOS peuvent être récupérées sur :

http://folk.uio.no/kaeaeb/publications/huggel_earsel05.pdf

REFERENCES

[1] T. Luo, K. Kramer, S. Samson, A. Remsen, D. Goldgof, L. Hall, and T. Hopkins, "Active learning to recognize multiple types of plankton," in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, 2004, pp. 478-481.

[2] E. Pasolli and F. Melgani, "Active learning methods for electrocardiographic signal classification," *Information Technology in Biomedicine, IEEE Transactions on,* vol. 14, pp. 1405-1416, 2010.

[3] G. Riccardi and D. Hakkani-Tur, "Active learning: Theory and applications to automatic speech recognition," *Speech and Audio Processing, IEEE Transactions on,* vol. 13, pp. 504-511, 2005.

[4] M. Sugiyama and N. Rubens, "A batch ensemble approach to active learning with model selection," *Neural Networks,* vol. 21, pp. 1278-1286, 2008.

[5] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," *The Journal of Machine Learning Research,* vol. 2, pp. 45-66, 2002.

[6] D. A. Cohn, Z. Ghahramani, and M. I. Jordan, "Active learning with statistical models," *Journal of artificial intelligence research,* 1996.

[7] M. M. Crawford, D. Tuia, and H. L. Yang, "Active learning: Any value for classification of remotely sensed data?," *Proceedings of the IEEE,* vol. 101, pp. 593-608, 2013.

[8] S. Rajan, J. Ghosh, and M. M. Crawford, "An active learning approach to hyperspectral data classification," *Geoscience and Remote Sensing, IEEE Transactions on,* vol. 46, pp. 1231-1242, 2008.

[9] D. Tuia, F. Ratle, F. Pacifici, M. F. Kanevski, and W. J. Emery, "Active learning methods for remote sensing image classification," *Geoscience and Remote Sensing, IEEE Transactions on,* vol. 47, pp. 2218-2232, 2009.

[10] O. Chapelle, B. Schölkopf, and A. Zien, "Semi-supervised learning," 2006.

[11] X. Zhu and A. B. Goldberg, "Introduction to semi-supervised learning," *Synthesis lectures on artificial intelligence and machine learning,* vol. 3, pp. 1-130, 2009.

[12] B. Settles, "Active learning literature survey," *University of Wisconsin, Madison,* vol. 52, p. 11, 2010.

[13] J. Li, J. M. Bioucas-Dias, and A. Plaza, "Semisupervised hyperspectral image classification using soft sparse multinomial logistic regression," *Geoscience and Remote Sensing Letters, IEEE,* vol. 10, pp. 318-322, 2013.

[14] X. Zhu, J. Lafferty, and R. Rosenfeld, "Semi-supervised learning with graphs," Carnegie Mellon University, Language Technologies Institute, School of Computer Science, 2005.

[15] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *The Journal of Machine Learning Research,* vol. 7, pp. 2399-2434, 2006.

[16] E. Pasolli, F. Melgani, D. Tuia, F. Pacifici, and W. J. Emery, "SVM active learning approach for image classification using spatial information," *Geoscience and Remote Sensing, IEEE Transactions on,* vol. 52, pp. 2217-2233, 2014.

[17] Y. Bazi, N. Alajlan, F. Melgani, H. AlHichri, S. Malek, and R. R. Yager, "Differential evolution extreme learning machine for the classification of hyperspectral images," *Geoscience and Remote Sensing Letters, IEEE,* vol. 11, pp. 1066-1070, 2014.

[18] M. A. Bencherif, Y. Bazi, A. Guessoum, N. Alajlan, F. Melgani, and H. AlHichri, "Fusion of Extreme Learning Machine and Graph-Based Optimization Methods for Active Classification of Remote Sensing Images," 2015.

[19] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: a new learning scheme of feedforward neural networks," in *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, 2004, pp. 985-990.

[20] Y. Lan, Y. C. Soh, and G.-B. Huang, "Ensemble of online sequential extreme learning machine," *Neurocomputing,* vol. 72, pp. 3391-3395, 2009.

[21] Q.-Y. Zhu, A. K. Qin, P. N. Suganthan, and G.-B. Huang, "Evolutionary extreme learning machine," *Pattern recognition,* vol. 38, pp. 1759-1763, 2005.

[22] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on,* vol. 42, pp. 513-529, 2012.

[23] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing,* vol. 70, pp. 489-501, 2006.

[24] G.-B. Huang and C.-K. Siew, "Extreme learning machine with randomly assigned RBF kernels," *International Journal of Information Technology,* vol. 11, pp. 16-24, 2005.

[25] G.-B. Huang, D. H. Wang, and Y. Lan, "Extreme learning machines: a survey," *International Journal of Machine Learning and Cybernetics,* vol. 2, pp. 107-122, 2011.

[26] Y. Lan, Z. Hu, Y. C. Soh, and G.-B. Huang, "An extreme learning machine approach for speaker recognition," *Neural Computing and Applications,* vol. 22, pp. 417-425, 2013.

[27] R. Zhang, G.-B. Huang, N. Sundararajan, and P. Saratchandran, "Multicategory classification using an extreme learning machine for microarray gene expression cancer diagnosis," *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB),* vol. 4, pp. 485-495, 2007.

[28] T. Helmy and Z. Rasheed, "Multi-category bioinformatics dataset classification using extreme learning machine," in *Evolutionary Computation, 2009. CEC'09. IEEE Congress on*, 2009, pp. 3234-3240.

[29] Y. Miche, A. Sorjamaa, P. Bas, O. Simula, C. Jutten, and A. Lendasse, "OP-ELM: optimally pruned extreme learning machine," *Neural Networks, IEEE Transactions on,* vol. 21, pp. 158-162, 2010.

[30] B. Luo, J. Chanussot, S. Douté, and L. Zhang, "Empirical automatic estimation of the number of endmembers in hyperspectral images," *Geoscience and Remote Sensing Letters, IEEE,* vol. 10, pp. 24-28, 2013.

[31] D. B. Heras, F. Argüello, and P. Quesada-Barriuso, "Exploring ELM-based spatial–spectral classification of hyperspectral images," *International Journal of Remote Sensing,* vol. 35, pp. 401-423, 2014.

[32] R. Moreno, F. Corona, A. Lendasse, M. Graña, and L. S. Galvão, "Extreme learning machines for soybean classification in remote sensing hyperspectral images," *Neurocomputing,* vol. 128, pp. 207-216, 2014.
[33] C. Chen, W. Li, H. Su, and K. Liu, "Spectral-spatial classification of hyperspectral image based on kernel extreme learning machine," *Remote Sensing,* vol. 6, pp. 5795-5814, 2014.
[34] N. Alajlan, Y. Bazi, F. Melgani, and R. R. Yager, "Fusion of supervised and unsupervised learning for improved classification of hyperspectral images," *Information Sciences,* vol. 217, pp. 39-55, 2012.
[35] J. Gantz and D. Reinsel, "The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east," *IDC iView: IDC Analyze the Future,* vol. 2007, pp. 1-16, 2012.
[36] M. Anthony and J. Shawe-Taylor, "A result of Vapnik with applications," *Discrete Applied Mathematics,* vol. 47, pp. 207-217, 1993.
[37] D. Haussler, "Decision theoretic generalizations of the PAC model for neural net and other learning applications," *Information and computation,* vol. 100, pp. 78-150, 1992.
[38] D. H. Wolpert and R. Waters, "The relationship between PAC, the statistical physics framework, the Bayesian framework, and the VC framework," in *In*, 1994.
[39] J. Ratsaby and V. Maiorov, "On the value of partial information for learning from examples," *Journal of Complexity,* vol. 13, pp. 509-543, 1997.
[40] M. A. Figueiredo and A. K. Jain, "Unsupervised learning of finite mixture models," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 24, pp. 381-396, 2002.
[41] S. Y. Ho and S. H. Kwok, "The attraction of personalized service for users in mobile commerce: an empirical study," *ACM SIGecom Exchanges,* vol. 3, pp. 10-18, 2002.
[42] A. M. Kaplan, "If you love something, let it go mobile: Mobile marketing and mobile social media 4x4," *Business Horizons,* vol. 55, pp. 129-139, 2012.
[43] V. Vandewalle, "Estimation et sélection en classification semi-supervisée," Université des Sciences et Technologie de Lille-Lille I, 2009.
[44] J. Holmgren and Å. Persson, "Identifying species of individual trees using airborne laser scanner," *Remote Sensing of Environment,* vol. 90, pp. 415-423, 2004.
[45] C. A. de Sousa, V. Souza, and G. E. Batista, "Time Series Transductive Classification on Imbalanced Data Sets: An Experimental Study," in *Pattern Recognition (ICPR), 2014 22nd International Conference on*, 2014, pp. 3780-3785.
[46] S. De Vito, G. Fattoruso, M. Pardo, F. Tortorella, and G. Di Francia, "Semi-supervised learning techniques in artificial olfaction: A novel approach to classification problems and drift counteraction," *Sensors Journal, IEEE,* vol. 12, pp. 3215-3224, 2012.
[47] T. Joachims, "Transductive inference for text classification using support vector machines," in *ICML*, 1999, pp. 200-209.
[48] D. Banks and R. Olszewski, "Estimating local dimensionality," in *Proceedings of the Statistical Computing Section of the American Statistical Society. ASA*, 1997, pp. 159-197.

[49] T. Danisman, I. M. Bilasco, J. Martinet, and C. Djeraba, "Intelligent pixels of interest selection with application to facial expression recognition using multilayer perceptron," *Signal Processing,* vol. 93, pp. 1547-1556, 2013.
[50] J. Korczak and A. Quirin, "Evolutionary approach to discovery of classification rules from remote sensing images," in *Applications of Evolutionary Computing*, ed: Springer, 2003, pp. 388-398.
[51] Y. Zhang, X. Liao, and L. Carin, "Detection of buried targets via active selection of labeled data: Application to sensing subsurface UXO," *Geoscience and Remote Sensing, IEEE Transactions on,* vol. 42, pp. 2535-2543, 2004.
[52] Q. Liu, X. Liao, and L. Carin, "Detection of unexploded ordnance via efficient semisupervised and active learning," *Geoscience and Remote Sensing, IEEE Transactions on,* vol. 46, pp. 2558-2567, 2008.
[53] J. Li, J. M. Bioucas-Dias, and A. Plaza, "Semisupervised hyperspectral image segmentation using multinomial logistic regression with active learning," *Geoscience and Remote Sensing, IEEE Transactions on,* vol. 48, pp. 4085-4098, 2010.
[54] E. Pasolli, F. Melgani, and Y. Bazi, "Support vector machine active learning through significance space construction," *IEEE Geoscience and Remote Sensing Letters,* vol. 8, pp. 431-435, 2011.
[55] D. Hakkani-Tur, G. Riccardi, and A. Gorin, "Active learning for automatic speech recognition," in *Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*, 2002, pp. IV-3904-IV-3907.
[56] D. Yu, B. Varadarajan, L. Deng, and A. Acero, "Active learning and semi-supervised learning for speech recognition: A unified framework using the global entropy reduction maximization criterion," *Computer Speech & Language,* vol. 24, pp. 433-444, 2010.
[57] S. Tong and E. Chang, "Support vector machine active learning for image retrieval," in *Proceedings of the ninth ACM international conference on Multimedia*, 2001, pp. 107-118.
[58] A. K. McCallumzy and K. Nigamy, "Employing EM and pool-based active learning for text classification," in *Machine Learning: Proceedings of the Fifteenth International Conference, ICML*, 1998.
[59] S. C. Hoi, R. Jin, J. Zhu, and M. R. Lyu, "Batch mode active learning and its application to medical image classification," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 417-424.
[60] L. Grady, "Random walks for image segmentation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 28, pp. 1768-1783, 2006.
[61] A. Top, G. Hamarneh, and R. Abugharbieh, "Active learning for interactive 3D image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2011*, ed: Springer, 2011, pp. 603-610.
[62] R. Burbidge, J. J. Rowland, and R. D. King, "Active learning for regression based on query by committee," in *Intelligent Data Engineering and Automated Learning-IDEAL 2007*, ed: Springer, 2007, pp. 209-218.
[63] J.-H. Lee and Y.-C. Hsueh, "Texture classification method using multiple space filling curves," *Pattern Recognition Letters,* vol. 15, pp. 1241-1244, 1994.

[64] E. Nowak, F. Jurie, and B. Triggs, "Sampling strategies for bag-of-features image classification," in *Computer Vision–ECCV 2006*, ed: Springer, 2006, pp. 490-503.
[65] M. Li and I. K. Sethi, "Confidence-based active learning," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 28, pp. 1251-1261, 2006.
[66] J. Li, J. M. Bioucas-Dias, and A. Plaza, "Hyperspectral image segmentation using a new Bayesian approach with active learning," *Geoscience and Remote Sensing, IEEE Transactions on,* vol. 49, pp. 3947-3960, 2011.
[67] K. Brinker, "Incorporating diversity in active learning with support vector machines," in *ICML*, 2003, pp. 59-66.
[68] M. Kalakech, "Sélection semi-supervisée d'attributs: Application à la classification de textures couleur," Thèse de doctorat, Université Lille 1, Sciences et Technologies, 2011.
[69] Z. Wu and R. Leahy, "An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 15, pp. 1101-1113, 1993.
[70] C. Bilgin, C. Demir, C. Nagi, and B. Yener, "Cell-graph mining for breast tissue modeling and classification," in *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*, 2007, pp. 5311-5314.
[71] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *Applied and Computational Harmonic Analysis,* vol. 30, pp. 129-150, 2011.
[72] D. Conte, P. Foggia, C. Sansone, and M. Vento, "Thirty years of graph matching in pattern recognition," *International journal of pattern recognition and artificial intelligence,* vol. 18, pp. 265-298, 2004.
[73] G. Camps-Valls, T. Bandos Marsheva, and D. Zhou, "Semi-supervised graph-based hyperspectral image classification," *Geoscience and Remote Sensing, IEEE Transactions on,* vol. 45, pp. 3044-3054, 2007.
[74] L. Gómez-Chova, G. Camps-Valls, J. Munoz-Mari, and J. Calpe, "Semisupervised image classification with Laplacian support vector machines," *Geoscience and Remote Sensing Letters, IEEE,* vol. 5, pp. 336-340, 2008.
[75] B. Sirmacek and C. Unsalan, "Urban-area and building detection using SIFT keypoints and graph theory," *Geoscience and Remote Sensing, IEEE Transactions on,* vol. 47, pp. 1156-1167, 2009.
[76] M. Ahmadlou, H. Adeli, and A. Adeli, "New diagnostic EEG markers of the Alzheimer's disease using visibility graph," *Journal of neural transmission,* vol. 117, pp. 1099-1109, 2010.
[77] A. Statnikov, C. F. Aliferis, I. Tsamardinos, D. Hardin, and S. Levy, "A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis," *Bioinformatics,* vol. 21, pp. 631-643, 2005.
[78] T. Korenius, J. Laurikkala, and M. Juhola, "On principal component analysis, cosine and Euclidean measures in information retrieval," *Information Sciences,* vol. 177, pp. 4893-4905, 2007.
[79] Y. Zhang, J. Callan, and T. Minka, "Novelty and redundancy detection in adaptive filtering," in *Proceedings of the 25th annual international ACM*

SIGIR conference on Research and development in information retrieval, 2002, pp. 81-88.

[80] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Computer Vision, 1998. Sixth International Conference on*, 1998, pp. 839-846.

[81] V. Baladi and B. Vallée, "Euclidean algorithms are Gaussian," *Journal of Number Theory,* vol. 110, pp. 331-386, 2005.

[82] J. A. Bondy and U. S. R. Murty, *Graph theory with applications* vol. 290: Macmillan London, 1976.

[83] N. B. Ellison, "Social network sites: Definition, history, and scholarship," *Journal of Computer-Mediated Communication,* vol. 13, pp. 210-230, 2007.

[84] J. Weng and B.-S. Lee, "Event Detection in Twitter," *ICWSM,* vol. 11, pp. 401-408, 2011.

[85] S. L. a. M. Lipson, *Schaum's Outlines: Discrete Mathematics*, 3rd ed.: McGraw-Hill, 2007.

[86] A. Blum and S. Chawla, "Learning from labeled and unlabeled data using graph mincuts," 2001.

[87] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing,* vol. 17, pp. 395-416, 2007.

[88] M.-B. Li, G.-B. Huang, P. Saratchandran, and N. Sundararajan, "Fully complex extreme learning machine," *Neurocomputing,* vol. 68, pp. 306-314, 2005.

[89] G.-B. Huang and L. Chen, "Enhanced random search based incremental extreme learning machine," *Neurocomputing,* vol. 71, pp. 3460-3468, 2008.

[90] S. Suresh, R. V. Babu, and H. Kim, "No-reference image quality assessment using modified extreme learning machine classifier," *Applied Soft Computing,* vol. 9, pp. 541-552, 2009.

[91] I. Marques and M. Graña, "Face recognition with lattice independent component analysis and extreme learning machines," *Soft Computing,* vol. 16, pp. 1525-1537, 2012.

[92] L.-C. Shi and B.-L. Lu, "EEG-based vigilance estimation using extreme learning machines," *Neurocomputing,* vol. 102, pp. 135-143, 2013.

[93] Z.-H. You, Y.-K. Lei, L. Zhu, J. Xia, and B. Wang, "Prediction of protein-protein interactions from amino acid sequences with ensemble extreme learning machines and principal component analysis," *BMC bioinformatics,* vol. 14, p. S10, 2013.

[94] Y. Song and P. Liò, "A new approach for epileptic seizure detection: sample entropy based feature extraction and extreme learning machine," *Journal of Biomedical Science and Engineering,* vol. 3, p. 556, 2010.

[95] S.-J. Lin, C. Chang, and M.-F. Hsu, "Multiple extreme learning machines for a two-class imbalance corporate life cycle prediction," *Knowledge-Based Systems,* vol. 39, pp. 214-223, 2013.

[96] S. Decherchi, P. Gastaldo, A. Leoncini, and R. Zunino, "Efficient digital implementation of extreme learning machines for classification," *Circuits and Systems II: Express Briefs, IEEE Transactions on,* vol. 59, pp. 496-500, 2012.

[97] G. Huang, S. Song, J. N. Gupta, and C. Wu, "Semi-supervised and unsupervised extreme learning machines," 2014.

[98] P. L. Bartlett, "The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the

network," *Information Theory, IEEE Transactions on,* vol. 44, pp. 525-536, 1998.
[99] M. T. Hanna, "The revised direct batch evaluation algorithm of optimal eigenvectors of the DFT matrix using the notion of Moore-Penrose matrix pseudoinverse," in *Communications, Control and Signal Processing (ISCCSP), 2014 6th International Symposium on*, 2014, pp. 433-436.
[100] K. Tanabe, "Projection method for solving a singular system of linear equations and its applications," *Numerische Mathematik,* vol. 17, pp. 203-214, 1971.
[101] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics,* vol. 12, pp. 55-67, 1970.
[102] G.-B. Huang, L. Chen, and C.-K. Siew, "Universal approximation using incremental constructive feedforward networks with random hidden nodes," *Neural Networks, IEEE Transactions on,* vol. 17, pp. 879-892, 2006.
[103] G. Gordon and R. Tibshirani, "Karush-kuhn-tucker conditions," *Optimization,* vol. 10, p. 725.
[104] R. Storn and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of global optimization,* vol. 11, pp. 341-359, 1997.
[105] C. J. Ter Braak, "A Markov Chain Monte Carlo version of the genetic algorithm Differential Evolution: easy Bayesian computing for real parameter spaces," *Statistics and computing,* vol. 16, pp. 239-249, 2006.
[106] K. Price, R. M. Storn, and J. A. Lampinen, *Differential evolution: a practical approach to global optimization*: Springer Science & Business Media, 2006.
[107] C. Couprie, L. Grady, L. Najman, and H. Talbot, "Power watershed: A unifying graph-based optimization framework," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 33, pp. 1384-1399, 2011.
[108] G. H. Golub and C. F. Van Loan, "Matrix computations. 1996," *Johns Hopkins University, Press, Baltimore, MD, USA,* pp. 374-426, 1996.
[109] R. Merris, "Laplacian matrices of graphs: a survey," *Linear algebra and its applications,* vol. 197, pp. 143-176, 1994.
[110] H. Yu, Y. Chen, J. Liu, and G.-B. Huang, "An Adaptive and Iterative Online Sequential ELM-Based Multi-Degree-of-Freedom Gesture Recognition System," *IEEE intelligent systems,* vol. 28, pp. 55-59, 2013.
[111] J. R. Landis and G. G. Koch, "An application of hierarchical kappa-type statistics in the assessment of majority agreement among multiple observers," *Biometrics,* pp. 363-374, 1977.
[112] A. B. Santos, A. de Albuquerque Araújo, and D. Menotti, "Combining multiple classification methods for hyperspectral data interpretation," *Selected Topics in Applied Earth Observations and Remote Sensing, IEEE Journal of,* vol. 6, pp. 1450-1459, 2013.
[113] I. Dópido, J. Li, P. R. Marpu, A. Plaza, J. B. Dias, and J. A. Benediktsson, "Semi-supervised self-learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens,* vol. 51, pp. 4032-4044, 2013.
[114] J. A. Benediktsson, J. A. Palmason, and J. R. Sveinsson, "Classification of hyperspectral data from urban areas based on extended morphological profiles," *Geoscience and Remote Sensing, IEEE Transactions on,* vol. 43, pp. 480-491, 2005.
[115] L. Chapel, T. Burger, N. Courty, and S. Lefèvre, "Classwise hyperspectral image classification with PerTurbo method," in *Geoscience and Remote*

Sensing Symposium (IGARSS), 2012 IEEE International, 2012, pp. 6883-6886.

[116] O. Ozdemir and Y. Y. Cetin, "The effect of training data on hyperspectral classification algorithms," in *Signal Processing and Communications Applications Conference (SIU), 2013 21st*, 2013, pp. 1-4.

[117] M. Dalla Mura, A. Villa, J. A. Benediktsson, J. Chanussot, and L. Bruzzone, "Classification of hyperspectral images by using extended morphological attribute profiles and independent component analysis," *Geoscience and Remote Sensing Letters, IEEE,* vol. 8, pp. 542-546, 2011.

[118] B. Song, J. Li, M. Dalla Mura, P. Li, A. Plaza, J. M. Bioucas-Dias, J. Atli Benediktsson, and J. Chanussot, "Remotely sensed image classification using sparse representations of morphological attribute profiles," *Geoscience and Remote Sensing, IEEE Transactions on,* vol. 52, pp. 5122-5136, 2014.

[119] F. Melgani and Y. Bazi, "Markovian fusion approach to robust unsupervised change detection in remotely sensed imagery," *Geoscience and Remote Sensing Letters, IEEE,* vol. 3, pp. 457-461, 2006.

[120] D. Tuia, M. Volpi, L. Copa, M. Kanevski, and J. Munoz-Mari, "A survey of active learning algorithms for supervised remote sensing image classification," *Selected Topics in Signal Processing, IEEE Journal of,* vol. 5, pp. 606-617, 2011.

[121] J. Carletta, "Assessing agreement on classification tasks: the kappa statistic," *Computational linguistics,* vol. 22, pp. 249-254, 1996.

[122] Y. Bengio and Y. Grandvalet, "No unbiased estimator of the variance of k-fold cross-validation," *The Journal of Machine Learning Research,* vol. 5, pp. 1089-1105, 2004.

[123] N. Liu and H. Wang, "Ensemble based extreme learning machine," *Signal Processing Letters, IEEE,* vol. 17, pp. 754-757, 2010.