

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Université de TLEMCEM



Faculté des Sciences  
Département d'Informatique

## MEMOIRE

Pour l'obtention du diplôme de **Magister en Informatique**

Option : Intelligence Artificielle et Aide à la Décision (IAAD)

Sujet :

# Amélioration de la Robustesse d'un Classifieur Flou par Apprentissage

Présenté et soutenu publiquement par :

**Saidi Mohammed El-Fatih**

Devant le jury composé de :

<b>Président</b>	Mr Bessaid Abdelhafid	Professeur	Université de Tlemcen
<b>Encadreur</b>	Mr Chikh Mohammed Amine	Professeur	Université de Tlemcen
<b>Examineur</b>	Mr Abderrahim Mohammed El Amine	MCA	Université de Tlemcen
<b>Invité</b>	Mr Benazzouz Mortada	MCB	Université de Tlemcen

Année universitaire 2014-2015

*Je dédie ce modeste travail*

*~ A mes chers parents pour leurs encouragements, leur soutien moral, spirituel et leur tolérance durant toutes mes années d'études, tous les mots restent faibles pour exprimer mes sentiments, qu'ils trouvent a travers ce travail la récompense de leurs efforts. J'espère qu'Allah me donne la force et le courage pour que je puisse rendre leurs sacrifices*

*~ A tous ceux qui m'ont aidé de loin ou de près.*

*FATEH.*

# *Remerciements*

*En premier lieu, je tiens à exprimer ma gratitude et reconnaissance envers mon directeur de thèse, Professeur Mr CHIKH M.A, pour m'avoir accueilli, encadré et soutenu tout au long de mes études à l'Université Tlemcen. Je le remercie de m'avoir apporté toute son expérience et ses compétences sans lesquelles ce travail n'aurait pas pu aboutir.*

*Ensuite, je désire adresser mes sincères remerciements à Mr Bessaid Abdelhafid, Professeur à l'Université de Tlemcen. Son savoir, sa méthodologie et son expérience m'ont beaucoup aidé lors de ma formation de Magister. Et je tiens à lui témoigner toute ma gratitude pour la réussite de cette formation. Merci d'avoir accepté de présider ce jury.*

*Je voudrais remercier, de façon particulière Mr Abderrahim Mohammed El Amine, maitre de conférences à l'université de Tlemcen, en tant que Examineur, pour ses nombreuses suggestions malgré ses charges administratives et pédagogiques, et surtout pour sa patience et ses judicieux conseils.*

*J'adresse mes profonds remerciements à Mr Benazzouz Mortada maitre de conférences à l'université de Tlemcen, pour l'intérêt porté à ce travail et sa présence parmi les membres du jury.*

# Tables des matières

Tables des Matières .....	i
Liste des figures.....	iv
Liste des tableaux .....	v
<b>Introduction Générale.....</b>	<b>1</b>
<b>CHAPITRE 1: Classification et Interprétabilité.</b>	
Introduction .....	3
1. Classification dans le domaine médicale.....	3
1.1 Aide au diagnostic .....	4
1.2 Interprétabilité .....	4
1.3 Contexte de notre travail .....	5
1.3.1 Définition diabète .....	5
2. Classification supervisés .....	6
2.1 Notions de classification .....	7
2.2 Le problème de la généralisation.....	7
2.3 Evaluation d'une hypothèse de classification .....	7
3. Système d'inférence flou.....	8
4. Avantage d'utilisation d'un système basé Règle-flou pour classification .....	9
5. Conclusion.....	10
<b>CHAPITRE 2: Etats de l'art</b>	
Introduction .....	12
1. Adaptative Neuro-Fuzzy Inférence Système.....	12
1.1 Théorie.....	12
1.2 Architecture d'ANFIS .....	13
1.3 Reconnaissance Neuro-floue du diabète dans la littérature.....	15

2. Algorithme génétique .....	17
2.1 Principe de fonctionnement .....	17
2.1.1 Le codage.....	20
2.1.2 La fonction d'évaluation (fitness) .....	20
2.1.3 Génération de la population initiale .....	21
2.1.4 Sélection .....	21
2.1.4.1 Sélection proportionnelle.....	22
2.1.4.1 Sélection par rang .....	23
2.1.4.1 Sélection par tournoi.....	23
2.1.5 Les opérateurs de variation.....	23
2.1.5.1 Croisement.....	24
2.1.5.2 Mutation .....	25
2.1.6 Test d'arrêt.....	26
3. L'apport des Algorithmes génétiques dans la classification flou.....	26
3.1 Approche (GF-Miner) .....	27
3.1.1 GF-Miner extension de Fuzzy Miner .....	28
3.1.2 Le processus génétique GF-Miner .....	29
3.1.3 Résultats obtenus .....	31
3.2 Approche (un cadre général pour la conception) .....	32
3.2.1 Résultats expérimentales .....	33
4. Synthèse.....	35
5. Conclusion.....	36

## **CHAPITRE 3: Implémentation**

Introduction .....	37
1. Mise en œuvre .....	37
1.1 Base de données .....	38
2. Phase1 (construction de notre classifieur).....	40
2.1 Partition flou .....	40
2.2 Dérivation des règles floues .....	41
2.3 L'inférence floue pour la classification.....	43
2.4 Résultats obtenus .....	44
2.4.1 Résultat1 .....	44
2.4.2 Résultat2 .....	45

2.4.2.1 Discussion d'un cas FN (faux-négative) .....	51
2.4.2.2 Discussion d'un cas FP (faux-positive).....	51
3. Phase2 (Amélioration par Algorithme Génétique).....	52
3.1 Codage .....	52
3.2 Opérateurs génétiques .....	53
3.2.1 Sélection .....	53
3.2.2 Croisement.....	53
3.2.3 Mutation .....	54
3.2.4 Test d'arrêt.....	54
3.3 Résultat obtenu .....	55
4. Conclusion.....	55
<b>Conclusion Générale .....</b>	<b>57</b>
<b>Bibliographie.....</b>	<b>59</b>

# Liste des figures

## CHAPITRE 1:

FIGURE 1.1 Projection du nombre de personnes diabétiques dans différentes régions du monde.....	6
Figure 1.2 Structure d'un Système Flou .....	9

## CHAPITRE 2:

Figure 2.1 System Neuro-Flou (NFS).....	13
Figure 2.2 Architecture ANFIS .....	13
Figure 2.3 Schéma de principe des AG.....	19
Figure 2.4 le codage .....	20
Figure 2.5 modèle de la roulette.....	22
Figure 2.6 méthode SUS .....	22
Figure 2.7 Croisement un point.....	24
Figure 2.8 Croisement deux points .....	24
Figure 2.9 Croisement uniforme .....	25
Figure 2.10 Croisement bit-flip.....	25
Figure 2.11 Architecture GF-Miner .....	28
Figure 2.12 Fonction d'appartenance (triangulaire et trapézoïdale.....	29

## CHAPITRE 3:

Figure 3.1 Schéma représentatif de notre procédure.....	37
Figure 3.2 Histogrammes des 9 paramètres de la base de données.....	39
Figure 3.3 distribution des diabétiques et non diabétiques dans les paramètres de la base ....	39
Figure 3.4 Table représentant des règles floues correspondant à L=2 et L=3 .....	41
Figure 3.5 Exemple fonction d'appartenance avec partition en 03 ens-flou.....	42
Figure 3.6 Exemple représentation d'un chromosome.....	52
Figure 3.7 Exemple de mutation .....	54
Figure 3.8 Meilleur Chromosome Obtenu .....	55

# Liste des Tableaux

## CHAPITRE 2:

Tableau 2.1 Quelques travaux avec ANFIS .....	16
Tableau 2.2 Paramétrés pour fonctions d'appartenance.....	29
Tableau 2.3 Résultats expérimentales .....	32
Tableau 2.4 Taux classification obtenus (utilisant toutes les attributs).....	34
Tableau 2.5 classification obtenus (comparaison avec l'approche sélection des attributs)...	34
Tableau 2.6 Nombre des règles et d'attributs utilisant les différents ensembles donnés .....	34

## CHAPITRE 3:

Tableau 3.1 résultats obtenus avec différentes type degré de compatibilité et différentes nombre de S-Ens-Flou.....	45
Tableau3.2 Nombre de règles dérivées.....	46
Tableau 3.3 Echantillon de règles floues dérivées .....	46
Tableau 3.4 Performances notre classifieur floue après phase Test.....	47
Tableau3.5 Exemple I1 avec les 02 règles .....	49
Tableau3.6 Exemple I2 avec les 02 règles .....	49
Tableau3.7 Exemple I3 avec les 02 règles .....	50
Tableau3.8 Exemple I4 avec les 02 règles .....	50
Tableau3.9 Exemple I5 avec les 02 règles .....	50
Tableau3.10 Exemple I7 avec les 02 règles .....	50

## Introduction

La modélisation floue, développée par les systèmes basés sur les règles floues linguistiques, nous permet de traiter des systèmes en établissant un modèle linguistique qui pourrait devenir interprétable par des êtres humains. L'interprétabilité permet de vérifier la plausibilité d'un système, conduisant à une maintenance aisée de ce dernier. Il peut également être utilisé pour acquérir des connaissances à partir d'un problème caractérisé par des exemples numériques.

La modélisation floue linguistique vient avec deux conditions contradictoires : Interprétabilité et Précision. Ces dernières années l'intérêt des chercheurs d'obtenant des modèles flous linguistiques plus interprétables s'est développé. L'interprétabilité d'une base de règles est généralement liée à la continuité, cohérence et l'exhaustivité. La continuité garantit que de légères variations dans l'entrée ne peut provoquer de grandes variations dans le résultat. La cohérence signifie que si deux ou plusieurs règles sont déclenchées à la fois, leurs conclusions seront cohérentes. L'exhaustivité signifie que pour tout vecteur d'entrée possible, au moins une règle est activée sans rupture d'inférence.

Indiscernabilité et Incomplétude des partitions floues, Inconsistance des règles floues, trop de sous-ensembles flous. Sont plusieurs facteurs qui résultent une perte sur l'interprétabilité et la précision d'un classifieur flou, afin d'améliorer la robustesse des classifieurs flous, nous proposons dans ce mémoire de Magister d'automatiser l'apprentissage structurel et paramétriques par les Algorithmes Génétiques qui offrent des possibilités fortes de recherche dans des espaces complexes. L'idée fondamentale est de préserver une population de chromosomes qui évolue par un processus de concurrence et de variation commandée.

L'objectif principal été alors d'utilisé Les algorithmes évolutionnaires comme une approche complémentaire afin d'optimiser la connaissance extraite par notre classifieur floue avec un minimum de règles et de partitions floues tout en gardant la performance à un niveau satisfaisant. L'évaluation de cette approche sera établie sur la base de données Diabète Indiennes PIMA du dépôt d'UCI Machine Learning Database.

Ce travail de Magister se situe dans le contexte général de l' « Aide au Diagnostic Médical », qui a pour but de réaliser un classifieur de diabète lisible et transparent en améliorant la robustesse. Le plan de mémoire est composé de :

- Chapitre 1 Classification et Interprétabilité, présente le concept de classification floue avec l'importance de l'indice d'interprétabilité.
- Chapitre 2 Etats de L'art, fait le tour des différents travaux existants dans la Classification flou et hybridation avec d'autres approches, réseau de neurones et génétique algorithmique.
- Chapitre 3 Implémentation, décrivant l'implémentation de notre classifieur floue avec l'approche utilisé comme hybridation, les résultats obtenus sont discutés.
- En dernier lieu, une conclusion générale et des perspectives de ce travail de Magister seront présentées.

## Chapitre 01

### Classification et Interprétabilité

#### *Introduction*

L'objectif de la classification automatique est de former des groupes d'individus ou de variables afin de structurer un ensemble de données. On cherche souvent des groupes homogènes c'est à dire que les objets sont ressemblant a l'intérieur d'un même groupe. Les méthodes de classification se distinguent entre autre par la structure de classification obtenue (partition, recouvrement, hiérarchie, pyramide).

Les réseaux neurologiques et les machines de vecteur de soutien SVM sont probablement les techniques de classification de données les plus populaires. Cependant, les classificateurs basés sur ces techniques ne sont pas assez transparents et sont souvent considérés en tant que « boîtes noires ». Le transparent et l'Interprétabilité sont des normes qui ont une grande importance dans quelques domaines d'application, tels que d'aide à la décision médical ou le contrôle de qualité. En revanche, des systèmes basés sur les règles flous et les arbres de décision floue sont connus pour leur transparent et capacité de l'incertitude d'explication. ANFIS, ARTMAP flou, FCM sont des exemples des systèmes basés sur la logique floue les plus en avant. Il est bien connu que concevoir des systèmes basés sur les règles flous dans les espaces dimensionnels élevés soit plutôt problématique. Cependant, il y a beaucoup de problèmes caractérisés par un petit ou modéré nombre de variables. D'ailleurs, les données dimensionnelles tout à fait souvent élevées varient dans un nombre beaucoup plus peu élevé des dimensions si comparé à la dimensionnalité d'un espace d'entrée.

Le but principal des différentes méthodes de classification est d'automatiser le processus de la prise de décision. Elles s'appliquent à un grand nombre d'activités humaines et conviennent en particulier au problème d'aide au diagnostic dans le domaine médical (regrouper des patients afin de définir une thérapeutique adaptée a un type particulier de malades).

#### *1. Classification dans le domaine médicale :*

La classification est un domaine de recherche qui a été développé aux années soixante. C'est l'une parmi les techniques les plus utilisées en médecine. Tel que les systèmes d'aide au diagnostic constituent aujourd'hui des moyens techniques indispensables dans le domaine médical, Il s'agit d'affecter un ensemble d'objets (tissus, enregistrements, etc.) à un ensemble de classes selon la description de celles-ci. Cette description est effectuée grâce à des

propriétés ou des conditions typiques propres aux classes. Les objets sont alors classifiés suivant qu'ils vérifient ou non ces conditions ou propriétés. [Mess05]

### ***1.1 Aide au diagnostic :***

Vue le nombre important d'individus et la complexité d'interprétation des paramètres, il est utile et important de faire appel aux systèmes de classification des données. D'où la nécessité de conception d'un système d'aide au diagnostic pour seconder le médecin d'une façon complémentaire et à réduire au minimum les erreurs possibles qui peuvent survenir ; et cela en examinant des données médicales dans un temps plus court et d'une façon plus détaillée et plus précise. [Set11]

La démarche système expert est une procédure qui a été largement utilisée ou employée dans les années 80 dans le domaine médical, le cadre de notre travail consiste à établir un système d'aide au diagnostic médical à partir des paramètres ou description clinique d'un patient.

Dans un système expert la connaissance d'un expert ou plusieurs experts est décrite sous forme de règles cette dernière est utilisée pour classifier des nouveaux cas. Ce type de méthode dépend fortement de la capacité à extraire et à formaliser les connaissances de l'expert.

La classification automatique propose un autre processus qui permet de tirer d'une manière automatique les règles à partir d'un ensemble d'exemples bien définis (Un exemple consiste en la description d'un cas avec la classification correspondante)

L'une des motivations pour l'utilisation de la classification automatique dans les systèmes d'aide au diagnostic médicale c'est pour faciliter la tâche de la prise de décision en augmentant la précision, cependant ces systèmes orienté précision entraîne généralement une réduction de l'interprétabilité du système, sachant que l'interprétabilité est une question clé pour les systèmes d'aide au diagnostic.

### ***1.2 Interprétabilité :***

Le concept de l'interprétabilité apparait dans la littérature sous différents noms : transparence, intelligibilité, compréhensibilité. L'interprétabilité d'un système qui est la facilité d'analyse de son comportement, dépend à la fois du nombre de ses variables et la complexité du modèle, ainsi qu'aux mécanismes d'apprentissage.

L'interprétabilité représente la force motrice la plus importante derrière la mise en œuvre des systèmes de classification automatique pour les problèmes d'application médicale. L'expert devrait être capable de comprendre le classifieur et d'évaluer ses résultats, elle permet de vérifier la plausibilité d'un système, conduisant à une maintenance aisée de ce dernier. Il peut également être utilisé pour acquérir des connaissances à partir d'un problème caractérisé par des exemples numériques [DS06]. Une amélioration de l'interprétabilité peut augmenter les performances de généralisation lorsque l'ensemble des données est petit.

### ***1.3 Contexte de notre travail :***

L'OMS a estimé le nombre de diabétiques dans le monde en 2000 à plus de 177 millions de cas. La projection en 2030 serait de 350 millions de cas, que seul un remède pourra freiner [Dia02]. La nécessité de conception d'un système d'aide au diagnostic pour aider le médecin d'une façon complémentaire est très important vue la complexité d'interprétation des paramètres de cette maladie. La gravité de cette maladie réside dans ses impacts néfastes sur plusieurs organes du corps d'où la nécessité de faire appel à un système de classification des données avec comme objectif de réduire les erreurs possibles qui peuvent survenir, et cela en examinant des données médicales dans un temps plus court et d'une façon plus détaillée et plus précise.

#### ***1.3.1 Définition diabète :***

Le diabète est une condition chronique caractérisée par une insuffisance absolue ou relative de la sécrétion de l'insuline par le pancréas. L'insuline est une hormone qui régule la concentration de sucre dans le sang. L'hyperglycémie, ou concentration sanguine élevée de sucre, est un effet fréquent du diabète non contrôlé qui conduit avec le temps à des atteintes graves de nombreux systèmes organiques et plus particulièrement des nerfs et des vaisseaux sanguins. [Set11]

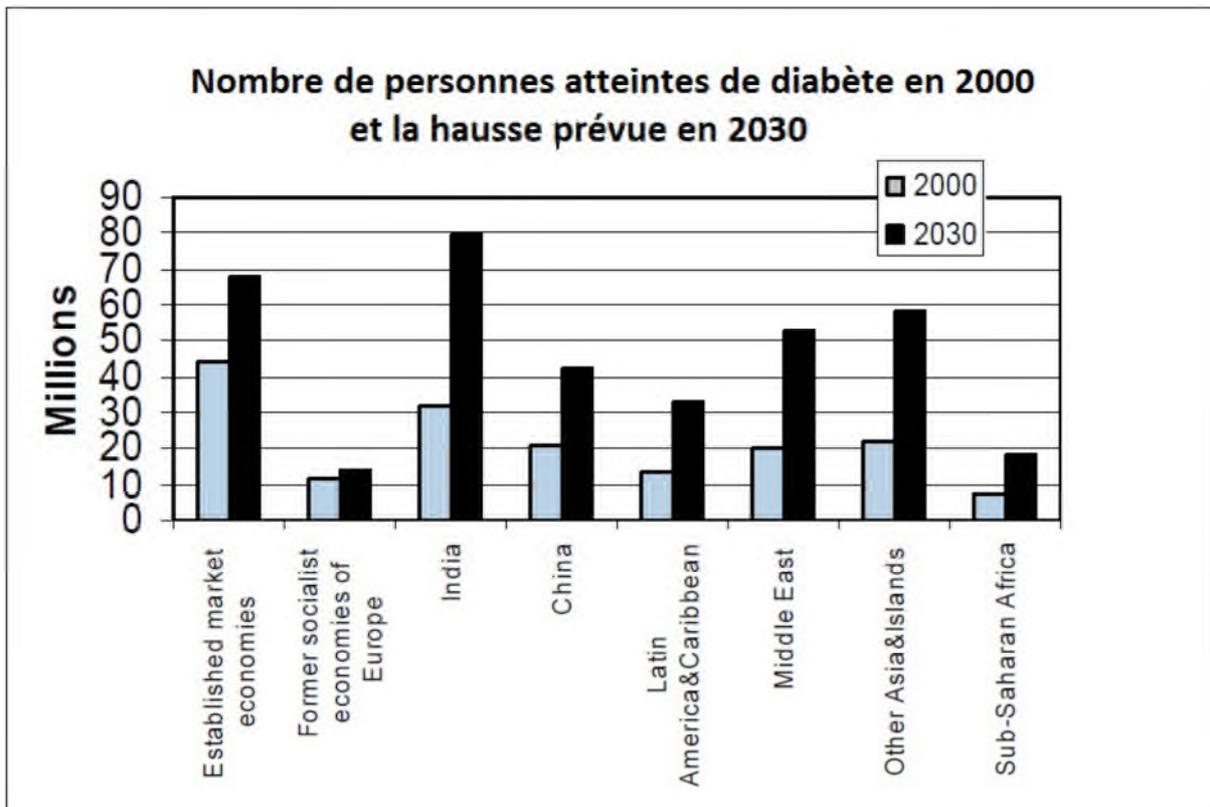


FIGURE 1.1 – Projection du nombre de personnes diabétiques dans différentes régions du monde. [Dia11]

## 2. Classification supervisé :

Nous rappelons tout d'abord une définition du problème de classification supervisée ou reconnaissances de formes. L'objectif de la classification est d'identifier les classes auxquelles appartiennent des objets à partir de leurs caractéristiques ou attributs descriptifs. Le terme de classification peut désigner deux approches distinctes : la classification supervisée et la classification non-supervisée (automatic classification et clustering en anglais). Les méthodes non supervisées ont pour but de constituer des groupes d'exemples (ou des groupes d'attributs) en fonction des données observées, sans connaissance a priori. En revanche les méthodes supervisées utilisent la connaissance a priori sur l'appartenance d'un exemple à une classe pour construire un système de reconnaissance de ces classes. L'objectif de la classification supervisée est d'apprendre à l'aide d'un ensemble d'entraînement une procédure de classification qui permet de prédire l'appartenance d'un nouvel exemple à une classe.

## ***2.1 Notions de classification :***

Dans le cadre de la classification supervisée, les classes sont connues et l'on dispose d'exemples de chaque classe.

**Définition 1.1 (Exemple)** Un exemple est un couple  $(x, y)$ , où  $x \in X$  est la description ou la représentation de l'objet et  $y \in Y$  représente la supervision de  $x$ .

**Définition 1.2 (Classification supervisée)** Soit un ensemble d'exemples de  $n$  données étiquetées:  $S = \{(x_1, y_1), \dots, (x_n, y_n)\}$ . Chaque donnée  $x_i$  est caractérisée par  $P$  attributs et par sa classe  $y_i$ . On cherche une hypothèse  $h$  telle que :

1.  $h$  satisfait les échantillons  $\forall i \in \{1, \dots, n\} h(x_i) = y_i$
2.  $h$  possède de bonnes propriétés de généralisation.

Le problème de la classification consiste donc, en s'appuyant sur l'ensemble d'exemples à prédire la classe de toute nouvelle donnée  $x$

## ***2.2 Le problème de la généralisation:***

L'objectif de la classification est de fournir une procédure ayant un bon pouvoir prédictif c'est-à-dire garantissant des prédictions fiables sur les nouveaux exemples qui seront soumis au système. La qualité prédictive d'un modèle peut être évaluée par le risque réel ou espérance du risque, qui mesure la probabilité de mauvaise classification d'une hypothèse  $h$ .

## ***2.3 Évaluation d'une hypothèse de classification :***

Pour avoir une estimation non optimiste de l'erreur de classification, il faut recourir à une base d'exemples qui n'ont pas servi pour l'apprentissage : il s'agit de la base de test.

La base de test contient elle aussi des exemples étiquetés qui permettent de comparer les prédictions d'une hypothèse  $h$  avec la valeur réelle de la classe. Cette base de test est généralement obtenue en réservant une partie des exemples supervisés initiaux et en ne les utilisant pas pour la phase d'apprentissage. [Ed08]

### *3. Système d'inférence flou*

C'est à Lotfi Zadeh, spécialiste renommée en théorie des systèmes, que revient le mérite de faire la jonction en 1965 entre la logique multi-valuées et les travaux sur les concepts du « vague », grâce a la notion d'ensemble flou et aux logiques associés. Celle-ci est en effet une extension de la logique booléenne, les ensembles flous et la théorie des possibilités sont a la base d'une méthodologie de représentation et d'utilisation de connaissances imprécises, vagues et incertaines, que Zadeh appelle théorie du raisonnement approché, plus connue sous le nom « logique flou ».

L'approche par la logique floue permet l'introduction du concept de degré d'appartenance, qui détermine les « forces » avec lesquelles un individu appartient aux différentes classes. Cela repose sur le fait que le concept flou ne cherche pas un point de rupture qui décide de l'appartenance d'un individu à une classe, mais qu'elle raisonne plutôt sur la base d'un intervalle de valeurs.

L'idée qui soutient l'approche est la possibilité d'appartenance à la fois à plusieurs classes. Partant dans ce sens, ce sont exclus toutes les méthodes de classification « dure » contraignant les individus à être membre d'une, et une seule, classe. Quoique la probabilité d'appartenance des objets aux classes ne soit pas évidemment une exclusivité des techniques de classification floue, il semble que les résultats obtenus des approches floues sont plus pertinents en ce sens qu'ils donnent une matrice des degrés d'appartenance de chaque individu à chaque classe, ce qui n'est pas tout à fait le cas dans les autres analyses « classiques » de classification. [Mess05]

### *Structure Interne d'un système d'inférence flou*

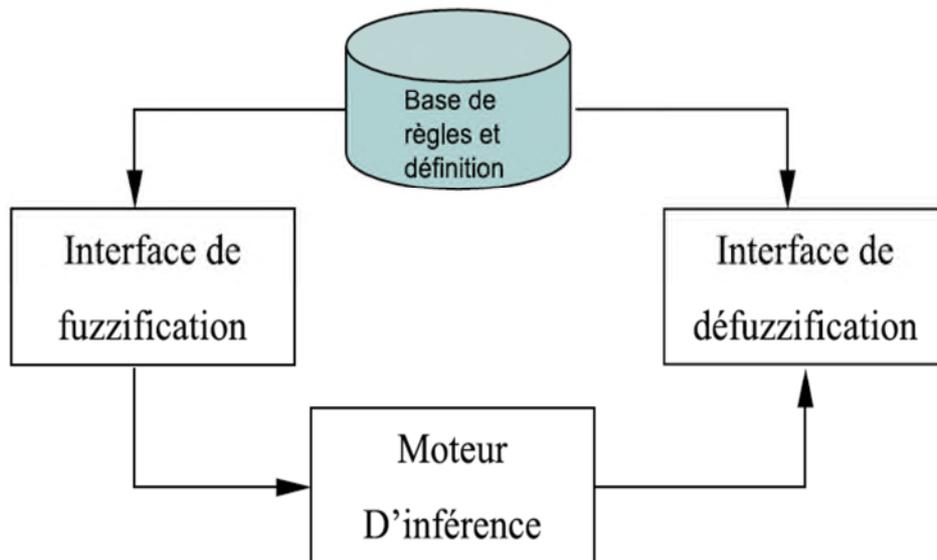


Figure1.2 Structure d'un Système Flou [Reb03]

Le traitement réalisé par le système d'inférence flou (SIF) est divisé en trois tâches essentielles (voir figure 1.2)

- 1- La fuzzification transforme une valeur numérique  $x_0$  de l'entrée en une valeur floue.
- 2- L'inférence, produit l'image de la partie floue issue de la fuzzification par une relation floue R, généralement construite à partir de règles.
- 3- La défuzzification transforme la partie floue issue de l'inférence en valeur numérique Y en sortie. La défuzzification constitue alors une prise de décision.

#### ***4. Avantage d'utilisation d'un système basé Règle-flou pour classification***

Les méthodes de classification ont pour but d'identifier les classes auxquelles appartiennent des objets à partir de certains traits descriptifs. Elles s'appliquent à un grand nombre d'activités humaines et conviennent en particulier au problème de la prise de décision automatisée [GT00]. Un classifieur basé sur la logique floue est un classifieur qui utilise des règles floues pour assigner des étiquettes de classe aux objets.

Les humains possèdent la capacité remarquable d'identifier des objets en dépit de la présence d'information incertaine et inachevée. Alors ce type de classifieurs (basé Règle-Flou) fournissent des moyens appropriés de traiter ce genre d'information bruyante, imprécise ou inachevée qui dans beaucoup de problèmes de classification est plutôt la norme que l'exception. Ils font un effort fort de réconcilier la précision empirique des techniques traditionnelles de technologie avec l'interprétabilité de l'intelligence artificielle.

Des systèmes de classification peuvent être divisés en deux groupes principaux selon la manière dans laquelle ils ont employé : systèmes de classification qui sont supposés travailler de façon autonome, et tels qui sont prévus pour être des outils pour soutenir la prise de décision d'un utilisateur humain. Dans le cas des classificateurs autonomes, l'objectif de base du processus de conception est la performance, c.-à-d., le pourcentage des classifications correctes. D'autres critères tels que la compréhensibilité, la robustesse, la polyvalence, et la concordance avec la connaissance précédente sont secondaires, car ils sont simplement appropriés à augmenter l'acceptation par les utilisateurs.

Nous cherchons dans notre travail qui s'inscrit dans le contexte d'aide au diagnostic médicale d'avoir de l'interprétabilité, dans tels problèmes de classification, le rôle fondamental des règles floues est de rendre le procédé de classification transparent à l'utilisateur dans un cadre formel et réalisable (Zadeh, 1977).[ Cor01]

### ***Conclusion :***

Les méthodes utilisées par les systèmes d'apprentissage sont très nombreuses et sont issues de domaines scientifiques variés. Les méthodes issues de l'intelligence artificielle sont des méthodes non paramétriques. On distingue les méthodes symboliques (la procédure de classification produite peut être écrite sous forme de règles) des méthodes non symboliques ou adaptatives (la procédure de classification produite est de type «boite noire»). Les méthodes symboliques, les plus utilisées sont basées sur les systèmes de règles floues. Pour les méthodes adaptatives, nous citons en particulier les réseaux de neurones.

Dans les systèmes d'aide au diagnostic médicale nous avons besoin d'un modèle qui garantit l'interprétabilité et la lisibilité. Des l'origine, les règles linguistiques dans les systèmes d'inférence floue ont été reconnues pour leurs avantages : conception simple et rapide d'un système, facilité de maintenance et d'adaptation. Leur utilisation s'inscrit dans la perspective de proposer des modèles utilisant un langage de description aussi proche que possible de celui employé par les experts humains. La logique floue constitue un cadre commode pour l'expression et la formalisation de la connaissance experte. D'autre part, le formalisme flou est à la base de nombreuses méthodes d'apprentissage. Dans les cas où des techniques de modélisation plus classiques peuvent s'avérer performantes, on attend des systèmes d'inférence floue qu'ils apportent des éléments de connaissance qualitative sur le

phénomène étudié, qu'ils soient interprétables, [Jos10]. Cela signifie que l'utilisateur pourra comprendre le fonctionnement du système dans ses moindres détails et qu'il pourra se représenter son comportement. Lorsque le langage de modélisation utilise des règles linguistiques floues, la compréhension implique une lecture facile de chacune des règles et des termes linguistiques associés ainsi qu'une représentation du comportement de la base de règles prise dans son ensemble.

L'interprétabilité ne doit pas nous faire oublier la précision : ces deux pôles sont essentiels pour la qualité des systèmes d'inférence floue. L'essentiel est de parvenir à un compromis. Et grâce à l'utilisation d'autres techniques (les réseaux neurones, Algorithmes Génétiques....) qui se trouve dans la littérature a permet le maintien de l'interprétabilité et la lisibilité de notre classifieur flou en assurant la précision.

Les systèmes d'aide au diagnostic constituent aujourd'hui des moyens techniques indispensables dans le domaine médical. Nous sommes intéressés dans notre travail qui s'inscrit dans le contexte de domaine d'aide au diagnostic médicale à réaliser un système d'apprentissage automatique structurel et paramétrique qui améliore la robustesse de notre classifieur flou en essayant d'augmenter les performances du notre classifieur sur le plan précision et interprétabilité.

## Chapitre 02

### Etats de L'art

#### *Introduction*

Les techniques informatiques d'intelligence telles que la logique floue, les réseaux neurones, et les algorithmes génétiques (GA) sont des domaines populaires de recherches, puisqu'elles peuvent confronter des problèmes complexes de technologie. On propose dans notre travail un cadre général pour concevoir un système de classification basé sur les règles flou. Notre objectif principal est de réaliser un système d'apprentissage qui permet d'améliorer la robustesse de notre classifieur à travers l'optimisation structurelle et paramétrique avec une précision de diagnostic élevée et une transparence maximale.

Dans la littérature plusieurs techniques d'amélioration de performance des classifieurs flous ont été testées et étudiés, les réseaux de neurones et les algorithmes restent les plus utilisées. Enfin dans ce chapitre on va présenter quelques travaux réalisés dans la littérature qui abordent ces différentes approches.

Dans un premier lieu on va s'intéresser aux différentes approches hybridation des réseaux de neurones avec la logique floue représentée par [Set11] dans son mémoire de magister. [Set11] est intéressée aux approches qui concernent la reconnaissance du diabète :

#### *1. Adaptive Neuro-Fuzzy Inférence Système ANFIS*

##### *1.1 Théorie*

###### *La modélisation Neuro-floue*

Il existe plusieurs méthodes pour mettre en œuvre la technique neuro-floue de modélisation. En général, l'hybridation neuro-floue est faite de deux façons (Mitra et Hayashi [SM00]) :

- Un réseau de neurones équipé de la capacité de manutention du traitement de l'information floue, est appelé un réseau de neurones flou (FNN)
- Un système flou combiné avec les réseaux de neurones pour l'amélioration de certaines de ses caractéristiques, comme la flexibilité, la vitesse et la capacité d'adaptation, est appelé un système neuro flou (NFS).

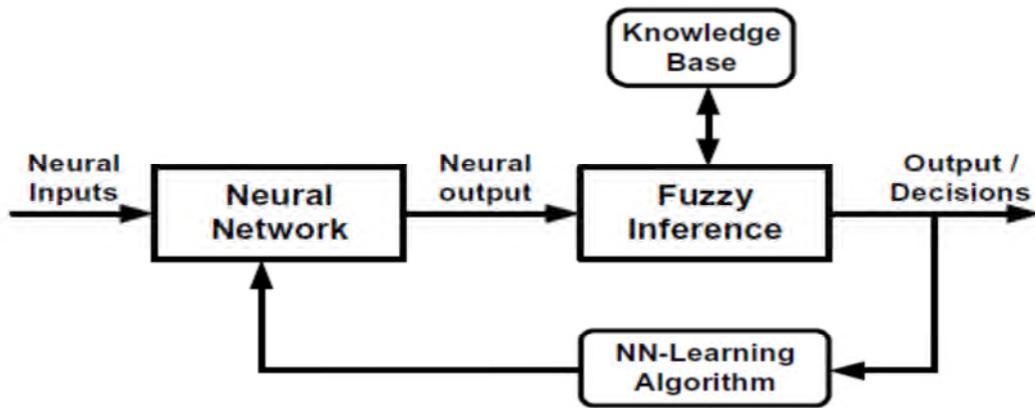


Figure 2.1 System Neuro-Flou (NFS)

Dans ce modèle le bloc de réseau de neurones (voire figure 2.1) entraîne le système d'inférence floue pour générer les décisions correspondantes.

### 1.2 Architecture d'ANFIS

Le modèle ANFIS est un réseau de neurones flou proposé par Jang [Jan93]. Ce réseau est composé de cinq couches représentées sur la figure 2.2 ci-dessous avec deux entrées X et Y. Pour simplifier la présentation de l'architecture d'ANFIS, deux règles floues « si alors » du modèle de Sugeno de premier ordre sont considérées :

Règle 1 : Si (x est A1) et (y est B1) alors (f1 = a<sub>1</sub>x + b<sub>1</sub>y + c<sub>1</sub>)

Règle 2 : Si (x est A2) et (y est B2) alors (f2 = a<sub>2</sub>x + b<sub>2</sub>y + c<sub>2</sub>)

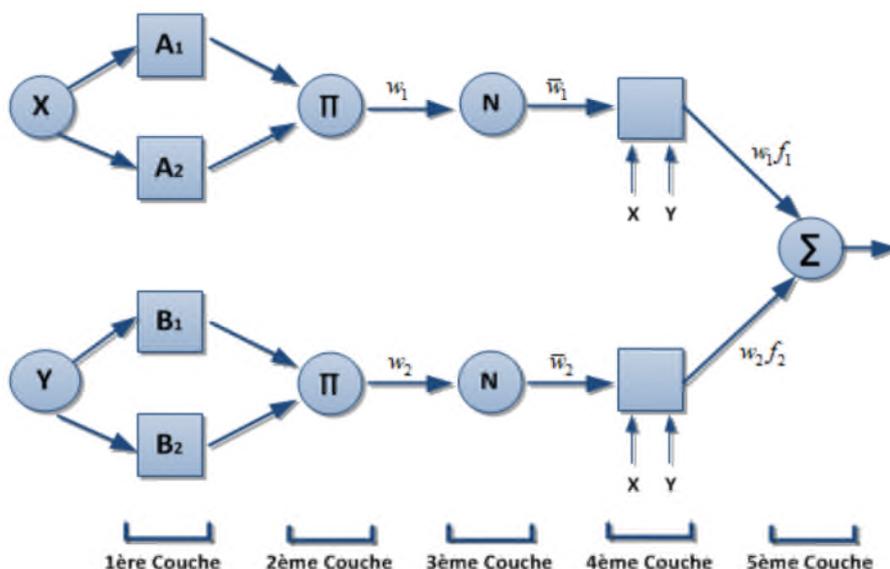


Figure 2.2 – Architecture ANFIS.

**1ère couche : Fuzzyfication** Cette couche permet de convertir les variables d'entrées dans l'espace flou à l'aide des fonctions d'appartenance.

$$O_i^1 = \mu_{A_i}(\chi) \quad \text{Avec : } i = 1, 2$$

$$O_i^1 = \mu_{B_{i-2}}(\gamma) \quad \text{Avec : } i = 3, 4$$

Où  $\mu_{A_i}(\chi)\mu_{B_{i-2}}(\gamma)$  sont les degrés d'appartenance du ième terme de la partition floue des variables d'entrée. Différents types de fonctions d'appartenance peuvent être utilisés : triangle, trapèze ou gaussienne par exemple. Les paramètres définissant ces fonctions sont appelés prémisses.

**2ème couche : Pondération des règles floues** Chaque neurone est associé à une règle floue et permet de définir le poids de cette règle à l'aide d'une t-norme. La t-norme la plus employée est le produit :

$$O_i^2 = \omega_i = \mu_{A_i}(\chi) = \mu_{B_{i-2}}(\gamma) \quad i = 1, 2$$

**3ème couche : Normalisation** Cette couche a pour but de normaliser les différents poids.

$$O_i^3 = \varpi_i = \frac{\omega_i}{\omega_1 + \omega_2} \quad i = 1, 2$$

**4ème couche : Defuzzyfication** Cette couche donne une sortie définie par une fonction du premier ordre des entrées pondérées (approche de Takagi-Sugeno).

$$O_i^4 = \omega_i f_i \quad \text{Avec } f = a_i \chi + b_i \gamma + c_i \quad \text{ou } i = 1, 2$$

La fonction  $f_i$  peut aussi être constante. Dans ce cas, tous les  $a_i = b_i = 0$ . Les paramètres  $a_i$ ,  $b_i$  et  $c_i$  des fonctions  $f_i$  sont appelés paramètres conséquents.

**5ème couche : Calcul de la sortie** La sortie  $O_i^5$  est définie par la somme des sorties de la couche précédente :

$$O_i^5 = \sum_{i=1}^2 \omega_i f_i = \frac{\sum_{i=1}^2 \omega_i f_i}{\omega_1 + \omega_2}$$

### ***Phase d'Apprentissage :***

L'ajustement des paramètres de l'ANFIS est réalisé lors de la phase d'apprentissage. Pour cela un ensemble de données associant les séquences d'entrées et de sorties est nécessaire. Pour la réalisation de cette phase, l'algorithme d'apprentissage hybride est utilisé.

L'algorithme d'apprentissage hybride est une association de deux méthodes celle de descente de gradient et la méthode d'estimation des moindres carrés. La première permet d'ajuster les prémisses en fixant les paramètres conséquents tandis que la seconde LSM (Least square Method) ajuste les paramètres conséquents en fixant les prémisses. [BAKF06]

### ***1.3 Reconnaissance Neuro-floue du diabète dans la littérature***

Elif Derya Ubeyli [Ü10] utilise le classifieur ANFIS pour diagnostiquer la maladie de diabète avec un dispositif de données caractérisé par 8 descripteurs. Dans la base de données de diabète d'Indiens Pima, il y a un total de 768 personnes, dont 268 sont des diabétiques et 500 non diabétiques. Les performances obtenues par cette approche sont : Spécificité 98.58% Sensibilité 96.97% Taux de classification 98.14%.

Le réseau neuro flou ANFIS a été utilisé précédemment par Çelikyılmaz et al. [CB07] pour la classification des diabétiques où le taux de reconnaissance a atteint 82%. Vosouli et al. [VTM08] a employé l'algorithme génétique pour la sélection des paramètres d'entrées donnant un taux de classification de 81.30%. Polat et Gunes [PG07] ont proposé l'analyse en composantes principales pour réduire l'ensemble de données du diabète de 8 à 4 variables d'entrée d'ANFIS atteignant un taux de classification de 89.47%, cependant ces résultats ont été discrédités par Temurtas et al. [TNT09] car ils ont obtenu avec la même approche un taux de reconnaissance de 66.78%. De leurs côtés, Wang et al [LWA+09] ont proposé dans leur étude, la classification par l'inférence floue en se basant sur des agents d'ontologie qui créent une structure des paramètres du diabète, réalisant des performances de 74.20% pour la reconnaissance de vrais positifs, 59.20% faux négatifs, et 71.50% de faux positifs.

AUTEURS	ARTICLES / RAPPORTS	APPROCHE	EXPERIENCE	RÉSULTATS
Narissara Eiamkarnitchat, Nipon Theera-Umpon, Sansanee Auephanwiriyaikul 2010 [ETUA10]	A Novel Neuro-Fuzzy Method for Linguistic Feature Selection and Rule-Based Classification	La méthode proposée met en application un réseau neuro flou qui peut choisir les bonnes caractéristiques et les utiliser pour créer des règles de classification efficace. Les fonctions d'appartenance des variables linguistiques ont également été réalisées dans le processus d'apprentissage où un seul groupe de variables linguistiques va représenter chaque élément d'origine.	La méthode de validation croisée (10-fold cross validation) a été appliquée à l'ensemble de données diabète Pima (768). Où 5 des meilleurs paramètres informatifs de tous les ensembles de données ont été sélectionnés en fonction de leurs poids (plus le poids augmente, plus il est instructif) pour chaque classe.	Un taux de classification de 72.01% avec un nombre de règles réduit de 24 à 5 règles.
WANG Yuan-zhen, ZHANG Zhi-bing, YI Bao-li, LI Hua-yang. 2005 [WZYL05]	Data Mining Based on Computational Intelligence	Cet article combine des outils de calculs intelligents comme le réseau de neurones, la logique floue et l'algorithme génétique pour développer une architecture de data mining (NFGDM). Les données d'entrée sont prétraitées par fuzzyfication, ensuite présentées au réseau probabiliste pour reconnaître les données selon les classes considérées. Une technique d'extraction de règles est ensuite appliquée afin d'extraire des connaissances explicites à partir des réseaux de neurones et la représente dans la forme de règles floues si-alors. Dans la phase finale, l'algorithme génétique est utilisé pour l'optimisation de ces dernières.	2 / 3 de la base de données ont été utilisés pour l'apprentissage et 1 / 3 pour les tests. La précision de la base de règles et la lisibilité ont été appliquées comme critères de départ pour l'évaluation de l'architecture.	Les résultats des tests de précision des bases de règles <b>Originales :</b> Nombre de règles : 21 TC=80.7% <b>Extraites :</b> Nombre de règles : 8 TC=83.1%
Nesma Settouti, Tadj. E Fandi, M. Amine Chikh. 2009 [SCF09]	Reconnaissance par Neuro-Floue du Diabète	Le modèle de Takagi-sugeno est adopté pour le réseau neuro-flou dans le cadre de la reconnaissance du diabète, avec l'application de différentes configurations en collaboration avec des experts. L'apprentissage est appliqué sur 262 personnes et a permis une optimisation des fonctions d'appartenance en modifiant ses points modaux.	Les performances du classifieur ont été testées sur les 130 patientes restantes qui englobent 91 non diabétiques et 39 diabétiques.	Age, Glu : Nbre règles=4 Tc= 86.92%, Age, Glu, BMI, PED : Nbre règles=24 Tc= 84.17% Age, Glu, BMI : Nbre règles=8 Tc= 82.17%

Tableau 2.1 : Quelques travaux avec ANFIS [Set11]

## ***2. Algorithmes génétiques :***

L'avènement des métaheuristiques a permis de résoudre plusieurs problèmes pour lesquels il n'existe pas de méthodes spécifiques. Les algorithmes génétiques, la programmation génétique, les stratégies d'évolution, et ce que l'on appelle maintenant en général les algorithmes évolutionnaires, sont des techniques d'optimisation stochastiques inspirées de la théorie de l'évolution selon Darwin. Les algorithmes évolutionnaires consistent à faire évoluer une population à l'aide de différents opérateurs: sélection, croisements et mutations, ils sont utilisés pour résoudre des problèmes d'optimisation difficile. Nous donnons ici une vision globale de ces techniques, en insistant sur l'extrême flexibilité du concept d'évolution artificielle. Cet outil a un champ très vaste d'applications, qui ne se limite pas à l'optimisation pure. Leur mise en œuvre se fait cependant au prix d'un coût de calculatoire important, d'où la nécessité de bien comprendre ces mécanismes d'évolution pour adapter et régler efficacement les différentes composantes de ces algorithmes. Par ailleurs, on note que les applications-phares de ce domaine sont assez souvent fondées sur une hybridation avec d'autres techniques d'optimisation.

Les algorithmes évolutionnaires ne sont donc pas à considérer comme une méthode d'optimisation concurrente des méthodes d'optimisation classiques, mais plutôt comme une approche complémentaire.

Les algorithmes génétiques sont le type d'algorithme le plus connue et le plus utilisé des algorithmes évolutionnaires ; développée dans les années 70 par J.Holland puis approfondis par Goldberg en 1989, les algorithmes génétiques possèdent la particularité qu'ils font évoluer des populations d'individu codés en binaire, ils utilisent généralement un opérateur de sélection proportionnel, un remplacement générationnel et l'opérateur de croisement comme opérateur principale c'est-à-dire les algorithmes génétiques donnent de forts taux de recombinaison et de faible taux de mutation Goldberg propose dans son livre 80% de croisement et 3% de mutation.

### ***2.1 Principe de fonctionnement :***

Nous essayons ici de traiter le principe de fonctionnement des algorithmes génétiques fondés sur le Néo-Darwinisme, c'est-à-dire l'union de la théorie de l'évolution et de la génétique moderne. Ils s'appuient sur différentes techniques dérivées de cette dernière : croisements, mutation, sélection..., Le principe de fonctionnement d'un algorithme génétique

est identique à celui d'un algorithme évolutionnaire à la différence qu'un algorithme génétique utilise un codage binaire et un remplacement générationnel.

Donc le génotype est constitué d'une chaîne de symboles binaires, le phénotype est une solution du problème dans une représentation naturelle. Le génotype subit l'action des opérateurs génétiques : sélections et variations, tandis que le phénotype ne sert qu'à l'évaluation de la performance d'un individu. [Siar03]

Un algorithme génétique recherche le ou les extrema d'une fonction définie sur un espace de données. Pour l'utiliser, on doit disposer les cinq éléments suivants :

- 1) Un principe de codage de l'élément de population. Cette étape associe à chacun des points de l'espace d'état une structure de données. Elle se place généralement après une phase de modélisation mathématique du problème traité. La qualité du codage des données conditionne le succès des algorithmes génétiques. Les codages binaires ont été très utilisés à l'origine. Les codages réels sont désormais largement utilisés, notamment dans les domaines applicatifs pour l'optimisation de problèmes à variables réelles.
- 2) Un mécanisme de génération de la population initiale. Ce mécanisme doit être capable de produire une population d'individus non homogène qui servira de base pour les générations futures. Le choix de la population initiale est important car il peut rendre plus ou moins rapide la convergence vers l'optimum global. Dans le cas où l'on ne connaît rien du problème à résoudre, il est essentiel que la population initiale soit répartie sur tout le domaine de recherche.
- 3) Une fonction à optimiser. Celle-ci retourne une valeur appelée fitness ou fonction d'évaluation de l'individu.
- 4) Des opérateurs permettant de diversifier la population au cours des générations et d'explorer l'espace d'état. L'opérateur de croisement recompose les gènes d'individus existant dans la population, l'opérateur de mutation a pour but de garantir l'exploration de l'espace d'états.
- 5) Des paramètres de dimensionnement : la Taille de la population, nombre total de générations ou critère d'arrêt, probabilités d'application des opérateurs de croisement et de mutation.

Les algorithmes génétiques fournissent des solutions aux problèmes n'ayant pas de solutions calculables en temps raisonnable de façon analytique ou algorithmique. Le principe

général du fonctionnement d'un algorithme génétique est représenté par la figure suivante (voir figure 2.3) :

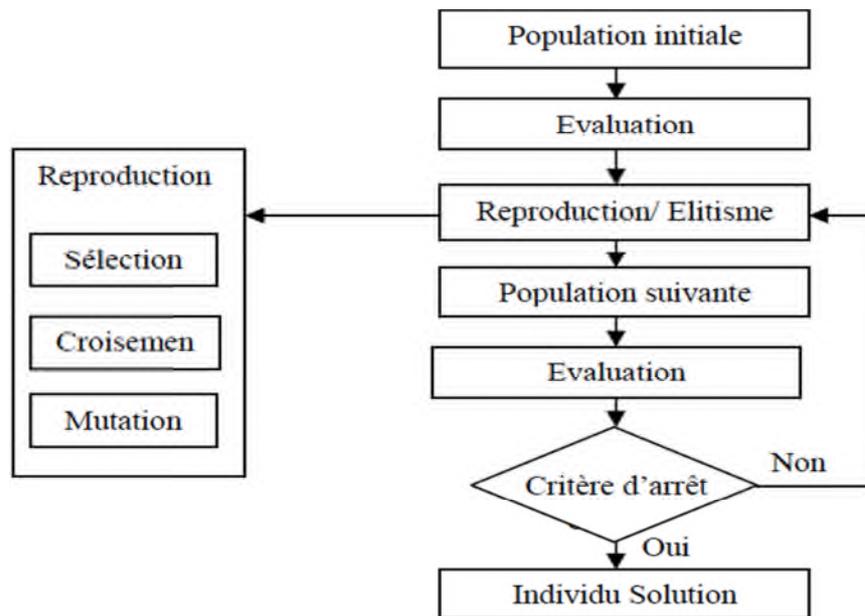


Figure 2.3 Schéma de principe des AG.

Structure d'un algorithme génétique canonique :

1. Initialiser la population initiale  $P(0)$  aléatoirement et mettre  $i = 0$ ;
2. REPETER
  - a) Evaluer la fitness pour chaque individu dans  $P(i)$ ;
  - b) Sélection des Parents dans  $P(i)$  par le calcul de leur fitness ;
  - c) Appliquer l'opérateur de croisement sur  $P(i)$  pour sélectionner les parents;
  - d) Appliquer l'opérateur de mutation pour la reproduire de nouveaux individus;
  - e) Remplacer les Anciens de  $P$  par leurs Descendants (progéniture) pour la création de la génération  $P(i + 1)$ ;
3. jusqu'à un critère d'arrêt satisfaisant.

En effet, l'utilisation des algorithmes génétiques, ne requiert pas la connaissance de la nature du problème, il est seulement nécessaire de fournir une fonction permettant de coder une solution sous forme de gènes (et donc de faire le travail inverse) ainsi que de fournir une fonction permettant d'évaluer la pertinence d'une solution au problème donné. [BENDIB08]

### 2.1.1 Le codage :

La première étape est de définir et de coder convenablement le problème. Cette étape associe une structure de données à chacun des points de l'espace de recherche. Elle se place généralement après une phase de modélisation mathématique du problème traité.



Figure 2.4- le codage.

Le codage permet de représenter l'individu sous la forme d'un chromosome. Ce chromosome est constitué de gènes qui prennent des valeurs dans un alphabet binaire ou non. Le choix du codage des données dépend de la spécificité du problème traité, L'efficacité de l'algorithme génétique va donc dépendre du choix du codage d'un chromosome.

### 2.1.2 La fonction d'évaluation (*fitness*) :

Pour mieux concrétiser le processus d'évolution, il est nécessaire de pouvoir faire la distinction entre les chromosomes les plus adaptés et les moins adaptés. Ceci est possible par l'assignation d'une valeur d'adaptation à chaque chromosome. Ce qui permet d'évaluer si un individu est mieux adapté qu'un autre à son environnement, Ainsi on peut faire une comparaison entre les individus.

Cette fonction, propre au problème, est souvent simple à formuler lorsqu'il y a peu de paramètres. Au contraire, lorsqu'il y a beaucoup de paramètres ou lorsqu'ils sont corrélés, elle est plus difficile à définir. Dans ce cas, la fonction devient une somme pondérée de plusieurs fonctions.

La mise au point d'une bonne fonction d'adaptation (*fitness function*) doit respecter plusieurs critères qui se rapportent à sa complexité ainsi qu'à la satisfaction des contraintes du problème. Si le problème doit satisfaire des contraintes et que les chromosomes produits par les opérateurs de recherche codent les individus invalides, une solution parmi d'autres est

d'attribuer une mauvaise fitness à l'élément qui a violé les contraintes afin de favoriser la reproduction des individus valides.

### ***2.1.3 Génération de la population initiale:***

Le choix de la population initiale d'individus conditionne fortement la rapidité de l'algorithme. Si la position de l'optimum dans l'espace d'état est totalement inconnue, il est naturel de générer aléatoirement des individus en faisant des tirages uniformes dans chacun des domaines associés aux composantes de l'espace d'état en veillant à ce que les individus produits respectent les contraintes. Si par contre, des informations à priori sur le problème sont disponibles, il paraît bien évidemment naturel de générer les individus dans un sous-domaine particulier afin d'accélérer la convergence. Dans l'hypothèse où la gestion des contraintes ne peut se faire directement, les contraintes peuvent être incluses dans le critère à optimiser sous forme de pénalités. Ainsi, un individu qui viole une contrainte se verra attribuer une mauvaise fitness et sera donc éliminé par le processus de sélection. Disposant d'une population d'individus non homogène, la diversité de la population doit être entretenue au cours des générations afin de parcourir le plus largement possible l'espace d'état. C'est le rôle des opérateurs de croisement et de mutation. Quel doit être la taille de cette population ?

Une population trop petite évoluera probablement vers un optimum local peu intéressant. Une population trop grande sera inutile car le temps de convergence sera excessif. La taille de la population doit être choisie de façon à réaliser un bon compromis entre temps de calcul et qualité du résultat.

Mais il faut être conscient que cette taille de population dépend de la puissance de calcul dont on dispose, des méthodes utilisées (sélection, opérateurs génétiques...), du nombre de variables considérées et de la fonction d'adaptation. Si la fonction à optimiser comporte peu d'optima locaux et un optimum global net, la population nécessaire sera plus petite que dans le cas d'une fonction beaucoup plus compliquée comportant de nombreux optima locaux.

### ***2.1.4 Sélection:***

La sélection permet de déterminer la capacité de chaque individu à persister dans la population et à se diffuser. En règle générale, la probabilité de survie d'un individu sera directement reliée à sa performance relative au sein de la population. La sélection permet

d'identifier statistiquement les meilleurs individus d'une population et d'éliminer les mauvais. Il existe plusieurs méthodes de sélection, dont on cite quelques unes :

### 2.1.4.1 Sélection proportionnelle:

Ce type de sélection a été conçu à l'origine par J.Holland. Le nombre espéré de sélection d'un individu est proportionnelle à sa fitness, un individu ayant une forte valeur d'adaptation à alors plus de chance à être sélectionner, pour mettre en évidence cette méthode de sélection, deux techniques sont définit :

#### La méthode RWS (Roulette Wheel Selection)

La méthode RWS s'inspire des roulettes de casino (voir figure 2.5), à chaque individu on associe un secteur d'une roue, l'angle du secteur est proportionnel a la qualité de l'individu, on fait tourné la roue, l'individu sélectionné est désigné par l'arrêt de la bille sur sa case.

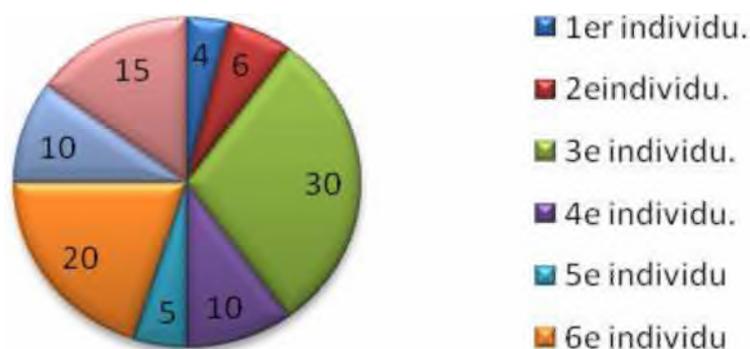


Figure 2.5 modèle de la roulette. [BENDIB08]

#### La méthode SUS (Stochastic Universal Simpling)

On considère un segment de droite partitionné en autant de zones qu'il y a d'individu dans la population, chaque zone étant proportionnel à la performance.

On place sur ces zones des points équidistants, un individu est sélectionné s'il possède au moins un point sur ça zone.

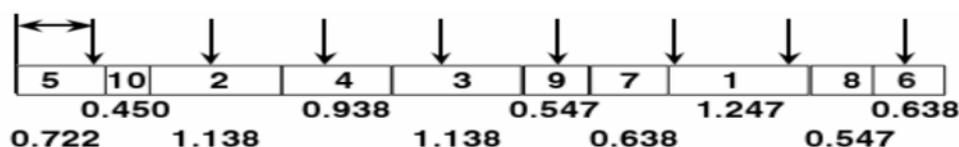


Figure 2.6- méthode SUS. [Siar03]

### ***2.1.4.2 Sélection par rang:***

Pour ce type de sélection on associe à chaque individu un rang proportionnellement à leur qualité. Les individus de moins bonne qualité obtiennent un rang faible (à partir de 1), Le rang N est attribué au meilleur individu (N étant la taille de la population). La suite de la méthode consiste uniquement en l'implémentation d'une roulette basée sur les rangs des individus. L'angle de chaque secteur de la roue sera proportionnel au rang de l'individu qu'il représente

### ***2.1.4.3 Sélection par tournoi:***

Le principe de la sélection par tournoi consiste à choisir k individus aléatoirement, l'individu sélectionné est le vainqueur du tournoi c'est-à-dire celui qui possède la meilleure performance, donc il y aurait autant de tournoi que d'individu à sélectionner.

### ***2.1.5 Les opérateurs de variation (reproduction):***

Les opérateurs de variations sont classés en deux catégories :

- Les opérateurs de croisement, qui utilisent plusieurs parents (souvent deux) pour créer un ou plusieurs descendants.
- Les opérateurs de mutation qui transforme un seul individu.

Ces deux opérateurs permettent de créer de nouveaux individus dans une population qui héritent des caractéristiques d'individus parents, ils doivent être capables d'assurer deux fonctionnalités importantes durant la recherche d'un optimum :

- Une bonne exploration de l'espace de recherche, afin d'en découvrir les régions qui ont de grandes chances de contenir les optimums globaux.
- Une bonne exploitation de ces régions afin de découvrir les optimums avec la précision requise.

### 2.1.5.1 Croisement:

Il existe trois variantes de croisement :

- le croisement "un point".
- le croisement "deux points".
- le croisement uniforme.

Un couple d'individu est constitué par tirage aléatoire de la population, Le croisement "un point" se déroule en deux étapes :

1. choix aléatoire d'un point de coupure identique sur les 2 chaînes binaire.
2. coupure des 2 chaînes et échange des 2 fragments situés à droite.

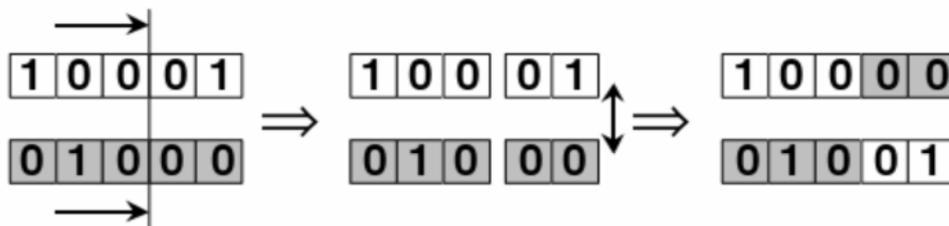


Figure 2.7- Croisement un point. [Siar03]

Ce processus produit deux descendants à partir de 2 parents, une généralisation de cette méthode consiste à choisir deux ou plusieurs points de coupure, la figure 2.7 illustre un croisement deux points.

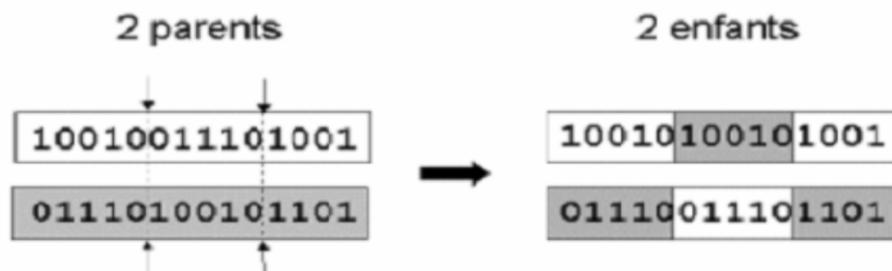


Figure 2.8- Croisement deux points. [Siar03]

Dans le croisement uniforme, on utilise un "masque de croisement" qui est un mot binaire de même taille que les individus, si la nième position du masque est "0" on laisse inchangés les symboles à la nième position des 2 chaînes, un "1" déclenche un échange des symboles correspondants. Le masque est engendré aléatoirement pour chaque couple d'individus, ce principe est illustré dans la figure 2.8 suivante.

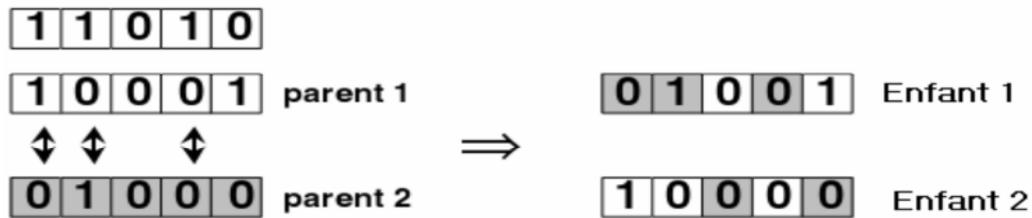


Figure2.9- Croisement uniforme. [Siar03]

### 2.1.5.2 Mutation:

L'opération de mutation consiste à modifier la valeur d'un génotype aléatoirement avec un faible taux de mutation, il existe plusieurs variantes dans les plus connues sont la mutation déterministe et la mutation bit-flip, la mutation bit-flip consiste à choisir un bit dans la chaîne binaire et l'inverser tandis que la mutation déterministe consiste à choisir au hasard un nombre fixé de bits et a les inverser, l'exemple de la figure 2.9 montre que le 4ime bit est modifié.

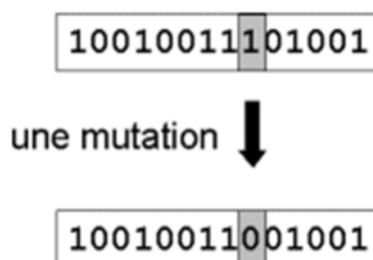


Figure 2.10- Croisement bit-flip. [Saidi08]

L'opérateur de mutation modifie donc de manière complètement aléatoire les caractéristiques d'une solution, ce qui permet d'introduire et de maintenir la diversité au sein

de notre population de solutions. Cet opérateur joue le rôle d'un "élément perturbateur", il introduit du "bruit" au sein de la population. En effet, une mutation pouvant intervenir de manière aléatoire au niveau de n'importe quel locus, on a la certitude mathématique que n'importe quelle permutation de notre chaîne de bits peut apparaître au sein de la population et donc que tout point de l'espace de recherche peut être atteint. On notera que la mutation règle donc le problème exposé après le croisement.

### ***2.1.6 Test d'arrêt:***

Le test d'arrêt joue un rôle primordial dans le jugement de la qualité des individus. Son but est de nous assurer l'optimalité, de la solution finale obtenue par l'algorithme génétique.

Les critères d'arrêts sont de deux natures :

1. Arrêt après un nombre fixé a priori de générations. C'est la solution retenue lorsqu'une durée maximale de temps de calcul est imposée.
2. Arrêt lorsque la population cesse d'évoluer ou n'évolue plus suffisamment. Nous sommes alors en présence d'une population homogène dont on peut penser qu'elle se situe à la proximité de l'optimum. Ce test d'arrêt reste le plus objectif et le plus utilisé.

Il est à noter qu'aucune certitude concernant la bonne convergence de l'algorithme n'est assurée. Comme dans toute procédure d'optimisation l'arrêt est arbitraire, et la solution en temps fini ne constitue qu'une approximation de l'optimum. [saidi08] [Siar03]

Après avoir une idée générale sur la notion ou bien le mode de fonctionnement des algorithmes génétiques, dans ce qui suit nous allons présenter quelques approches qui se trouvent dans la littérature abordant le thème de l'hybridation des algorithmes génétiques avec les classifieurs flous.

### ***3. L'apport des Algorithmes génétique dans la classification flou :***

Les techniques d'intelligence Artificielles telles que la logique floue, les réseaux neurones, et les algorithmes génétiques (GA) sont des domaines populaires de recherches, puisqu'elles peuvent confronter des problèmes complexes de technologie. Les algorithmes génétiques sont des algorithmes de recherche, basés sur la génétique naturelle qui offrent des possibilités fortes de recherche dans les espaces complexes. L'idée fondamentale est de

préserver une population des chromosomes qui évolue par un processus de concurrence et de variation commandée. D'autre part, les règles floues sont une collection de rapports linguistiques qui décrivent comment un système d'inférence flou devrait prendre une décision concernant la classification d'une entrée. Elles combinent deux ou plus d'entrées floues et associent à eux un résultat. Beaucoup de chercheurs ont employé GA pour optimiser les systèmes basés sur les règles floues (FRBS), qui sont connus en tant que système basé sur les règles floues génétique (GFRBS).

### 3.1 Approche (*GF-Miner*) : [VNY09]

Cette approche propose un classificateur flou génétique sous le nom **GF-Miner** qui est basé sur un système qui utilise les règles floues pour des données numériques, **Fuzzy Miner** qui a été proposé dans le travail de [PTK05]. **Fuzzy Miner** applique une méthode floue heuristique pour la classification des données numériques. **GF-Miner** adopte une méthode de groupement non supervisé améliorée pour réussir une division floue plus normale de l'espace d'entrée. En outre, un processus génétique est conçu qui se débarrasse des règles inutiles de l'ensemble de règles initial et raffine en même temps la base des règles en éliminant des termes inutiles des règles floues. En soi, le **GF-Miner** optimise non seulement le nombre de règles produites mais également leur taille construisant de petites règles avec des différentes tailles qui sont plus compréhensibles par l'utilisateur et obtiennent une précision plus élevée de classification. Alors le **GF-Miner** permet à l'analyste de spécifier un différent nombre d'ensembles flous par variable d'entrée. En second lieu, le nombre de règles produites est optimisé par Algorithme Génétique et il n'est pas prédéterminé par l'utilisateur. Troisièmement, une fonction fitness plus efficace est adoptée, suivant les indications des expériences, il obtient de meilleurs résultats en termes de taux de classification, alors que ceci est réussi en employant une base de règles plus petite.

Les principales contributions dans ce travail sont :

a) **Fuzzy Miner** [PTK05] est amélioré en proposant une nouvelle méthode de partition floue d'espace d'entrée utilisant Fuzzy-C-Means (FCM) l'algorithme de regroupement (clustering).

b) un algorithme génétique (c.-à-d. *GA<sub>sr</sub>*) est convenablement conçu pour la réduction de la taille des règles qui sont produites par le **Fuzzy Miner** amélioré.

c) le nombre de règles dans la base règles est réduit en employant un deuxième algorithme génétique (c.-à-d. *GAnr*).

d) l'efficacité de cette approche a été démontrée par une évaluation expérimentale étendue utilisant l'ensemble de données (IRIS benchmark).

### 3.1.1 *GF-Miner* comme extension de *Fuzzy Miner*:

**Fuzzy Miner** se compose de quatre composants principaux (voir figure 2.10) : une interface de fuzzification, une base de connaissance, Système d'inférence flou (une logique de prise de décision) et une interface defuzzification. Le **GF-Miner** est basé sur le **Fuzzy Miner** [PTK05]. Plus spécifiquement, il améliore **Fuzzy Miner** en incorporant une manière plus flexible et non supervisé pour diviser l'espace d'entrée en ensembles flous et en optimisant son sortie avec l'utilisation d'Algorithme Génétique.

L'architecture du **GF-Miner** est montrée sur le schéma suivant.

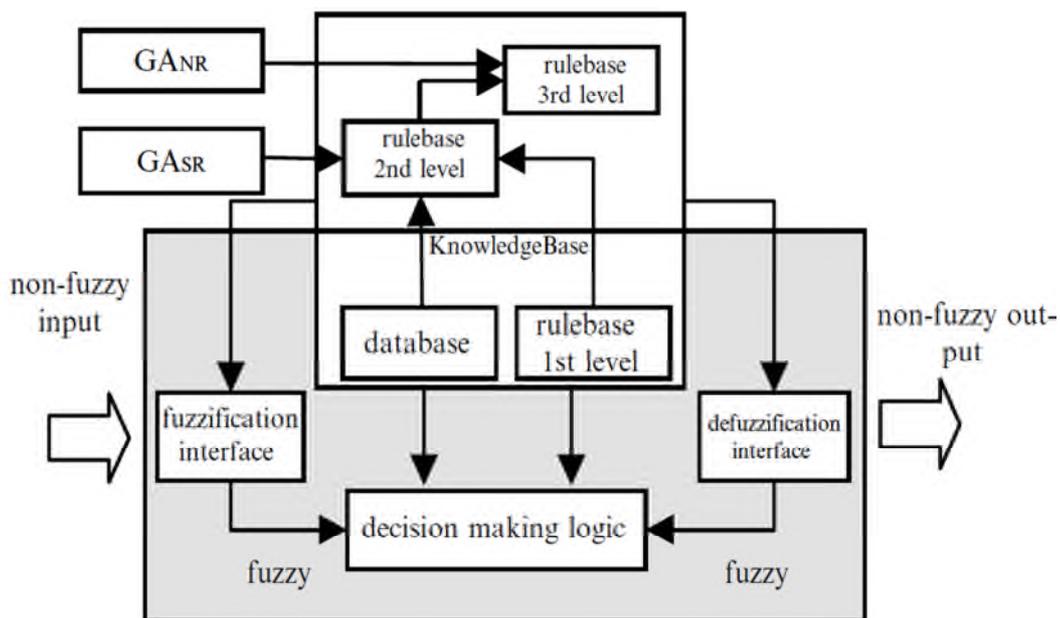


Figure. 2.11. Architecture *GF-Miner*

Base de connaissance (KnowledgeBase): se compose d'une base de données (database) et la base règles (rulebase). Il y a deux facteurs qui déterminent une base de données, c.-à-d., une partition floue de l'espace d'entrée et les fonctions d'appartenance des ensembles flous des antécédents.

Le **GF-Miner** supporte deux types de fonctions d'appartenance, (triangulaire et trapézoïdale). Pour définir la fonction d'appartenance en GF-Miner emploi une manière non supervisé (l'algorithme de regroupement de FCM). Dans le tableau 1 nous montrons comment nous employons les centres de surface des clusters  $V_i = \{V_1, V_2, \dots, V_{K_i}\}$ , où  $K_i$  est le nombre d'ensembles flous pour la variable d'entrée  $X_i$ , pour déterminer les paramètres pour chaque ensemble flou (voir tableau 2.2):

	Triangular	Trapezoidal
First fuzzy set:	$-\infty, x_{\min}, V_2$	$-\infty, -\infty, x_{\min}, V_2$
The next fuzzy sets: For $j=1$ to $K_i-2$ :	$V_j, V_{j+1}, V_{j+2}$	$V_j, V_{j+1}-(V_{j+1}-V_j)/3, V_{j+1}+(V_{j+2}-V_{j+1})/3, V_{j+2}$
Last fuzzy set:	$V_{K_i-1}, x_{\max}, +\infty$	$V_{K_i-1}, x_{\max}, +\infty, +\infty$

Tableau2.2: Paramétrés pour fonctions d'appartenance.

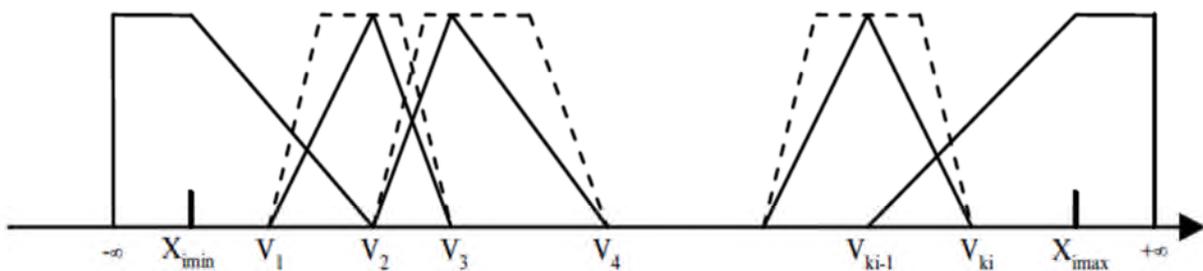


Figure 2.12 Fonction d'appartenance (triangulaire et trapézoïdale)

Ce qui nous intéresse dans ce travail c'est le processus génétique utilisé.

### 3.1.2 Le processus génétique dans le GF-Miner :

Dans ce papier on trouve l'utilisation de deux phases d'algorithme génétique pour optimiser les règles qui a été créés, **GA<sub>sr</sub>** (Génétique Algorithm Size Rule) réduit la taille des règles dans la base règles initial sans éliminer aucune d'entre elles, alors que **GA<sub>nr</sub>** (Génétique Algorithm Number Rule)réduit le nombre de règles qui ont été produites après que l'algorithme de **GA<sub>sr</sub>** ait été appliqué.

### ***GAsr* algorithme :**

*Représentation individuelle* : Le chromosome d'un individu représente la partie antécédente de la règle.  $K$  le nombre de règles produites à l'étape précédente. Alors un chromosome se compose de  $K$  gènes, où chaque gène correspond à une règle. Chaque gène  $i$  est divisé dans  $n$  champs binaires où  $n$  est le nombre de variables d'entrée. Nous employons 0 quand la variable d'entrée spécifique n'est pas importante pour la règle.

Le **GAsr** procède comme suit :

*Population initiale* : Il est produit aléatoirement et nous présentons en plus un chromosome qui représente toutes les règles, celui est tous les gènes recevrons la valeur 1.

*Fonction de fitness* : C'est le nombre de modèles correctement classifiés par la base règle flou codée dans ci correspondant de chromosome :

$$\mathbf{FitnessFunction} (Ci) = \mathbf{NPC} (Ci)$$

Où NPC (Ci) est le nombre de modèles correctement Classifié.

*Opérateurs génétiques* : Pour le choix il a été employé la sélection par tournoi. En outre, le croisement uniforme a été utilisé parce que cette opération de croisement ne tient pas compte de la position de chaque gène et les modifications sont aléatoirement apportées. La mutation est faite aléatoirement selon la probabilité mutation et transforme 0 à 1 ou 1 à 0. L'état d'arrêt un nombre maximum des générations M. La nouvelle base de règle est représentée par le meilleur chromosome avec la meilleure valeur de fitness dans la dernière génération. En conséquence, ce GA réduit le nombre conditions de règle qui ne sont pas importants, ainsi leur absence n'affectant pas le nombre de modèles correctement classifiés.

### ***GAnr* algorithme :**

Le but de **GAnr** est d'éliminer certaines règles qui se trouvent dans la base de règle floue qui a été créé après **GAsr**, Les chromosomes sont codés comme un ordre des éléments binaires avec la même longueur que le nombre de règles produites à l'étape précédente. Chaque gène est associé à une règle Si l'élément binaire est 1 que la règle est en activité et la règle liée à ce

gène sera dans le final base ; autrement pas. Croisement, mutation et test d'arrêt sont identiques que nous avons employé dans *GAnr*.

Fonction fitness employée :

$$\text{FitnessFunction}(Ci) = CR(Ci)^2 * (L - NR(Ci) + 1)$$

Tel que  $CR(Ci)$  est le taux de classification,  $L$  est le nombre de règles produites à l'étape précédente et  $NR(Ci)$  est le nombre de règles actives. Le de l'utilisation de ce type de la fonction fitness est assuré un taux élevé de classification on diminue le nombre de règles actives.

### 3.1.3 Résultats obtenus :

Cette approche a été évaluée sur l'ensemble de données d'IRIS obtenu à partir du dépôt d'UCI des bases de données d'apprentissage [Rep98], qui se compose de 50 échantillons provenant de chacune de 3 espèces des fleurs d'iris (*setosa*, *virginica* et *versicolor*) et 4 attributs ont été mesurés pour chaque échantillon : la longueur et la largeur du sépale et du pétale.

L'expérimentation a été effectuée sur 02 partitions de la base d'IRIS :

**Partition A (70%) :** 105 exemples pour l'apprentissage et 45 exemples pour test.

**Partition B (50%) :** 75 exemples pour l'apprentissage et 75 exemples pour test.

Il a été employé trois ensembles flous pour chaque variable avec la fonction d'appartenance triangulaire. Le taux de classification et le nombre de règles sont calculés comme la moyenne après 10 itérations. Nous montrons également le minimum et les valeurs maximum réalisés par chaque partition.

Le tableau 2.3 suivant représente les résultats obtenus et comparaisons avec d'autres travaux qui se trouvent dans la même littérature [PH05] et [CL06]

Tableau 2.3 Résultats expérimentales

Partition	Approche	Taux classification	# Règles
<b>A</b>	GF-Miner	Avg: 97,56 Max: 97,78Min: 95,56	Avg: 4,9 Max: 7 Min: 3
	Fuzzy Miner	95,56	15
	[PH05]	96,9	4
<b>B</b>	GF-Miner	Avg: 97,87 Max: 98,67 Min: 97,33	Avg: 4,6 Max: 6 Min: 4
	Fuzzy Miner	93,33	14
	[CL06]	96,33	3

### 3.2 Approche (un cadre général pour la conception) :

Dans [GVGB09] propose un cadre général pour la conception d'un système de classification fondé sur des règles floues est proposé. L'Algorithme Génétique développé partitionne l'espace de recherche et permet de fixer à la fois la structure et le calcul des paramètres du classifieur. Des simulations sur 4 bases de données (vote, cancer du sein, Diabète, inspection de surface) ont montré que les performances obtenues par le classifieur sont comparables, voire supérieures à la meilleure performance réalisée par l'utilisation d'autre approche.

L'approche comprend les étapes suivantes :

- 1- Diviser l'ensemble de données en apprentissage et test
- 2- Grouper l'ensemble des données d'apprentissage en appliquant évolution SOM Tree.
- 3- Basé sur l'arbre évolué, produire une population des sous-arbres. Chaque sous-arbre définit la structure initiale d'un classificateur basé sur les règles flou. La génération est accomplie en coupant aléatoirement des branches de l'arbre développé dans l'étape 2. Le découpage se produit approximativement entre 25 et 75% de la profondeur d'arbre.
- 4- Représenter chaque nœud dans la population de sous-arbre par un ensemble d'ensembles flou avec les fonctions gaussiennes d'appartenance.
- 5- Coder la structure de chaque sous-arbre de la population dans un chromosome séparé.
- 6- Prendre un sous-arbre (classificateur d'une structure donnée) de la population de sous-arbre et coder les attributs (dispositifs) (utilisés/non utilisés) et les paramètres des

fonctions d'appartenance du classificateur dans un chromosome. En codant, permettre la sélection d'attributs indépendamment pour chaque règle flou.

- 7- Produire une population des chromosomes codant différents classificateurs de la structure donnée. Les différents classificateurs diffèrent en d'attributs (dispositifs) et valeurs des paramètres.
- 8- appliquer l'algorithme LVQ-3 modifié aux individus de la population et examiner fitness des individus.
- 9- Appliquer les opérations génétiques (aux dispositifs et aux paramètres) et produire une nouvelle population.
- 10- Répéter les étapes 8-9 jusqu'à la convergence.
- 11- Prendre le meilleur individu de la structure donnée.
- 12- Répéter les étapes 6-11 à la population entière des sous-arbres.
- 13- Appliquer les opérations génétiques (à la structure des sous-arbres) et produire une nouvelle population des sous-arbres.
- 14- Répéter les étapes 6-13 pour un nombre donné de générations.

Maintenant, nous décrivons brièvement les matières principales de la technique. Comme SOM, l'arbre en évolution de SOM [JJO04] présente la propriété de self-organisation. La structure arborescente en évolution permet à l'arbre de SOM de traiter efficacement des problèmes de large échelle. D'ailleurs, il n'y a aucun besoin de choisir la taille de carte à l'avance. Comme dans SOM ordinaire, chaque nœud de l'arbre de SOM a vecteur de poids WI. Quand l'apprentissage d'arbre, pour chaque vecteur X de formation, la meilleure unité d'assortiment (BMU) est trouvée par une recherche d'arbre avide. BMU est toujours un nœud de feuille.

### ***3.2.1 Résultats expérimentales :***

Dans la série d'expériences, la sélection des descripteurs a été activée et les caractéristiques propres de chaque patiente ont été trouvées par l'algorithme génétique (AG). La classification a été obtenue en utilisant les paramètres sélectionnés ainsi que la totalité des 8 paramètres pour des fins de comparaison.

1. Dans le premier ensemble d'expériences, la sélection d'attributs n'a pas été appliquée et l'exactitude de classification obtenue à partir du classificateur basé sur les règles floues a été comparée à la précision réalisée par d'autres techniques. Le perceptron

multicouche (MLP), le k-NN, et les classificateurs LVQ-3 ont été employés pour la comparaison (voir tableau 2.4).

Data set \ classifier	k-NN	MLP	LVQ-3	Proposed
Voting	91.24	93.78	77.41	94.68
Breast cancer	88.73	97.18	75.87	98.54
Diabetes	71.19	71.49	64.22	71.92
Surface inspection	77.30	81.63	78.13	84.13

Tableau 2.4. Taux classification obtenus (utilisant toutes les attributs)

2. Dans le prochain ensemble d'expériences, la sélection d'attributs a été activée et des attributs spécifiques pour chaque règle ont été trouvés par la recherche génétique.

Data set \ features	All features	Selected features
Voting	94.68	98.68
Breast cancer	98.54	99.02
Diabetes	71.92	75.92
Surface inspection	84.13	99.63

Tableau 2.5 Taux classification obtenus (comparaison avec l'approche sélection des attributs)

Data set	# Rules	# Features
Voting	15 (25)	05–09 (16)
Breast cancer	10 (14)	12–20 (30)
Diabetes	10 (16)	02–08 (8)
Surface inspection	09 (13)	03–04 (5)

Tableau 2.6 Nombre des règles et d'attributs utilisant les différents ensembles donnés.

Nous remarquons dans ce travail que la classification avec les paramètres sélectionnés par l'AG donnent de meilleurs résultats, avec une diminution de nombre de règles activés sans détériorer taux de classification. D'ailleurs, taux de classification des données de test a augmenté en raison de la réduction.

#### 4. Synthèse :

L'identification de structure de système et l'optimisation de paramètre sont deux thèmes principaux à considérer en concevant un système basé sur les règles floues. La partition floue, le choix variable, et le raisonnement flou sont les tâches qui doivent être résolues pour identifier la structure du système. Diverses approches ont été employées pour traiter les deux principales issues de conception de système basé sur les règles floues. La structure initiale de système, souvent nommée comme la partition floue, est habituellement identifiée à travers K-Means, Fuzzy C-Means (FCM) comme dans [Bez81], Learning Vector Quantization (LVQ) ou SOM-based-clustering [ZK07], aussi bien que le groupement par accroissement incrémental (clustering) ou en construisant un arbre de décision.

Il semble que la descente de gradient, de correction d'erreurs et la recherche génétique sont les techniques d'optimisation les plus populaires de paramètre utilisées dans diverses études. L'optimisation combinée de la structure et les paramètres a été également considérée en appliquant les algorithmes génétiques (GA), apprentissage non supervisé et par renforcement, ou l'heuristique simple. Dans [TRK07] le processus de la recherche génétique se concentre sur les points de données « durs » en assignant un poids plus élevé à de tels points. Dans [HY07], l'optimisation multi-objective génétique a été appliquée pour concevoir un système basé sur les règles floues. La tâche était de maximiser  $f1(s)$ , réduire au minimum le  $f2(s)$ , et réduire au minimum le  $f3(s)$ , où  $S$  est un ensemble de règles floues,  $f1$  représente l'ensemble des échantillons qui sont correctement classifiés,  $f2$  le nombre de règles floues dans  $S$ , et le  $f3$  est le nombre de toutes les conditions antécédentes dans  $S$ . Ainsi, le  $f3$  peut être considéré comme la longueur totale de règle. On note que l'optimisation dans ce travail commence par toutes les règles possibles dans l'espace de recherche défini par le processus d'apprentissage.

En parcourant la littérature scientifique impliquant les différentes méthodes, on observe que la connaissance réalisée a été mise à l'écart, ce qui limite la plus d'entre eux juste sur les performances de classification sans montrer la capacité d'interprétation des résultats. Rares sont les cas où la connaissance réalisée est abordée mais demeure inexploitée.

Notre travail dans le cadre de ce mémoire de magister consiste à réaliser un système d'aide au diagnostic médical, il s'oriente vers une caractérisation et une reconnaissance interprétable des données du diabète. La transparence et la lisibilité sont des objectifs principaux des systèmes d'aide au diagnostic médical, notre objectif s'oriente vers la réalisation d'un système d'apprentissage automatique structurel et paramétrique en utilisant les

algorithmes génétiques afin d'augmenter les performances du notre classifieur floue sur le plan précision et interprétabilité en renforçant la connaissance extraite.

### ***5. Conclusion :***

Dans la littérature et en particulier dans le domaine médical. Les méthodes dites intelligentes comme les réseaux de neurones, la logique floue, les algorithmes génétiques et leurs hybridations ont été largement utilisés. L'utilisation de la logique floue dans le domaine de la classification permet d'avoir des systèmes lisible et interprétable qui permet de préserver la connaissance extraite .Au contraire des systèmes basé sur les réseaux de neurones (dites boîtes noires) l'apport des algorithmes génériques peut être d'une grande importance dans l'optimisation de la connaissance.

## Chapitre 03

### Implémentation

#### *Introduction*

La mise en œuvre en pratique c'est le moyen le plus efficace pour caractériser et acquérir un concept théorique. Notre cadre de travail dès le début se base sur l'élaboration d'un système d'aide au diagnostic médicale (classifieur flou), nous utilisons la logique floue due à sa capacité de traiter un modèle linguistique interprétable et lisible. Sachant qu'il existe un autre concept très important dans ce domaine qui est la précision, alors nous sommes confrontés à un problème du compromis entre l'interprétabilité et la précision. En parcourant l'état de l'art nous avons choisis l'algorithme génétique pour améliorer la robustesse de notre classifieur (amélioration structurelle et paramétrique). Notre travail comprend deux phases:

- Phase de réalisation de notre classifieur flou.
- Phase d'amélioration structurelle et paramétrique (algorithme génétique).

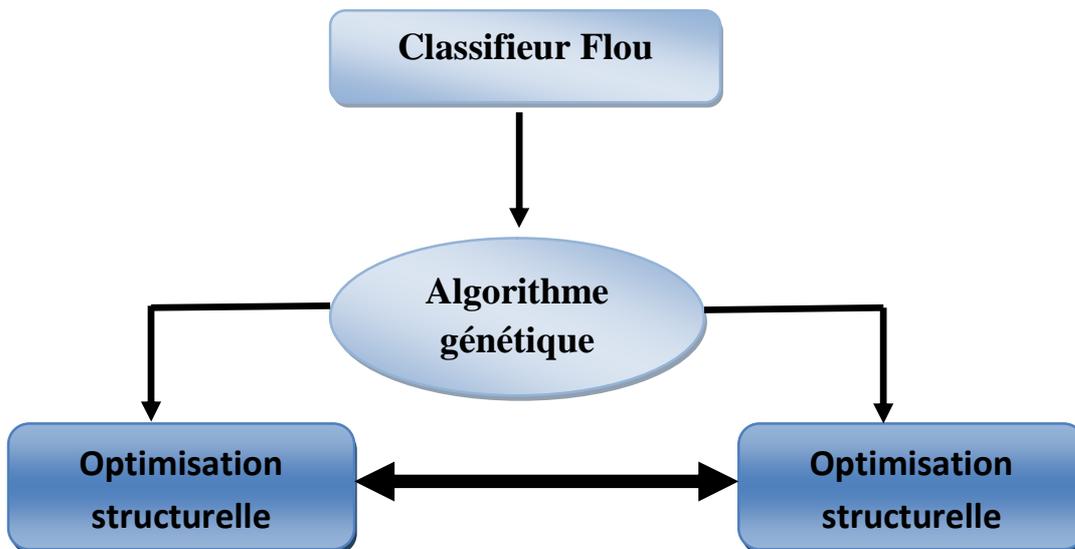


Figure 3.1 – Schéma représentatif de notre procédure

#### *1. Mise en œuvre:*

En vertu des innombrables avantages du langage java. Notre choix de langage de programmation s'est naturellement porté sur ce langage, l'avantage le plus important c'est sa portabilité, la formule (un seul code+une seule compilation = parfaite portabilité).

On a choisi de mettre en place notre application sur un environnement NetBeans.

NetBeans est à l'origine un EDI Java. NetBeans fut développé à l'origine par une équipe d'étudiants à Prague, racheté ensuite par Sun Microsystems. Quelque part en 2002, Sun a décidé de rendre NetBeans open-source. Mais NetBeans n'est pas uniquement un EDI Java. C'est également une plateforme, nous permettant d'écrire nos propres applications Swing. Sa conception est complètement modulaire : Tout est module, même la plateforme. Ce qui fait de NetBeans une boîte à outils facilement améliorable ou modifiable.

### ***1.1 Base de données :***

Dans ce mémoire de magister nous utilisons la base de données médicale réelle (Indians Diabetes Pima). L'ensemble de données a été choisi du dépôt d'UCI [FA10b] qui réalise une étude sur 768 femmes Indiennes Pima, Ces mêmes femmes, qui ont stoppé leurs migrations en Arizona, États-Unis, adoptant un mode de vie occidentalisé, développent un diabète dans presque 50% des cas. [Set11]

Dans les 768 cas présentés Il y a 500 échantillons provenant des patients qui n'ont pas le diabète et 268 échantillons provenant des patients qui sont connus d'avoir le diabète. Le diagnostic est une valeur binaire variable «classe» qui permet de savoir si le patient montre des signes de diabète selon les critères de l'Organisation Mondiale de la Santé. Les huit descripteurs cliniques sont :

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

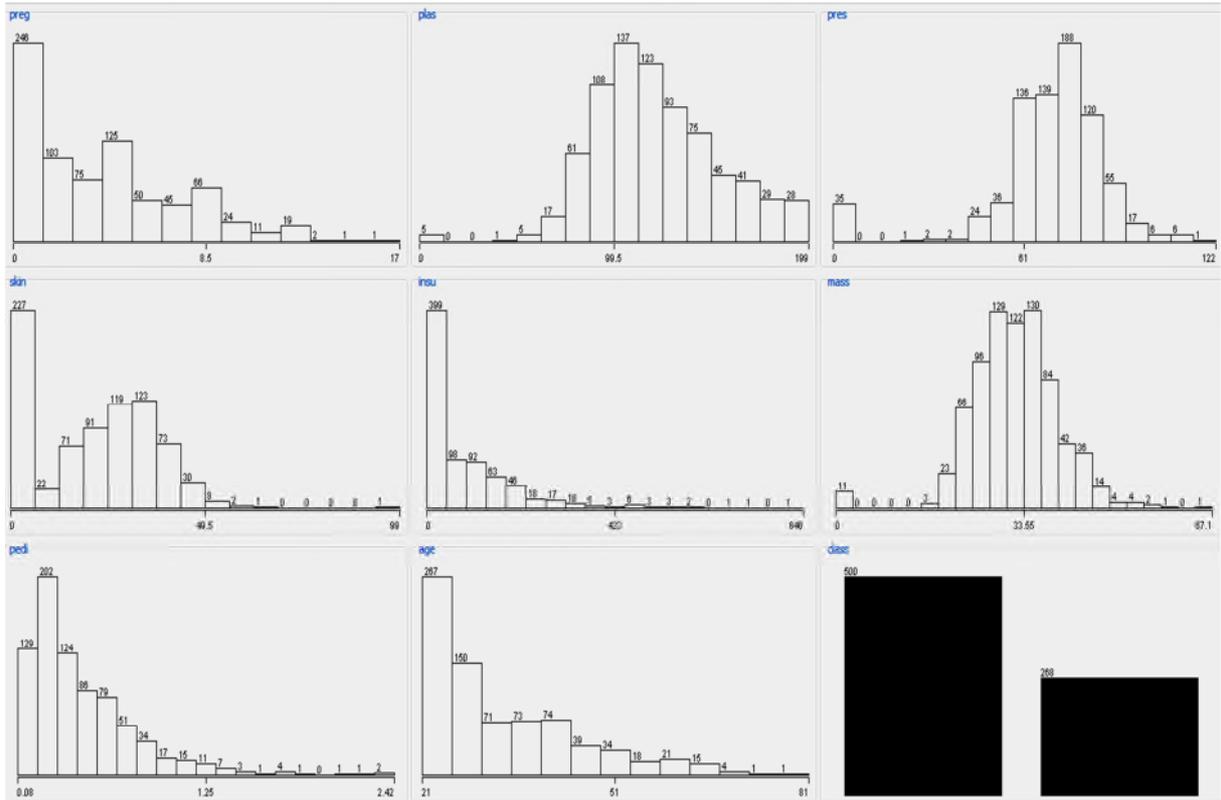


Figure 3.2 - Histogrammes des 9 paramètres de la base de données

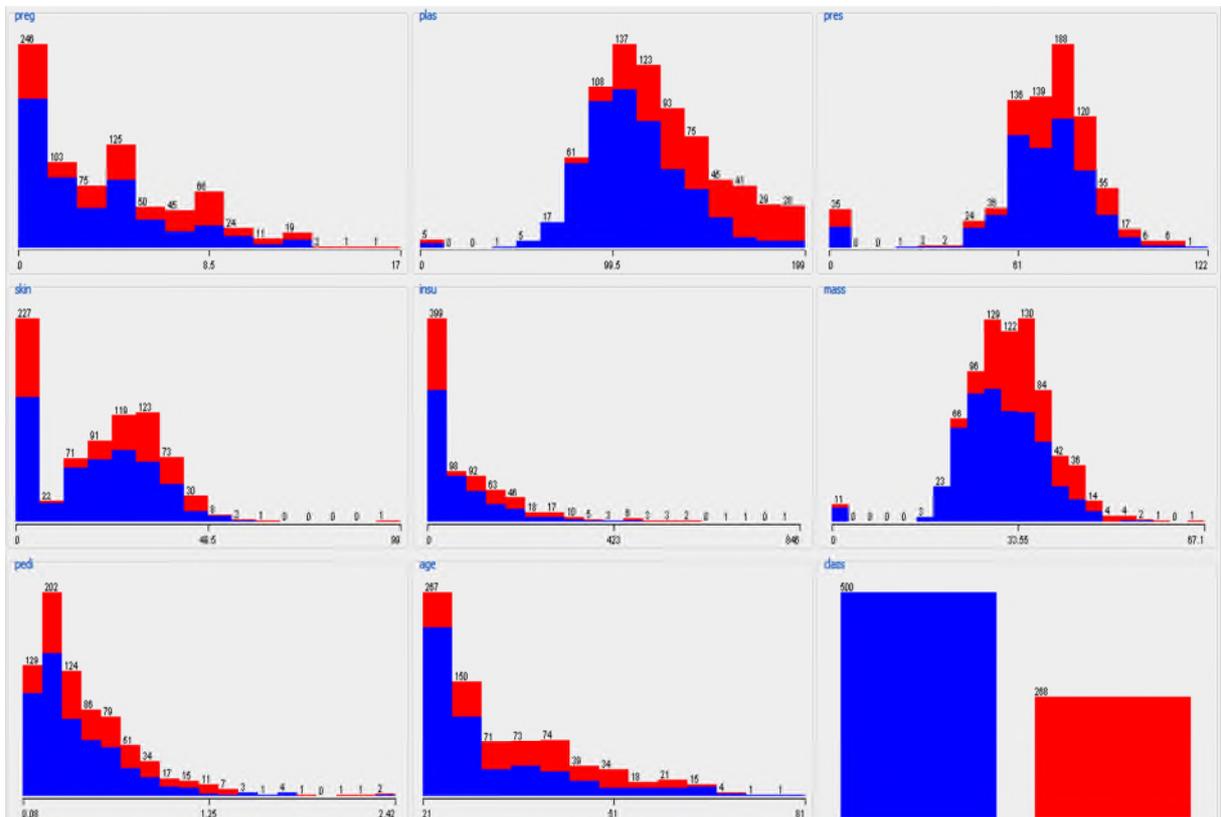


Figure 3.3 - distribution des diabétiques et non diabétiques dans les paramètres de la base.

## 2. Phase1 (construction de notre classifieur) :

Le but de cette première partie de travail est de construire notre classifieur flou (générer les règles floues Si-Alors) à partir des données numériques pour un problème de classification, elle comprend deux phases :

- La partition de l'espace de données en sous-espaces flous.
- La détermination de la règle floue pour chaque sous-espace flou.

La performance de notre système de classification basée sur les règles floues, dépend de choix de la partition flou, si une partition est trop brute la performance peut être dégradé avec la possibilité de plusieurs modèles peuvent être classifiés faux. Et si une partition trop étroite, beaucoup des règles ne peuvent pas être produites en raison du manque de modèle d'apprentissage dans les sous-espaces flous correspondants. Alors le choix de la partition flou est très important, pour cette raison nous avons utilisé l'approche Grille Flou (FuzzyGrid) pour générer nos règles floues, cette approche est proposée la première fois dans un article [HKH92]

### 2.1 Partition flou:

Cette partie est consacrée à l'explication de la première partie (partition flou) dans le processus de construction d'un classifieur flou à partir des données numériques.

Supposons qu'on a un espace de modèle d'unité  $[0.1] \times [0.1]$  (avec deux attributs pour chaque modèle). Nous supposons également que la même partition floue  $(A_1, A_2, \dots, A_L)$  est employée pour chacune des deux axes dans l'espace de modèle où  $L$  est le nombre de sous-espaces flous à chaque axe. Dans ce cas-ci, des règles floues ordinaires peuvent être représentées comme suit :

$$\text{Si } X_1 \text{ Est } A_i \text{ Et } X_2 \text{ Est } A_j \text{ Alors } \dots, i, j = 1, 2, \dots, L. (1)$$

Le nombre total de règles floues dans (1) est  $L^2$ . Dans la figure suivante, nous présentons la structure de tables des règles floue correspondant à  $L=2$  et  $L=3$ .

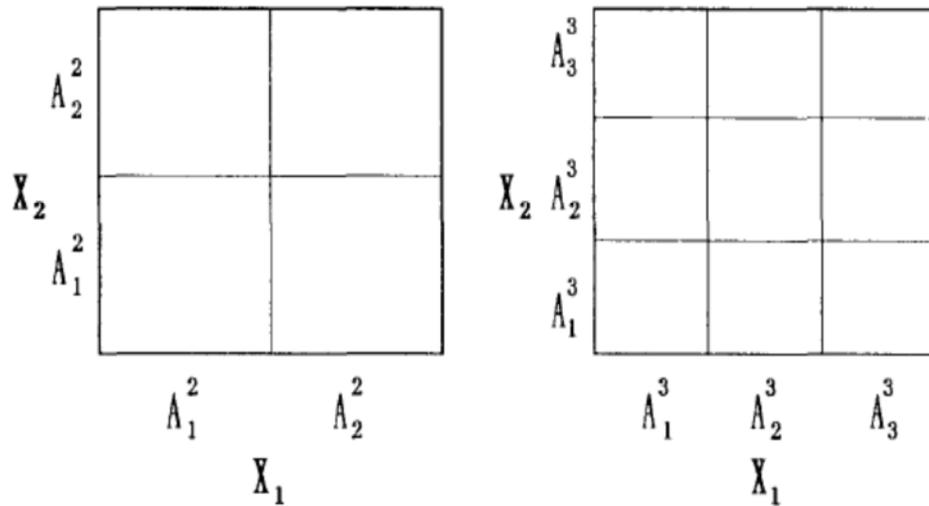


Figure 3.4 – Table représentant des règles floues correspondant à  $L=2$  et  $L=3$

## 2.2 Dérivation des règles floues :

Après la terminaison de la première partie il reste maintenant la dérivation des règles floues, Alors le but de cette deuxième partie est de déterminer la classe conséquente de chaque règle et le leur degré de vérité.

Il faut noter que chaque attribut est divisé en  $K$  ens-flou avec une fonction d'appartenance triangulaire :

$$\mu_i^k = \text{Max} \left\{ 1 - \frac{|x - a_i^k|}{b^k}, 0 \right\}, \quad i=1,2,\dots,K \quad (2)$$

Telque :

$$a_i^k = \frac{(i-1)}{(K-1)} \quad i=1,2,\dots,K; \quad b^k = 1/(k-1) \quad (3)$$

Dans la fonction d'appartenance (2),  $a_i^k$  est le centre où le degré de l'appartenance est égal à 1 et le  $b^k$  est la diffusion de la fonction d'appartenance.

*Degré d'appartenance*

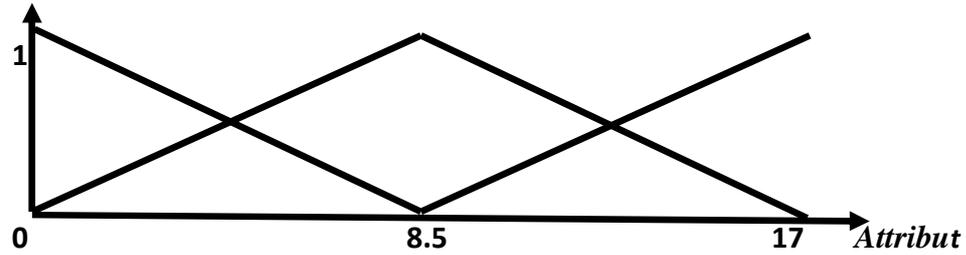


Figure 3.5- Exemple fonction d'appartenance avec partition en 03 ens-flou.

Chaque règle floue pour le problème de classification est représentée :

$$\text{Si } X_{1p} \text{ Est } A_{ik} \text{ Et } X_{2p} \text{ Est } A_{jk} \text{ Alors } X_p \text{ a } C_{ij}^K \text{ avec } CF = CF_{ij}^K \quad (4)$$

Avec  $C_{ij}^K$  est soit  $G1$  ou  $G2$  (deux classes) et  $CF_{ij}^K$  est le degré de vérité de la règle. Comme méthode pour dériver  $C_{ij}^K$  et le degré  $CF_{ij}^K$  a partir exemples données

$Xp = (X_{1p}, X_{2p}), p=1,2,\dots,m$ , la procédure suivante est proposée:

**[Dérivation de règles floues]**

- (i) Calcule la somme de compatibilités des modèles données dans chaque classe avec la prémisses règle floue (4) :

$$\beta_{G1} = \sum_{p \in G1} \mu_i^k(X_{1p}) \cdot \mu_j^k(X_{2p}) \quad (5)$$

$$\beta_{G2} = \sum_{p \in G2} \mu_i^k(X_{1p}) \cdot \mu_j^k(X_{2p})$$

- (ii) Si  $\beta_{G1} = \beta_{G2}$  alors nous ne produisons pas de la règle floue correspondant au sous-espace flou  $(A_i^K, A_j^K)$ , sinon  $C_{ij}^K$  est déterminés comme suit.

$$\text{Si } \beta_{G1} > \beta_{G2} \text{ alors } C_{ij}^K = G1 \quad (6)$$

$$\text{Si } \beta_{G1} < \beta_{G2} \text{ alors } C_{ij}^K = G2 \quad (7)$$

(iii) Pour le cas  $\beta_{G1} \neq \beta_{G2}$ ,  $CF_{ij}^K$  est déterminée :

$$CF_{ij}^K = |\beta_{G1} - \beta_{G2}| / |\beta_{G1} + \beta_{G2}| \quad (8)$$

Pour le degré de compatibilité on peut utiliser d'autres techniques comme la somme des minimums, on va voir la déférence entre ces techniques dans la partie des résultats expérimentales obtenus, Il convient de noter que la règle floue ne peut pas être produite par ce procédé s'il n'y a aucun modèle dans le sous-espace floue  $(A_i^K, A_j^K)$ .

### 2.3 L'inférence floue pour la classification

En se basant sur ce qui précède, nous pouvons dériver des règles floues à partir des données présentées et par conséquent l'inférence floue des modèles inconnus utilisant ces règles floues ordinaires. Si nous employons la partition floue  $(A_1^L, A_2^L, \dots, A_L^L)$ , nous pouvons dériver les règles floues suivantes.

$$\begin{aligned} & \text{Si } X_{1p} \text{ Est } A_i^L \text{ Et } X_{2p} \text{ Est } A_j^L \text{ Alors } X_p \text{ a } C_{ij}^L \text{ avec } CF = CF_{ij}^L \\ & \text{Avec } i, j = 1, 2, \dots, L \quad (9) \end{aligned}$$

Nous proposons la procédure suivante pour classifier un modèle inconnu  $X_p$  utilisant les règles floue ordinaires(9) :

1- Calculer  $\alpha_{G1}$  et  $\alpha_{G2}$  :

$$\alpha_{G1} = \text{Max}\{\mu_i^L(X_{1p}) \cdot \mu_j^k(X_{2p}) \cdot CF_{ij}^L | C_{ij}^L = G_1 : i, j = 1, 2, \dots, L\} \quad (10)$$

$$\alpha_{G2} = \text{Max}\{\mu_i^L(X_{1p}) \cdot \mu_j^k(X_{2p}) \cdot CF_{ij}^L | C_{ij}^L = G_2 : i, j = 1, 2, \dots, L\} \quad (11)$$

2- a) Si  $\alpha_{G1} > \alpha_{G2}$  alors infer  $X_p \in G1$  avec Grade de support  $\alpha_{G1} - \alpha_{G2}$

b) Si  $\alpha_{G1} < \alpha_{G2}$  alors infer  $X_p \in G2$  avec Grade de support  $\alpha_{G2} - \alpha_{G1}$

c) Si  $\alpha_{G1} = \alpha_{G2}$  alors  $X_p$  ne peut pas être classifié.

Dans ce procédé, le résultat de l'inférence floue est le conséquent de la règle floue qui a le produit maximum de la compatibilité et de la certitude. Le degré de support au résultat de l'inférence floue est donné comme valeur dans l'intervalle [0.1]. Il convient de noter que le  $X_p$

inconnu ne peut pas être classifié s'il n'y a aucune règle floue qui a une compatibilité positive au  $X_p$ . C'est-à-dire, si  $\mu_i^k(X_{1p}) \cdot \mu_j^k(X_{2p}) = 0$  pour toutes les règles floues,  $X_p$  inconnu ne peut pas être classifié.

## 2.4 Résultats obtenus :

Pour faire l'évaluation on a utilisé la base de données de diabète (The PimaIndiansdiabetes) qui contient 768 exemples pris des patients qui peuvent montrer des signes de diabète. Chaque échantillon est décrit par huit attributs. Il y a 500 échantillons provenant des patients qui n'ont pas le diabète et 268 échantillons provenant des patients qui sont connus pour avoir le diabète.

L'ensemble de données a été divisé aléatoirement dans deux sous-ensembles apprentissage 75% et test 25%.

### 2.4.1 Résultat1 :

Dans un premier temps nous avons essayé de voir les résultats obtenus avec un nombre de S-ensemble flou pour chaque attribut qui varié entre 02 jusqu'au 07 S-ensemble flou.

Pour enrichir notre travail nous avons proposées d'utiliser aussi deux types de degré de compatibilité lors de la phase dérivation des règles floues dans (5) :

- La somme de produit :

$$- \beta_{G1} = \sum_{p \in G1} \mu_i^k(X_{1p}) \cdot \mu_j^k(X_{2p})$$

-

$$- \beta_{G2} = \sum_{p \in G2} \mu_i^k(X_{1p}) \cdot \mu_j^k(X_{2p})$$

- La somme de minimum :

$$- \beta_{G1} = \sum_{p \in G1} \text{Min} [\mu_i^k(X_{1p}), \mu_j^k(X_{2p})]$$

-

$$- \beta_{G2} = \sum_{p \in G2} \text{Min} [\mu_i^k(X_{1p}), \mu_j^k(X_{2p})]$$

**Taux de classification :** le pourcentage des cas correctement classifié sur l'ensemble de la base utilisé pour le Test.

Le tableau suivant représente les résultats obtenus :

Nombre S-Ensemble-Flou		2	3	4	5	6	7
Taux classification	<b>Produit</b>	68.22	78.12	77.08	72.39	70.31	63.02
	<b>Minimum</b>	67.70	70.83	76.56	74.47	70.83	65.62

Tableau3.1 résultats obtenus avec différentes type degré de compatibilité et différentes nombre de S-Ens-Flou

D'après les résultats obtenus nous pouvons dire que le choix de la partition flou est très important, nous avons remarqué que le taux de classification a augmenté on va de 02 S-ens Flou puis il a diminué d'une manière considérable nous arrivons à 07 S-ens Flou.

- Comme avec type de degré compatibilité **Produit** le taux de classification a débuté avec un taux de classification 68.22%, il a atteint le maximum avec 03 S-Ens Flou 78.12% et il a chuté avec 07 S-Ens Flou 63.02%. presque la même observation a été constatée pour type de degré compatibilité **Minimum**.

Ce qui confirme qu'une partition qui est trop brute peut dégradé la performance avec la possibilité de plusieurs modèles peuvent être classifiée faux. Et si une partition trop étroite, beaucoup des règles ne peuvent pas être produites en raison du manque de modèle d'apprentissage dans les sous-espaces flous correspondants. Le choix de la partition flou est très important.

#### **2.4.2 Résultat2 :**

Dans la partie précédente nous avons obtenu le meilleur taux de classification 78.12%, en utilisant le type de degré de compatibilité **Produit** et avec 03 S-Ens Flou dans cette partie nous avons essayé de discuter ce résultat pour bien comprendre notre mécanisme de classification:

- Après la phase dérivation des règles flous (Base d'apprentissage):

<i>Classe conséquente</i>	<b>Nombre de règles</b>
<i>Diabétique (Positive)</i>	47
<i>Non diabétique (Négative)</i>	65
<b>Total</b>	112

Tableau 3.2- Nombre de règles dérivées

Dans cette partie de notre expérimentation nous avons proposé 03 S-Ens Flou pour chaque attribut (total : 08 attributs) :

- *Petit* : représenté par F1 dans la règle.
- *Moyen* : représenté par F2 dans la règle.
- *Grand* : représenté par F3 dans la règle.

Dans la partie conclusion de chaque règle il y'a deux classes :

- *tested\_negative* : résultat Non Diabétique.
- *tested\_positive* : résultat Diabétique

Le tableau suivant représente un échantillon de règles dérivés après la phase d'apprentissage :

<b>Partie prémisse</b>	<b>Classe</b>	<b>Degré Vérité</b>
Si F1 Et F2 Et F1 Et F1 Et F1 Et F2 Et F1 Et F2	Alors tested_negative	Avec 0.13
Si F1 Et F2 Et F1 Et F1 Et F1 Et F2 Et F2 Et F1	Alors tested_negative	Avec 0.058
Si F1 Et F2 Et F1 Et F1 Et F1 Et F2 Et F2 Et F2	Alors tested_negative	Avec 0.484
Si F1 Et F2 Et F1 Et F1 Et F1 Et F2 Et F2 Et F3	Alors tested_negative	Avec 1
Si F1 Et F3 Et F1 Et F1 Et F1 Et F2 Et F1 Et F2	Alors tested_positive	Avec 0.625
Si F1 Et F3 Et F1 Et F1 Et F1 Et F2 Et F2 Et F1	Alors tested_positive	Avec 0.833
Si F1 Et F3 Et F2 Et F1 Et F1 Et F2 Et F1 Et F1	Alors tested_negative	Avec 0.249
Si F1 Et F3 Et F2 Et F1 Et F1 Et F2 Et F1 Et F2	Alors tested_positive	Avec 0.253
Si F1 Et F3 Et F2 Et F1 Et F1 Et F2 Et F1 Et F3	Alors tested_negative	Avec 0.16
Si F1 Et F3 Et F2 Et F1 Et F1 Et F2 Et F2 Et F1	Alors tested_positive	Avec 0.008

Tableau3.3 Echantillon de règles floues dérivées.

Pour la phase de test nous avons utilisé 192 exemples (Diabétiques et non diabétiques) pour évaluer le classifieur en termes de précision, sensibilité et spécificité.

**Sensibilité** : on appelle sensibilité (Se) du test sa capacité de donner un résultat positif quand la maladie est présente. Représente ceux qui sont correctement détectés parmi tous les événements réels.

$$\text{Sensibilité} = TP / (TP + FN)$$

**Spécificité** : on appelle spécificité du test cette capacité de donner un résultat négatif quand la maladie est absente. Elle est représentée pour détecter les patients non diabétiques.

$$\text{Spécificité} = TN / (TN + FP)$$

Où TP, TN, FP et FN sont notés respectivement :

- TP : diabétique classé diabétique ; - FP : non diabétique classé diabétique ;
- TN : non diabétique classé non diabétique ; - FN : diabétique classé non diabétique.

Les résultats tirés de la phase de test sont :

- Instances correctement classée 150 78.125 %
- Instances incorrectement classée 42 21.875 %, Erreur absolue moyenne 0.2188

<b>Nombre paramètres</b>	08 paramètres
<b>Taux classification%</b>	78.125
<b>Se%</b>	38.70
<b>Sp%</b>	96.92
<b>TP</b>	24
<b>TN</b>	126
<b>FP</b>	4
<b>FN</b>	38
<b>Nombre règles</b>	112

Tableau 3.4- Performances notre classifieur flou après phase Test.

Pour bien comprendre notre mécanisme de la classification et d'essayer en même temps de trouver une explication des résultats obtenus qui peuvent nous aider à améliorer notre classifieur, nous avons essayé dans cette partie de discuter ces résultats:

La première observation déduite c'est que le nombre d'instances positives (des patients qui ont connus pour avoir le diabète) males classifiées (FN) est plus élevé par rapport au nombre total d'instances positives utilisé dans la phase de test, elle représente environ 63% et les instances classifiées juste (TP) représente 38%.

Par contre le nombre d'instances négatives males classifiées (FN) ne représente que 3% du nombre total d'instances négatives utilisées dans la phase de test.

Pour la compréhension de ces résultats il reste à prendre des exemples de l'échantillon de test utilisé et étudier pour chaque Instance les deux règles, la règle qui a été activé ou solliciter pour classifier l'exemple et celle qui a donné le degré d'appartenance le plus élevé pour toutes les paramètres de chaque instance (conséquence positive P et négative N) selon la vrai classe de chaque instance.

1- Nous allons prendre aussi des exemples qui ont été classifiés correctement :

Avec **Actual** c'est la classe de l'exemple et **Pred** est la classe donnée par notre classifieur.

Les tableaux suivants représentent les règles correspondantes à chaque exemple :

- Ra : règle activée pour donner le résultat de la classe (phase de test).

Instances	Enc	Glu	Ten	Peau	Insu	Index	Ped	Age	Actual	Pred
<b>I1</b>	2	90	80	14	55	24.4	0.249	24	<b>N</b>	<b>N</b>
<b>I2</b>	1	172	68	49	579	42.4	0.702	28	<b>P</b>	<b>P</b>
<b>01 cas TP et 01 cas TN</b>										

Ins	Règles	Enc	Glu	Ten	Peau	Insu	Index	Ped	Age	Classe	Degré
<b>I1</b>	<b>Ra</b>	F1	F2	F2	F1	F1	F2	F1	F1	N	0.745

L'exemple ou le patient I1 présente une valeur Glucose et insuline inférieur par rapport à d'autres exemples, notre classifieur permet d'activé la règle qui a donné une bonne réponse (classer le patient comme non diabétique).

Ins	Règles	Enc	Glu	Ten	Peau	Insu	Index	Ped	Age	Classe	Degré
I2	Ra	F1	F3	F2	F2	F2	F2	F2	F1	P	0.229

Par contre, l'exemple ou le patient I2 présente une valeur Glucose et insuline très grande, notre classifieur permet d'activer la règle qui a donné une bonne réponse (classer le patient comme diabétique).

Autres exemples mal classés :

Instances	Enc	Glu	Ten	Peau	Insu	Index	Ped	Age	Actual	Pred
I1	0	105	84	0	0	27.9	0.741	62	P	N
I2	2	155	52	27	540	38.7	0.24	25	P	N
I3	3	173	82	48	465	38.4	2.137	25	P	N
I4	4	141	74	0	0	27.6	0.244	40	N	P
I5	9	154	78	30	100	30.9	0.164	45	N	P
I6	8	118	72	19	0	23.1	1.476	46	N	P
03 cas FN et 03 cas FP										

Avec **Actual** c'est la classe de l'exemple et **Pred** est la classe donné par notre classifieur.

Les tableaux suivants représentent les règles correspondantes à chaque exemple :

- Ra : règle activée pour donner le résultat de la classe (phase de test).
- Rv : la vraie règle, son conséquence est le même avec l'exemple avec la grande compatibilité.

Ins	Règles	Enc	Glu	Ten	Peau	Insu	Index	Ped	Age	Classe	Degré
I1	Ra	F1	F2	F2	F1	F1	F2	F2	F3	N	0.496
	Rv	F1	F3	F2	F1	F1	F2	F1	F2	P	0.252

Tableau3.5 Exemple I1 avec les 02 règles.

Ins	Règles	Enc	Glu	Ten	Peau	Insu	Index	Ped	Age	Classe	Degré
I2	Ra	F1	F2	F2	F1	F2	F2	F1	F1	N	0.742
	Rv	F1	F3	F2	F2	F2	F2	F1	F2	P	0.467

Tableau3.6 Exemple I2 avec les 02 règles.

Ins	Règles	Enc	Glu	Ten	Peau	Insu	Index	Ped	Age	Classe	Degré
I3	Ra	F1	F3	F2	F2	F3	F2	F3	F1	N	1
	Rv	F1	F3	F2	F2	F2	F2	F2	F1	P	0.228

Tableau3.7 Exemple I3 avec les 02 règles.

Ins	Règles	Enc	Glu	Ten	Peau	Insu	Index	Ped	Age	Classe	Degré
I4	Ra	F2	F3	F2	F1	F1	F2	F1	F2	P	0.319
	Rv	F1	F2	F2	F1	F1	F2	F1	F2	N	0.258

Tableau3.8 Exemple I4 avec les 02 règles.

Ins	Règles	Enc	Glu	Ten	Peau	Insu	Index	Ped	Age	Classe	Degré
I5	Ra	F2	F3	F2	F2	F1	F2	F1	F2	P	0.246
	Rv	F2	F2	F2	F2	F1	F2	F1	F2	N	0.287

Tableau3.9 Exemple I5 avec les 02 règles.

Ins	Règles	Enc	Glu	Ten	Peau	Insu	Index	Ped	Age	Classe	Degré
I6	Ra	F2	F3	F2	F1	F1	F2	F2	F2	P	0.393
	Rv	F2	F2	F2	F1	F1	F2	F2	F2	N	0.041

Tableau3.10 Exemple I7 avec les 02 règles.

A partir des différents exemples, nous constatons que les règles floues sont sollicitées en fonction des paramètres du vecteur d'entrée. Ensuite le moteur d'inférence floue évalue ses entrées en utilisant la base de connaissances pour diagnostiquer la classe du patient.

### 2.4.2.1 Discussion d'un cas FN (faux-négative) :

Dans le cas du patient I2, il est mal classé, c'est à dire diabétique reconnu comme non diabétique. La règle floue Ra (voir Tableau3.5) est activée avec un degré de vérité d'environ 0.496. Rv représente la règle qui doit donner le bon résultat, les différences qui existe avec Ra :

- Il présente une valeur un peu élevé de glucose, dans Ra elle est représentée par S-ens Flou *Moyen* (F2), par contre dans la règle Rv par S-ens Flou *Grand* (F3).
- La valeur Ped un peu petit, dans Ra elle est représentée par S-ens Flou *Moyen* (F2), par contre dans la règle Rv par S-ens Flou *Petit* (F1).
- L'attribut d'âge est élevé, dans Ra elle est représentée par S-ens Flou *Grand* (F3), par contre dans la règle Rv par S-ens Flou *Moyen* (F3).
- Le manque d'information pour les attributs Insuline et Peau.

### 2.4.2.2 Discussion d'un cas FP (faux-positive) :

Dans le cas du patient I4, il est mal classé, c'est à dire non diabétique reconnu comme diabétique. La règle floue Ra (voir Tableau3.8) est activée avec un degré de vérité d'environ 0.319. Rv représente la règle qui doit donner le bon résultat, les différences qui existe avec Ra :

- Il présente une valeur élevé de glucose, dans Ra elle est représentée par S-ens Flou *Grand* (F3), par contre dans la règle Rv par S-ens Flou *Moyen* (F3).
- La valeur Enc (nombre de grossesses) dans Ra elle est représentée par S-ens Flou *Moyen* (F2), par contre dans la règle Rv par S-ens Flou *Petit* (F1).
- Le manque d'information pour les attributs Insuline et Peau.

D'après les résultats obtenus nous avons observé qu'il faut établir une procédure d'optimisation qui permet de donner une partition floue adéquate et de choisir les paramètres les plus pertinents pour avoir un bon modèle de classification.

L'identification de structure du système et l'optimisation des paramètres sont deux thèmes principaux à considérer en concevant un système basé sur les règles flous. La partition floue, le choix de variable, et

le raisonnement flou sont les tâches résolues pour identifier la structure de système. Diverses approches ont été employées pour traiter les deux principales issues de conception de système basé sur les règles floues. L’algorithme génétique reste parmi les techniques les plus utilisées.

Dans la partie suivante nous allons proposer une approche qui utilise un algorithme génétique qui a pour objectif d’optimiser le nombre d’attributs et sélectionner les paramètres les plus pertinents avec une amélioration de la robustesse de notre classifieur flou.

### 3. Phase2 (Amélioration par Algorithme Génétique) :

Le but de cette partie est d’essayer d’améliorer notre système de classification par une des techniques informatiques bien établie qui a été utilisée pour traiter les problèmes d’optimisation : Algorithme Génétique.

Nous allons établir ou concevoir un algorithme génétique ayant comme but d’optimiser le nombre d’attributs (optimisation structurelle).

#### 3.1 Codage :

**Représentation individuelle** : Le chromosome d'un individu représente tous les attributs (dans notre cas 08 gènes), on donne comme valeur

- 0: l’attribut n’est pas présent, 1: l’attribut est active.

1	2	3	4	5	6	7	8
0	1	1	1	1	1	0	1

Figure 3.6 Exemple représentation d’un chromosome.

Dans cet exemple la première et la septième position dans le chromosome contient 0 alors nous allons éliminer les attributs 1 et 7 de notre base d’apprentissage.

**Notre AG procède comme suit :**

**Population initiale** : Elle est produite aléatoirement et nous présentons en plus un chromosome qui représente tous les attributs, tous les gènes recevrons la valeur 1.

**Fonction de fitness** : C'est le nombre de modèles correctement classifiés par la base règle flou générée à partir des attributs codés dans le chromosome correspondant:

$$\mathit{FitnessFunction} (Ci) = \mathit{NPC} (Ci),$$

Où  $\mathit{NPC} (Ci)$  est le nombre de modèles correctement Classifiés.

### 3.2 Opérateurs génétiques :

#### 3.2.1 Sélection :

Pour la sélection nous employons la méthode RWS (Roulette Wheel Selection), le nombre espéré de sélection d'un individu est proportionnel à sa fitness, un individu ayant une forte valeur d'adaptation a alors plus de chance à être sélectionné. La probabilité de sélection d'un individu est égale au ratio de son évaluation par rapport à la somme des évaluations de chaque individu. Donc chaque individu aurait une probabilité de sélection  $P$  tel que :

$$P_i = H_i/T, \text{ où } T \text{ est la somme.}$$

Après on classe les individus par probabilité croissante et on calcul pour chacun une Probabilité cumulative :

$$Q_i = \sum_{h=1}^i P_h$$

Et après nous utilisons la méthode RWS, nous générons une valeur aléatoire  $\beta$  dans  $[0,1]$ .

- Si  $\beta < Q_i$ , nous sélectionnons l'individu  $i$ , sinon nous sélectionnons l'individu  $j$  tel que

$$Q_{j-1} < \beta \leq Q_j.$$

#### 3.2.2 Croisement :

Pour le croisement nous avons choisis le croisement à un point tel que nous affectons à chaque individu la probabilité de croisement généralement supérieur à 70%. Alors pour qu'un

individu soit sélectionné pour le croisement, nous générons une valeur aléatoire  $\beta$  dans  $[0,1]$ , que nous allons comparer à la probabilité de croisement.

- Si  $\beta < Pc$ , ( $Pc$  : probabilité de croisement), alors l'individu est sélectionné pour le croisement, sinon il passe directement à la population qui participe à la mutation. Pour chaque pair des éléments sélectionnés nous définissons de façon aléatoire le point de croisement  $h$  (nous utilisons le croisement "un point").

### 3.2.3 Mutation :

Comme pour le croisement, tout d'abord nous affectons une probabilité de mutation  $Pm$  à chaque individu, cette valeur est généralement comprise entre 0.25% et 1%, nous générons ensuite une valeur aléatoire  $M$  dans  $[0,1]$ , que nous allons comparer à la probabilité de mutation.

- Si  $M < Pm$  nous sélectionnons l'individu pour la mutation, sinon l'individu est sélectionné dans la génération suivante.

Pour appliquer sur l'individu la mutation, il suffit de prendre deux positions  $p$  et  $q$  au hasard, et de permuter les éléments correspondants.

Voici un exemple : Position 2 et 6.

0	1	0	0	1	0	0	1
	↓				↓		
1	0	0	0	1	1	0	1

Figure 3.7 Exemple de mutation

**3.2.4 Test d'arrêt :** Le test d'arrêt d'une manière générale est défini selon le type de problème étudié (heuristique), dans notre cas nous allons utiliser comme test le nombre d'itération pour gagner dans le temps de calcul.

### 3.3 Résultat obtenu :

Après 100 itérations (test d'arrêt), et 50 chromosomes choisis aléatoirement comme population initiale nous avons obtenu le résultat suivant :

Enc	Glu	Ten	Peau	Insu	Index	Ped	Age
<b>0</b>	<b>1</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>1</b>

Figure 3.8 Meilleur Chromosome Obtenu

Le chromosome représente le meilleur classifieur flou obtenu par notre Algorithme Génétique qui a la Fonction de fitness la plus élevée, le nombre de modèles correctement classifiés par la base de règle flou générée à partir des attributs codés dans le chromosome correspondant.

***Fonction de fitness= 86.97%.***

Les attributs les plus pertinents (04 paramètres) trouvés sont :

- 1. Glu : concentration du glucose plasmatique,
- 2. BMI : index de masse corporelle,
- 3. PED : fonction de pedigree de diabète (l'hérédité).
- 4. Age : âge

Avec **04 paramètres** sélectionnés et **47 règles**, nous constatons que l'utilisation d'algorithme génétique pour l'optimisation d'un classifieur flou a amélioré la robustesse de notre classifieur d'une manière très considérable.

De nombreuses études dans le même contexte ont été réalisées comme dans [GVGB09] qui a obtenu avec seulement 02 paramètres et 10 règles floues activées un taux de classification 75.92%, on peut citer d'autres études [HCH04] et [ME07] qui recourent aux mêmes méthodes sur la même base de données ont obtenu 75.19% et 75.00% respectivement.

Les résultats réalisés dans ce mémoire de Magister sont très intéressants par rapport aux études citées dans la littérature, puisque l'apport d'utilisation d'algorithme génétique a permis de réduire le nombre de règles floues qui peuvent être interprétées linguistiquement. Cela permettra au clinicien d'obtenir des informations sur le Diagnostic Automatique.

### 4. Conclusion :

Notre étude propose un modèle de classification flou du diabète. Deux critères principaux à considérer en concevant un système basé sur les règles floues. Le premier est la

bonne précision des performances du classifieur, le second touche la compréhensibilité des résultats obtenus. L'outil proposé par les algorithmes génétique pour la sélection des paramètres les plus pertinents a permis d'augmenter de manière significative la précision de classification, ce qui a montré qu'on peut éliminer un grand nombre de règles sans détériorer le taux de classification.

## Conclusion et Perspectives :

Les techniques d'intelligence Artificiels telles que les réseaux neurologiques artificiels, la logique floue, et les algorithmes génétiques (GAs) sont des sujets populaires de recherches, puisqu'elles peuvent traiter des problèmes complexes difficilement résolu par des méthodes classiques, Les approches hybrides ont attiré une attention considérable à la communauté du recherche en informatique. Une des approches les plus populaires est l'hybridation entre la logique floue et les algorithmes génétiques menant aux systèmes génétiques floues (GFSs).

Les systèmes flous sont l'un des domaines les plus importants pour l'application de la théorie des ensembles floue. On le considère une structure de modèle basés sur des règles flous (FRBSs). Ils constituent une prolongation aux systèmes basés sur les règles classiques, parce qu'ils traitent « IF-Then » règles, dont les antécédents et les conséquents se composent de rapports de logique floue. Ce type de système a démontré ça capacité dans un nombre important d'applications.

Dans un processus de diagnostic un classifieur flou peut être un outil pratique, Il se compose de règles linguistiques qui sont faciles à interpréter par l'expert humain, au contraire des réseaux de neurones qui se sont considérés comme des boîtes noires ce classifieur peut contrôler la vraisemblance. Ceci est très important pour les systèmes de prise de décision, car les experts n'acceptent pas une évaluation sur ordinateur, à moins qu'ils comprennent pourquoi et comment une recommandation a été donnée.

Dans un classifieur flou un résultat de classification est donné par les degrés de vérité (activations) de plusieurs règles. un exemple peut appartenir à plusieurs classes à différents degrés d'appartenance. Le résultat idéal serait qu'un modèle appartient clairement à une seule classe, i.e le degré d'appartenance à une classe est sensiblement plus grand que les degrés d'appartenance à d'autres classes, Cette information peut être exploitée pour évaluer la qualité du résultat de la classification.

L'avantage des classifieurs flous réside dans l'interprétabilité, grâce à ça manipulation intuitive et la simplicité, qui sont des facteurs importants pour l'acceptation et l'utilisation d'une solution. Un classifieur flou ne résout pas les problèmes de classification mieux que d'autres approches, tel que : les statistiques, les arbres de décision ou les réseaux de neurones. Alors nous sommes confrontés a un problème de compromis entre l'interprétabilité et la précision d'un classifieur. Ce travail aborde le problème de la création d'un classifieur flou à partir des données médicales, l'objectif principal est d'améliorer la robustesse du notre classifieur sans détériorer son interprétabilité

La définition automatique d'un classifieur flou peut être vue comme problème d'optimisation ou de recherche, les Algorithmes Génétiques sont une technique globale bien connue et employée couramment de recherche avec la capacité d'explorer un grand espace de recherche pour les solutions appropriées exigeant seulement une mesure d'exécution. En plus de leur capacité de trouver des solutions optimales dans les espaces de recherche complexes, la structure de code et les dispositifs indépendants génériques d'exécution du GAs leur incitent les candidats appropriés pour incorporer la connaissance a priori. Dans le cas d'un classifieur flou, cette connaissance a priori peut être sous forme de variables linguistiques, de paramètres flous de fonction d'appartenance, de règles floues, de nombre de règles, etc. Ces possibilités ont prolongé l'utilisation des Algorithmes Génétiques.

L'outil proposé dans notre travail par l'utilisation des algorithmes génétiques pour la sélection des attributs les plus pertinents a permis d'augmenter la précision de classifieur si comparé au cas d'employer toutes les variables d'entrée disponible. Ça a montré quand peut diminuer le nombre de règles sans détériorer l'interprétabilité de classification. Ce travail de recherche nous permet d'apporter une valeur ajoutée dans les systèmes d'Aide au Diagnostic Médical.

Beaucoup d'approches restent en perspective pour l'amélioration des classifieurs flous par le biais des algorithmes génétiques (optimisation structurelle et paramétrique). Structurelle par l'optimisation de nombre de S-ens flou pour chaque attribut d'entrée (la partition floue), Paramétrique par l'optimisation des paramètres qui concerne les fonctions d'appartenance.

***Bibliographie :***

[BAKF06] N. Benamrane, A. Aribi, L. Kraloua, and A. Freville. *Interprétation des images médicales par une approche neuro-flou-genétique*. In *Modélisation, Optimisation et Simulation des Systèmes : Défis et Opportunités*. 6e Conférence Francophone de Modélisation et Simulation -MOSIM06 - Rabat - Maroc, 3 au 5 avril 2006.

[BENDIB08] Toufik BENDIB, «*Modélisation et simulation du transistor DGMOSFET en utilisant les Algorithmes Génétiques* », UNIVERSITE DE BATNA, 2008.

[Bez81] Bezdek JC, *Pattern recognition with fuzzy objective function algorithms*. Plenum Press, New York (1981).

[CB07] I. Burhan Aktac Ramazan Douganay M. Mete Ceylan Celikyilmaz, Asli Türkçen and N. Bacak. *A new classifier design with fuzzy functions*. In *Proceedings of the 11th International Conference on Rough Sets, Fuzzy Sets, Data Mining and Granular Computing, RSFDGrC '07*, pages 136–143, Berlin, Heidelberg, 2007. Springer- Verlag.

[CL06] Chen SM, Lin HL, *Generating weighted fuzzy rules from training instances using genetic algorithms to handle the Iris data classification problem*. *Journal of Information Science and Engineering* 22: 175-188, (2006).

[Cor01] Oscar Cordon, Francisco Herrera, Frank Hoffmann, Luis Magdalena, " *Genetic Fuzzy Systems - Evolutionary Tuning and Learning of Fuzzy Knowledge Bases* ", *ADVANCES IN FUZZY SYSTEMS — APPLICATIONS AND THEORY* Vol. 19, Publier 2001 par World Scientific Publishing Co. Pte. Ltd.

[Dia02] OMS Diabète. *Le coût du diabète*. Technical report, Aide-mémoire No.236, 2002.

[Dia11] OMS Diabète. *Diabète. Technical report*, Aide-mémoire No.312, Janvier 2011.

- [DS06] K.-L. Du and M. N. S. Swamy. *Neural Networks in a Soft computing Framework*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.
- [Ed08] Edmundo BONILLA HUERTA, " *LOGIQUE FLOUE ET ALGORITHMES GÉNÉTIQUES POUR LE PRÉ-TRAITEMENT DE DONNÉES DE BIOPUCES ET LA SÉLECTION DE GÈNES* ". THÈSE DE DOCTORAT, Laboratoire d'Étude et de Recherche en Informatique d'Angers, Soutenu : le 13 novembre 2008.
- [ETUA10] Narissara Eiamkanitchat, Nipon Theera-Umpon, and Sansanee Auephanwiriyaikul. *A novel neuro-fuzzy method for linguistic feature selection and rule-based classification*. IEEE ASSP Magazine, 2 :247–252, 2010.
- [FA10b] A. Frank and A. Asuncion. UCI Machine Learning Repository, [<http://archive.ics.uci.edu/ml>]. University of California, Irvine, CA: School of Information and Computer Science. (Access 2010).
- [GT00] R. Gilleron and M. Tommasi. *Découverte de connaissances à partir de données*. Technical report, Technical report, Grappa – Université de Lille 3, 2000.
- [GVGB09] Jonas Guzaitis, Antanas Verikas, Adas Gelzinis, and Marija Bacauskiene. *A framework for designing a fuzzy rule-based classifier*. In Proceedings of the 1st International Conference on Algorithmic. Springer-Verlag Berlin Heidelberg 2009.
- [HCH04] Ho SY, Chen HM, Ho SJ, *Design of accurate classifiers with a compact fuzzy-rule base using an evolutionary scatter partition of feature space*. 2010 IEEE Trans Syst Man Cybern Part B34(2):1031–1043.
- [HKH92] H. Ishibuchi, K. Nozaki, H. Tanaka. *Pattern Classification by Distributed Representation of Fuzzy Rules, Fuzzy set and systems*. 1992 IEEE.
- [HY07] Ishibuchi, H., Nojima, Y.: *Analysis of interpretability-accuracy tradeoff of fuzzy systems by multi-objective fuzzy genetics-based machine learning*. International Journal of Approximate Reasoning 44(1), 4–31 (2007).

- [Jan93] J. S. R. Jang. ANFIS: *adaptive-network-based fuzzy inference system*. Systems, Man and Cybernetics, IEEE Transactions on, 23(3):665–685, August 1993.
- [JJO04] Pakkanen J, Iivarinen J, Oja E, *The evolving tree—A novel self-organizing network for data analysis*. 2004 Neural Process Lett 20:199–211.
- [Jos10] J.M. Alonso, S. Guillaume, L. Magdalena. *Un indice d'interprétabilité des bases de règles floues*. Rencontres LFA francophones sur la Logique Floue et ses Applications (LFA), Oct 2006, Toulouse, France. Cépadués Editions, p. 37 - p. 44.
- [LWA+09] Chang-Shing Lee, Mei-Hui Wang, Giovanni Acampora, Vincenzo Loia, and Chin-Yuan Hsu. *Ontology-based intelligent fuzzy agent for diabetes application*. pages 16–22, 2009.
- [ME07] Zolghadri MJ, Mansoori EG, *Weighting fuzzy classification rules using receiver operating characteristics (ROC) analysis*. Inf Sci 177:2296–2307, (2007).
- [Mess05] Ali MESSAOUD, Mohamed BEN MESSAOUD, Abdennaceur KACHOURI et Faïçal SELLAMI, "*Classification des arythmies cardiaques par logique floue à partir de signaux ECG*", SETIT2005.
- [PG07] Kemal Polat and Salih Güneş, *An expert system approach based on principal component analysis and adaptive neuro-fuzzy inference system to diagnosis of diabetes disease*. Digit. Signal Process, 17:702–710, July 2007.
- [PH05] Castro P, Camargo H, *Improving the genetic optimization of fuzzy rule base by imposing a constraint condition on the number of rules*, V Artificial Intelligence National Meeting (ENIA), São Leopoldo, Rio Grande de Sul 972-981, (2005).
- [PTK05] Pelekis N, Theodoulidis B, Kopanakis I, Theodoridis Y, *Fuzzy Miner: Extracting Fuzzy Rules from Numerical Patterns*. International Journal of Data Warehousing and Mining 57-81, (2005).

- [Reb03] El-hadi Rebaa, " *Génération automatique et optimisation de système a inférence flou* ", Thèse Doctorat présenté a l'université Paris XII Val De Marne, Soutenue le 18 Décembre 2003.
- [Rep98] Blake CL, Merz CJ, (1998) UCI Repository of machine learning databases, Irvine, University of California, Department of Information and Computer Science.  
<http://www.ics.uci.edu/~mllearn/MLRepository.html>. Accessed January 19 2009
- [saidi08] SAIDI Khayreddine, « *Contrôle des systèmes non linéaires par la logique floue optimisée par les algorithmes génétiques* », MEMOIRE EN VUE DE L'OBTENTION DU DIPLOME DE MAGISTER, Université des Sciences et de la Technologie d'Oran, 2008.
- [Koza92] J. R. Koza. *Genetic Programming*. MIT Press, 1992
- [SCF09] Nesma Settouti, Amine Chikh, and Tadj Eddine Fandi. *Reconnaissance par Neuro-Floue du Diabète*. Research report, hal.inria.fr, 07/2009. Ce rapport de recherche fait parti du projet fin d'étude d'ingénieur d'état en Génie Bio-Médical.
- [Set11] Settouti Nesma, " *Renforcement de l'Apprentissage Structurel pour la Reconnaissance du Diabète* ", mémoire magister. Soutenu : le 28 Juin 2011.
- [Siar03] Johan Dréo, Alain Pérowski, Patrick Siarry, Eric Taillard, « *Métaheuristique pour l'optimisation difficile* », Eyrolles 2003.
- [SM00] Y. Hayashi S. Mitra. *Neuro-fuzzy rule generation: survey in soft computing framework*. IEEE Trans. on Neural Networks, 11(3):748–768, 2000.
- [TNT09] Hasan Temurtas, Yumusak Nejat, and Feyzullah Temurtas. *A comparative study on diabetes disease diagnosis using neural networks*. Expert Syst. Appl., 36:8610–8615, May 2009.
- [TRK07] Ozyer, T., Alhaji, R., Barker, K.: *Intrusion detection by integrating boosting genetic fuzzy classifier and data mining criteria for rule pre-screening*. Journal of Network and Computer Applications 30(1), 99–113 (2007)

- [Ü10] ElifDeryaÜbeyli. *Automatic diagnosis of diabetes using adaptive neuro-fuzzy inference systems*. Expert Systems, 27, Issue 4,:259–266, September 2010.
- [VNY09] Tsikolidaki, V., Pelekis, N. and Theodoridis, Y, in IFIP International Federation for Information Processing, Volume 296; Artificial Intelligence Applications and Innovations III; Eds. Iliadis, L.,Vlahavas, I., Bramer, M.; (Boston: Springer), pp. 529–534, (2009).
- [VTM08] A. Vosoulipour, M. Teshnehlab, and H. A. Moghadam. *Classificationon diabetes mellitus data-set based-on artificial neural networks and anfis*. 21 :27–30, 2008.
- [WZYL05] Yuan-zhen Wang, Zhi-bing Zhang, Bao-li Yi, and Hua-yang Li. *Data mining based on computational intelligence*. Wuhan University Journal of Natural Sciences, 10, No.2:371–374, 2005.
- [ZK07] Zhou E, Khotanzad A, *Fuzzy classifier design using genetic algorithms*. Pattern Recognit 40(12):3401–3414, (2007).

# Résumé

La précision et l'interprétabilité sont deux approches importantes dans la conception d'un système de classification flou. Dans beaucoup d'applications de monde réel, comme dans le domaine médical L'expert devrait être capable de comprendre le classifieur et d'évaluer ses résultats. Les modèles à base de règles floues sont particulièrement adaptés, car ils sont constitués de simples règles linguistiques interprétables. Ce mémoire de Magister traite la possibilité d'utilisation des Algorithmes Génétiques afin d'augmenter la robustesse de notre classifieur flou a travers d'un processus d'apprentissage structurelle et paramétrique. La partition floue, le choix de nombre de variables pertinents et le raisonnement flou utilisés sont des taches les plus étudiés dans plusieurs travaux qui existent dans la littérature. Nos résultats expérimentaux appliqués sur la base de données du diabète (UCI Machine Learning) montrent une augmentation dans la précision sans détérioration de l'interprétabilité. L'approche proposée est simple et efficace pour clarifier la décision finale du classifieur.

## **Mots Clés**

Classification Supervisé ; Interprétabilité ; Règles floues ; Génétique Algorithme ; Précision.

# **Abstract**

Accuracy and the interpretability are two important approaches in the design of a fuzzy classification system. In much of applications of real-world, as in the medical field the expert should be able to include/understand the classifier and to evaluate his results. The models containing fuzzy rules are particularly adapted, because they are made up simple interpretable linguistic rules. This memory of Magister treats a potential use of the Genetic Algorithms in order to increase the robustness of our fuzzy classifier through process of structural and parametric optimization. The fuzzy partition, the choice of variables and the fuzzy reasoning used are spots the most studied in several work which exists in the literature. Our experimental results applied to the database of diabetes (UCI Learning Machine) show an increase in the precision without deterioration of the interpretability. The approach suggested is simple and effective to clarify the final decision of the classifier.

## **Keywords**

Supervised Classification; Interpretability; Fuzzy rules; Genetics Algorithm; Precision.

# ملخص

إن الدقة والتفسير هما عاملين مهمين في تصميم أي نظام للتصنيف يعتمد على مبدأ الضبابية. في العديد من التطبيقات في العالم الحقيقي، كما هو الحال في المجال الطبي يجب أن يكون الخبير قادراً على فهم المصنف وتقييم نتائجه. النماذج القائمة على الضبابية هي مثالية لأنها تتكون من القواعد اللغوية البسيطة التأويل. تتناول ذاكرة هذه الماجستير إمكانية استخدام الخوارزميات الجينية لزيادة متانة وصلابة هذا المصنف من خلال عملية التعلم على مستوى هيكله وخصائصه. التقسيم الضبابي واختيار عدد المتغيرات ذات الصلة والمنطق الضبابي المستخدم هي أكثر النواحي دراسة في العديد من الأعمال التي توجد في هذا الجانب. النتائج التجريبية المتحصل عليها والتي تم تطبيقها على قاعدة بيانات السكري تظهر زيادة في الدقة دون تدهور للتفسير. النهج المقترح هو بسيط وفعال لتوضيح القرار النهائي للمصنف.

## كلمات مفتاحيه

تصنيف تحت الإشراف, التفسير, قواعد ضبابية, الخوارزمية الجينية, الدقة.

