

République Algérienne Démocratique et Populaire
Université Abou Bakr Belkaid– Tlemcen
Faculté des Sciences
Département d'Informatique

Mémoire de fin d'études

pour l'obtention du diplôme de Master en Informatique

Option: Modèle Intelligent et Décision(M.I.D)

Thème

La Sélection de Variables pour la Reconnaissance du Diabète en utilisant une approche Neuronale

Réalisé par :

- Mr Bekhti Mohammed Anès

Présenté le 01 juillet 2012 devant le jury composé de MM.

- Mr Benazzouz Mortada (Président)
- Mr Chikh Mohammed Amine (Encadreur)
- Mlle Settouti Nesma (Coencadreur)
- Mr Hadjila Fethallah (Examineur)
- Mme L.Chaouche (Examineur)

Remerciements

Je remercie tout d'abord Dieu pour l'accomplissement de ce mémoire.

Ensuite, j'adresse mes sincères remerciements à Monsieur Chikh Mohammed Amine, pour son encadrement, sa disponibilité, ses orientations, et sa compréhension.

Je remercie également Mlle Settouti Nesma pour sa disponibilité, ses conseils, ses suggestions, et son encadrement.

Je remercie aussi Mr M.Benazzouz, Mr F.Hadjila et Mme L.Chaouche pour avoir accepté d'examiner ce travail.

Je remercie bien évidemment ma famille, mes parents, mon frère, ma sœur et mes amis de m'avoir toujours soutenu et à qui je dois tout.

Enfin je tiens à remercier tous mes professeurs du département d'informatique pour leurs enseignements.

*Je dédie ce modeste travail à
ceux qui ont fait de moi l'homme que je suis aujourd'hui. mes très chers parents,
que dieu les récompense et les garde.*

Mon frère et ma sœur.

Tous mes amis.

Mes collègues de promotion.

A tous ceux qui me connaissent de près ou de loin.

Merci d'être toujours là pour moi.

Table des matières

Remerciements	
Table des matières	ii
Table des figures	v
Liste des tableaux	vii
Glossaire	viii
Introduction générale	1
I Problématique	3
1 Introduction	4
2 Définition du diabète	4
3 Statistiques sur le Diabète	4
3.1 Dans le Monde	4
3.2 En Algérie	6
4 Les types du diabète	6
4.1 Le diabète de type 1	6
4.2 Le diabète de type 2	7
4.3 Le diabète gestationnel	7
5 Le diagnostic du diabète	8
5.1 Le diagnostic du type 1	9
5.2 Le diagnostic du type 2	9
6 Les causes du diabète	9
6.1 Les causes du diabète de type 1	9
6.2 Les causes du diabète de type 2	10

7	Les symptômes du diabète	10
8	Les conséquences du diabète	11
9	L'aide au diagnostic du diabète	12
9.1	Les systèmes experts	12
9.2	Les systèmes de classification	13
10	Problématique	13
11	Conclusion	14
II Les réseaux de neurones et la sélection de variables		15
1	Introduction	16
2	Les réseaux de neurones artificiels	16
2.1	Le neurone	16
2.2	Topologie en couches	17
2.3	Apprentissage	19
2.4	Les réseaux de neurones et la reconnaissance du diabète dans la littérature	23
3	La sélection de variables	25
3.1	Définition	26
3.2	Critères d'évaluation	26
3.3	Procédures de recherche	27
3.4	Critères d'arrêt	29
3.5	Les différentes approches de sélection de variables	29
3.6	Les méthodes connexionnistes de sélection de variables	33
3.7	La sélection de variables et la classification du diabète dans la littérature	38
4	Contribution	42
5	Conclusion	43
III Expérimentation		44
1	Introduction	45
2	Approche proposée	45

3	Modélisation	47
3.1	Définition des besoins de l'utilisateur	47
3.2	Les acteurs principaux de l'application	47
3.3	Les diagrammes de cas d'utilisation	48
3.4	Les diagrammes de séquence	50
3.5	Le diagramme des classes	52
4	Expérimentations	55
4.1	Base de données PIMA	55
4.2	Repartitionnement de la base	57
4.3	Les critères d'évaluation	58
4.4	Résultats et interprétation	59
5	Comparaison avec des travaux de la littérature	62
6	Conclusion	63
	Conclusion générale	64
	Bibliographie	65
	Annexe A : Perceptron multicouche	70
1	Rétro propagation du gradient de l'erreur	71
2	Propriétés mathématiques	73
3	problème de sur-apprentissage	75
	Annexe B : Outils	76
1	Manuel d'utilisation	76
1.1	La base de données	76
1.2	Le réseau de neurones	77
1.3	Sélection et optimisation	77
2	Outils utilisés dans ce mémoire	78
2.1	LaTeX	78
2.2	MiKTeX	79
2.3	JabRef	79
2.4	NetBeans	80
2.5	JDK	81

Table des figures

I.1	Projection en terme de personnes atteintes du diabète (20-79 ans) [2011- 2030] [1].	5
I.2	Proportion de la mortalité mondiale (% des décès à cause des maladies non transmissibles) [4]	5
I.3	Proportion de la mortalité en Algérie (% des décès) [5]	6
I.4	Les deux types du diabète dans le monde (% d'apparition) [2]	7
II.1	Architecture d'un perceptron multicouche	18
II.2	Arrêt par la méthode de validation croisée [24]	23
II.3	Le principe général d'une méthode de sélection de type <i>Filter</i> [40]	31
II.4	Le principe général d'une méthode de sélection de type <i>wrapper</i> [40]	32
II.5	Le principe général d'une méthode de sélection de type <i>embedded</i>	33
III.1	Diagramme de cas d'utilisation « création d'un classifieur »	49
III.2	Diagramme de cas d'utilisation « aide au diagnostic »	50
III.3	Diagramme de séquence « création d'un modèle validé »	51
III.4	Diagramme de séquence « utilisation d'un modèle validé »	52
III.5	Diagramme des classes	55
III.6	Histogramme des 9 paramètres de la base de données	57
III.7	Architecture utilisée dans l'expérimentation	59
III.8	Architecture trouvée après l'application de la méthode OCD	62
A.1	La sigmoïde et sa dérivée	73
A.2	Exemple de sur-apprentissage	75
B.1	Partie base de données	76
B.2	Partie réseau de neurones	77

B.3 Partie sélection et optimisation 78

Liste des tableaux

I.1	Les critères cliniques pour le diagnostic d'une anomalie de la glycémie à jeune, de l'intolérance au glucose et du diabète [8]	8
II.1	Tableau des travaux consultés pour la classification	25
II.2	Tableau des travaux consultés pour la sélection de variables	41
III.1	Informations sur les descripteurs de la base	56
III.2	Informations sur les exemples de la base	56
III.3	Repartitionnement de la base	58
III.4	Matrice de confusion -1-	60
III.5	Matrice de confusion -2-	61
III.6	Tableau des performances	61
III.7	Tableau de comparaison avec des travaux de la littérature	62

Glossaire

- ACP : Analyse de Composant Principal
- AG : Algorithme Génétique
- ARTMAP : Adaptive Resonance Theory Map
- ARTMAP-IC : Adaptive Resonance Theory Map Instances Counter
- ARD : Automatic Relevance Determination
- Att : Attribut
- BDD : Base De Données
- BIS : Bidirectional Selection
- BMI : Body Mass Index
- BP : Diastolic Blood Pressure
- BS : Backward Selection
- CFS : Correlation Feature Selection
- EASD : European Association for the Study of Diabetes
- EBD : Early Brain Damage
- EBS : Early Brain Surgeon
- FCBF : Fast Correlation Based Filter
- FID : Fédération Internationale du Diabète
- FN : Faux Négatifs
- FP : Faux Positifs
- FS : Forward Selection
- GP : Programmation Génétique
- GP-CPS : Programmation Génétique avec Comparatif des Partenaires
- GR : Gain Ratio
- GRNN : General Regression Neural Network
- KNN : K Nearly Neighbors

- MNT : Maladies Non Transmissibles
- Npreg : Number of Times Pregnant
- OBD : Optimal Brain Damage
- OBS : Optimal Brain Surgeon
- OCD : Optimal Cell Damage
- OMG : Unified Modeling Language
- OMS : Organisation Mondiale de la Santé
- PED : Diabetes Pedigree Function
- PMC : Perceptron Multi Couche
- RBF : Réseau de neurones à Fonctions Radiales de base
- RNA : Réseau de Neurones Artificiels
- RNMC : Réseau de Neurones Multi Couche
- SE : Sensibilité
- SP : Spécificité
- SV : Sélection de Variables
- SVM : Support Vector Machines
- TC : Taux de Classification
- TS : Tabu Search
- UCI : Université de Californie Irvine
- UML : Unified Modelling Language
- VN : Vrais Négatifs
- VP : Vrais Positifs

Introduction générale

Plusieurs travaux sont effectués afin de développer des outils de classification et d'aide au diagnostic médicale. L'utilisation des méthodes dites intelligentes pour effectuer cette classification sont de plus en plus fréquentes. Même si la décision du médecin est le facteur le plus déterminant dans le diagnostic, l'aide au diagnostic médical s'est développé et a gagné en popularité. Ces systèmes sont même considérés comme étant essentiels dans beaucoup de disciplines médicales car ils permettent d'assurer un diagnostic plus exact et réduisent au maximum les erreurs dues à la fatigue et les doutes du médecin.

Dans la création de ces systèmes, la quantité et la qualité d'information sont essentielles. Nous pouvons même se dire que plus nous disposerons d'un ensemble d'information important, plus le système que nous conserverons sera performant et fiable. Si l'information est vitale pour de tels systèmes, il s'avère que la redondance d'information nuit au bon fonctionnement de ceux-ci et ajoute de surcroit du calcul. Ces variables font partie des informations « néfastes » : celles qui n'apportent rien à la définition du phénomène étudié, comme une variable restant constante ou bien n'ayant aucune corrélation avec celui-ci. Tout ça confirme que la sélection de variables reste un processus très important pour le bon fonctionnement du système. Elle cherche à réduire le nombre de variables à recueillir pour le déploiement du système et améliorer la connaissance du phénomène de causalité du problème étudié, ce qui est fondamental si nous voulons interpréter les résultats et assurer la qualité de la prédiction.

Ce travail de fin d'étude se situe dans le contexte général d'aide au diagnostic médical et a pour but de réaliser un système capable d'aider le médecin pour le diagnostic du diabète et de sélectionner les variables les plus pertinentes pour la

reconnaissance de cette maladie.

Il est composé de trois chapitres :

- Chapitre I : concerne la problématique, il détaille le contexte, la problématique et les objectifs de ce travail.
- Chapitre II : concerne l'état de l'art, il présente les techniques utilisées avec un aperçu sur les travaux les plus récents dans ce domaine.
- Chapitre III : concerne l'expérimentation, il présente l'approche proposée, la base de données utilisée, les expérimentations réalisées, les résultats obtenus avec leurs interprétations et une étude comparative avec d'autres résultats de certains travaux portant sur le même sujet.
- En dernier lieu, une conclusion générale et des perspectives pour ce travail sont présentées.

Chapitre I

Problématique

1 Introduction

La fédération internationale du diabète estime qu'une personne meurt du diabète toutes les sept secondes et que le nombre de personnes vivant avec le diabète dans le monde atteint 366 millions. Ce nombre devrait atteindre 552 millions de personnes d'ici 2030, ce qui correspond à la découverte d'environ trois cas de diabète de plus toutes les 10 secondes [1]. Ces estimations confirment que le diabète continue de frapper de manière disproportionnée tous les pays du monde. Les systèmes de santé de bon nombre de pays parmi eux l'Algérie sont actuellement armés pour faire face au fardeau croissant du diabète. ce qui montre que le diabète devient une épidémie mondiale.

Dans ce chapitre, nous présentons le diabète avec des statistiques de cette maladie, les différents types du diabète, comment faire le diagnostic du diabète, les causes du diabète, les symptômes et les conséquences de cette maladie. Ensuite, nous aborderons l'aide au diagnostic du diabète et nous finirons par la présentation de notre problématique.

2 Définition du diabète

Le diabète est une maladie chronique qui apparaît lorsque le pancréas ne produit pas suffisamment d'insuline ou que l'organisme n'utilise pas correctement l'insuline qu'il produit. 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 les nerfs et les vaisseaux sanguins [2].

3 Statistiques sur le Diabète

3.1 Dans le Monde

A l'occasion du 47ème congrès d'EASD [3], La Fédération internationale du diabète a annoncé l'existence de 366 millions de personnes atteintes de diabète dans

le monde [1]. Une personne meurt du diabète toutes les sept secondes. Elle estime que le diabète et ses complications causent la mort de 4,6 millions de personnes chaque année et que les secteurs de santé dépensent plus de 465 milliards de dollars annuellement pour le traitement du diabète. La (Figure I.1) représente les régions de la FID et les projections mondiales en termes de nombre de personnes atteintes du diabète.

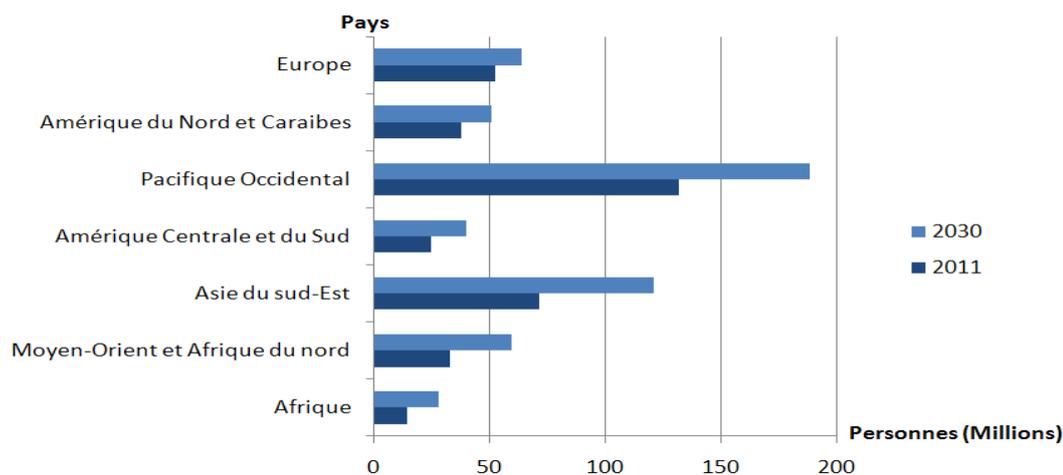


FIGURE I.1 – Projection en terme de personnes atteintes du diabète (20-79 ans) [2011- 2030] [1].

Le diabète est responsable de 4% des décès des maladies non transmissibles dans le monde, La (Figure I.2) montre la proportion de décès des maladies non transmissibles parmi les personnes de moins de 70 ans, selon la cause.

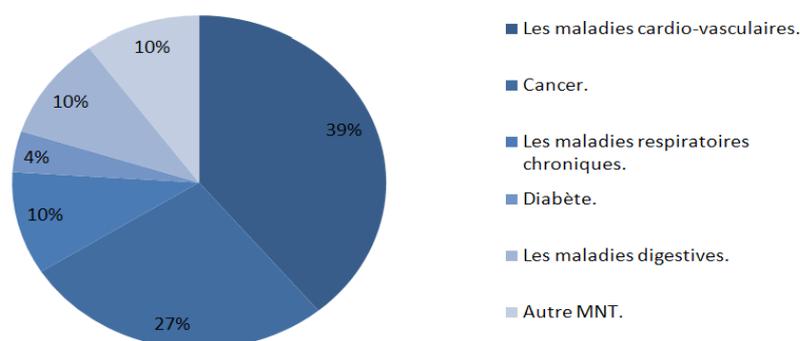


FIGURE I.2 – Proportion de la mortalité mondiale (% des décès à cause des maladies non transmissibles) [4]

3.2 En Algérie

Pour les praticiens algériens, le diabète est un véritable fléau. Il est considéré comme un sérieux problème de santé publique. Les diabétologues naviguent à vue car aucune étude épidémiologique n'a été lancée pour recenser les malades. Les seuls chiffres disponibles sont lancés par des experts du système de comptage de l'Organisation mondiale de la santé (OMS) qui estimaient que le nombre de diabétiques en Algérie est plus de 3 millions en 2011 dont 300 000 insulino-dépendants (diabète de type 1) [5] et que le diabète est responsable de 4% des décès dans le pays (Figure I.3).

Cette maladie est en progression continue à cause du changement rapide du mode de vie qui a tendance à s'aligner sur le modèle occidental de consommation alimentaire à base de graisse et de sucre.

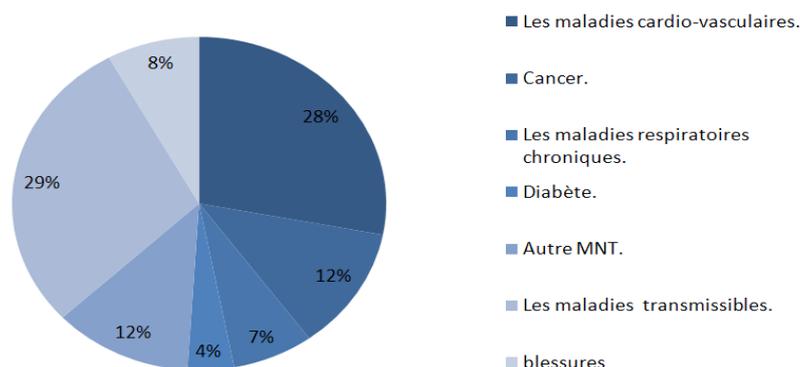


FIGURE I.3 – Proportion de la mortalité en Algérie (% des décès) [5]

4 Les types du diabète

Il existe plusieurs types de diabète dans le monde. Les plus connus sont :

4.1 Le diabète de type 1

(dit "diabète insulino-dépendant") touche généralement les enfants et les adultes de moins de 30/40 ans (sujets jeunes). L'apparition de ce type de diabète est brutale. Les personnes touchées maigrissent du fait de leur diabète et sont traitées à vie par

insulinothérapie.

4.2 Le diabète de type 2

(dit "diabète non insulino-dépendant") est aussi appelé diabète de la maturité et à l'inverse du diabète de type 1 apparaît de façon très insidieuse chez des personnes généralement plus âgées (classiquement > 40 ans) et en sur-poids [6].

Le diabète de type 2 est beaucoup plus répandu que le diabète de type 1. Il représente près de 90% des diabètes dans le monde (Figure I.4). Les rapports faisant état d'un diabète de type 2 chez les enfants – autrefois rares – sont de plus en plus nombreux. Dans certains pays, ce diabète représente près de la moitié des nouveaux cas diagnostiqués chez l'enfant et l'adolescent.

4.3 Le diabète gestationnel

Le diabète gestationnel est une hyperglycémie apparue ou décelée pour la première fois pendant la grossesse. Les symptômes du diabète gestationnel sont les mêmes que ceux du diabète de type 2. Il est très souvent diagnostiqué au cours du dépistage prénatal et non pas suite à des symptômes [7].

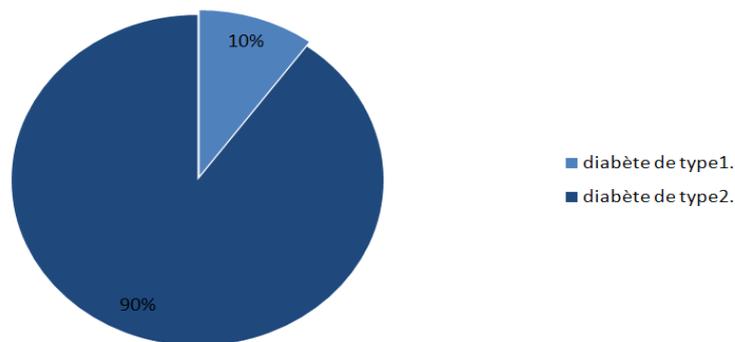


FIGURE I.4 – Les deux types du diabète dans le monde (% d'apparition) [2]

5 Le diagnostic du diabète

Les mesures de la glycémie (Table I.1) servent à déterminer si une personne est atteinte de diabète ou si elle risque de le développer. Les personnes dont les taux de glycémie sont supérieurs à la normale, font souvent l'objet d'un diagnostic de pré-diabète. Ce dernier est caractérisé par une anomalie de la glycémie à jeune, une intolérance au glucose ou les deux.

	Glycémie à jeune (mmol/L)		Glycémie deux heures après l'ingestion de 75 g de glucose (mmol/L)		Glycémie aléatoire (mmol/L)
Anomalie de la glycémie à jeune	6.1 – 6.9	et	< 7.8		
Intolérance au glucose	< 6.1	et	7.8 – 11.0		
Anomalie de la glycémie à jeune et intolérance au glucose	6.1 – 6.9	et	7.8 – 11.0		
Diabète	≥ 7.0	ou	≥ 11.1	ou	≥ 11.1 accompagné de symptômes du diabète (polyurie, polydipsie et perte de poids inexplicée)

TABLE I.1 – Les critères cliniques pour le diagnostic d'une anomalie de la glycémie à jeune, de l'intolérance au glucose et du diabète [8]

- Glycémie à jeune : aucun apport calorique depuis au moins huit heures.
- Glycémie aléatoire : à tout moment de la journée, sans égard au moment du dernier repas.
- 1,0 mmol/L(millimole)= 0,18 g/L.

5.1 Le diagnostic du type 1

Le diagnostic du diabète de type 1 se fait en recherchant les marqueurs de la destruction auto-immune des cellules du pancréas produisant de l'insuline. Ces marqueurs comprennent les anticorps anti-îlots de Langerhans, les anticorps anti-insuline, les anticorps acide glutamique décarboxylase et les anticorps anti-tyrosine phosphatase [9].

5.2 Le diagnostic du type 2

Souvent, l'apparition du diabète de type 1 se fait assez rapidement, alors que, dans le cas du diabète de type 2, l'augmentation de l'hyperglycémie se produit progressivement. Dans ce cas, des années peuvent s'écouler avant que les taux de glycémie soient suffisamment élevés pour présenter des symptômes donnant lieu à un diagnostic du diabète de type 2 [10]. Par conséquent, de nombreuses personnes sont atteintes de diabète de type 2 et n'ont aucun symptôme [11] [12]. C'est pour cette raison que des complications du diabète apparaissent avant même que les personnes ne soient diagnostiquées.

6 Les causes du diabète

Les causes du diabète sont complexes, mais sont en grande partie dues à l'obésité, au vieillissement et même à la génétique. Ces causes ce diffèrent pour chaque type du diabète.

6.1 Les causes du diabète de type 1

il s'installe quand le système immunitaire (système de défense naturelle de l'organisme contre les virus et les bactéries) se détériore et entraîne une autodestruction des cellules du pancréas responsables de la production d'insuline. La ou les causes d'une telle affection ne sont pas pour l'instant clairement établies.

Certains facteurs externes (infections virales) pourraient jouer un rôle. Il existe une susceptibilité génétique du diabète de type 1 mais il faut souligner que dans la grande

majorité des cas (90%) les cas de diabètes sont isolés, non familiaux.
Le mode d'interaction de tous ces facteurs reste inconnu [13].

6.2 Les causes du diabète de type 2

il s'installe lorsque 2 anomalies sont réunies :

- Les cellules de l'organisme deviennent moins sensibles à l'insuline qui ne peut remplir son rôle. Cela survient le plus souvent chez les sujets en sur-poids, sédentaires et ayant une alimentation trop riche. En conséquence de cette inefficacité, le pancréas essaie de produire plus d'insuline pour obtenir les mêmes effets.
- Le pancréas devient progressivement défaillant, et après quelques années, n'arrive plus à produire suffisamment d'insuline pour compenser la diminution de sensibilité des cellules à l'insuline. Cette insuffisance est plus prononcée durant les repas. C'est à ce moment qu'apparaît le diabète de type 2. Cette défaillance du pancréas est probablement héréditaire

Ces deux anomalies sont nécessaires pour développer le diabète de type 2. Par exemple, seuls 30% des patients en sur-poids sont diabétiques. Les autres ont un pancréas qui fonctionne tout à fait normalement et qui arrive à compenser l'inefficacité de l'insuline.

7 Les symptômes du diabète

Dans les deux types de diabète, une glycémie très élevée peut être à l'origine de symptômes comme :

- Soif excessive.
- Fréquente envie d'uriner (polyurie).
- Fatigue.
- Troubles de la vue.
- Perte de poids.
- Envie de manger (polyphagie).

Ces symptômes révèlent généralement le diabète de type 1. Cependant, environ 50% des personnes atteintes de diabète de type 2 sont asymptomatiques et le diabète

est découvert de façon fortuite lors d'un bilan systématique ou lorsque le patient développe des complications. Le diabète est une maladie chronique et grave qui induit des complications sévères pouvant affecter les yeux, les reins, les nerfs, les artères et pouvant conduire à une cécité, une amputation, une insuffisance rénale ou un infarctus du myocarde. Ces complications peuvent être prévenues ou retardées par un diagnostic précoce, par un bon contrôle de la glycémie et par un traitement approprié.

8 Les conséquences du diabète

Avec le temps, le diabète peut endommager le cœur, les vaisseaux sanguins, les yeux, les reins et les nerfs.

- Le diabète augmente le risque de cardiopathie et d'accident vasculaire cérébral. 50% des diabétiques meurent d'une maladie cardio-vasculaire (principalement cardiopathie et accident vasculaire cérébral).
- Associée à une diminution du débit sanguin, la neuropathie qui touche les pieds augmente la probabilité d'apparition d'ulcères des pieds et, au bout du compte, d'amputation des membres.
- La rétinopathie diabétique est une cause importante de cécité et survient par suite des lésions des petits vaisseaux sanguins de la rétine qui s'accumulent avec le temps. Au bout de 15 ans de diabète, près de 2% des sujets deviennent aveugles et environ 10% présentent des atteintes visuelles graves.
- Le diabète figure parmi les principales causes d'insuffisance rénale. 10 à 20% des diabétiques meurent d'une insuffisance rénale.
- La neuropathie diabétique fait suite aux lésions nerveuses dues au diabète et touche jusqu'à 50% des diabétiques [2]. Bien que de nombreux problèmes différents puissent résulter d'une neuropathie diabétique, les symptômes courants sont les suivants : fourmillement, douleur, engourdissement ou faiblesse au niveau des pieds et des mains.
- Le risque général de décès chez les diabétiques est au minimum deux fois plus important que chez leurs pairs qui n'ont pas de diabète.

9 L'aide au diagnostic du diabète

Le diagnostic du diabète semble s'adapter difficilement aux contraintes de l'informatique, si l'on en juge par le petit nombre de programmes existant actuellement.

Cela se comprend aisément car :

- L'évocation d'un diagnostic fait intervenir non seulement des éléments objectifs mais aussi des données subjectives et interprétatives.
- il n'existe pas une seule forme d'une même maladie mais plusieurs, que nous appelons dans le jargon médical « formes cliniques ». Celles-ci se rencontrent plus souvent que la forme classique.
- Il existe de nombreuses formes d'une même affection liées à son degré de gravité et à son évolutivité. Certains signes peuvent apparaître secondairement ou tardivement.
- Les signes principaux d'une maladie peuvent manquer sans que nous puissions rejeter la possibilité de cette affection.
- Bien que les signes principaux de l'affection manquent, quelques signes secondaires doivent nous faire évoquer le diagnostic.

Plusieurs catégories de systèmes informatiques peuvent résoudre ce type de problème tels que : les systèmes experts et les systèmes de classification.

9.1 Les systèmes experts

Un système expert est un outil d'aide à la décision capable de reproduire les mécanismes cognitifs d'un expert ou d'un groupe d'experts. Il se compose de 3 parties :

- une base de faits.
- une base de règles.
- un moteur d'inférence.

Donc, un système expert est un logiciel capable de répondre à des questions en effectuant un raisonnement à partir de faits et de règles connus [14] .

Les champs d'interrogation à remplir par l'utilisateur peuvent être plus ou moins nombreux et les concepts affichés dans ces champs seront traités par le moteur de recherche du logiciel en utilisant le langage booléen "ET", "OU", "NON".

9.2 Les systèmes de classification

Ce sont des systèmes basés sur des méthodes de classification. Ces méthodes de classification ont pour but d'identifier les classes auxquelles appartiennent des objets à partir de certains traits descriptifs. Ces types de systèmes s'appliquent à un grand nombre d'activités humaines et conviennent en particulier au problème de la prise de décision automatisée [15].

L'utilisation de l'informatique pour la réalisation de cette classification devient de plus en plus fréquente. Même si la décision de l'expert est le facteur le plus important lors du diagnostic, les systèmes de classification fournissent une aide substantielle car ils réduisent les erreurs dues à la fatigue et le temps nécessaire pour le diagnostic. Dans le cadre de la classification, les réseaux de neurones artificiels (RNA) ont largement fait leurs preuves au sein de la communauté scientifique et industrielle. À l'origine, ils constituent une tentative de modélisation formelle du cerveau humain. Ils représentent aussi une technique d'apprentissage artificiel qui a connu une évolution constante depuis les travaux de Mac Culloch et PiUsen 1943 [16].

10 Problématique

La médecine est une discipline scientifique mais aussi une discipline d'action qui requiert souvent une prise de décision. Ce processus résulte de la confrontation d'un problème réel à l'expérience acquise et à un corpus de connaissances théoriques. Un diagnostic médical est le résultat du raisonnement d'un médecin avec une décision très souvent prise à partir d'informations incertaines et/ou incomplètes qui sont le résultat d'un interrogatoire et d'un examen clinique. Le médecin puise dans ses connaissances pour trouver la maladie en cause.

Pour assurer un diagnostic exacte, un système d'aide au diagnostic aide le médecin à réduire au minimum les erreurs possibles qui peuvent survenir pendant le diagnostic d'une maladie. Pour le cas du diabète, le système va aider le médecin à classer les patients suivant deux catégories « diabétique » ou « non diabétique » en analysant les différentes valeurs des paramètres pris en considération pour le dépistage de cette maladie.

Des techniques telles que les réseaux de neurones sont les plus couramment utili-

sées dans les systèmes de classification et sont développées par de grand nombres d'équipes de recherches. Mais avec la présence d'attributs redondants ou d'attributs fortement bruités dans les bases de données, la performance du système peut se déclinier. Ce qui nécessite l'utilisation des techniques de sélection de variables qui ont pour objectif de sélectionner ou d'extraire un sous-ensemble optimal de caractéristiques ou de paramètres les plus pertinents pour faire un meilleur apprentissage et assurer une bonne performance du système de classification.

11 Conclusion

Le diabète s'augmente à l'échelle international et occupe la quatrième place parmi les maladies chroniques dans notre pays et dans le monde. Ses causes sont complexes mais sont en grande partie dues aux augmentations rapides enregistrées dans la fréquence du sur-poids, de l'obésité et de la sédentarité. Même si le diabète à de graves conséquences sur le corps humain, une grande proportion de ses cas et de ses complications pourrait être prévenue par un bon contrôle de la glycémie et par un diagnostic précoce. Ce qui nécessite l'utilisation d'un système d'aide au diagnostic pour faciliter la prise de décision et réduire au maximum l'incertitude quant à l'état actuel ou futur du patient.

Chapitre II

Les réseaux de neurones et la sélection de variables

1 Introduction

Les réseaux de neurones ont connu une large utilisation dans le domaine de la classification grâce à la simplicité de leur raisonnement et de leur performance d'apprentissage inspiré du raisonnement humain. Pour faire un bon apprentissage, le modèle du réseau de neurones sera choisi d'une manière expérimentale. Cela dépend du nombre de couches cachées, de neurones dans chaque couche, de nombre d'entrées et de sorties. Pour avoir de bons résultats, le modèle choisi peut avoir une architecture très complexe. La sélection de variables dans le domaine connexionniste permet de sélectionner l'ensemble de variables les plus pertinents pour faire le bon apprentissage et au même temps de choisir le meilleur modèle du réseau de neurones tout en assurant sa simplicité.

Dans ce chapitre, nous abordons en premier lieu les réseaux de neurones et leur principe d'apprentissage. En second lieu, nous présentons les différents types de méthodes de sélection de variables. Ensuite, nous présenterons en détail les méthodes connexionnistes tout en passant en revue l'état de l'art dans ce domaine et nous finirons par la présentation de notre contribution.

2 Les réseaux de neurones artificiels

Un réseau de neurones artificiels est un modèle de calcul dont la conception est très schématiquement inspirée du fonctionnement des neurones biologiques. Les réseaux de neurones sont généralement optimisés par des méthodes d'apprentissage de type probabiliste, Ils sont placés d'une part dans la famille des applications statistiques, et d'autre part dans la famille des méthodes de l'intelligence artificielle auxquelles ils fournissent un mécanisme perceptif indépendant des idées propres du programmeur.

2.1 Le neurone

Le Perceptron de Rosenblatt [17] transpose le comportement des neurones en une équation intégrant les grands principes observés dans la nature. A partir d'un stimulus représenté par un vecteur d'observations $x \in \mathbf{R}^n$, chaque composante x_i ,

$i \in [1, n]$ est multipliée par un poids de connexion w_i . La somme de ces entrées pondérées est ensuite faite en y ajoutant un biais θ (ou seuil d'activation). Pour des raisons de commodité de notation, on transforme ce biais en un nouveau neurone de sortie 1 avec un lien de poids w_0 tel que $\theta = w_0$. Pour finir, la somme (ou état d'activation) est passée dans une fonction f (ou fonction de seuil) non linéaire de type escalier. Au final la sortie d'un neurone s'écrit de manière synthétique (Eq II.1) :

$$y = f\left(\sum_{i=0}^n w_i x_i\right) \quad (\text{II.1})$$

Seul, le Perceptron peut être vu comme une fonction de discrimination entre deux classes pour un problème de classification : il partitionne l'espace d'entrée en deux régions avec une frontière de décision linéaire. Avec des poids w_i bien étudiés, cette surface linéaire permet de représenter des fonctions logiques comme le ET, le OU et le NON. Le Perceptron ne peut pas simuler le OU exclusif (XOR) car dans ce cas, la surface de décision est non linéaire. La critique du cas du XOR par [18] a d'ailleurs provoqué une temporaire mais historique désaffection pour le Perceptron. Bien qu'un neurone informatique ne soit pas une modélisation parfaite de sa version biologique, il n'en reste pas moins proche expérimentalement des phénomènes observés. De plus, agrégé en réseau, les capacités pourtant limitées du neurone artificiel produisent des résultats très intéressants tant au niveau purement fonctionnel qu'au niveau de la modélisation. Plusieurs algorithmes ont été proposés pour déterminer les poids w_i , à commencer par la méthode de [17] à base d'expériences, la technique des moindres carrés [19] et enfin les techniques à base de descente de gradient comme dans le cas du Perceptron multicouche.

2.2 Topologie en couches

Si les unités élémentaires sont souvent très proches dans la plupart des systèmes neuronaux, c'est au niveau d'architecture de ces neurones que les systèmes se différencient. Il a été constaté que si les neurones sont placés en couches successives (les sorties d'un certain nombre de neurones sont les entrées des suivants et ainsi de suite jusqu'à la sortie), alors l'ensemble du réseau est capable de décider des problèmes

plus complexes et peut aussi simuler n'importe quelle fonction booléenne. Ce type d'organisation (Figure II.1) est appelé Perceptron multicouche (PMC).

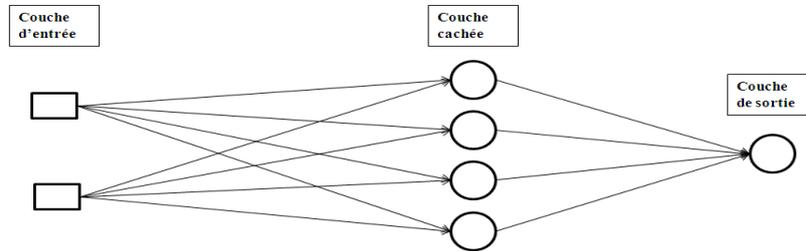


FIGURE II.1 – Architecture d'un perceptron multicouche

Les unités de calcul ne sont plus appelées Perceptrons mais plus simplement neurones ou encore noeuds. Outre la topologie en couches, la principale différence avec la version de [17] vient de l'utilisation de fonctions d'activation dérivables et non linéaires telles que la sigmoïde (Annexe A, Figure A.1), encore appelée fonction logistique.

L'idée d'une telle topologie est ancienne et il a fallu attendre un certain nombre d'années pour voir apparaître des algorithmes permettant de calculer les poids d'un tel réseau en particulier à cause de l'introduction des couches cachées. Proposé pour la première fois par [20] en 1974, l'utilisation de la rétro propagation du gradient de l'erreur dans des systèmes à plusieurs couches sera de nouveau mise au-devant de la scène en 1986 par [21], et simultanément, sous une appellation voisine, chez [22] durant sa thèse.

Ces réseaux sont souvent totalement connectés, ce qui signifie que chaque neurone d'une couche i est connecté à tous les neurones de la couche $i+1$. Par contre, dans un schéma classique, les neurones d'une même couche ne sont jamais reliés entre eux.

Les PMC sont essentiellement employés à deux tâches : le partitionnement d'un espace de formes pour des problèmes de classification et l'approximation de fonctions. Contrairement au Perceptron de [17], le PMC peut représenter n'importe quelle fonction booléenne à n variables, bien que certaines puissent requérir un nombre

exponentiel en n de neurones dans les couches cachées. Du fait de la non-linéarité de la sigmoïde comme fonction d'activation, les frontières de séparation s'adaptent mieux à chaque classe dans le cas d'un problème de classification. Cette propriété se retrouve aussi dans le cas de l'approximation de fonctions qui produit des courbes continues et lisses à la fois.

Les PMC possèdent des propriétés mathématiques intéressantes. Beaucoup d'entre-elles sont valables pour des réseaux à seulement deux couches cachées, ce qui témoigne de la puissance potentielle des PMC. Il est à noter que ces propriétés sont rarement constructives dans le sens où bien qu'il soit démontré qu'un certain nombre de neurones soit suffisant pour réaliser une tâche, la propriété ne donne aucune information sur la topologie à choisir afin de résoudre le problème (Annexe A .Propriétés mathématiques).

La majeure partie des propriétés sont prouvées sans l'hypothèse de l'utilisation de la sigmoïde, il suffit simplement que la fonction d'activation soit bornée (majorée et minorée), croissante et continue.

La difficulté d'utilisation de ce réseau réside dans le fait qu'il faille déterminer sa topologie, il s'agit de définir le nombre de neurones des différentes couches ainsi que leurs interconnexions.

Si le nombre de neurones cachés est trop faible, l'algorithme d'apprentissage n'arrivera pas à construire une représentation intermédiaire du problème qui soit linéairement séparable et certains des exemples ne seront pas appris correctement. Inversement, si ce nombre est trop élevé, il y a risque d'apprentissage par cœur du problème : le réseau reconnaît parfaitement les exemples d'apprentissage mais donnera des résultats médiocres sur des nouvelles données qu'il n'a pas vues durant l'apprentissage.

2.3 Apprentissage

L'approche la plus connue pour l'apprentissage d'un PMC est la technique de descente de gradient. En effet, l'utilisation de fonctions d'activation différentiables permet d'utiliser cette technique à la fois simple à mettre en œuvre et surtout très efficace sur le plan calculatoire.

Nous utiliserons dans la suite de cette section les notations suivantes :

- P le nombre de formes dans la base d'apprentissage.
- $x_p, p \in [1, P]$ la forme n° p de la base d'apprentissage.
- L le nombre de couches du réseau (y compris la couche d'entrée et de sortie).
- $N_l, l \in [0, L - 1]$ le nombre de neurones dans la couche n° l .
- $o_{l,j}$ la sortie calculée du neurone n° j dans la couche n° l .
- $d_j^i(x_p)$ la composante n° j de la sortie attendue pour la forme x_p .
- $w_{l,j,i}$ le poids de la connexion entre le neurone n° i dans la couche $l - 1$ et le neurone n° j dans la couche l .
- f la fonction d'activation.

La sortie d'un neurone quelconque est donnée par (Eq II.2) :

$$o_{l,j} = f\left(\sum_{i=0}^{N_{l-1}} w_{l,j,i} o_{l-1,i}\right) \quad (\text{II.2})$$

La fonction de cout E à minimiser dans le cas d'un apprentissage est une mesure de l'erreur entre la sortie souhaitée pour une forme et la sortie calculée par le réseau. L'erreur sur une forme p se quantifie généralement par une erreur quadratique $E_p(w)$ (Eq II.3) :

$$E_p(w) = \frac{1}{2} \sum_{q=1}^{N_L} (o_{L,q}(x_p) - d_q(x_p))^2 \quad (\text{II.3})$$

L'erreur pour l'ensemble des formes $E_p(w)$ est donc (Eq II.4) :

$$E(w) = \sum_{p=1}^P E_p(w) \quad (\text{II.4})$$

Le problème se résume donc à :

$$\min E(w) \quad (\text{II.5})$$

Pour résoudre ce type de problème, une technique classique d'optimisation issue de la recherche opérationnelle consiste à déterminer par itérations successives les valeurs du paramètre w . Au regard des objets à manipuler, la descente de gradient est une réponse adéquate à ce problème. Elle consiste à utiliser un point existant w_0

et lui faire effectuer un déplacement dans la direction de l'anti gradient. Le nouveau point obtenu par la translation $w \longrightarrow w + \mu x_p \sigma_p$ a une plus petite valeur pour la fonction objectif. Le paramètre μ est un pas positif appelé dans le cas présent pas d'apprentissage et σ est le gradient de l'erreur. L'opération de translation est répétée jusqu'à l'obtention d'une solution satisfaisante. En utilisant la rétro propagation du gradient de l'erreur (Annexe A. 1), le résumé du déroulement de la méthode est donné par (Algorithme 1).

Algorithm 1 Apprentissage d'un PMC par rétro-propagation du gradient

```

1: Initialisation aléatoire des poids du réseau
2: répéter
    pour chaque échantillon de la base d'apprentissage
    faire
        - Propager l'échantillon dans le réseau
        - Calcule de l'erreur sur la couche de sortie
        - Propagation de l'erreur sur les couches
          inférieures
        - Ajustement des poids
    fin
    Mise à jour de l'erreur totale
jusqu'à Critère d'arrêt
    
```

Bien que l'erreur soit minimisée localement, la technique permet de converger vers un minimum et donne de bons résultats pratiques. Dans la plupart des cas, peu de problèmes dus aux minima locaux sont rencontrés. Il persiste cependant deux problèmes que l'on rencontre dans une application réelle qui sont d'une part la lenteur de la convergence si μ est mal choisi et d'autre part le possible risque de converger vers un minimum local et non global de la surface d'erreur.

Le principal défaut de cette méthode est un temps de convergence restant assez long qui dépend de différents paramètres comme l'initialisation à l'instant $t = 0$ des poids synaptiques ou de la valeur initiale du paramètre μ . Il n'en reste pas moins qu'elle donne de bons résultats expérimentaux.

Dans une implémentation de l'algorithme de rétro propagation de l'erreur, il est aussi difficile de déterminer quand l'ajustement des poids du PMC doit s'achever. Plusieurs critères d'arrêt sont employés : les itérations cessent quand la norme du gradient est proche de zéro (les poids ne varient alors que très peu), ou bien alors

dés que l'erreur en sortie est en dessous d'un certain seuil.

Le premier critère est plus intéressant mathématiquement car il correspond à la stabilisation de la solution dans un minimum, le second est plus proche de critères réels (interprétables) de bonne corrélation entre solution calculée et solution attendue. Dans ce dernier cas, si le problème étudié concerne une tâche de classification, on peut considérer que l'apprentissage s'achève quand toutes les formes sont classifiées, ce qui permet de s'affranchir de la détermination du taux d'erreur à ne pas dépasser.

En pratique, nous allions ce dernier critère d'arrêt à un deuxième qui tient compte d'un nombre maximum d'itérations à ne pas franchir. En effet, il n'est pas garanti que le réseau puisse classifier toutes les formes, même avec un nombre infini d'itérations. La combinaison des deux conditions permet d'obtenir une solution correcte dans un temps raisonnable.

La validation croisée est une technique s'assurant principalement de la bonne généralisation du réseau, c'est-à-dire de son bon fonctionnement sur de nouveaux échantillons. Elle consiste à utiliser deux bases : l'une pour l'apprentissage et l'autre pour le test d'arrêt (Figure II.2). La première, comme son nom l'indique, sert uniquement à l'algorithme de rétro-propagation du gradient, la seconde permet de tester, à la fin de chaque itération, la qualité du réseau. Tant que l'erreur globale du réseau sur la base de test diminue, les itérations continuent. Dès que l'erreur augmente, l'apprentissage est stoppé même s'il aurait été possible de diminuer encore l'erreur sur la base d'apprentissage. Cette solution, quand on dispose d'un grand nombre d'échantillons permet d'éviter le phénomène de sur-apprentissage (overfit) sur les données d'apprentissage ayant comme conséquence une mauvaise généralisation [23] (Annexe A Figure A.2). Les résultats en termes de taux de lecture ou de taux d'erreur sont alors donnés pour une troisième base, indépendante des deux autres, nommée base de validation.

Reste le choix des échantillons lors de l'apprentissage qui est aussi un problème crucial. Il existe dans ce domaine peu de résultats théoriques concernant la création d'une «bonne» base d'apprentissage. Il est évident que dans un cas réel, afin d'avoir une bonne fiabilité et un grand pouvoir de généralisation, les exemples doivent être

d'autant plus nombreux que le problème est complexe et sa topologie peu structurée. Et pour éviter des phénomènes de sur-apprentissage de certaines classes, il est recommandé de fournir au réseau un nombre d'exemples similaire pour chacune des classes et de les présenter de manière aléatoire lors de l'apprentissage [24].

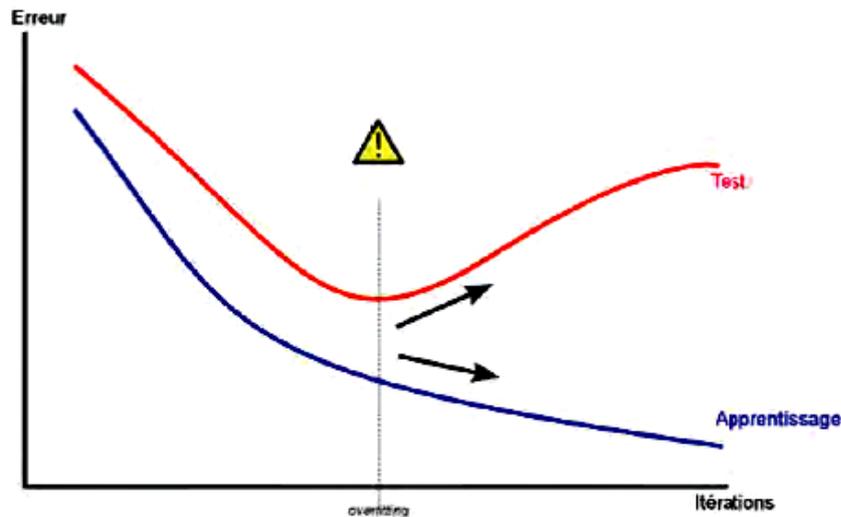


FIGURE II.2 – Arrêt par la méthode de validation croisée [24]

2.4 Les réseaux de neurones et la reconnaissance du diabète dans la littérature

Dans cette partie, nous citons les différents travaux réalisés pour la reconnaissance du diabète. Parmi les travaux consultés, il y a ceux qui utilisent les différents types des réseaux de neurones comme l'ARTMAP (Table II.1) (1), le GRNN, RBF, PMC (Table II.1) (2) et ceux qui utilisent d'autres approches comme les algorithmes génétiques et les hybridations basées sur les AGs (Table II.1) (3-4). Le tableau suivant présente les différents travaux consultés :

Auteurs	Titre	Approche Utilisée	Expérimentation	Résultats obtenus
1- Gail A. Carpenter, Natalya-Markuzon	ARTMAP-IC and medical diagnosis :Instance counting and inconsistent cases. [25]	Une amélioration du réseau de neurones ARTMAP par l'ajout d'un compteur d'instance (ARTMAP-IC). Les réseaux de type ARTMAP ont la capacité de choisir les exemples les plus pertinents pour effectuer la tâche d'apprentissage.	Ils ont testé la méthode proposée (ARTMAP-IC) sur quatre bases de données d'UCI, parmi eux la base du diabète PIMA.	Le taux de classification trouvé par le (ARTMAP-IC) est :81%.
2- K. Kayaer and T. Yildirim.	Medical diagnosis on pima indian diabetes using general regression neural networks. [26]	l'utilisation des général regression neural networks (GRNN) pour un diagnostic médical. Le GRNN rapproche n'importe quelle fonction arbitraire entre l'entrée et les vecteurs de rendement tirant l'évaluation de fonction directement des données d'apprentissage. Il n'exige pas un procédé itératif d'apprentissage comme dans le PMC.	Ils ont appliqué le (GRNN) sur la base PIMA Indian Diabète. Et les résultats obtenus sont comparés avec Plusieurs approches.	-Les résultats de la classification obtenus sont les suivants : PMC :77.60% RBF :68.23% KNN :77% ARTMAP-IC :81% GRNN :80.21%
3- Jonas Guzaitis, Antanas Verikas, Adas Gelzinis, And Marija Ba-causkiene	A Framework for Designing a Fuzzy Rule-Based Classifier [27]	La conception d'un système de classification fondé sur des règles floues et un AG pour partitionner l'espace de recherche qui permet de fixer à la fois la structure et le calcul des paramètres du classifieur de type PMC.	Ils ont appliqué l'approche proposée sur 4 bases de données (vote, cancer du sein, diabète, inspection de surface).	Le taux de reconnaissance trouvé par cette méthode pour la base PIMA égale 71.49%.

<p>4- Muhammad Waqar Aslam and Asoke Kumar Nandi</p>	<p>DETECTION OF DIABETES USING GENETIC PROGRAMMING [28]</p>	<p>L'utilisation de la programmation génétique (GP) et une variation de programmation génétique appelée GP avec le choix comparatif des partenaires (CPS) pour la détection de diabète. GP et GP+CPS se diffèrent dans la manière que des parents sont croisés les uns avec les autres pour produire des enfants.</p>	<p>Le système proposé se compose de deux étapes. Dans la première phase ils ont employé la programmation génétique pour produire un modèle à partir des données d'apprentissage. Dans la 2eme phase, ils ont utilisé des données pour tester le modèle trouvé.</p>	<p>Ils ont fait 2 expérimentations, les taux de classification obtenus sont : ExpI : Standard-GP :78.2% GP-CPS :78.4% ExpII : Standard-GP :77.4% GP-CPS :78.5%</p>
--	---	---	--	---

TABLE II.1 – Tableau des travaux consultés pour la classification

3 La sélection de variables

La sélection de variables est un problème difficile qui a été étudié depuis les années 70. Il revient dans l'actualité scientifique avec l'apparition des grandes bases de données et les systèmes de fouille de données «Data Mining» [29].

La sélection de variables a fait l'objet de plusieurs recherches en statistique, et plus particulièrement dans des domaines comme la classification, la modélisation de séries chronologiques et l'identification de processus [30]. Dans le domaine de l'apprentissage, l'étude de la problématique de la sélection de variables est assez récente. Dans le domaine de l'apprentissage connexionniste [31], [32], la sélection de variables a été abordée à partir d'un problème d'optimisation et de choix d'architectures des modèles, ainsi des approches très intéressantes ont émergé.

La sélection de variables est une problématique complexe et d'une importance cruciale pour les systèmes d'apprentissage [33]. Afin de mettre en évidence les deux aspects du processus de la sélection de variables, difficulté et importance, nous présentons les éléments essentiels que nécessite généralement ce processus. Une définition de la sélection de variables peut s'énoncer de la façon suivante :

3.1 Définition

La sélection de variables est un procédé permettant de choisir un sous-ensemble optimal de variables pertinentes, à partir d'un ensemble de variables original, selon un certain critère de performance [31].

A partir de cette définition, on peut se poser trois questions essentielles :

- Comment mesurer la pertinence des variables ?
- Comment former le sous-ensemble optimal ?
- Quel critère d'optimalité utiliser ?

Ces trois questions définissent les éléments essentiels d'une procédure de sélection de variables. En effet, le problème de la sélection de variables consiste à identifier les variables permettant une meilleure séparation entre les différentes classes dans le cas d'un classement. On parle alors de "pouvoir discriminant", pour désigner la pertinence d'une variable. La réponse à la première question consiste à trouver une mesure de pertinence ou **un critère d'évaluation** $J(X)$ permettant de quantifier l'importance d'une variable ou d'un ensemble de variables X . La deuxième question évoque le problème du choix de **la procédure de recherche** ou de constitution du sous-ensemble optimal des variables pertinentes. La dernière question demande la définition **d'un critère d'arrêt de la recherche**. Le critère d'arrêt est généralement déterminé à travers une combinaison particulière entre la mesure de pertinence et la procédure de recherche.

3.2 Critères d'évaluation

L'amélioration des performances d'un système d'apprentissage par une procédure de sélection de variables nécessite dans un premier temps la définition d'une mesure de pertinence. Dans le cas d'un problème de classement, on teste, la qualité de discrimination du système en présence ou en absence d'une variable. Par contre, pour un problème de régression, on teste plutôt la qualité de prédiction par rapport aux autres variables.

Pertinence d'une variable : Une variable pertinente est une variable telle que sa suppression entraîne une détérioration des performances - pouvoir de dis-

crimination en classement ou la qualité de prédiction en régression - du système d'apprentissage [31].

Plusieurs critères d'évaluation ont été proposés, basés sur des hypothèses statistiques ou sur des heuristiques. Pour un problème de classement - discrimination -, les critères d'évaluation sont souvent basés sur les matrices de dispersion intra et inter classes. En effet, ces matrices sont directement liées à la géométrie des classes et donnent une information significative sur la répartition des classes dans l'espace des formes.

On trouve aussi des critères d'évaluation qui utilisent des distances probabilistes ou des mesures d'entropie. Le critère dans ce cas est basé sur l'information mutuelle entre le classement et l'ensemble de variables. Dans le cas des systèmes d'apprentissage connexionnistes, l'évaluation des variables se fait en fonction de l'importance des poids qui est définie comme le changement de l'erreur - de classement ou de régression - dû à la suppression de ces poids.

3.3 Procédures de recherche

En général, nous ne connaissons pas le nombre optimal m de variables à sélectionner. Ce nombre dépendra de la taille et de la qualité de la base d'apprentissage - la quantité et la qualité d'information disponible - et de la règle de décision utilisée - le modèle. Pour un ensemble de n variables il existe $(2^n - 1)$ combinaisons de variables possibles où **2** représente deux choix : sélectionner ou ne pas sélectionner une variable. La recherche d'un sous-ensemble de m variables parmi n engendre un nombre de combinaison égal à (Eq II.6) :

$$\binom{n}{m} = \frac{n!}{(n-m)!m!} \quad (\text{II.6})$$

En grande dimension - très grande -, le nombre de combinaison à examiner devient très élevé et une recherche exhaustive n'est pas envisageable. La recherche d'un sous-ensemble optimal de variables est un problème **NP-difficile**.

Cependant, la plupart des critères d'évaluation utilisés pour la sélection ne sont pas monotones et dans ce cas on a recours à la seule alternative basée sur des méthodes sous-optimales comme les procédures séquentielles [34] :

- Stratégie ascendante : *Forward Selection (FS)*.
- Stratégie descendante : *Backward Selection (BS)*.
- Stratégie bidirectionnelle : *Bidirectional Selection (BiS)*.

La méthode FS

procède par agrégations successives - par ajouts successifs de variables. Au départ l'ensemble des variables sélectionnées est initialisé à l'ensemble vide. À chaque étape k , on sélectionne la variable qui optimise le critère d'évaluation $J(X_k)$ et on la rajoute à l'ensemble des variables sélectionnées X_k . Soit X l'ensemble des variables, on sélectionne la variable x_i telle que :

$$J(X_k) = \max_{x_i \in X \setminus X_{k-1}} J(X_{k-1} \cup \{x_i\}) \quad (\text{II.7})$$

L'ordre d'adjonction des variables à l'ensemble des variables sélectionnées produit une liste ordonnée des variables selon leur importance. Les variables les plus importantes sont les premières variables ajoutées à la liste. Néanmoins, il faut aussi se rappeler que des variables individuellement peu pertinentes peuvent être très informatives si on les utilise conjointement.

La méthode BS

est une procédure inverse de la précédente - par retraits successifs de variables. On part de l'ensemble X complet des variables et on procède par élimination. À chaque étape la variable la moins importante selon le critère d'évaluation est éliminée. Le procédé continue jusqu'à ce qu'il reste qu'une seule variable dans l'ensemble des variables de départ. À l'étape k , on supprime la variable x_i telle que :

$$J(X_k) = \max_{x_i \in X_{k+1}} J(X_k \setminus \{x_i\}) \quad (\text{II.8})$$

Une liste ordonnée selon l'ordre d'élimination des variables est ainsi obtenue. Les variables les plus pertinentes sont alors les variables qui se trouvent dans les dernières positions de la liste.

La procédure BiS

effectue sa recherche dans les deux directions – *Forward et Backward*- d'une manière concurrentielle. La procédure s'arrête dans deux cas :

- Quand une des deux directions a trouvé le meilleur sous-ensemble de variables avant d'atteindre le milieu de l'espace de recherche.
- Quand les deux directions arrivent au milieu. Il est clair que les ensembles de variables sélectionnés trouvés respectivement par *FS* et par *BS* ne sont pas égaux à cause de leurs différents principes de sélection.

Néanmoins, cette méthode réduit le temps de recherche puisque la recherche s'effectue dans les deux directions et s'arrête dès qu'il y a une solution quelle que soit la direction.

3.4 Critères d'arrêt

Le nombre optimal de variables n'est pas connu a priori, l'utilisation d'une règle pour contrôler la sélection de variables permet d'arrêter la recherche lorsqu'aucune variable n'est plus suffisamment informative. Le critère d'arrêt est souvent défini comme une combinaison de la procédure de recherche et du critère d'évaluation. Une heuristique, souvent utilisée, consiste à calculer pour les différents sous-ensembles de variables sélectionnées une estimation de l'erreur de généralisation par validation croisée. Le sous-ensemble de variables sélectionnées est celui qui minimise cette erreur de généralisation.

3.5 Les différentes approches de sélection de variables

Il existe trois familles de techniques dans la littérature dont la finalité est d'effectuer une sélection de sous-ensembles de variables [24]. Il y a d'un côté les méthodes par filtre (*filter methods*) qui consistent à effectuer la sélection indépendamment du classement pendant une étape de prétraitement. D'un autre côté, les approches à adaptateur (*wrapper methods*) utilisent le système de classification comme une boîte noire uniquement pour évaluer le pouvoir prédictif d'un groupe de variables. Dans un esprit similaire, la famille des méthodes Intégrées (*embedded methods*) utilise

également le classifieur mais en sélectionnant les variables durant la phase d'apprentissage.

Méthodes filtres - *Filters*

La sélection de variables ne s'effectue ni pour, ni par un classifieur à apprentissage (Figure II.3). Le filtrage, comme son nom l'indique, est utilisé en phase de prétraitement, ce qui permet de réduire à la fois la dimension des entrées et de se prémunir dans certains cas du phénomène de sur-apprentissage.

La simplicité et l'efficacité de l'approche par filtre est souvent mise en avant, comme par exemple dans [35] où, bien qu'un SVM (Support Vector Machines) non linéaire soit utilisé en tant que classifieur. La méthode de sélection se fait elle aussi par un SVM linéaire en amont de la reconnaissance.

D'autres systèmes neuronaux comme le PMC sont utilisés pour effectuer une sélection. Dans [36], les auteurs tentent d'éliminer les variables inutiles, le procédé se déroule en deux phases : la première, additive qui entraîne des PMC candidats avec un nombre croissant de neurones dans les couches cachées, puis, les candidats obtenus, la seconde, une phase de sélection par des tests de Fisher est effectuée sur les réseaux. Le processus stoppe dès l'obtention du plus petit réseau dont les variables et les neurones cachés ont une contribution significative pour le problème de régression-classification. Une discussion sur le même problème est donnée par [37].

Les méthodes par filtre ne reposent généralement pas sur des méthodes assez complexes, ce sont souvent des méthodes statistiques. Dans [38], les auteurs proposent pour accroître les performances de plusieurs classifieurs, la sélection, via une heuristique de sélection par corrélation, d'un sous-ensemble de variables pour les classifieurs. L'heuristique tient compte de l'utilité individuelle de chaque variable et de sa corrélation avec les autres variables. Dans le même ordre d'idée, [39] les auteurs proposent aussi une heuristique nommée FCBF (pour fast correlation-based filter) qui introduit la notion de corrélation de prédominance. Elle a l'avantage d'avoir une complexité quasi linéaire au lieu d'une complexité au moins quadratique comme il est fréquent d'en rencontrer dans les méthodes se basant sur la corrélation.

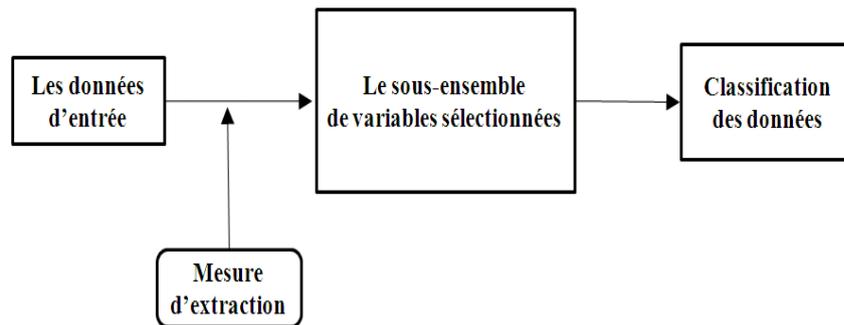


FIGURE II.3 – Le principe général d’une méthode de sélection de type *Filter* [40]

Méthodes à adaptateur - *Wrappers*

Le principe de base de ces méthodes est le suivant : nous considérons déjà acquis le classifieur et, sans aucune information a priori à son sujet, il est utilisé tel quel pour qualifier l’utilité d’un groupe de variables (Figure II.4). La difficulté de la méthode provient de la génération de ces groupes qui doit être renouvelée pour obtenir des ensembles plus performants à chaque itération. Le choix de la fonction objective et du critère d’arrêt sont eux aussi des problèmes difficiles à résoudre car ils sont la clé pour l’obtention d’une solution satisfaisante générée en un minimum d’itérations.

Pour disposer d’une méthode exacte (c’est-à-dire obtenir l’optimum), il serait toujours envisageable d’énumérer toutes les combinaisons de sous-ensembles et de les tester à travers le classifieur. Dans un cas pratique, sauf si l’on considère un nombre très restreint de variables, le problème est de classe NP et n’est donc pas réalisable. On utilise donc plus généralement des stratégies standard pour réduire le nombre de sous-ensembles à générer par des techniques de type séparation et évaluation, de recuit simulé, ou bien encore des méthodes basées sur des algorithmes génétiques [41]. Ces derniers sont d’ailleurs les plus puissants et les plus utilisés depuis quelques années. Dans [42], [43] les auteurs montrent comment les adapter dans le cas de la reconnaissance de chiffres manuscrits, la solution proposée est même renforcée dans [44] avec l’utilisation d’une version multi-objective.

Les méthodes à adaptateur sont des méthodes assez simples à mettre en œuvre et donnant de bons résultats. Mais le côté «force brute» peut être toutefois un frein à leur utilisation, si le fait de considérer le classifieur comme une boîte noire permettant de conserver une sorte d’universalité du résultat obtenu. Le nombre de combinaisons à tester est souvent très important et ne donne en général pas de meilleurs résultats que ceux des méthodes Intégrées.

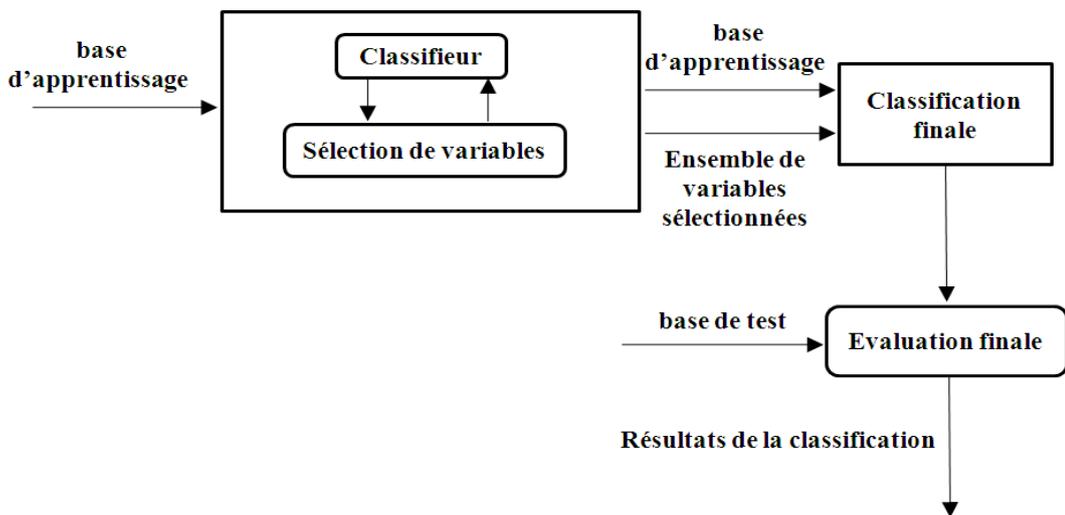


FIGURE II.4 – Le principe général d’une méthode de sélection de type *wrapper* [40]

Méthodes Intégrées - *Embeddeds*

Les méthodes Intégrées sont très utilisées ces dernières années. Ce type de méthodes choisit le groupe de variables durant l’apprentissage du classifieur. La recherche du meilleur sous-ensemble est guidée par l’apprentissage avec par exemple la mise à jour de la fonction objectif (Figure II.5). Des techniques déjà évoquées [45] peuvent être utilisées pour supprimer les variables non pertinentes.

Le problème des méthodes Intégrées est lié au fait qu’il faille justement adopter une stratégie en adéquation avec le type du classifieur : l’Optimal Brain Damage [45] ne s’applique qu’aux Perceptrons multicouches. Pour les SVM (Support Vector Ma-

chines, séparateurs à vastes marges [46].

Les méthodes connexionnistes représentent une grande majorité parmi les méthodes de type Intégrées. En générale, elles se basent sur des critères heuristiques permettant d'estimer l'importance d'une ou de plusieurs variables sur la performance globale du système. Dans la partie suivante nous dévoilons en détail ce type de méthodes très attrayantes.

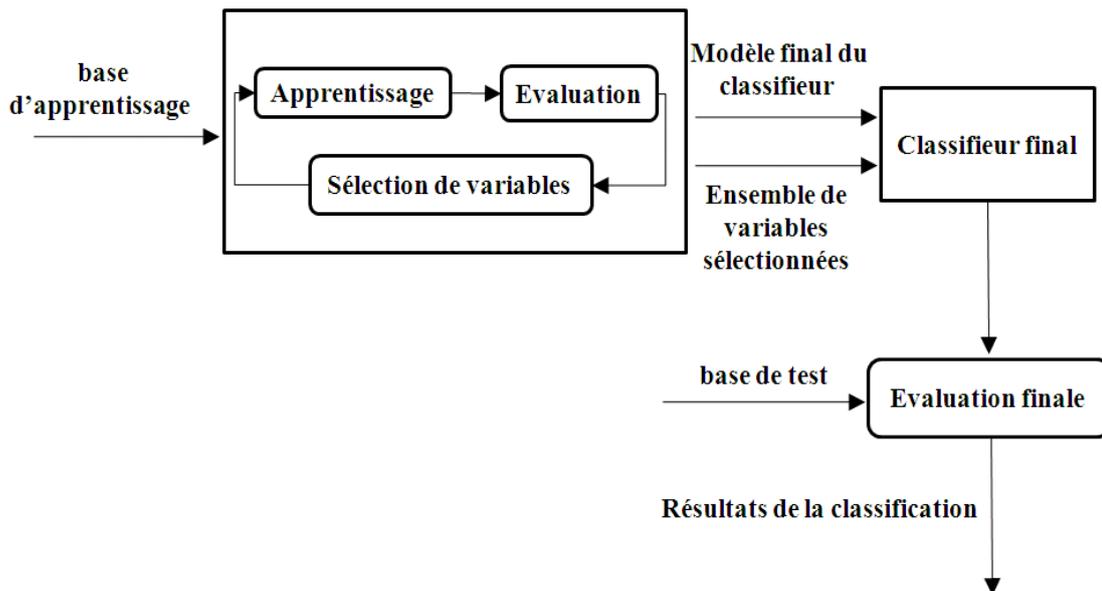


FIGURE II.5 – Le principe général d'une méthode de sélection de type *embedded*

3.6 Les méthodes connexionnistes de sélection de variables

La sélection de variables dans le domaine connexionniste est très attrayante et soulève de nombreux enjeux à la fois théoriques et applicatifs fondamentaux. En effet, dans le cas des réseaux connexionnistes, le processus de la sélection de variables peut être effectué parallèlement au processus de classement. Le sous-ensemble de variables ainsi sélectionnées sera choisi de façon à optimiser le critère d'apprentissage [34]. En plus, le nombre de variables est directement lié à l'architecture et à la complexité de la fonction réalisable par le modèle connexionniste.

Plusieurs approches ont été proposées dans la littérature. La plupart de ces tech-

niques emploient la première ou la deuxième dérivée de la fonction de coût par rapport aux poids pour estimer l'importance des connexions.

Les méthodes les plus largement employées sont : Optimal Brain Damage (OBD) proposée par Le Cun et al. [45], et Optimal Brain Surgeon (OBS) [47] par Hassibi et Stork qui est une amélioration de la précédente. Pedersen et al. ont proposé γ OBD et γ OBS [48], où l'estimation de l'importance d'un poids est basée sur le changement associé dans l'erreur de généralisation si le poids est élagué.

D'autres variantes d'OBD et d'OBS ont été proposées : Early Brain Damage (EBD) et Early Brain Surgeon (EBS) [49]. On peut citer aussi Optimal Cell Damage (OCD) développée par Cibas et al. Dans [50] qui est une extension d'OBD pour l'élagage des variables d'entrée. Ces méthodes se basent sur l'estimation systématique de l'importance d'une connexion qui est définie comme le changement de l'erreur causé par la suppression de ce poids.

Dans les paragraphes qui suivent, nous détaillons quelques méthodes en les regroupant par type. Les méthodes connexionnistes de sélection de variables sont en général de type "backward". L'idée générale est de faire converger un réseau jusqu'à un minimum local en utilisant toutes les variables et de faire ensuite la sélection. L'étape de sélection consiste à trier les variables par ordre croissant de pertinence, supprimer la ou les variables les moins pertinentes et ré-entraîné le réseau avec les variables restantes. Ce processus continue tant qu'un certain critère d'arrêt n'est pas satisfait. Les méthodes qui suivent cette procédure comportent donc deux phases : une phase d'apprentissage et une phase d'élagage qui peuvent être alternées.

Nous disons qu'une "vraie" procédure connexionniste de sélection de variables suit l'algorithme général suivant (Algorithme 2) :

Algorithm 2 algorithme général d'une méthode connexionniste de sélection de variables

- 1: Atteindre un minimum local
 - 2: Calculer la pertinence de chaque entrée
 - 3: Trier les entrées par ordre croissant de pertinence
 - 4: Supprimer les entrées dont la pertinence cumulée est inférieure à un seuil fixé
 - 5: Recommencer en 1. Tant que les performances estimées sur une base de validation ne chutent pas
-

Les méthodes de sélection de variables en apprentissage connexionniste peuvent se regrouper en trois grandes familles :

- Les méthodes d'ordre zéro.
- Les méthodes du premier ordre.
- Les méthodes du second ordre.

Méthodes d'ordre zéro

Pour estimer la pertinence d'une variable, les mesures d'ordre zéro utilisent les valeurs des paramètres du système d'apprentissage (les valeurs des connexions, la structure,...). Par exemple la mesure de pertinence HVS [51] repose sur les paramètres et la structure du réseau connexionniste. Dans le cas d'un Perceptron multicouches à une seule couche cachée, cette mesure est définie par :

- pertinence d'une variable	$\zeta_i = \sum_{j \in Hidden} \left[\frac{ w_{ji} }{\sum_{i' \in Input} w_{ji'} } * \sum_{k \in Output} \frac{ w_{kj} }{\sum_{j' \in Hidden} w_{kj'} } \right]$
- critère d'évaluation	$J(X_k) = \sum_{x_i \in X_k} \zeta_i$
- procédure de recherche	<i>Backward</i> + réapprentissage.
- critère d'arrêt	test statistique

Une autre méthode d'ordre zéro très efficace a été proposée par [52] : Automatic Relevance Determination (ARD). Dans cette méthode la pertinence d'une variable

est estimée par la variance de ses poids : la variable est éliminée si la variance correspondante est faible.

Méthodes du premier ordre

La dérivée de la fonction ψ que représente un système d'apprentissage connexionniste - un réseau - par rapport à chacune de ses variables est très utilisé comme mesure de pertinence des variables. Si une dérivée est proche de zéro pour tous les exemples, alors la variable correspondante n'est pas utilisée par le réseau, et peut donc être supprimé.

Dans le cas des PMC - Perceptrons multicouches -, cette dérivée peut se calculer comme une extension de l'algorithme d'apprentissage. Comme ces dérivées peuvent prendre aussi bien des valeurs positives que négatives, produisant une moyenne proche de zéro, c'est la moyenne des valeurs absolues qui est généralement utilisée- ce sont les grandeurs des dérivées qui nous intéressent. On trouve beaucoup de mesures de pertinences basées sur cette approche. La sensibilité de l'erreur à la suppression de chaque variable est utilisée par Moody dans [53]. Une mesure de sensibilité est calculée pour chaque variable x_i pour évaluer la variation de l'erreur en apprentissage si cette variable est supprimée du réseau. Le remplacement d'une variable par sa moyenne supprime son influence sur la sortie du réseau. La définition de la pertinence est :

$$\left| \begin{array}{l} - \zeta_i = R(w) - \tilde{R}(\bar{x}_i, w) \\ - \text{avec } \tilde{R}(\bar{x}_i, w) = \frac{1}{N} \sum_{k=1}^N \|y^k - \psi(x_1^k, \dots, \bar{x}_i^k, \dots, x_n^k)\|^2 \end{array} \right.$$

N est la taille de la base d'apprentissage. Quand cette taille est très grande, Moody propose d'utiliser une approximation qui donne la méthode de sélection suivante :

– pertinence d'une variable	$\zeta_i \cong^{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N [(x_i^k - \bar{x}_i) (y^k - \psi(x^k, w)) \frac{\partial \psi(x^k, w)}{\partial x_i}]$
– critère d'évaluation	$J(X_k) = \sum_{x_i \in X_k} \zeta_i$
– procédure de recherche	<i>Backward</i>
– critère d'arrêt	variation des performances en test

Méthodes du second ordre

Pour estimer la pertinence d'une variable, les méthodes du second ordre calculent la dérivée seconde de la fonction de coût par rapport aux poids. Ces mesures sont des extensions des techniques d'élagage des poids. La technique d'élagage la plus populaire est Optimal Brain Damage (OBD) proposée par Le Cun et al. [45]. OBD est basée sur l'estimation de la variation de la fonction de coût $R(w)$ lorsqu'un poids est supprimé du réseau. Cette variation peut être approximée à l'aide d'un développement en série de Taylor (Eq II.9) :

$$\delta \tilde{R}(w_i) = \sum_i \frac{\partial \tilde{R}(w)}{\partial w_i} \delta w_i + \frac{1}{2} \sum_i \sum_j \frac{\partial^2 \tilde{R}(w)}{\partial w_i \partial w_j} \delta w_i \delta w_j + O(\delta w^3) \quad (\text{II.9})$$

Tresp et al. [49] proposent deux extensions d'OBD et d'OBS : EarlyBrain Damage (EBD) et EarlyBrain Surgeon (EBS). EBD et EBS peuvent être utilisées avec le "earlystopping" comme critère d'arrêt de l'apprentissage. Dans EBD, par exemple, la sensibilité d'un poids est donnée par la formule suivante :

$$pertinence(w_i) = \frac{1}{2} \frac{\partial^2 \tilde{R}(w)}{\partial w_{ji}^2} w_{ji}^2 - \frac{\partial \tilde{R}(w)}{\partial w_{ji}} w_{ji} + \frac{(\frac{\partial \tilde{R}(w)}{\partial w_{ji}})^2}{\frac{\partial^2 \tilde{R}(w)}{\partial w_{ji}^2}} \quad (\text{II.10})$$

A partir de cette définition de pertinence , [54] propose la méthode ECD (EarlyCell Damage) :

- pertinence d'une variable $\zeta_i = \frac{1}{2} \sum_{j \in fan-Out(i)} \frac{\partial^2 \hat{R}(w)}{\partial w_{ji}^2} w_{ji}^2 - \frac{\partial \hat{R}(w)}{\partial w_{ji}} w_{ji} + \frac{(\frac{\partial \hat{R}(w)}{\partial w_{ji}})^2}{\frac{\partial^2 \hat{R}(w)}{\partial w_{ji}^2}}$
- critère d'évaluation $J(X_k) = \sum_{x_i \in X_k} \zeta_i$
- procédure de recherche *Backward*
- critère d'arrêt test statistique

3.7 La sélection de variables et la classification du diabète dans la littérature

Dans cette partie, nous citons les différents travaux réalisés dans le domaine de sélection de variables avec les réseaux de neurones de manière générale (Table II.2) (1-2). Après, nous passons aux travaux de sélection consacrés pour la reconnaissance du diabète (Table II.2) (3-10). Le Tableau suivant présente les différents travaux consultés :

Auteurs	Titre	Approche Utilisée	Expérimentation	Résultats obtenus
1-T. Cibas, F. Fogelman Soulié, P. Galinari, and S. Raudys.	Variable Selection with Optimal Cell Damage [50]	Une Méthode de type Embedded appelée OCD :(Optimal Cell Damage) évalue l'importance des variables d'entrée dans un RNMC et élague les moins utiles	Évaluation de la méthode sur deux problèmes : -prédiction d'une série chronologique synthétique -Classification des formes d'onde.	Les résultats expérimentaux montrent que la méthode OCD a prouvé son efficacité sur les deux problèmes artificiels relativement complexes.
2- J. Proriol	Selection of variables for neural network analysis Comparisons of several methods	Cinq méthodes différentes pour choisir les variables les plus importantes sont comparées :test-F, analyse de composant principal (ACP), Arbre de décision (CART),	Les variables choisies par les différentes approches sont appliquées avec le classifieur pour évaluer la performance de chaque méthode.	Le taux d'apprentissage est calculé dans 2 cas : le cas de 3 classes (b, c, uds) et le cas de 2 classes (b, udsc) le classement des cinq méthodes étudiées est le suivant : pour 3 classes :

	with high energy physics data [55]	évaluation de poids, et optimal cell damage (OCD) en vue de classifier des données de physique de haute énergie avec les réseaux neuronaux.		OCD-(PMC-RBF), OCD-(PMC), F-test-(PMC-RBF), ACP-(PMC), CART-(PMC), WEIGHT-(PMC), F-test-(PMC). pour 2 classes : ACP-(PMC), OCD-(PMC-RBF), F-test-(PMC-RBF), CART-(PMC), OCD-(PMC), WEIGHT-(PMC), F-test-(PMC) - nous remarquons que la méthode OCD est classée en 2 premières positions dans la 1er expérimentation et en 2eme position dans la 2eme expérimentation ce qui confirme la performance de cette méthode.
3- SarojiniBala krishnan, RamarajNaraswamy	Feature Selection Using FCBF in Type II Diabetes Databases [56]	Une Méthode de sélection de variables : <i>Fast Correlation-Based Filter</i> (FCBF) basée sur une approche de type filter utilise l'incertitude symétrique d'attributs comme évaluateur et SVM comme classifieur	La démarche est validée par la moyenne d'amélioration des performances du classifieur SVM en utilisant la base de données PIMA.	Les performances obtenues sont : - Att TC% -SVM 8 77.474 -(FCBF+ 4 SVM) 77.995
-4 Asha Gowda Kare gowda,A. S. Manjunath	Comparative study of attribute selection using gain ratio And	deux Méthodes de type filtre utilisées : -L'arbre C4.5 utilise le ratio de gain (GR) pour déterminer les grands	Les deux types d'approches ont été utilisés pour illustrer l'importance de la sélection d'un sous-ensemble de	Les performances obtenues sont : - Att TC% -RBF 8 81.20 -GR+RBF 5 86.46

et M.A. Jaya-ram	correlation based feature selection [57]	écarts et sélectionner les variables importantes. -Algorithme génétique(GA) basé sur la corrélation pour la sélection de variables (CFS). Evaluation des deux approches par (PMC) et un (RBF).	caractéristiques pour la classification de base de données PIMA diabétiques Indien .	-GA-CFS 4 88.00 +RBF - -PMC 8 72.88 -GR+PMC 5 78.21 -GA-CFS 4 79.5 +PMC
5- Md. Monirul Kabir a, Md.Monirul Islam b, Kazuyuki Murase	A new wrapper feature selection approach using neural network [58]	Un Algorithme de sélection de variables basé sur l'approche wrapper en utilisant un RNMC, il est nommé CAFS. Son principe est la détermination automatique des architectures des RNs durant le processus de la SV	Ils ont évalué la performance du CAFS sur huit bases de données parmi eux la base PIMA.	Les performances obtenues sont : - Att TC% -PMC 8 75.95 -(CAFS+ 6 76.18 PMC)
6- Sancho Salcedo-Sanz, Gustavo Camps-Valls, Fernando Pérez-Cruz, José Sepúlveda-Sanchis, and Carlos Bousoño-Calzón.	Enhancing Genetic Feature Selection Through Restricted Search and Walsh Analysis [40]	Deux améliorations au niveau d'un algorithme génétique 1er :(m-features operator) pour fixer dans chaque itération le nombre de dispositifs à choisir et réduire l'espace de recherche (approche wrapper). 2eme : l'expansion de Walsh de la fonction de forme physique (approche filter).	Trois vrais ensembles de données biologiques sont employés pour examiner le comportement des deux approches proposées parmi eux la base PIMA du diabète.	Les performances obtenues sont : - Att TC% -RN 8 74.95 -RN+AAG 5 78.28 -RN+B' 5 79.55

<p>7- Cheng-Lung Huang, Chieh-Jen Wang</p>	<p>A GA-based feature selection and parameters optimization for support vector machines [59]</p>	<p>Une approche Wrapper basée sur un algorithme génétique pour la sélection de variables et un SVM pour la classification.</p>	<p>l'approche proposée est testée sur plusieurs bases du monde réel (11 bases d'UCI parmi eux la base PIMA du diabète). Les résultats obtenus sont comparés avec un algorithme appelé (Grid algorithme).</p>	<p>Les performances obtenues sont :</p> <table border="0"> <tr> <td>-</td> <td>Att</td> <td>TC%</td> </tr> <tr> <td>-GA</td> <td>4</td> <td>81.5</td> </tr> <tr> <td>-Grid</td> <td>8</td> <td>77.3</td> </tr> </table>	-	Att	TC%	-GA	4	81.5	-Grid	8	77.3									
-	Att	TC%																				
-GA	4	81.5																				
-Grid	8	77.3																				
<p>8- Jihoon Yang and VasantHonnar</p>	<p>Feature Subset Selection Using A Genetic Algorithm. [60]</p>	<p>Une approche wrapper basée sur une méthode de sélection de variables multicritère. Ils ont utilisé un AG pour la sélection et un RN pour la classification.</p>	<p>Les expériences rapportées ici ont employé des ensembles de données réels pour explorer la praticabilité d'employer des algorithmes génétiques pour le choix de sous-ensemble de dispositif pour des classificateurs de réseaux neurologiques.</p>	<p>Les performances obtenues sont :</p> <table border="0"> <tr> <td>-</td> <td>Att</td> <td>TC%</td> </tr> <tr> <td>-RN</td> <td>8</td> <td>76.3</td> </tr> <tr> <td>-GA</td> <td>4</td> <td>79.5</td> </tr> </table>	-	Att	TC%	-RN	8	76.3	-GA	4	79.5									
-	Att	TC%																				
-RN	8	76.3																				
-GA	4	79.5																				
<p>9- Yong Wang, Lin Li , Jun Ni , Shuhong Huang</p>	<p>Feature selection using tabu search with long-term memories and probabilistic neural networks [61]</p>	<p>Une hybridation de la recherche taboue et les réseaux de neurones probabilistes pour la sélection de variables</p>	<p>Quatre algorithmes hybrides (SFS + PNN, SFFS + PNN, TS1 + PNN, et TS2 + PNN) sont implémentés en utilisant des ensembles de données du monde réel.</p>	<p>Les performances obtenues sont :</p> <table border="0"> <tr> <td>-</td> <td>Att</td> <td>TC%</td> </tr> <tr> <td>-SFS+RN</td> <td>4</td> <td>74.95</td> </tr> <tr> <td>-SFFS+RN</td> <td>4</td> <td>78.28</td> </tr> <tr> <td>-TS2+RN</td> <td>4</td> <td>79.55</td> </tr> <tr> <td>-TS1+RN</td> <td>4</td> <td>79.55</td> </tr> <tr> <td>-TS1+KNN</td> <td>4</td> <td>79.55</td> </tr> </table>	-	Att	TC%	-SFS+RN	4	74.95	-SFFS+RN	4	78.28	-TS2+RN	4	79.55	-TS1+RN	4	79.55	-TS1+KNN	4	79.55
-	Att	TC%																				
-SFS+RN	4	74.95																				
-SFFS+RN	4	78.28																				
-TS2+RN	4	79.55																				
-TS1+RN	4	79.55																				
-TS1+KNN	4	79.55																				

TABLE II.2 – Tableau des travaux consultés pour la sélection de variables

4 Contribution

Beaucoup de travaux ont été réalisés en employant une approche neuronale pour la classification et ont abouti à des résultats intéressants. Cela dit, chaque modèle de réseau de neurones diffère d'un autre par ses paramètres expérimentaux, sa complexité d'implémentation et son domaine d'applicabilité. Ainsi, chaque modèle possède ses avantages et ses inconvénients, faisant qu'il soit plus approprié pour une tâche donnée qu'une autre.

Pour la phase de sélection, plusieurs travaux dans ce domaine sont consultés et nous avons déduit que les approches connexionnistes (ou intégrées) sont meilleures par rapport aux autres approches par ce que dans une approche filter (sélection après classification) « l'ensemble de variables sélectionnées peut ne pas donner de bons résultats de classification » et dans une approche de type wrapper (sélectionner un ensemble de variables ensuite évaluer la performance du classifieur par l'ensemble de variables sélectionnées et répéter cette opération jusqu'à un critère d'arrêt) « le choix du critère d'arrêt est un problème très difficile à résoudre car l'opération de sélection va prendre beaucoup de temps pour atteindre une bonne performance du système et peut ne jamais l'atteindre ».

Cette conclusion nous a amené à proposer pour la résolution de notre problématique une approche basée sur un réseau de neurones comme classifieur amélioré par une tâche de sélection de variables de type embedded. La méthode de sélection choisie est la méthode OCD (Optimal Cell Damage) [50], [62]. Elle se base sur la notion de pertinence des variables. Ce choix est justifié par l'efficacité de cette méthode car la recherche du meilleur sous-ensemble est guidée par l'apprentissage du classifieur, ce qui génère une optimisation de l'architecture du modèle du réseau de neurones et une amélioration des performances de la classification.

5 Conclusion

La mise en œuvre des systèmes de classification peut heurter à plusieurs difficultés parmi elles la présence d'attributs redondants ou non pertinents dans les bases de données. Cependant, en apprentissage artificiel le nombre de variables est essentiel et doit être le plus important possible. C'est dans ce contexte que la tâche d'extraction et de sélection de variables s'introduit. Ce qui entraîne forcément une réduction de la dimensionnalité du problème et améliore considérablement la précision des performances de classification.

Chapitre III

Expérimentation

1 Introduction

Les données utilisées par les applications réelles qui intègrent des techniques d'apprentissage renferment souvent de nombreux attributs redondants ou bruités. Cette redondance peut nuire à la découverte de structures correctes par les algorithmes de classification automatique. Intuitivement, une information redondante, qui est représentée par de nombreux attributs, risque d'en occulter une autre qui bien qu'elle soit potentiellement pertinente est moins présente. Dans le pire des cas, l'information pertinente est noyée parmi de nombreux attributs qui expriment tous une même idée sans intérêt pour l'utilisateur. Cette situation extrême risque de conduire à une classification sans réel intérêt pour l'utilisateur.

Dans ce chapitre nous présentons en premier lieu l'approche proposée pour effectuer la tâche de sélection, ensuite nous aborderons la partie de modélisation de notre application, après nous passerons à l'expérimentation réalisée et les résultats obtenus par cette expérimentation avec leurs interprétations. Enfin nous faisons une comparaison avec les résultats de certains travaux portant sur le même sujet de manière à situer le travail effectué par rapport à ces travaux et mettre en évidence les contributions qu'il amène.

2 Approche proposée

Pour la résolution de notre problématique, nous avons proposé une méthode de sélection de variables de type (*Embedded*). C'est une méthode de second ordre des méthodes connexionnistes, elle est appelée OCD (Optimal Cell Damage), elle est utilisée avec les réseaux de neurones de type Perceptron Multicouches. Cette méthode est basée sur la mesure de pertinence des connexions, où l'importance de chaque variable s'obtient en sommant les importances des connexions qui partent de celle-ci.

La pertinence d'une connexion est définie par (Eq III.1).

$$Pertinence(w_j) = \frac{1}{2} H_{jj} w_j^2 = \frac{1}{2} \frac{\partial^2 J}{\partial w_j^2} w_j^2 \quad (\text{III.1})$$

Où J est la fonction de coût (erreur quadratique), H sa matrice hessienne (dérivée seconde par rapport aux paramètres w). Le Hessien de la fonction de coût est utilisé pour calculer la dépendance du modèle par rapport aux poids.

Pour utiliser cette mesure de pertinence d'une connexion comme critère de sélection d'une variable, il faut calculer la pertinence d'un neurone de la couche d'entrée en faisant l'approximation suivante (Eq III.2) :

$$Pertinence(x_i) = \sum_{j \in FanOut(i)} Pertinence(w_j) \quad (III.2)$$

Où $FanOut(i)$ représente l'ensemble des poids partant de la variable i .

En utilisant les équations (Eq III.1) et (Eq III.2), nous obtenons (Eq III.3) :

$$S_i = Pertinence(x_i) = \frac{1}{2} \sum_{j \in FanOut(i)} \frac{\partial^2 J}{\partial w_j^2} w_j^2 \quad (III.3)$$

La technique OCD considère que, H le Hessien de la fonction de coût, est une matrice diagonale. En faisant l'approximation du coût par une fonction quadratique. Cela revient à supposer que l'influence sur le coût de la suppression simultanée de deux poids est la somme des influences des deux suppressions réalisées individuellement. Dans le cas d'un RN, le Hessien "diagonal" peut alors être estimé en $O(N)$ (avec O représente la complexité et N est le nombre d'exemples).

Cibas et al. [50] [62] ont proposé un algorithme de sélection de variable (Algorithme 3) basé sur l'équation (Eq III.3), cet algorithme utilise une procédure de recherche de type (*backward*).

Algorithm 3 OCD

- 1: Répéter la rétro-propagation (chapitreII, Algorithme1) jusqu'à convergence de la variation du coût estimé sur des données de validation.
 - 2: Calculer la pertinence de chaque entrée grâce à l'équation (III.3)
 - 3: Trier les entrées par ordre croissant de pertinence
 - 4: Supprimer les entrées dont la pertinence cumulée $S'_i = \sum_{j=1}^i S_j$ est inférieure à un seuil fixé q
 - 5: Recommencer en 1 tant que les performances estimées sur une base de test ne chutent pas.
-

3 Modélisation

Notre approche de conception suit le modèle UML (Unified Modelling Language). Plusieurs raisons nous ont conduit à ce choix :

- La première est sa normalisation par l'OMG [63]. L'historique a montré que la profusion des notations est, à moyen terme, préjudiciable aux entreprises comme à leurs fournisseurs. Toute norme doit donc être considérée avec le plus grand sérieux, en particulier les normes du domaine public, comme c'est le cas pour l'UML.
- La deuxième raison est l'intérêt montré par les informaticiens pour ce langage de modélisation.
- La troisième raison est la possibilité d'utiliser le même atelier de génie logiciel, de puis l'expression des besoins jusqu'à la génération de tout ou partie de l'application.
- La dernière raison, est d'utiliser les principes et concepts objets pour enrichir la démarche de conception des systèmes d'aide à la décision. On en attend des améliorations dans le sens tout à la fois de la richesse, d'une modularité, d'une cohérence et d'une rigueur accrues.

3.1 Définition des besoins de l'utilisateur

L'utilisateur de notre application est le médecin, il a besoin d'un outil pour l'aider à effectuer un diagnostic du diabète pour un patient, cette application a besoin d'avoir les caractéristiques suivantes :

- Une meilleure performance.
- Un minimum d'erreur.
- Demande le minimum d'informations pour faire le diagnostic (les plus pertinentes).
- Interprétable (donne la raison pour laquelle elle a trouvé ces résultats).

3.2 Les acteurs principaux de l'application

Un acteur représente un rôle joué par une personne, un groupe de personnes ou par une chose qui interagit avec le modèle. Les acteurs se recrutent parmi les

utilisateurs de l'application et aussi parmi les responsables de sa configuration et de sa maintenance. Dans notre modèle nous avons considéré trois acteurs principaux :

Administratif

Dans notre cas, l'administratif est un informaticien ou le programmeur, il est fortement impliqué dans l'organisation de l'application, il a comme rôle la gestion et le contrôle de l'application.

Expert

Dans notre cas, l'expert est un médecin ou un groupe de médecins (phase de conception), il joue un rôle essentiel dans la configuration de l'application, il s'agit de la validation du système de classification après plusieurs tests des différents modèles de classifieurs.

Médecin

Le médecin est l'utilisateur principale de l'application, il l'utilise pour l'aider à faire un diagnostic de la maladie du diabète pour un patient.

L'étude des cas d'utilisation a pour objectif de déterminer ce que chaque acteur attend du système. La détermination des besoins est basée sur la représentation de l'interaction entre l'acteur et le système. Donc un cas d'utilisation est une abstraction d'une partie de comportement du système, qui s'instancie à chaque utilisation du système par une instance d'un acteur.

3.3 Les diagrammes de cas d'utilisation

Les cas d'utilisation représentent un élément essentiel de la modélisation orientée objets : ils interviennent très tôt dans la conception et doivent en principe permettre de concevoir et de construire un système adapté aux besoins de l'utilisateur (Build the right system). Ils doivent également servir de fil rouge tout au long du développement, lors de la phase de conception, d'implémentation et de tests. Ils servent

donc aussi bien à définir le produit à développer, à modéliser le produit et à tester le produit réalisé. Dans notre application, nous avons considéré deux cas d'utilisation :

création du classifieur

Pour la création d'un classifieur, l'administratif importe la base de données, crée l'architecture initiale du réseau de neurones, fait entrer les paramètres d'apprentissage et les paramètres de sélection de variables. Ensuite le modèle fera l'apprentissage et la sélection en parallèle et donnera une nouvelle architecture du réseau de neurones. Après cette phase, l'administratif lancera un test pour l'évaluation des performances du modèle créé par l'application. Les résultats de la classification vont être consultés par l'expert pour valider le modèle du classifieur qui va être enregistré par l'administratif pour ensuite être utilisé par le médecin (l'utilisateur final). Voir (Figure III.1).

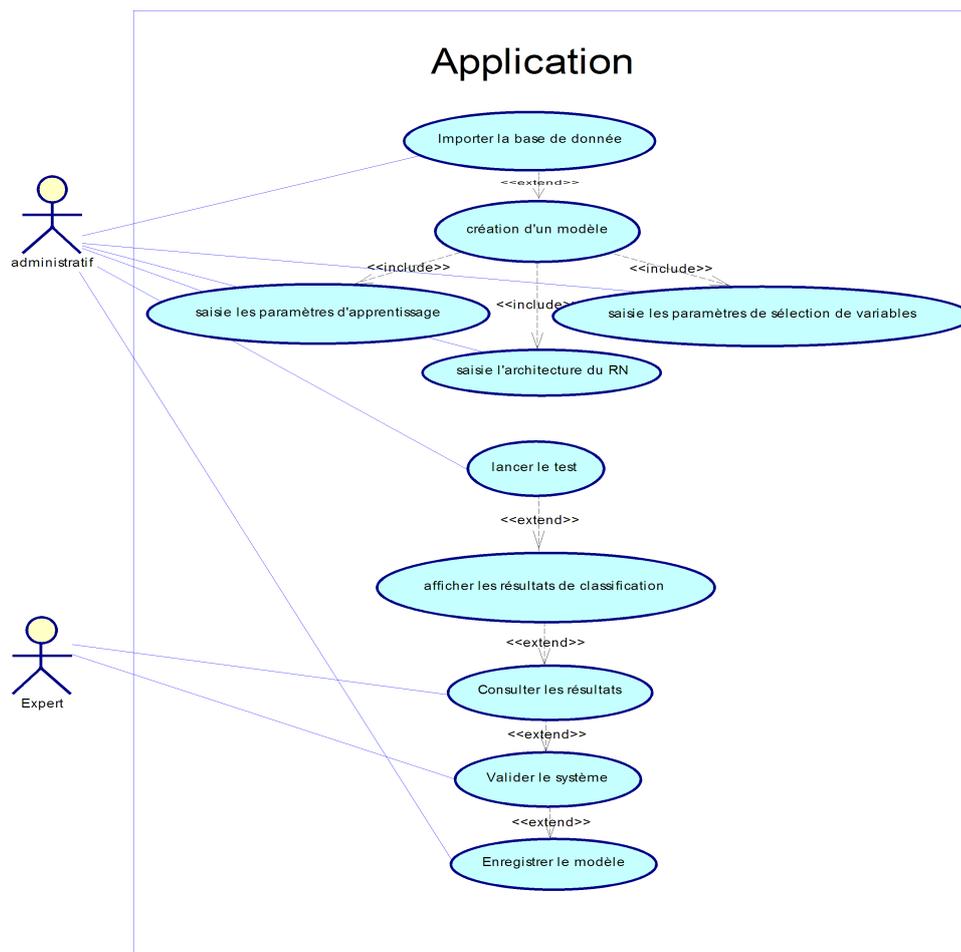


FIGURE III.1 – Diagramme de cas d'utilisation « création d'un classifieur »

aide au diagnostic

Pour faire un diagnostic, le médecin utilise un modèle validé par l'expert, il saisie les informations d'un patient et voir les résultats trouvés par le modèle ie :« diabétiques » ou « non diabétiques ». Ces résultats vont l'aider pour bien diagnostiquer la maladie chez le patient. Voir (Figure III.2).

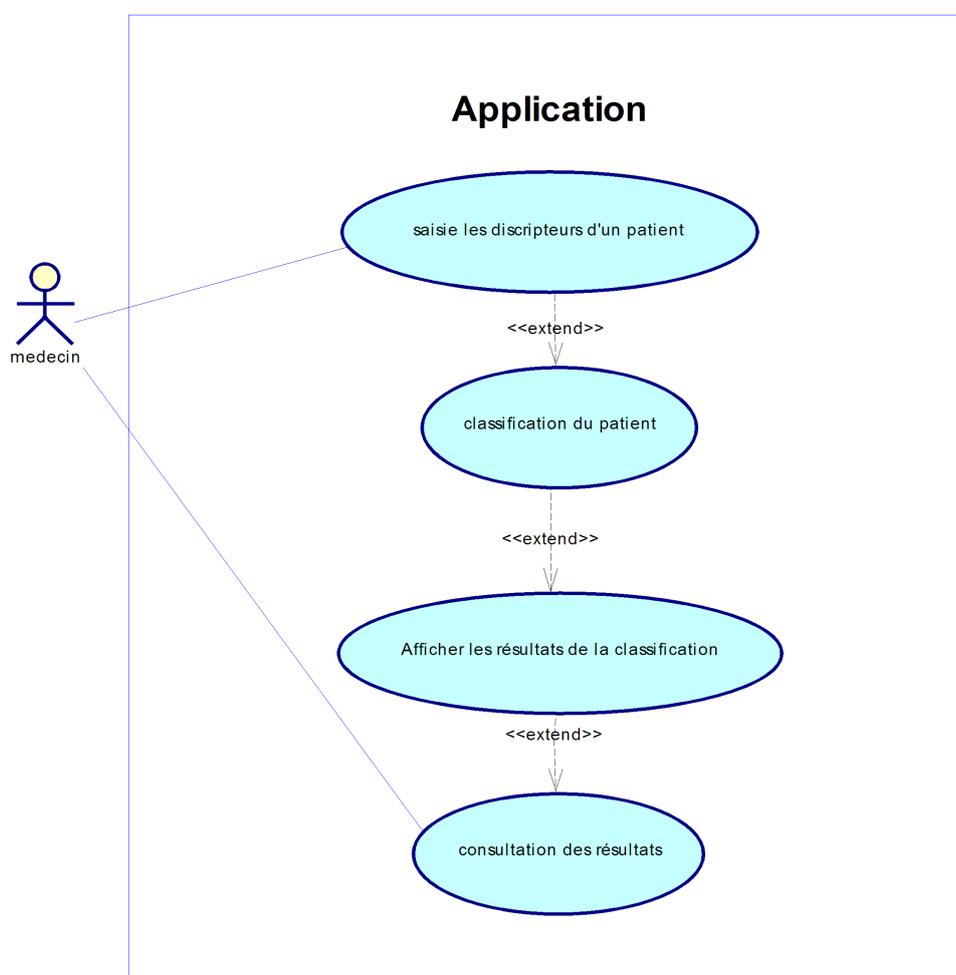


FIGURE III.2 – Diagramme de cas d'utilisation « aide au diagnostic »

3.4 Les diagrammes de séquence

Dans la réalisation de notre application, nous avons considéré deux diagrammes de séquence :

Création d'un modèle validé

Pour la création d'un modèle validé, l'administratif importe la base de données en utilisant l'interface de l'application. Ensuite, il saisira l'architecture du modèle qu'il désire créer. Après, le modèle fera l'apprentissage et la sélection en entrant les paramètres nécessaires pour effectuer cette tâche par l'administratif. Par la suite, le modèle retournera l'ensemble de variables sélectionnées et les résultats de classification. Ces résultats seront affichés par l'interface et seront consultés par l'expert pour valider le modèle créé. Voir (Figure III.3).

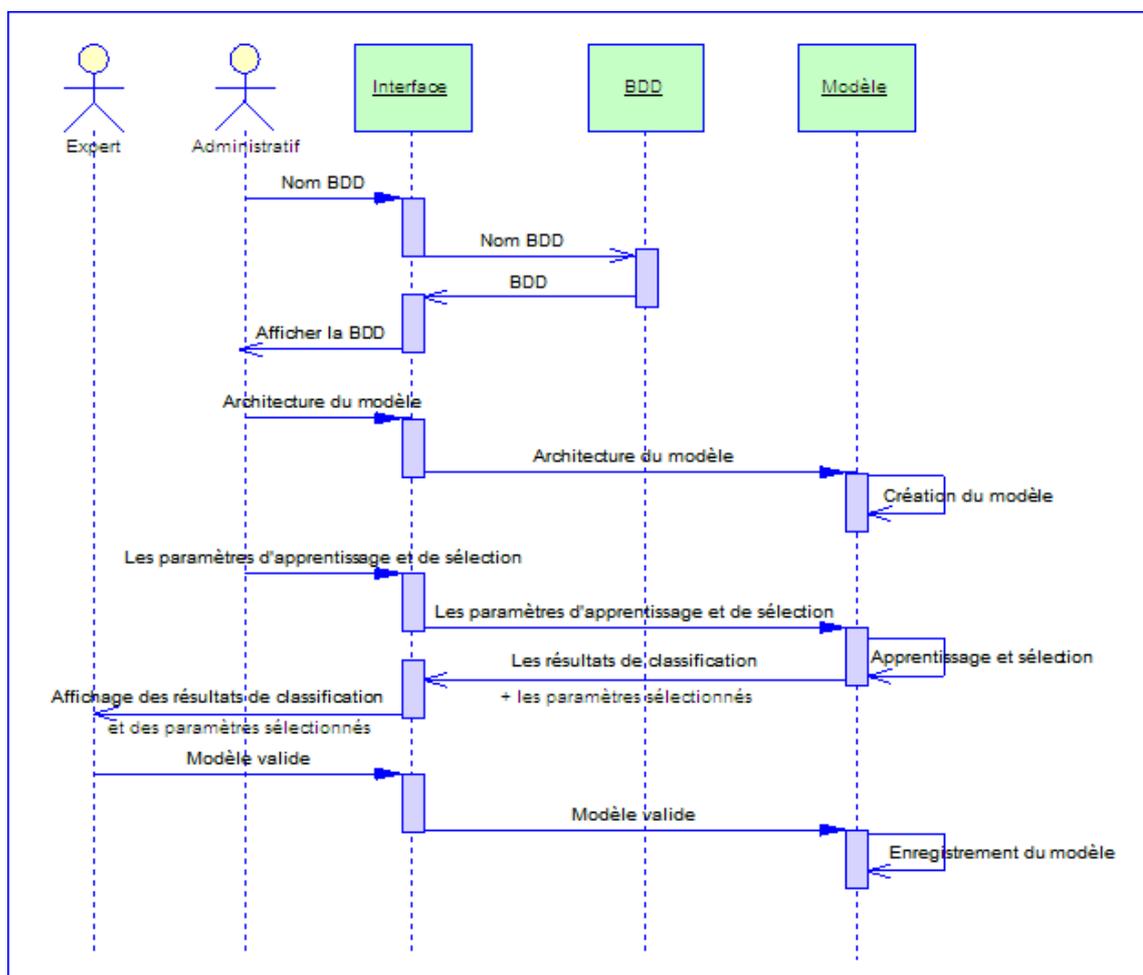


FIGURE III.3 – Diagramme de séquence « création d'un modèle validé »

Utilisation d'un modèle validé

Pour l'utilisation de l'application dans l'aide au diagnostic, le médecin remplit les informations d'un patient selon les paramètres pris en considération par le modèle validé. Ensuite le modèle fera la classification de ce patient et retournera le résultat trouvé. Ce résultat sera affiché par l'interface pour être ensuite utilisé comme une référence dans le diagnostic de la maladie chez le patient. Voir (Figure III.4).

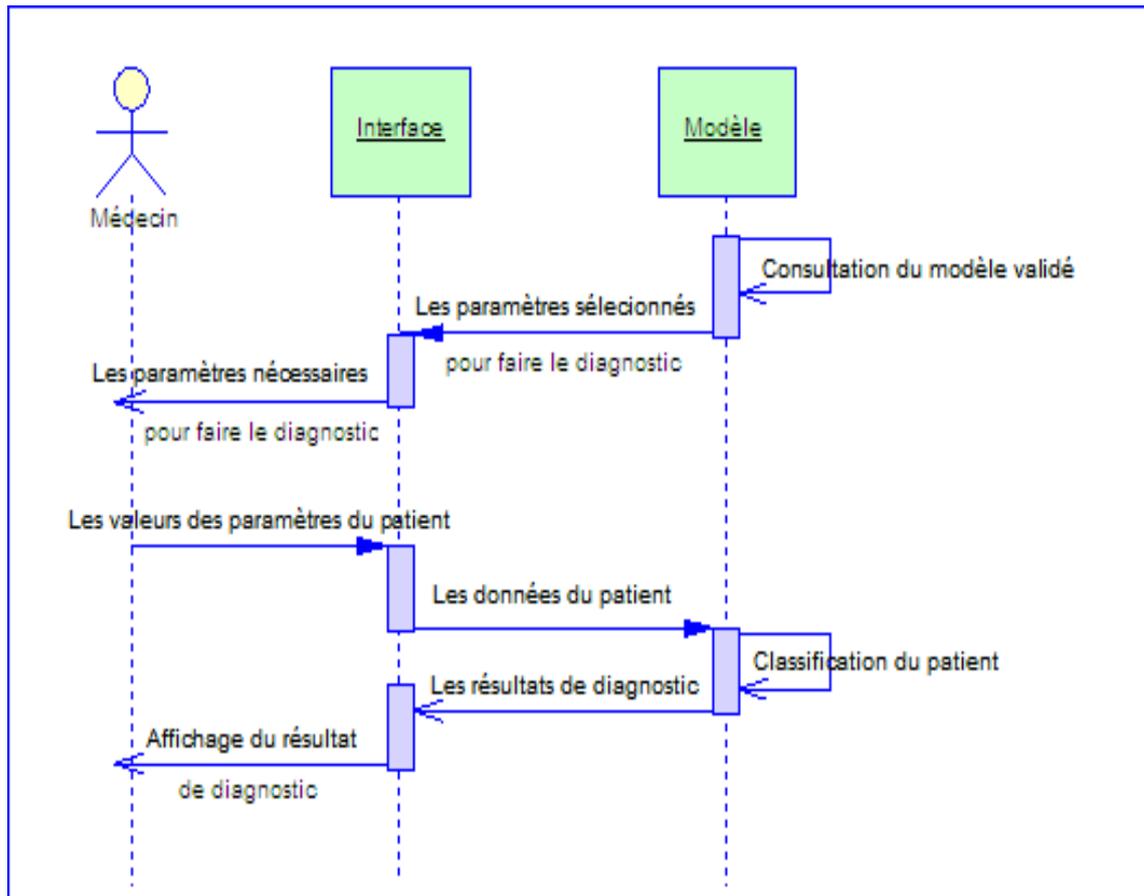


FIGURE III.4 – Diagramme de séquence « utilisation d'un modèle validé »

3.5 Le diagramme des classes

Dans la réalisation de notre application nous avons considéré trois grandes classes et une interface. Voir (Figure III.5) :

Classe BDD

Elle est utilisée pour importer le fichier de la base de données et pour analyser les données de cette base. Elle contient six méthodes :

- ChargerBase : une méthode publique utilisée pour charger le fichier de la base de données (fichier est de type .arff), elle a comme entrée le nom de la base.
- GetBdd : une méthode publique utilisée pour récupérer les données de la base chargée.
- GetNbrAttr : elle donne le nombre d'attributs ou de variables de la base.
- GetNbrInst : elle donne le nombre d'instances de la base.
- GetNbrPos : elle donne le nombre d'instances positives de la base.
- GetNbrNeg : elle donne le nombre d'instances négatives de la base.

Classe PMC

Elle est utilisée pour la classification des données par un perceptron multicouche. Cette classe contient six méthodes :

- CréerModele : une méthode publique utilisée pour la création d'un modèle du classifieur en donnant l'architecture du modèle comme entrée.
- ApprModele : une méthode publique utilisée pour l'apprentissage du modèle créé. Elle a comme entrées le pas d'apprentissage, le maximum d'itération, la base d'apprentissage et la base de validation. Elle retourne un tableau qui contient l'erreur d'apprentissage et l'erreur de validation.
- TestModele : une méthode publique utilisée pour le test du modèle après l'apprentissage. Elle a comme entrée la base de test et retourne le tableau de performance de la classification.
- InitialiseModele : une méthode publique pour initialiser aléatoirement le modèle créé.
- FctActivation : une méthode privée utilisée au niveau locale de la classe comme fonction d'activation des sorties du modèle. Cette fonction est de type sigmoïde.
- TestErrVal : une méthode privée utilisée pour tester le changement de l'erreur générée par la base de validation entre deux itérations a but de simuler le

principe de l'arrêt par validation croisée.

Classe OCD-PMC

Elle est utilisée pour la sélection de variables par la méthode OCD et en parallèle la classification par le PMC. Cette méthode contient toutes les méthodes de la classe précédente avec six autres méthodes en plus :

- PertiDesConnexions : une méthode privée utilisée pour calculer la pertinence des connexions sortantes de la couche d'entrée. Elle retourne une matrice diagonale qui représente le Hessien de la fonction de coût.
- PertiDesVariables : une méthode privée utilisée pour calculer la pertinence des variables. Elle a comme entrée la matrice des pertinences des connexions sortantes des variables d'entrées. Elle retourne un tableau qui contient la pertinence de chaque variable.
- SupriméVariable : une méthode privée utilisée pour supprimer les variables les moins pertinentes. Elle a comme entrée le tableau qui contient les pertinences des variables d'entrée.
- SupriméConnexions : une méthode privée utilisée pour supprimer les connexions sortantes d'une variable supprimée. Elle a comme entrée le tableau qui contient les indices des variables supprimées.
- GetVariablesSupprimés : une méthode publique utilisée pour récupérer le tableau qui contient les indices des variables supprimées.
- TestRequette : une méthode publique utilisée pour interroger le modèle sur une requête donnée. Cette requête contient les valeurs des paramètres pris en considération par le modèle chez un patient.

Interface

Elle est utilisée pour faire l'interaction entre les trois premières classes et utilise des objets de ces classes. Elle est utilisé aussi pour récupérer les données saisies par l'utilisateur et afficher les résultats obtenus. Pour plus de détails sur le fonctionnement de l'interface voir (Annexe B, Manuel d'utilisation).



FIGURE III.5 – Diagramme des classes

4 Expérimentations

4.1 Base de données PIMA

Les tests de la méthode proposée sont effectués sur la base de données Pima Indians Diabetes [64]. L'ensemble de données a été choisi du dépôt d'UCI qui réalise une étude sur 768 femmes Indiennes Pima (500 non diabétique 268 Diabétiques). Ces mêmes femmes qui ont stoppé leurs migrations en Arizona (Etats-Unis) adoptant un mode de vie occidentalisé développent un diabète dans presque 50% des cas. 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 descripteurs cliniques de la base

Les huit descripteurs cliniques sont :

- Npreg : nombre de grossesses.
- Glu : concentration du glucose plasmatique.
- BP : tension artérielle diastolique, (*mm Hg*).
- SKIN : épaisseur de pli de peau du triceps, (*mm*).
- Insuline : dose d'insuline, (*mu U/ml*).
- BMI : index de masse corporelle, (*poids en kg/(taille m)²*).
- PED : fonction de pedigree de diabète (l'hérédité).
- Age : âge (Année).

Analyse des données de la base

(Table III.1) contient les informations sur les paramètres pris en considération.

La (Figure III.6) représente l'histogramme de ces paramètres :

Nom d'attribut :	Type	Min / Max	Moyen	Écart type	Distincts
Npreg	Numérique	0 / 17	3 .845	3.37	17
Glu	Numérique	0 / 199	120.895	31.973	136
BP	Numérique	0 / 122	69.105	19.356	47
Skin	Numérique	0 / 99	20.536	15.952	51
Insu	Numérique	0 / 846	79.799	115.244	186
Bmi	Numérique	0 / 67.1	31.993	7.884	248
Ped	Numérique	0.078/2.42	0.472	0.331	517
Age	Numérique	21 / 81	33.241	11.76	52

TABLE III.1 – Informations sur les descripteurs de la base

(Table III.2) contient les informations sur les exemples de la base :

N° de la Classe	Label	Nombre
1	Tested _negative	500 exemple
2	Tested _positive	268 exemple

TABLE III.2 – Informations sur les exemples de la base

- un ensemble d'apprentissage.
- un ensemble de validation.
- et un ensemble de test.

La répartition entre les trois ensembles est faite selon l'ordre suivant. Voir (Table III.3) :

Apprentissage	Validation	Test
[positifs+Négatifs]	[positifs+Négatifs]	[positifs+Négatifs]
[98+158]=256	[87+169]=256	[83+173]=256

TABLE III.3 – Repartitionnement de la base

4.3 Les critères d'évaluation

Les performances de classification des données ont été évaluées par le calcul des vrais positifs (VP), vrais négatifs (VN), faux positifs (FP) et faux négatifs (FN), le pourcentage de sensibilité (SE), la spécificité (SP) et le taux de classification (TC), leurs définitions respectives sont les suivantes :

- VP : diabétique classé diabétique.
- VN : non diabétique classé non diabétique.
- FP : non diabétique classé diabétique.
- FN : diabétique classé non diabétique.

La Sensibilité : c'est la capacité de donner un résultat positif quand la maladie est présente. Elle est calculée par (Eq III.4) :

$$S_e = \frac{VP}{VP + FN} \quad (\text{III.4})$$

La Spécificité : c'est la capacité de donner un résultat négatif quand la maladie est absente. Elle est calculée par (Eq III.5) :

$$S_p = \frac{VN}{VN + FP} \quad (III.5)$$

Le Taux de classification : c'est le pourcentage des exemples correctement classés. Il est calculé par (Eq III.6) :

$$TC = \frac{VP + VN}{VP + VN + FP + FN} \quad (III.6)$$

4.4 Résultats et interprétation

Expérimentation 1

Dans la première expérimentation nous avons utilisé un perceptron multicouche avec la topologie [8 :6 :1] qui a comme entrée les huit paramètres de la base PIMA (Figure III.7) : avec un pas d'apprentissage=0.5 et Maximum d'itération=200.

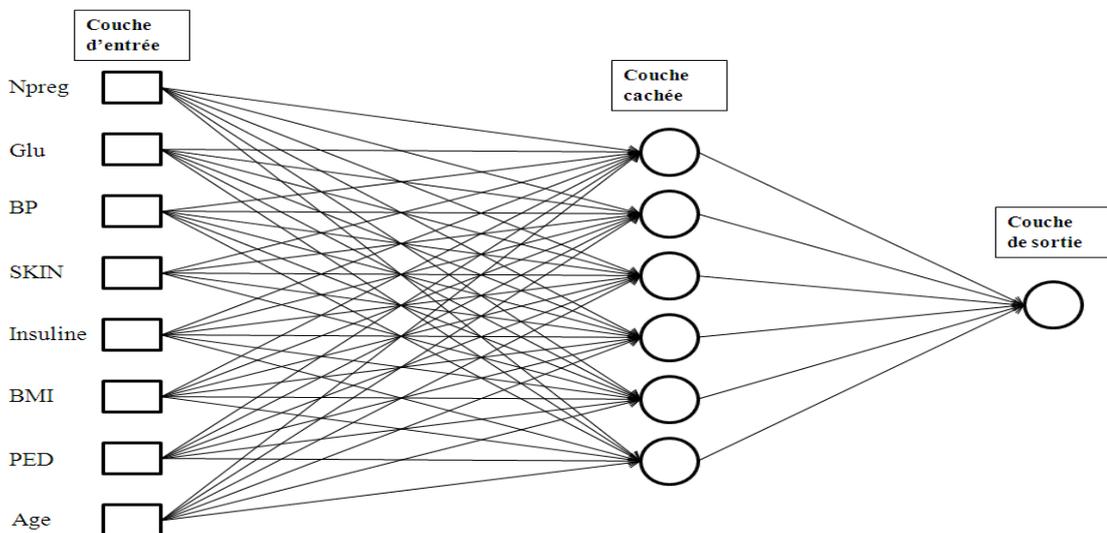


FIGURE III.7 – Architecture utilisée dans l'expérimentation

L'Erreur d'apprentissage trouvée =0.25, Erreur de Validation trouvée =0.28. (Table III.4) représente les résultats de test :

	Tested _positive	Tested _négative	Somme
Tested _positive	VP=30	FN=53	83
Tested _négative	FP=6	VN=167	173
Somme	36	220	256

TABLE III.4 – Matrice de confusion -1-

$$\text{La Sensibilité} = \frac{30}{30+53} = 0.3614 = 36.14\%$$

$$\text{La Spécificité} = \frac{167}{167+6} = 0.9653 = 96.53\%$$

$$\text{Le Taux de classification} = \frac{30+167}{30+167+6+53} = 0.7695 = 76.95\%$$

Les résultats de cette expérimentation montrent que la spécificité du système est très élevée ce qui veut dire que le système a fait un bon apprentissage pour les données négatives. Donc lorsqu'un patient est non diabétique notre modèle le détecte avec beaucoup de succès.

Par contre la sensibilité du système est très faible ce qui veut dire que le système a fait une mauvaise reconnaissance des données positives. Donc beaucoup de patients diabétiques ont été reconnu comme non diabétiques. Ce qui peut générer un risque majeur pour la santé du patient.

Avec ces performances, nous pouvons dire que le modèle a donné un taux de classification moyen et une bonne spécificité. Par contre il a donné une faible sensibilité. Ce qui reste un inconvénient à étudier.

Expérimentation 2

Dans le but d'améliorer la performance de l'expérimentation précédente, nous avons appliqué la méthode OCD à la même architecture de l'expérimentation 1 [8 :6 :1] (Figure III.7) avec un pas d'apprentissage=0.5 et un maximum d'itération=200.

L'Erreur d'apprentissage trouvée=0.23, L'Erreur de Validation=0.25. La nouvelle architecture obtenue est [3 :6 :1] (Figure III.8). (Table III.5) représente les résultats de test :

	Tested <u>positive</u>	Tested <u>négative</u>	Somme
Tested <u>positive</u>	VP=51	FN=32	83
Tested <u>négative</u>	FP=10	VN=163	173
Somme	61	195	256

TABLE III.5 – Matrice de confusion -2-

La Sensibilité= $\frac{51}{51+32} = 0.6144 = 61.44\%$

La Spécificité= $\frac{163}{163+10} = 0.9421 = 94.21\%$

Le Taux de classification= $\frac{51+163}{51+163+10+32} = 0.8359 = 83.59\%$

Cette méthode a sélectionné les variables (Glu :Glucose, BMI :la masse, PED :L'hérédité) comme des variables qui ont le pouvoir discriminant entre les deux classes. En revenant à la nature de la maladie du diabète nous pouvons confirmer que cette méthode a effectivement trouvé les variables les plus pertinentes : par ce que le changement de la concentration du glucose est le paramètre le plus utilisé pour le diagnostic de cette maladie et si nous revenons aux causes du diabète, nous remarquons que la majorité des gents diabétiques souffrent du problème de sur-poids et que le facteur génétique est responsable de la plupart des cas des défaillances du pancréas qui est l'anomalie principale d'un développement d'un diabète. Ces résultats montrent que le modèle a amélioré sa capacité de détection des positifs avec presque 50% de succès ce qui a généré une grande amélioration du taux de reconnaissance. Nous notons aussi une légère diminution de sa spécificité. (Table III.6) résume les résultats obtenus par les deux expérimentations :

Méthode	Nombre de variables	Nombre de connexions	Erreur	Sensibilité	Spécificité	Taux De classification
PMC	8	54	0.2305	36.14%	96.53%	76.95%
PMC+OCD	3	24	0.1641	61.44%	94.21%	83.59%

TABLE III.6 – Tableau des performances

En comparant les résultats des deux expérimentations, nous remarquons que la méthode proposée a diminuée le nombre de variables d'entrée de plus de 60% et que

le nombre de connexions s'est diminuées de plus de 55%, ce qui a donné une bonne optimisation de l'architecture du modèle et une forte amélioration des performances du classifieur (Figure III.8).

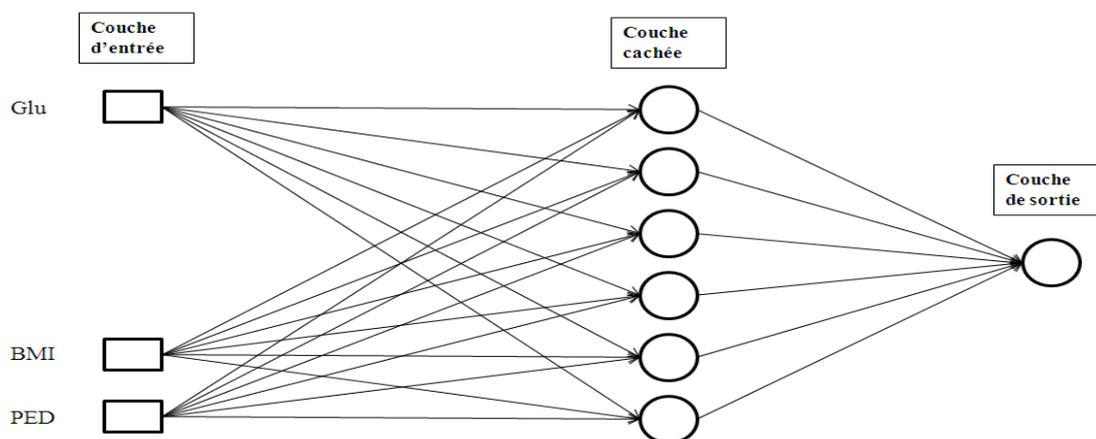


FIGURE III.8 – Architecture trouvée après l'application de la méthode OCD

5 Comparaison avec des travaux de la littérature

Afin de situer la performance de l'approche proposée nous avons réalisé une étude comparative entre les résultats obtenus et celles des travaux déjà réalisés dans ce domaine (étudiés dans l'état de l'art) avec la base PIMA. (Table III.7) résume la comparaison avec les autres travaux :

Nom de La Méthode	Type de la Méthode	Nombre de variables	Taux De Classification
FCBF+ SVM [56]	Filter	4	77.99%
GR+RBF [57]	Filter	5	86.46%
GR+PMC [57]	Filter	5	78.21%
GA+SVM [59]	Wrapper	4	81.50%
CAFS+PMC [58]	Wrapper	6	76.18%
TS1+PMC [61]	Wrapper	4	79.55%
OCD+PMC	Embedded	3	83.59%

TABLE III.7 – Tableau de comparaison avec des travaux de la littérature

Après plusieurs recherches sur cette problématique nous pouvons confirmer que la méthode proposée est la première méthode de type Embedded appliquée sur la

base de données PIMA et que le Nombre de variables trouvées par cette méthode est le plus petit nombre de variables sélectionnées jusqu'à présent. Nous remarquons aussi que le taux de classification est le meilleur parmi les méthodes qui utilisent les PMC et meme parmi les autres méthodes sauf dans le travail de M.Jayaram et al. [57] ils ont trouvé un taux meilleur mais en utilisant les RBF comme classifieur.

6 Conclusion

Nous avons présenté une méthode de sélection de variables qui a confirmé sa performance dans le test effectué et qui a donné une amélioration intéressante de l'erreur du modèle, de sensibilité et de taux de classification tout en optimisant son architecture.

Après la comparaison des résultats obtenus avec des travaux dans la littérature, nous avons remarqué que les résultats trouvés sont comparables ou des fois meilleurs par rapport aux autres résultats.

Pour la validation de ces résultats nous avons consulté des médecins diabétologues qui nous ont confirmé que les trois variables sélectionnées sont effectivement les plus pertinentes pour le diagnostic du diabète.

Conclusion générale

Afin de créer une application performante utilisée pour la reconnaissance du diabète, nous avons implémenté une méthode de sélection de variables qui a comme but d'éliminer les variables qui n'ont pas le pouvoir discriminant entre les classes et laisser juste celles qui portent les informations nécessaires pour faire le bon apprentissage. Cette méthode a sélectionné les variables (Glu :Glucose, BMI :la masse, PED :L'hérédité) comme les variables les plus pertinentes pour effectuer une meilleure classification. Pour la validation de ces résultats nous avons contacté des médecins diabétologues qui ont confirmé que ces trois variables sont effectivement les plus pertinentes pour le diagnostic du diabète.

Les résultats obtenus après l'utilisation de cette méthode sont très prometteurs et sont bien situés parmi les travaux déjà réalisés dans ce domaine ce qui confirme la rigueur de la contribution proposée pour la résolution de notre problématique.

Dans les perspectives d'avenir, nous prévoyons d'assurer l'interprétabilité des résultats du modèle en intégrant la notion du flou avec le classifieur. Nous voulons aussi généraliser cette modeste application sur tous les types de maladies afin de l'intégrer dans l'avenir dans un système d'aide au diagnostic applicable dans un hôpital ou dans un cabinet médical.

Bibliographie

- [1] International Diabetes Federation, DIABETES ATLAS, [<http://www.idf.org>], 2011.
- [2] organisation mondiale de la santé, “Aide-mémoire n 312,” *Technical report*, Septembre 2011.
- [3] E. A. for the Study of Diabètes, “Easd,” *Lisbonne (Portugal)*, 12 au 16 septembre 2011.
- [4] organisation mondiale de la santé, “ncd report chapter1,” *Burden : mortality, morbidity and risk factors*, pp. 9–31, 2010.
- [5] W. H. Organization, “Ncd country profiles,” *Algeria*, 2011.
- [6] Uffe Iversen, UIVE, Novo Nordisk Affiliate sites, Types de diabète, [<http://www.novon-ordisk.fr>], octobre 2003.
- [7] OMS, “la prévention du diabète sucré,” *série de rapport technique 844*, vol. Genève, 1994.
- [8] A. de la santé publique du Canada, “Lignes directrices de pratique clinique 2008 de l’association canadienne du diabète pour la prévention et le traitement du diabète au canada,” *Can J Diabetes*, vol. 32, pp. S1–S225, 2008.
- [9] F. R. KAUFMAN, “Type 1 diabetes mellitus,” *Pediatr Rev*, vol. 24, no 9, pp. 291–300, 2003.
- [10] L. A. LEITER, A. BARR, A. BÉLANGER, and coll., “Diabetes screening in canada (diascan) study : Prevalence of undiagnosed diabetes and glucose intolerance in family physician offices,” *Diabetes Care*, vol. 24, no 6, pp. 1038–1043, 2001.
- [11] M. I. HARRIS, K. M. FLEGAL, C. C. COWIE, and coll., “Prevalence of diabetes, impaired fasting glucose, and impaired glucose tolerance in u.s. adults : The third national health and nutrition examination survey, 1988-1994,” *Diabetes Care*, vol. 21, No 4, pp. 518–524, 1998.
- [12] T. K. YOUNG and C. A. MUSTARD., “Undiagnosed diabetes : Does it matter?,” *JAMC*, vol. 164, No 1, pp. 24–28, 2001.
- [13] Uffe Iversen, UIVE, Novo Nordisk Affiliate sites, Causes du diabète, [<http://www.novon-ordisk.fr>], octobre 2003.
- [14] A. Martel and R. Bouleis, *SYSTÉME EXPERT*. Edition BFG, 2005.
- [15] F. BOSSAERT, *Approche connexionniste pour le diagnostique des systèmes complexes : application au réseau téléphonique*. Université paris nord Ville-taneuse, 2009.

- [16] W. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *Bulletin of Mathematical Biophysics*, vol. 05, pp. 115–133, 1943.
- [17] F. Rosenblatt, "The perceptron : a probabilistic model for information storage and organization in the brain," *Psychological Review*, vol. 06, pp. 386–408, 1958.
- [18] M. L. Minsky and S. A. Papert, *Perceptrons*. Cambridge, 1969.
- [19] B. Widrow and M. E. Hoff, "Adaptive switching circuits," *Western Electronic Show and Convention*, vol. 04, pp. 96–104, 1960.
- [20] P. J. Werbos, *The Roots of Backpropagation : from Ordered Derivatives to Neural Networks and Political Forecasting*. Wiley-Interscience, 1994.
- [21] G. E. H. e. R. J. W. D. E. Rumelhart, "Learning internal representations by error propagation," *Parallel Data Processing : Explorations in the Microstructure of Cognition*, vol. 01, pp. 318–362, 1986.
- [22] Y. L. Cun, "Une procédure d'apprentissage pour réseau à seuil asymétrique (a learning scheme for asymmetric threshold networks)," *Proceedings of Cognitive*, pp. 599–604, 1985.
- [23] D. J. L. I. V. Tetko and A. I. Luik, "Neural network studies. 1. comparison of overfitting and overtraining," *Journal of Chemical Information and Computer Sciences*, vol. 35, No 5, pp. 826–833, 1995.
- [24] Y. Rangoni, *Réseau de neurones dynamique perceptif - Application à la reconnaissance de structures logiques de documents*. PhD thesis, université Nancy 2 France, 203, 2011.
- [25] N. M. Gail A. Carpenter, "Artmap-ic and medical diagnosis : Instance counting and inconsistent cases," *Center for Adaptive Systems and Department of Cognitive and Neural Systems*, vol. 01, pp. 323–336, 1998.
- [26] T. Y. Kamer Kayaer, *Medical Diagnosis on Pima Indian Diabetes Using General Regression Neural Networks*. Yildiz Technical University , Department of Electronics and Comm. Eng, 2003.
- [27] A. G. Jonas Guzaitis, Antanas Verikas and M. Bacauskiene, "A framework for designing a fuzzy rule-based classifier," *Proceedings of the 1st International Conference on Algorithmic Decision Theory, Berlin, Heidelberg*, vol. 09, pp. 434–445, 2009.
- [28] A. K. N. Muhammad Waqar Aslam, *DETECTION OF DIABETES USING GENETIC PROGRAMMING*. 18th European Signal Processing Conference (EUSIPCO-2010), Aalborg, Denmark, 2010.
- [29] H. Liu and H. Motoda, *Feature Selection for Knowledge Discovery and Data Mining*. Kluwer Academic Publishers, 1998.
- [30] D. Cakmakov and Y. Bennani, *Feature Selection for Pattern Recognition*. Informa Press, 2002.
- [31] Y. Bennani, "Systèmes d'apprentissage connexionnistes : sélection de variables," *Revue d'Intelligence Artificielle. Hermes Science Publications, Paris, France*, vol. 15, pp. 3–4, 2001.
- [32] Y. Bennani, *Apprentissage Connexionniste*. Editions Hermès Science, 2006.

- [33] M. N. I. Guyon, S. Gunn and L. Zadeh., “Feature extraction, foundations and applications, editors,” *Series Studies in Fuzziness and Soft Computing, Physica-Verlag. Springer*, vol. to appear, 2006.
- [34] S. Guerif, *Réduction de dimension en Apprentissage Numérique Non Supervisé*. PhD thesis, Université Paris 13, France ,148, 2006.
- [35] M. E. C. M. B. e. M. S. J. Bi, K. P. Bennett, “Dimensionality reduction via sparse support vector machines,” *Journal of Machine Learning Research*, vol. 03, pp. 1229–1243, 2003.
- [36] I. R. et L. Personnaz, “Mlps (mono-layer polynomials and multi-layer perceptrons) for nonlinear modeling,” *Journal of Machine Learning Research*, vol. 03, pp. 1383–1398, 2003.
- [37] R. D. H. Stoppiglia, G. Dreyfus and Y. Oussar, “Ranking a random feature for variable and feature selection,” *Journal of Machine Learning Research*, vol. 03, pp. 1399–1414, 2003.
- [38] M. A. Hall and L. A. . Smith., “Feature subset selection : a correlation based filter approach,” *International Conference on Neural Information*, vol. 1, No 4, pp. 855–858, 1997.
- [39] L. Y. et H. Liu, “Feature selection for high-dimensional data : a fast correlation-based filter solution,” *International Conference on Machine Learning*, vol. 01, No 12, pp. 856–863, 2003.
- [40] F. P.-C. J. S.-S. Sancho Salcedo-Sanz, Gustavo Camps-Valls and C. Bousoño-Calzon, “Enhancing genetic feature selection through restricted search and walsh analysis,” *IEEE transactions on systems, man, and cybernetics : applications and reviews*, vol. 34, no 4, 2004.
- [41] R. Kohavi and G. John, “Wrappers for feature selection,” *Artificial Intelligence*, vol. 1-2, pp. 272–324, 1997.
- [42] G. K. et S. Kim, “Feature subset selection using genetic algorithms for handwritten digit recognition,” *International Workshop on Frontiers in Handwriting Recognition*, vol. 1, No 7, pp. 103–112, 2000.
- [43] R. S. F. B. L. E. S. de Oliveira, N. Benahmed and C. Y. Suen, “Feature subset selection using genetic algorithms for handwritten digit recognition,” *Brazilian Symposium on Computer Graphics and Image Processing*, vol. 01, no 14, pp. 362–369, 2001.
- [44] M. M. e. R. S. L. E. S. de Oliveira, “Feature selection for ensembles applied to handwriting recognition,” *International Journal on Document Analysis and Recognition*, vol. 8 ,No 4, pp. 262–279, 2006.
- [45] J. D. Y. Le Cun and S. Solla, “Optimal brain damage,” *Advances in Neural Information Processing Systems*, vol. 02, pp. 598–605, 1990.
- [46] V. N. Vapnik, “The nature of statistical learning theory,” *Springer Verlag*, vol. 01, 1995.
- [47] B. Hassibi and D. Stork, “Second order derivatives for networks pruning : Optimal brain surgeon,” *Advances in Neural Information Processing Systems*, vol. 05, pp. 164–171, 1993.

- [48] L. H. M.W. Pedersen and J. Larsen, "Pruning with generalization based weight saliencies : yobd, yobs," *Advances in Neural Information Processing Systems*, vol. 08, 1996.
- [49] R. N. V. Tresp and H. G. Zimmermann, "Early brain damage," *Advances in Neural Information Processing Systems*, pp. 669–675, 1996.
- [50] P. G. T. Cibas, F. Fogelman and S. Raudys, "Variable selection with optimal cell damage," *Proceeding of the ICANN'94*, vol. 01, pp. 727–730, 1994.
- [51] M. Yacoub and Y. Bennani, "Hvs : A heuristics for variables selection in multilayer neural network classifiers," *ANNIE'97*, vol. 07, pp. 527–532, 1997.
- [52] D. MacKay, "Bayesian methods for backpropagation networks," *Springer-Verlag, New York, USA*, 1994.
- [53] J. Moody, "Prediction risk and architecture selection for neural networks," *Statistics to Neural Networks - Theory and Pattern Recognition Application*, 1994.
- [54] P. Leray and P. Gallinari, "De l'utilisation d'obd pour la sélection de variables dans les perceptrons multicouches," *Systèmes d'apprentissage connexionnistes : sélection de variables*, vol. 15, p. 373.
- [55] J. Proriot, "Selection of variables for neural network analysis comparisons of several methods with high energy physics data," *Nuclear Instruments and Methods in Physics Research*, vol. A 361, pp. 581–585, 1995.
- [56] R. N. Sarojini Balakrishnan, "Feature selection using fcbf in type ii diabetes databases," *International Conference on IT to Celebrate S. Charmonman's 72nd Birthday, Thailand*, pp. 50–58, 2009.
- [57] M. Asha Gowda Karegowda, A. S. Manjunath, "Comparative study of attribute selection using gain ratio and correlation based feature selection," *International Journal of Information Technology and Knowledge Management*, vol. 02, No 2, pp. 271–277, 2010.
- [58] K. M. Md. Monirul Kabir, Md. Monirul Islam, "A new wrapper feature selection approach using neural network," *Neurocomputing*, vol. 73, pp. 3273–3283, 2010.
- [59] C.-J. W. Cheng-Lung Huang, "A ga-based feature selection and parameters optimization for support vector machines," *Expert Systems with Applications*, vol. 31, pp. 231–240, 2006.
- [60] J. Yang and V. Honavar, "Feature subset selection using a genetic algorithm," *Artificial Intelligence Research Group Department of Computer Science 226 Atanasoff Hall Iowa State University Ames*, vol. IA 50011, p. 1997.
- [61] J. N. S. H. Yong Wanga, Lin Li, "Feature selection using tabu search with long-term memories and probabilistic neural networks," *Pattern Recognition Letters*, vol. 30, pp. 661–670, 2009.
- [62] P. G. T. Cibas, F. Fogelman-Soulié and S. Raudys, "Variable selection with neural networks," *Neurocomputing*, vol. 12, pp. 223–248, 1996.
- [63] OMG, "Unified modeling language specification," vol. Version1.5, 2003.
- [64] "Pima," A. Frank and A. Asuncion, 2010, Pima Indians Diabetes Data Set, UCI Machine Learning Repository, University of California, Irvine, School of Information and Computer Sciences, [<http://archive.ics.uci.edu/ml>].

- [65] J. H. S. e. T. M. J. E. R. Kandel, *Principles of Neural Science*. McGraw-Hill Medical, 2000.
- [66] L. Lamport, “Latex : A document preparation system,” *Addison-Wesley Professional*, vol. 2, p. 288, 1994.
- [67] S. R. e. F. M. Michel Goossens, “The latex graphics companion : Illustrating documents with tex and postscript,” *Addison-Wesley Professional*, p. 608, 1997.

Annexe A : Perceptron multicouche

Le neurone biologique est une cellule nerveuse qui comprend un corps cellulaire (noyau et protoplasme), un prolongement axial cylindrique (axone) et des prolongements secondaires (dendrites). Les neurones reçoivent et émettent des signaux excitateurs ou inhibiteurs. Ces signaux sont transmis le long des axones puis atteignent les dendrites d'un autre neurone par l'intermédiaire de neurotransmetteurs [65]. Il existe près de dix mille types différents de neurones, le cerveau humain en comporterait environ cent milliards et chaque neurone est relié à dix mille autres en moyenne. Devant ces chiffres, il est plus facile de comprendre pourquoi le cerveau humain est robuste et tolérant aux fautes, flexible et facilement adaptable. Il s'accommode d'informations incomplètes, incertaines, ou bruitées et il est capable d'apprentissage. Toutes ces caractéristiques font de lui un modèle très séduisant, c'est pourquoi, de nombreux systèmes informatiques de reconnaissance, actuels ou anciens, s'en inspirent.

Les neurones formels que nous utilisons tentent de reproduire les comportements des neurones biologiques. Ils s'en approchent dans la plupart des cas assez bien lorsqu'ils sont utilisés séparément. Le neurone informatique élémentaire peut être vu comme un automate qui reçoit des impulsions de ses autres neurones voisins, qui se charge à son tour de modifier cette information et de la renvoyer à ses voisins. La transmission de l'information se fait biologiquement par l'intermédiaire d'axones et de synapses qui sont simplement modélisés dans le modèle informatique par des paramètres scalaires (ou poids synaptiques) servant à pondérer les informations.

Le précurseur de la majorité des travaux sur les réseaux de neurones est le Perceptron de Rosenblatt [17]. Souvent à la base des réseaux statiques, le neurone formel

en tant qu'abstraction du neurone physiologique a été proposé par [16].

1 Rétro propagation du gradient de l'erreur

Il faut minimiser $E(x) = \sum_{p=1}^P E_p(w)$ avec $E_p(w) = \frac{1}{2} \sum_{q=1}^{N_L} (O_{L,q}(x_p) - d_q(x_p))^2$. Si l'on applique la descente de gradient à la variable w , il apparait pour chaque poids $w_{l,j,i}$.

$$w_{l,j,i} \leftarrow w_{l,j,i} - \mu \frac{\partial E(w)}{\partial w_{l,j,i}} \quad (\text{A.1})$$

$$w_{l,j,i} - \mu \sum_{p=1}^P \frac{\partial E_p(w)}{\partial w_{l,j,i}} \quad (\text{A.2})$$

$$w_{l,j,i} - \mu \sum_{p=1}^P \frac{\partial E_p(w)}{\partial o_{l,j}} \frac{\partial o_{l,j}}{\partial w_{l,j,i}} \quad (\text{A.3})$$

$$w_{l,j,i} - \mu \sum_{p=1}^P \frac{\partial E_p(w)}{\partial o_{l,j}} \frac{\partial}{\partial w_{l,j,i}} f\left(\sum_{m=0}^{N_{l-1}} w_{l,j,m} o_{l-1,m}\right) \quad (\text{A.4})$$

$$w_{l,j,i} - \mu \sum_{p=1}^P \frac{\partial E_p(w)}{\partial o_{l,j}} f'\left(\sum_{m=0}^{N_{l-1}} w_{l,j,m} o_{l-1,m}\right) \frac{\partial}{\partial w_{l,j,i}} \left(\sum_{m=0}^{N_{l-1}} w_{l,j,m} o_{l-1,m}\right) \quad (\text{A.5})$$

comme $\forall m \neq i, \frac{\partial}{\partial w_{l,j,i}} \left(\sum_{m=0}^{N_{l-1}} w_{l,j,m} o_{l-1,m}\right) = 0$, il en résulte :

$$w_{l,j,i} \leftarrow w_{l,j,i} - \mu \sum_{p=1}^P \frac{\partial E_p(w)}{\partial o_{l,j}} f'\left(\sum_{m=0}^{N_{l-1}} w_{l,j,m} o_{l-1,m}\right) o_{l-1,i} \quad (\text{A.6})$$

Si f est la sigmoïde, on a la propriété suivante :

$$\frac{\partial f(x)}{\partial x} = \frac{\partial}{\partial x} (1 + e^{-x})^{-1} = -(-e^{-x})(1 + e^{-x})^{-2} \quad (\text{A.7})$$

$$= f(x)e^{-x}(1 + e^{-x})^{-1} = f(x)(1 + e^{-x} - 1)(1 + e^{-x})^{-1} \quad (\text{A.8})$$

$$= f(x)(1 - f(x)) \quad (\text{A.9})$$

D'où :

$$w_{l,j,i} \leftarrow w_{l,i,j} - \mu \sum_{p=1}^P \frac{\partial E_p(w)}{\partial o_{l,j}} o_{l,j} (1 - o_{l,j}) o_{l-1,i} \quad (\text{A.10})$$

Le terme $\frac{\partial E_p(w)}{\partial o_{l,j}}$ peut être exprimé en fonction des neurones des couches suivantes :

$$\frac{\partial E_p(w)}{\partial o_{l,j}} = \sum_{m=1}^{N_{l+1}} \frac{\partial E_p(w)}{\partial o_{l+1,m}} \frac{\partial o_{l+1,m}}{\partial o_{l,j}} \quad (\text{A.11})$$

$$= \sum_{m=1}^{N_{l+1}} \frac{\partial E_p(w)}{\partial o_{l+1,m}} \frac{\partial}{\partial o_{l,j}} f\left(\sum_{q=0}^{N_l} w_{l+1,m,q} o_{l,q}\right) \quad (\text{A.12})$$

$$= \sum_{m=1}^{N_{l+1}} \frac{\partial E_p(w)}{\partial o_{l+1,m}} o_{l+1,m} (1 - o_{l+1,m}) w_{l+1,m,j} \quad (\text{A.13})$$

Le terme $\frac{\partial E_p(w)}{\partial o_{l+1,m}}$ peut lui aussi être calculé de la même manière et continue ainsi jusqu'à parvenir à la couche de sortie où, cette fois-ci le résultat est immédiat :

$$\frac{\partial E_p(w)}{\partial o_{L,j}} = o_{L,j}(x_p) - d_j(x_p) \quad (\text{A.14})$$

Grâce a cette erreur de sortie connue, on peut remonter les liens et corriger chaque poids jusqu'à la couche d'entrée.

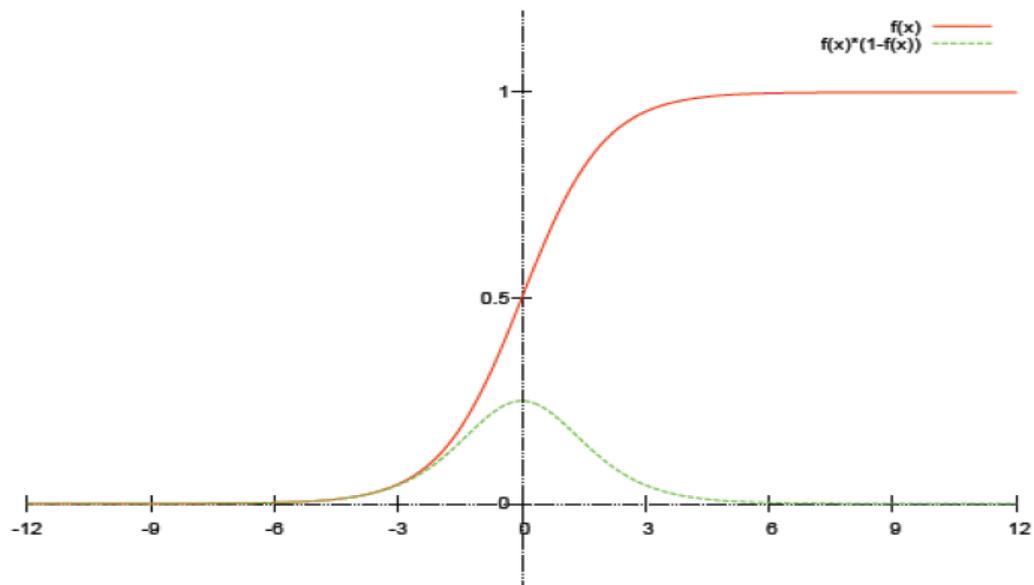


FIGURE A.1 – La sigmoïde et sa dérivée

2 Propriétés mathématiques

Les propriétés suivantes donnent un aperçu formel de la puissance des PMC. Les trois premières concernent uniquement un Perceptron seul, les suivantes sont relatives aux PMC [24] :

- un Perceptron seul, linéaire à seuil à \mathbf{n} entrées, divise l'espace des entrées \mathbb{R}^n en deux sous espaces délimités par un hyperplan. Réciproquement, tout ensemble linéairement séparable peut être discriminé par un Perceptron.
- toute fonction booléenne linéairement séparable sur \mathbf{n} variables peut être implantée par un Perceptron dont les poids synaptiques entiers w_i sont tels que $w_i \leq (n + 1)^{n + \frac{1}{2}}$.
- il existe des fonctions booléennes linéairement séparables sur \mathbf{n} variables qui requièrent des poids entiers supérieurs à $2^{n + \frac{1}{2}}$.

- toute fonction booléenne peut être calculée par un PMC linéaire à seuil comprenant une seule couche cachée.
- deux couches cachées suffisent à représenter des frontières de décision convexes.
- deux couches cachées suffisent à former une approximation arbitrairement proche de n'importe quelle surface de décision ou fonction continue non linéaire.
- toutes les fonctions continues bornées sont représentables, avec une précision arbitraire, par un réseau à une seule couche cachée.
- la plupart des fonctions numériques peuvent être approchées avec une précision arbitraire par des réseaux à une seule couche cachée.
- il est possible d'implémenter toute bijection avec deux couches cachées si elles ont suffisamment d'éléments.
- certains problèmes demandant un nombre exponentiel de neurones avec deux couches cachées et n'en demandent qu'un nombre polynomial si trois couches cachées sont utilisées.
- le nombre d'exemples nécessaires à l'apprentissage d'un PMC est au pire de l'ordre de $\mathbf{n \log(n)}$ si la fonction à approcher est booléenne de $\{0, 1\}^n \rightarrow \{0, 1\}$.
- si les exemples sont linéairement séparables, alors l'algorithme du Perceptron trouve la solution en un nombre fini d'itérations.

3 problème de sur-apprentissage

Dans l'exemple suivant (Figure A.2) la fonction linéaire et polynomiale approchent correctement les données présentées dans le plan. La fonction polynôme passe par la majorité de points et elle a moins d'excursions aux extrémités : elle a fait un sur-apprentissage.

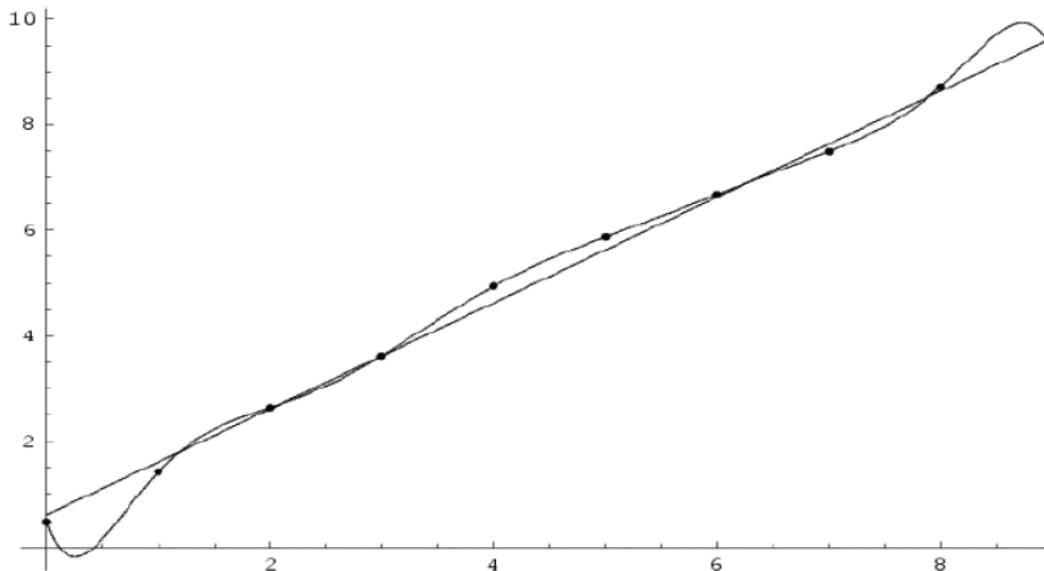


FIGURE A.2 – Exemple de sur-apprentissage

Annexe B : Outils

1 Manuel d'utilisation

L'interface de l'application est divisée en trois parties :

1.1 La base de données

Cette partie est consacrée à l'importation et l'affichage des données et des caractéristiques de la base de données. Voir (Figure B.1).

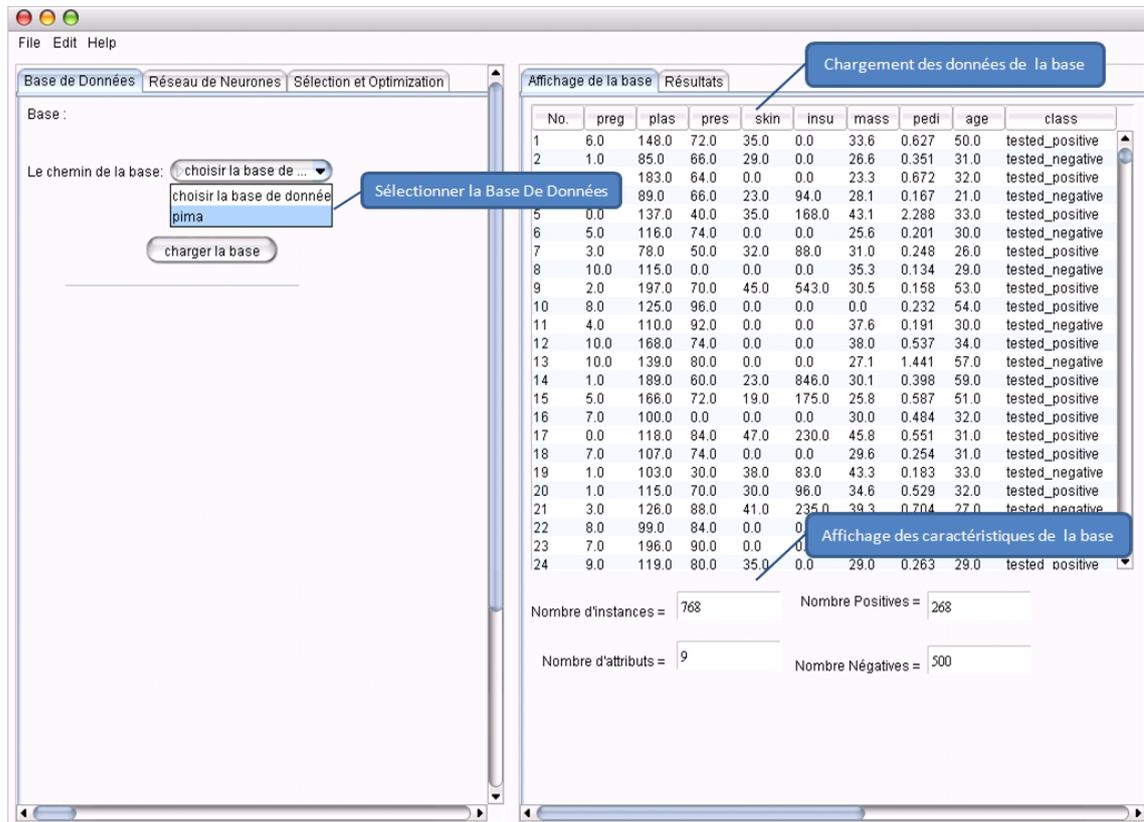


FIGURE B.1 – Partie base de données

1.2 Le réseau de neurones

Cette partie est consacrée à la création et le test d'un modèle initial du réseau de neurones. Voir (Figure B.2).

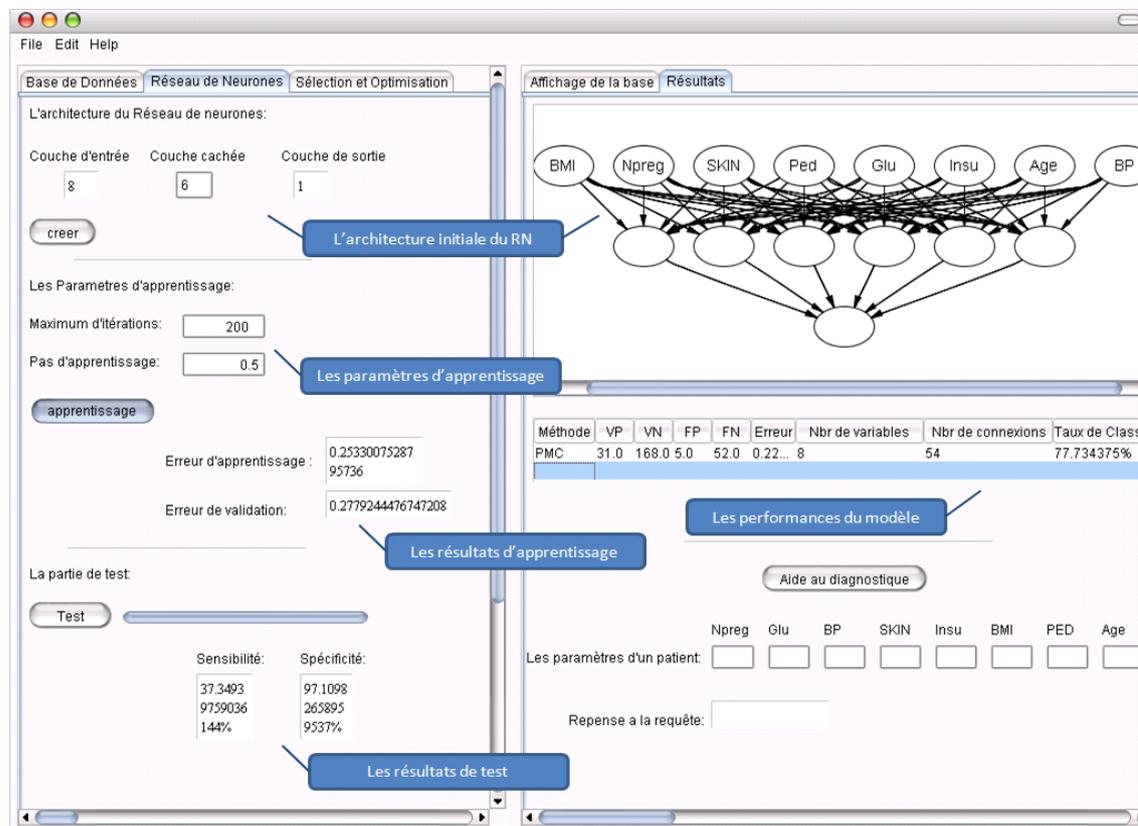


FIGURE B.2 – Partie réseau de neurones

1.3 Sélection et optimisation

Cette partie est consacrée à la sélection de variables et l'optimisation du modèle créé dans la partie précédente. Voir (Figure B.3).

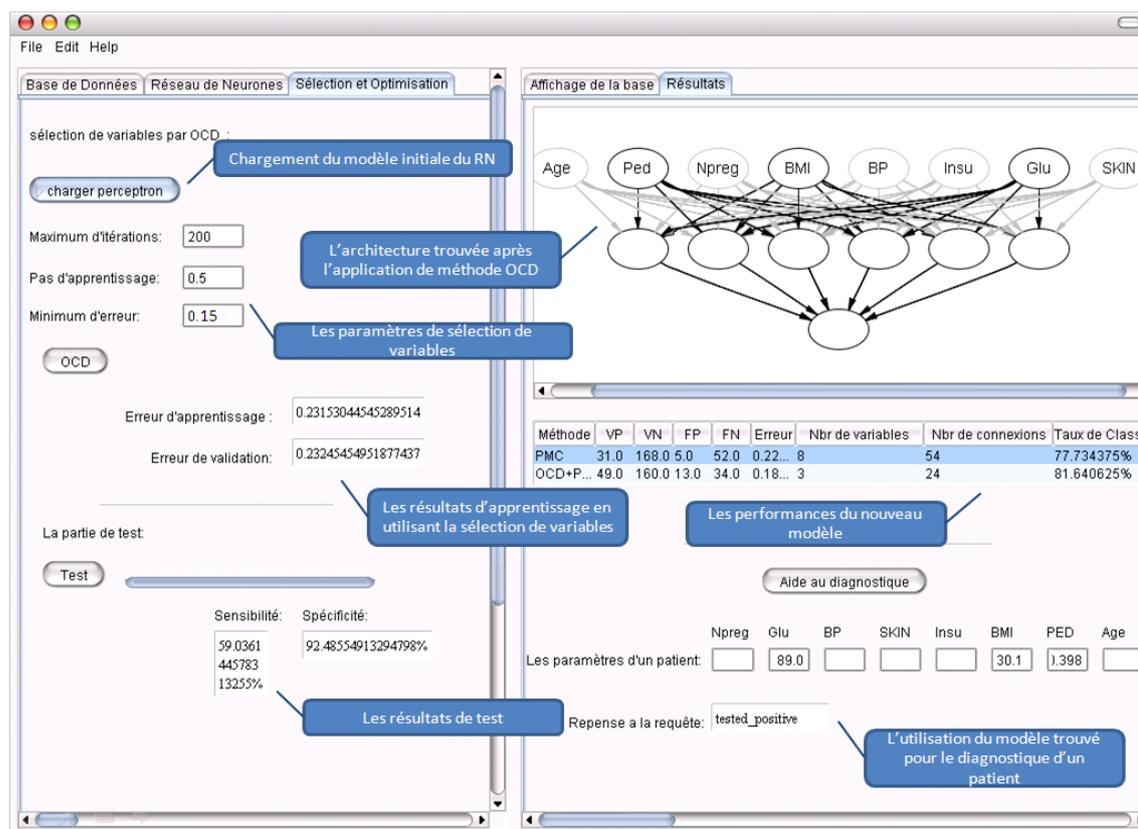


FIGURE B.3 – Partie sélection et optimisation

2 Outils utilisés dans ce mémoire

2.1 LaTeX

LaTeX est un langage et un système de composition de documents créé par Leslie Lamport en 1983 [66], [67]. Plus exactement, il s'agit d'une collection de macro-commandes destinées à faciliter l'utilisation du « processeur de texte » TeX de Donald Knuth. Depuis 1993, il est maintenu par le LATEX3 Project team. La première version utilisée largement, appelée LaTeX2.09, est sortie en 1984. Une révision majeure, appelée LaTeX2E est sortie en 1991.

LaTeX est distribué sous une licence libre de type LPPL et il est souvent utilisé par des personnes devant produire un contenu parfois complexe (équations) ayant une mise en forme standard, c'est-à-dire ne nécessitant pas de mise en forme particulière. Pour cette raison, il est particulièrement utilisé par des étudiants, éditeurs,

professeurs de mathématiques et de physique, scientifiques, principalement en informatique, en ingénierie, en mathématiques et en physique.



2.2 MiKTeX

MiKTeX est une distribution TeX libre pour Windows visant à fournir un environnement TeX/LaTeX complet et prêt à utiliser. Les points forts de MiKTeX sont sa facilité d'installation et son outil de téléchargement et de mises à jour de packages, en particulier sa fonction d'installation à la volée des fichiers manquants pendant la compilation d'un document.



2.3 JabRef

JabRef est un logiciel de gestion bibliographique libre qui utilise BibTeX comme format natif. JabRef fournit une interface conviviale pour éditer des fichiers BibTeX, pour importer des données à partir de bases de données scientifiques en ligne et pour gérer et effectuer des recherches sur des fichiers BibTeX.

- Développeur : The JabRef team
- Dernière version : 2.8 (14 mai 2012)
- Environnements : Multiplateforme
- Langues : Multilingue
- Type : Logiciel de gestion bibliographique
- Licence : GNU GPL
- Site web : jabref.sourceforge.net



2.4 NetBeans

NetBeans est un environnement de développement intégré (EDI) placé en open source par Sun en juin 2000 sous licence CDDL et GPLv2 (Common Development and Distribution License). En plus de Java, NetBeans permet également de supporter différents autres langages, comme Python, C, C++, JavaScript, XML, Ruby, PHP et HTML. Il comprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projets multi-langage, refactoring, éditeur graphique d'interfaces et de pages Web).

Conçu en Java, NetBeans est disponible sous Windows, Linux, Solaris (sur x86 et SPARC), Mac OS X ou sous une version indépendante des systèmes d'exploitation (requérant une machine virtuelle Java). Un environnement Java Development Kit JDK est requis pour les développements en Java.

NetBeans constitue par ailleurs une plate forme qui permet le développement d'applications spécifiques (bibliothèque Swing (Java)). L'IDE NetBeans s'appuie sur cette plate forme.

- Développeur : Oracle
- Première version : 1996, sous le nom de Xelfi
- Dernière version : 7.1.2 (26 avril 2012)
- Environnements : Plateforme Java
- Langues : Multilingue
- Type : IDE pour Java, PhP, C/C++, Fortran, JavaScript, Python, Ruby
- Licence : CDDL/ GPL
- Site web : netbeans.org



2.5 JDK

Le Java Development Kit (JDK) désigne un ensemble de bibliothèques logicielles de base du langage de programmation Java, ainsi que l'environnement dans lequel le code Java est compilé pour être transformé en bytecode afin que la machine virtuelle Java (JVM) puisse l'interpréter.

- Développeur : Sun Microsystems (possédé par Oracle Corporation)
- Dernière version : 7 (4/26 avril 2012)
- Environnements : Multiplateforme
- Langues : Multilingue
- Type : Logiciel de développement Kit
- Licence : Permis de Sun (plupart également sous le GPL)
- Site web : jdk6.dev.java.net



Résumé:

La médecine est une discipline scientifique mais aussi une discipline d'action qui requiert souvent une prise de décision. Des techniques telles que les réseaux de neurones artificiels sont les plus couramment utilisées pour l'aide au diagnostic médicale et sont développées par de grand nombre d'équipes de recherche grâce à leurs simplicités du raisonnement et de leurs performances d'apprentissage inspirés du raisonnement humain. Mais avec la présence d'attributs redondants ou d'attributs fortement bruités dans les bases de données, la performance du système peut se déclinier, ce qui nécessite l'utilisation des techniques de sélection de variables qui ont pour objectif de sélectionner ou d'extraire un sous-ensemble optimal de caractéristiques pertinentes pour faire un meilleur apprentissage et assurer une bonne performance du système de classification. Pour effectuer cette tâche nous avons proposé une méthode appelée OCD (Optimal Cell Damage). Elle fait sa sélection en parallèle avec le processus d'apprentissage du réseau de neurones artificiel en se basant sur la notion de pertinence des entrées du réseau. Nous avons testé cette méthode sur la base de données du diabète PIMA (UCI : machine learning). Les résultats obtenus sont très prometteurs.

Abstract:

Medicine is a scientific discipline but also a discipline of action often requires a decision. Techniques such as artificial neural networks, are the most commonly used in the medical diagnostic help and are developed by many research teams, through their simplicity of reasoning and their learning performance inspired by human reasoning. But with the presence of redundant attributes or attributes in highly noisy databases, system performance can be affected, requiring the use of variable selection techniques that aim to select or retrieve an optimal subset of relevant features to better learning and achieve successful performance of the classification system. To perform this task we proposed a method called OCD (Optimal Cell Damage), it made her selection in parallel with the learning process of artificial neural network based on the notion of relevance of network inputs. We tested this method on the database of diabetes PIMA (UCI machine learning), the results are very promising.

ملخص:

الطب هو انضباط علمي وأيضاً انضباط عملي وغالباً ما يتطلب اتخاذ قرارات صعبة. تقنيات مثل الشبكات العصبية الاصطناعية، هي الأكثر استخداماً في مجال المساعدة التقنية في التشخيص الطبي، ويتم تطويرها من قبل فرق بحثية عديدة، وذلك لسبب بساطتها في التفكير وأدائها في التعلم المستوحى من المنطق البشري. ولكن مع وجود خصائص زائدة عن الحاجة أو تحمل معلومات غير مفيدة في قواعد البيانات، أداء نظام المساعدة يضعف ويصبح بدون فائدة، مما يتطلب استخدام تقنيات اختيار الخصائص التي تهدف إلى تحديد أو استرداد المجموعة المثلى من المزايا أو الخصائص ذات الصلة المباشرة بالإشكالية المدروسة. وذلك لتحسين التعلم وتحقيق الأداء الناجح لنظام التصنيف. لأداء هذه المهمة اقترحنا طريقة تسمى أم خ (الأمثل لمحو الخلايا)، حيث تقوم باختيار أحسن الخصائص بالتوازي مع عملية التعلم في الشبكة العصبية وذلك باستعمال نظرية أهمية مداخل الشبكية. قمنا باختبار هذه الطريقة على قاعدة البيانات الخاصة بمرض السكري "بيما"، وتحصلنا على نتائج واعدة جداً.