Table des matières

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Table des figures

Liste des tableaux

Glossaire

ACO optimisation par colonies de fourmis

ABC colonies d'abeilles artificielles

AG algorithme génétique

App apprentissage

FO fonction objective

Fdd fouille de données

FCBF Fast Correlation Based Feature Selection

IA intelligence artificielle

OP-VAR Optimisation des VARiables

PSO optimisation par essaims de particules

RS recuit simulé

RO recherche opérationnelle

SV selection des variables

SVM machines à vecteurs de support

SE Sensibilité

SP Spécificité

TC Taux de Classification

UCI Irvine Machine Learning Repository

Rapport-Gratuit.com

Introduction Générale

Dans leurs activités, les ingénieurs et les décideurs sont confrontés à des problèmes de complexité grandissante, en vue de maximiser les bénéfices, minimiser les pertes…etc. Ces problèmes surgissent dans des domaines très divers, comme la conception et l'implémentation des systèmes d'aide à la décision, les réseaux informatiques, le traitement d'images, en robotique, en électronique…

Dans le domaine de diagnostic médical, la résolution des problèmes se base sur le traitement de données extraites à partir des informations acquises sur des patients, et structurées sous forme de vecteurs de caractéristiques. La qualité du système de diagnostic dépend énormément du bon choix du contenu de ces vecteurs. Cependant, dans de nombreux cas, la résolution pratique du problème devient presque impossible à cause de la dimensionnalité importante de ces vecteurs. Par conséquent, il est souvent utile, et parfois nécessaire, de réduire la taille de ces caractéristiques à une taille plus optimale, en utilisant des méthodes de résolution dédiées. Dans plusieurs cas, la résolution des problèmes complexes avec des descripteurs de grande taille pourrait être gérée, en utilisant peu de caractéristiques extraites des données initiales.

Ce problème peut souvent s'exprimer comme un problème d'optimisation qui occupe à nos jours, une place grandissante dans le domaine de la fouille de données. Ce type de problèmes est formulé en sélectionnant une ou plusieurs variables et en définissant une fonction objectif ou fonction coût, que l'on cherche à minimiser ou à maximiser par rapport à tous les variables et les contraintes concernés. Ceci peut augmenter la précision de la prédiction du système ou réduire le temps de traitement des données. Classiquement, la sélection d'attributs est définie comme le fait de sélectionner un sous-ensemble de M attributs à partir d'un ensemble N, tel que $M<N$ et que la fonction critère choisie soit optimale sur le sous-ensemble de taille M choisi.

Le choix d'une méthode efficace capable de produire une solution optimale dans un temps de calcul raisonnable, est la principale difficulté à laquelle est confronté un décideur pour faire face à un problème d'optimisation. Les méthodes existantes pour la sélection d'attributs utilisent des connaissances dans divers domaines : les statistiques, l'apprentissage, les heuristiques et les métaheuristiques, ... Dans ce mémoire de master, nous nous sommes plus particulièrement intéressés à la conception des métaheuristiques pour résoudre le problème de sélection de variables (SV).

Pour ce type de problèmes d'optimisation qui demeurent hors de portée des méthodes exactes, les méthodes heuristiques constituent le moyen le plus efficace pour obtenir une solution de bonne qualité et de se rapprocher le plus possible de la solution optimale en un temps raisonnable. Même si ces heuristiques ne fournissent aucune garantie d'optimalité, elles offrent l'avantage de ne parcourir qu'une fraction de l'espace de recherche pour parvenir à une solution acceptable ce qui leur permet d'être exploitées avec succès sur une large gamme de problèmes pratiques et théoriques.

Cependant, les approches heuristiques partagent l'inconvénient d'être spécifiques aux problèmes pour lesquels elles étaient conçues, ce qui rend leur développement, une tâche difficile, nécessitant une grande connaissance du domaine des heuristiques et du problème étudié. Contrairement aux heuristiques dédiées, les métaheuristiques peuvent être appliquées à n'importe quel type de problèmes sans avoir un changement profond des algorithmes. Ces approches présentent actuellement des alternatives intéressantes pour la résolution des problèmes d'optimisation difficile, pour lesquels on ne connaît pas d'algorithmes classiques plus efficaces.

Les métaheuristiques permettent d'envisager une résolution approchée de nombreux problèmes d'optimisation différents, avec un minimum d'adaptation réalisée pour chaque problème. Parmi ces méthodes, on trouve les algorithmes génétiques. Ces derniers sont des algorithmes d'exploration fondés sur les mécanismes de la sélection naturelle et de la génétique. À chaque génération, un nouvel ensemble de créatures artificielles (codées sous forme de chaînes de caractères) est construit à partir des meilleurs éléments de la génération précédente.

Dans le cadre de ce mémoire de master, nous proposons un modèle de sélection d'attributs qui permette de trouver un sous ensemble (de taille réduite) nécessaire et suffisant pour une amélioration des performances et de la qualité du notre système de classification. Ce modèle nommé OP-VAR (Optimisation des VARiables), est basée sur les algorithmes génétiques (GA) , pour explorer l'espace des sous-ensembles des attributs, où chaque gène détermine la présence ou l'absence d'un attribut dans le sous ensemble sélectionné. En utilisant l'approche enveloppante, les individus sont évalués en entraînant les classifieurs sur le sous-ensemble d'attributs, indiqué par le chromosome, et nous optons pour le taux d'erreur comme valeur pour sa fonction fitness. Le classifieur utilisé est les séparateurs à vaste marges (SVM) de type linéaire. Le modèle OP-VAR proposée est appliquée sur un échantillon des bases de données médicales issues de l'UCI (Irvine Machine Learning Repository). Les résultats expérimentaux obtenus dans ce cadre, montrent l'efficacité de notre système développée.

L'organisation de ce mémoire de master suit une progression ordonnée. Ce document se décompose en trois chapitres :

- Le chapitre 1 décrit le cadre général de l'optimisation et des métaheuristiques. Dans un premier temps, nous définissons les problèmes d'optimisation difficile, l'heuristique et le concept de métaheuristique. Ensuite, nous introduisant une présentation des principales métaheuristiques connues dans la littérature, en se basant sur les algorithmes génétiques (AG).

- Le chapitre 2décrit la notion de sélection d'attributs, en particulier : ses objectifs, les différentes approches de sélection ainsi que les différentes méthodes proposées dans la littérature et qui sont basés sur les métaheuristiques.

- Le troisième chapitre est consacré à l'implémentation de notre modèle Op-Var (Optimisation des Variables), pour la résolution du problème de sélection de variables (SV). Op-Var est basé sur une approche Wrapper qui fait l'hybridation entre l'algorithme génétique (AG) et les SVM. Nous exposant par la suite, une analyse détaillée de nos expérimentations obtenues sur les trois jeux de données médicales utilisées.

Enfin, nous terminons par une conclusion générale et quelques perspectives.

Chapitre 1 :

Généralités sur les métaheuristiques

1 Introduction

Un grand nombre de problèmes d'aide à la décision, comme les problèmes d'apprentissage et de sélection de variables, peuvent être décrits sous forme de problèmes d'optimisation. Ces derniers occupent actuellement une place grandissante dans la communauté scientifique. Ils peuvent être combinatoires (discrets) ou à variables continues, avec un seul ou plusieurs objectifs (optimisation mono ou multi-objectif), statiques ou dynamiques, avec ou sans contraintes. Cette liste n'est pas exhaustive et un problème peut être, par exemple, à la fois continu et dynamique. [1]

Ce chapitre décrit tout d'abord le cadre général de l'optimisation et des métaheuristiques dans lequel nous nous plaçons dans ce travail.

2 Définition d'un problème d'optimisation

L'optimisation se définit comme la sélection du meilleur élément (appelé optimum) parmi un ensemble d'éléments autorisés (appelé espace de recherche), en fonction d'un critère de comparaison. Un problème d'optimisation ***P*** peut être décrit comme un triple ***(S, C, F)*** ou :
S : est l'espace de recherche défini sur un ensemble de variables de décisions.
C : est l'ensemble de contraintes d'égalités ou inégalités qui doivent être satisfaites pour qu'une solution soit faisable.
F : est la fonction objective (fonction de cout) qui assigne une valeur du coût positive à chaque élément (ou solution) de *S*.

Plusieurs problèmes d'optimisation dépendent du choix de la meilleure configuration de l'ensemble de variables pour atteindre ses objectifs, ils peuvent se découper en deux catégories : les problèmes où les solutions sont codées avec des valeurs réelles (Problème Continu) et les problèmes où les solutions sont codées avec des variables discrètes (problème Combinatoire) que nous nous intéressons dans notre thèse. . Il existe aussi des problèmes *mixtes* qui utilisent à la fois des variables continues et discrètes.

Il existe autre classification des problèmes d'optimisation, dont nous citons les plus connues (figure1.1)

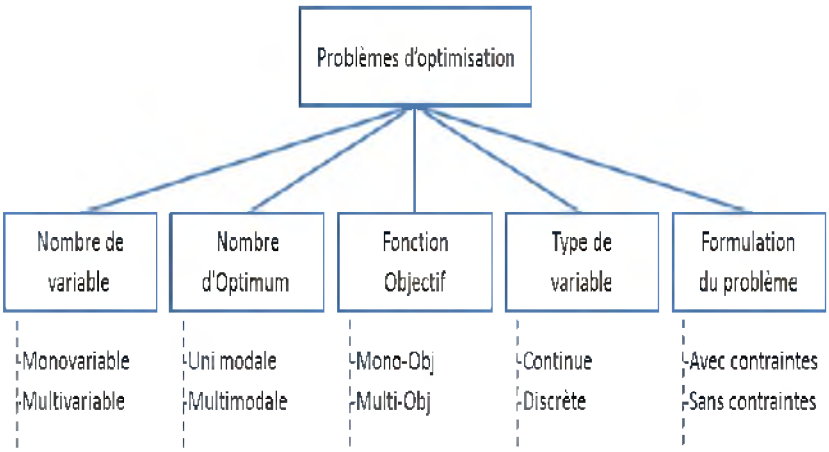

Figure1.1 : Classification des problèmes d'optimisation **[21]**

Un problème d'optimisation combinatoire est généralement caractérisé par un ensemble fini de solutions admissibles W et une fonction objectif F , associant une valeur à chaque solution admissible. La résolution du problème consiste à déterminer la ou les solution(s) de W minimisant ou maximisant F, c'est-à-dire l'optimum global [9].

Cependant il peut exister des solutions intermédiaires, qui sont également des optimums, mais uniquement pour un sous-espace restreint de l'espace de recherche : on parle alors d'optimums locaux. Cette notion est illustrée sur la figure 1.2.

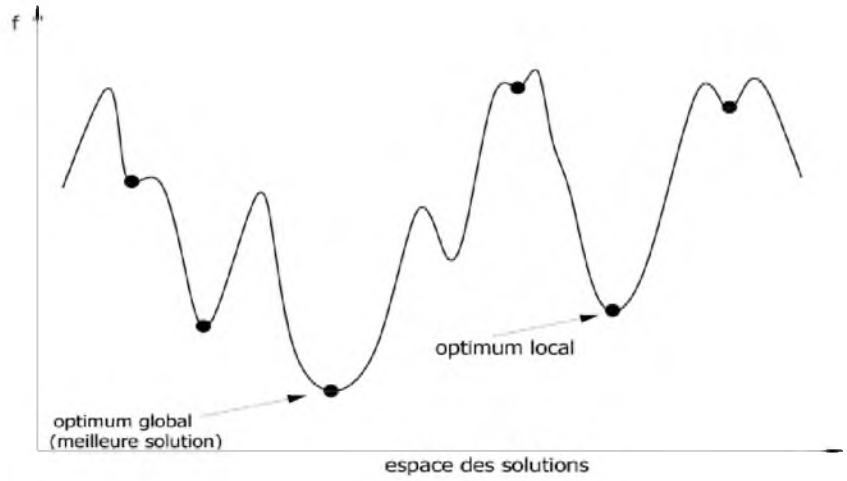

Figure 1.2 : optimum local et optimum global

3 Classification des méthodes d'optimisation

L'optimisation combinatoire occupe une place très importante en recherche opérationnelle, en mathématiques discrètes et en informatique. Bien que les problèmes d'optimisation combinatoire soient souvent faciles à définir, ils sont généralement difficiles à résoudre.
Etant donnée l'importance de ces problèmes, de nombreuses méthodes de résolution ont été développées en recherche opérationnelle (RO) et en intelligence artificielle (IA).
Ces méthodes peuvent être classées sommairement en deux grandes catégories illustrées dans la figure 1.3 :

- Les méthodes exactes (complètes) qui garantissent la complétude de la résolution.
- Les méthodes approchées (incomplètes) qui perdent la complétude pour gagner en efficacité. [4]

3.1 Les méthodes exactes :

Le terme de méthodes exactes regroupe l'ensemble des méthodes permettant d'obtenir la solution optimale d'un problème, en un temps "raisonnable". Elles s'opposent aux heuristiques, car les méthodes exactes permettent d'obtenir théoriquement la solution optimale et non une solution approchée. [2]

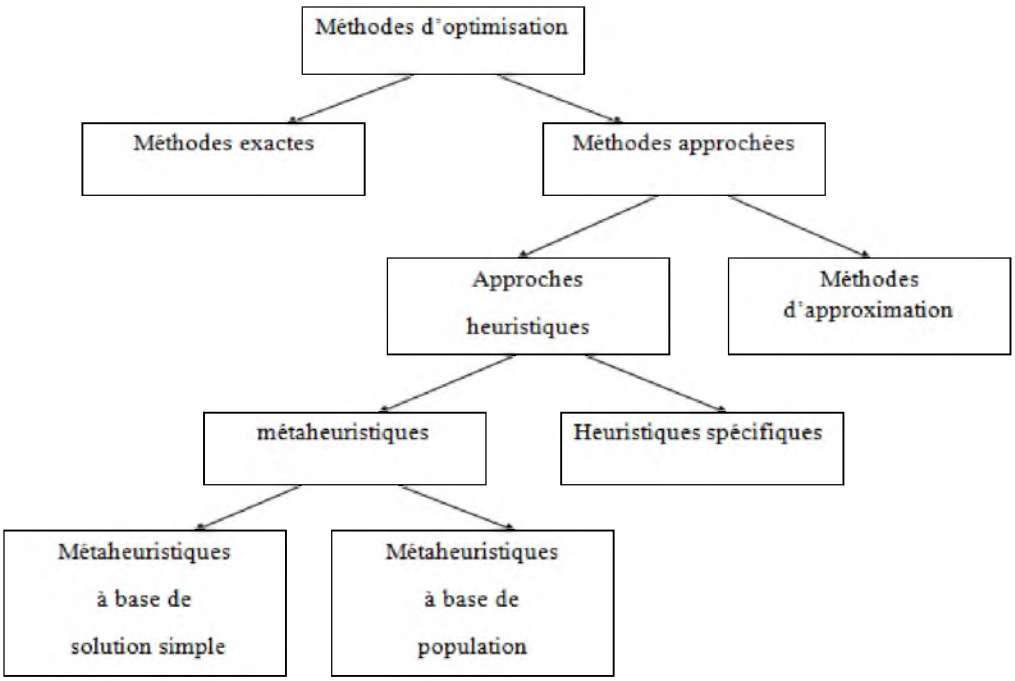

Figure 1.3 classification des méthodes d'optimisation

3.2 Les méthodes approchées :

Les méthodes exactes permettent de trouver une ou plusieurs solutions dont l'optimalité est garantie. Dans certaines situations, on peut obtenir de solutions de bonnes qualités sans garantie d'optimalité mais avec un temps de calcul réduit (les méthodes approchées) [3]

Les méthodes approchées constituent une alternative très intéressante pour traiter les problèmes d'optimisation de grande taille si l'optimalité n'est pas primordiale. En effet, ces méthodes sont utilisées depuis longtemps par de nombreux praticiens [4]
La figure 1.4 suivante décrit un schéma représentatif des méthodes approchées :

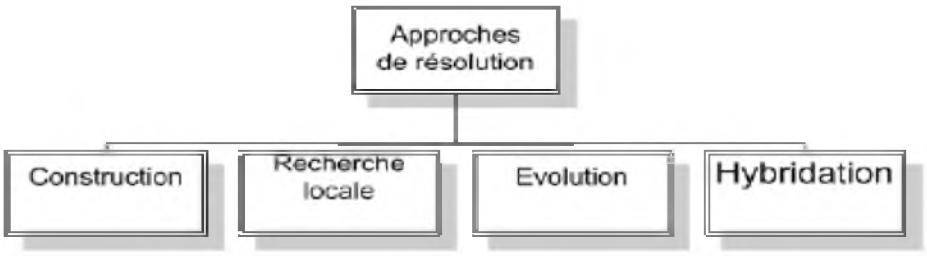

Figure1.4 paradigmes des méthodes approchées

- Construction : déterminer la qualité des composantes d'une solution avec une fonction heuristique, et à chaque étape retenir la 'meilleure' composante et l'ajouter
- Recherche locale (ou voisinage) : le principe d'une recherche locale est de partir d'une solution sinon approchée du moins potentiellement bonne et d'essayer de l'améliorer itérativement. Pour améliorer une solution on ne fait que de légers changements (on parle de changement local, où de solution voisine)
- Évolution : une population de solutions évolue par des opérateurs génétiques (sélection, croisement, mutation).

- Hybridation : mélange des approches précédentes

Depuis une dizaine d'années, des progrès importants ont été réalisés avec l'apparition d'une nouvelle génération de méthodes approchées puissantes et générales, souvent appelées **les métaheuristiques.**

Dans la section suivant, nous présentons en détail cette classe des méthodes d'optimisation (Métaheuristiques).

4 Les métaheuristiques

4.1 Introduction :

Au milieu des années 1970 des nouvelles méthodes sont apparues qui supervisent l'évolution de solutions fournies par des heuristiques. Ces méthodes assurent un compromis entre la diversification (possibilité de déterminer que la recherche se concentre sur de mauvaises zones de l'espace de recherche) et l'intensification (recherche des meilleures solutions dans la région de l'espace de recherche en cours d'analyse). Ces algorithmes sont appelés "métaheuristiques".

Au début des années 1980, ces méthodes ont été appliquées dans le but est de résoudre au mieux les problèmes dits d'optimisation difficile et de trouver des solutions dont la qualité est au-delà de ce qu'il aurait été possible de réaliser avec une simple heuristique. Elles partent de principes plus génériques que les heuristiques et sont susceptibles de s'appliquer à un cadre plus large de problèmes, tandis qu'une heuristique est particulière pour un problème donné [7].

4.2 Définitions des métaheuristiques:

Le terme **métaheuristique** vient des mots grecs méta (au delà) 'dans un niveau supérieur' et heuriskein et qui signifie (trouver).Une heuristique est une méthode, conçue pour un problème d'optimisation donné, qui produit une solution non nécessairement optimale lorsqu'on lui fournit une instance de ce problème [4].

D'après Blum et Roli [23] il n'existe actuellement aucune définition communément acceptée pour les métaheuristiques.

Définition(1) : Une métaheuristique est formellement défini comme une génération itérative processus qui guide une heuristique subordonnée en combinant intelligemment différents concepts pour l' exploration et l'exploitation de l' espace de recherche , les stratégies d'apprentissage sont utilisés pour structurer l'information afin de trouver des solutions de manière efficace quasi- optimales [8].

Définition (2) : Les métaheuristiques sont des méthodes génériques de résolution approchée de problèmes d'optimisation. Elles permettent d'envisager une résolution approchée de nombreux problèmes d'optimisation différents, avec un minimum d'adaptation réalisée pour chaque problème. Parmi ces méthodes on trouve les algorithmes génétiques [10].

Définition (3) : Les métaheuristiques sont définies comme un processus itératif maitre qui guide et modifie des heuristiques subordonnées dans le but d'efficacement produire des solutions de haute qualité. Une métaheuristique peut manipuler une ou plusieurs solution complètes(ou incomplètes) a chaque itération. Les heuristiques subordonnées peuvent être des procédures de haut(ou bas) niveau, ou de simple recherche locale ou juste des méthodes de construction [11].

4.3 Caractéristiques des métaheuristiques:

Les principales caractéristiques attachées aux métaheuristiques sont les suivantes:

- La plupart des métaheuristiques sont des algorithmes incertains utilisant des processus aléatoires comme moyens de récolter de l'information et de parcourir l'espace de recherche afin de trouver une solution satisfaisante.
- Les métaheuristiques ne donnent aucune garantie d'optimalité.
- Les métaheuristiques peuvent utiliser l'expérience acquise durant le processus de recherche pour guider les étapes suivantes du processus.
- En plus de cette base stochastique, les métaheuristiques sont généralement itératives, c'est à dire qu'un même schéma de recherche est appliqué plusieurs fois au cours de l'optimisation, et directes, c'est à dire qu'elles n'utilisent pas l'information du gradient de la fonction objectif.
- Elles sont inspirées par analogie avec la réalité : avec la physique (le recuit simulé), avec la biologie (les algorithmes génétiques) ou avec l'éthologie (les colonies de fourmis)...
- Les métaheuristiques peuvent contenir des mécanismes qui permettent d'éviter d'être piégé dans des zones de l'espace de recherche.
- Elles partagent aussi les mêmes inconvénients tels que la difficulté de réglage de ces paramètres

5 Classification des Métaheuristiques

Bien que les métaheuristiques partagent plusieurs caractéristiques communes, il existe cependant des points différences entre ces approches. Les métaheuristiques peuvent être classées selon :

- Le principe de fonctionnement durant la recherche de la solution (i.e La manipulation d'une solution à la fois (recherche locale ou méthodes de trajectoire) ou un ensemble des solutions (à base de population)).
- Leur origine (Inspiration de la nature ou non).
- Utilisation de l'historique de la recherche (mémoire).

5.1. Recherche Locale (Méthodes de trajectoire) :

Les métaheuristiques de recherche locale ou les méthodes itératives à solution unique sont basées sur un algorithme de recherche de voisinage qui commence avec une solution initiale, puis l'améliore à pas en choisissant une nouvelle solution dans son voisinage, le voisinage d'une solution est défini en fonction du problème à résoudre : passent d'une solution s à une autre dans l'espace des solutions candidates (l'espace de recherche) qu'on note S, jusqu'à ce qu'une solution considérée comme optimale soit trouvée ou que le temps réparti soit dépassé.
Nous présenterons ici les méthodes les plus classiques et les plus utilisées qui sont : le recuit simulé, la recherche tabou et la méthode de descente (recherche locale).

La figure1.5 illustre bien le fonctionnement de l'algorithme général des méthodes de recherche locale.

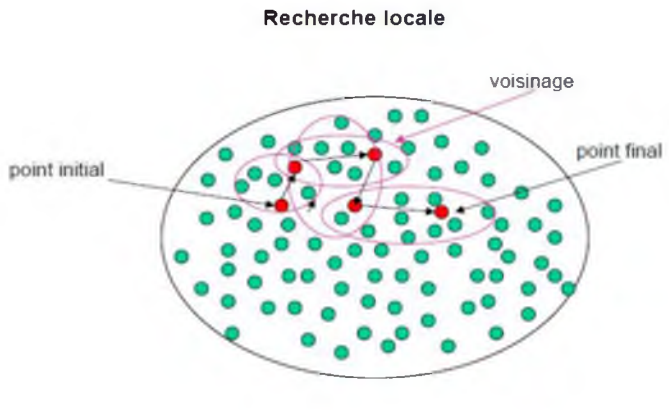

Figure 1.5 : Représentation de la procédure générale de la recherche locale.

Dans ce mémoire de master, nous nous s'intéressons à la méthode du recuit simulé, son fonctionnement et ces domaine d'applications qu'on va les citer par la suite.

5.1.1. Le Recuit simulé (RS):

a. Principes du R.S :

Les origines du recuit simulé remontent aux expériences réalisées par Metropolis et al. Dans les années 50 pour simuler l'évolution d'un tel processus de recuit physique [24]. Metropolis et al utilisent une méthode stochastique pour générer une suite d'états successifs du système en partant d'un état initial donné. Tout nouvel état est obtenu en faisant subir un déplacement (une pertur- bation) aléatoire à un atome quelconque.

En 1983, La méthode de recuit simulé a été conçue par Kirkpatrick et al. [15] qui étaient des spécialistes de physique. Depuis son apparition, elle a eu un impact majeur dans le domaine de la recherche heuristique à cause de sa simplicité et son efficacité dans la résolution de problèmes d'optimisation combinatoire et son extension pour résoudre d'autres types de problèmes [7].

La détermination numérique des configurations des systèmes physiques posait de redoutables problèmes d'optimisation. Cette méthode est issue d'une correspondance entre le phénomène physique de refroidissement lent d'un corps en fusion, qui le conduit à un état solide, de basse énergie. L'analogie entre un problème d'optimisation et un système physique est présentée dans le tableau suivant :

Problème d'optimisation	**Système physique**
Fonction objective	Energie libre
Paramètres du problème	Coordonnées des particules
Trouver une bonne configuration	Trouver les états à basse énergie

Tableau 1.1 : Analogie entre un problème d'optimisation et un système physique.

L'analogie historique s'inspire du recuit des métaux en métallurgie : un métal refroidi trop vite présente de nombreux défauts microscopiques, c'est l'équivalent d'un optimum local pour un problème d'optimisation combinatoire. Si on le refroidit lentement, les atomes se réarrangent, les défauts disparaissent, et le métal a alors une structure très ordonnée, équivalente à un optimum global [16].

b. Algorithme du RS :

Cette méthode d'optimisation est basée sur les travaux de Metropolis [26] qui permet de décrire le comportement d'un système dans l'équilibre thermodynamique à une certaine température. Dans l'algorithme de Metropolis, on part d'une configuration donnée, et on lui fait subir une modification aléatoire. Si cette modification fait diminuer la fonction objectif (ou énergie du système), elle est directement acceptée ; Sinon, elle n'est acceptée qu'avec une probabilité égale à ($\Delta E/T$) (avec E=énergie, et T=température), cette règle est appelée critère de Metropolis [24].

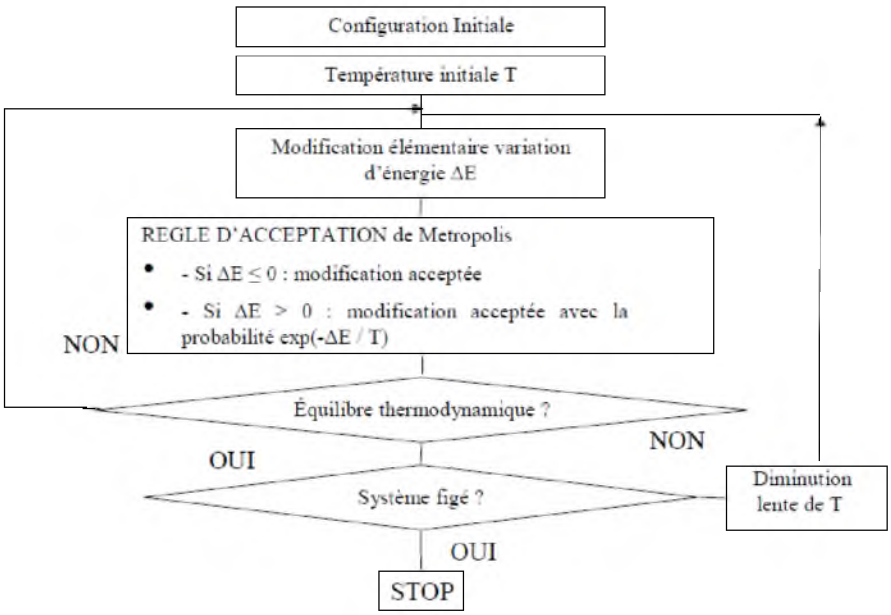

Figure1.6: l'organigramme du recuit simulé [25]

L'algorithme du RS part d'une solution donnée, et la modifie itérativement jusqu'au refroidissement du système. Les solutions trouvées peuvent améliorer le critère que l'on cherche à optimiser (la fonction objective à minimiser étant l'énergie E du matériel), on dit alors qu'on a fait baisser l'énergie du système, comme elles peuvent le dégrader. Si on accepte une solution qui améliore le critère, on tend ainsi à chercher l'optimum dans le voisinage de la solution de départ. Contrairement autres méthodes de recherche locale, le recuit simulé peut accepter des solutions dont la qualité est moins bonne en fonction de la dégradation de la solution considérée [17]. La température T est également introduite, qui est dans ce cas un paramètre de contrôle de l'algorithme.
L'organigramme du recuit simulé est présenté dans la figure 1.6

c. Domaines d'applications :

Depuis son apparition, la méthode du recuit simulé a prouvé son efficacité dans des différents domaines:

- la conception électronique : La conception des circuits intégrés (Kirkpatrick, et al. 1988) « problème de placement et de répartition ».
- Le traitement d'images : La segmentation d'images
- L'organisation des réseaux : Le routage des paquets dans les réseaux
- Le problème du voyageur de commerce.
- Et, le problème du sac à dos.

d. Avantages et inconvénients :

Avantage :

- Très simple, rapide et Facile à implémenter.
- Le principal avantage du RS est donc de pouvoir sortir d'un minimum local, en fonction d'une probabilité d'acceptation liée à une fonction exponentielle.
- Convergence vers un optimum global, la prédiction de la future à partir du présent ne nécessite pas la connaissance du passé : cette métaheuristique ne nécessite pas de mémoire (passé) afin de trouver les espaces de recherche locaux suivants (futur).
- Donne généralement de bonnes solutions par rapport aux algorithmes de recherche classiques.
- Peut être utilisé dans la plupart des problèmes d'optimisation.
- Elle a donnée d'excellents résultats pour nombres de problèmes, le plus souvent de grande taille.

Inconvénients :

- Très coûteuse en temps de calcul.
- non-convergence vers l'optimum peut se rencontrer assez vite.
- L'impossibilité de savoir si la solution trouvée est optimale.
- La difficulté de déterminer la température initiale :
  - Si elle est trop basse, la qualité de recherche sera mauvaise.
  - Si elle est trop haute, le temps de calcul sera élevé.
- Dégradation des performances pour les problèmes où il y a peu de minimas locaux (comparé avec les heuristiques classiques comme la descente du gradient par exemple).

5.1.2 Recherche tabou :

La recherche tabou a été formalisée par Glover en 1986. La principale particularité de cette méthode développée pour résoudre des problèmes combinatoires tient dans la mise en œuvre de mécanismes inspirés de la mémoire humaine. Contrairement au recuit simulé totalement dépourvu de mémoire, et donc incapable de tirer les leçons du passé, la méthode tabou n'a aucun caractère stochastique et utilise la notion de mémoire pour éviter de tomber dans un optimum local.

Le principe de base de cette méthode consiste à choisir la meilleure solution du voisinage de la solution courante, parfois il n'y a plus de meilleures solutions dans le voisinage, le meilleur voisin remplace la solution courante même si celui-ci est moins bon. Par conséquent, la recherche ne s'arrête pas au premier optimum local trouvé. Le danger est alors de revenir à des solutions déjà explorées. Pour s'échapper d'un optimum local, on mémorise les dernières solutions visitées dans une liste tabou(de longueur limitée l) . La liste tabou est donc une sorte de mémoire à court terme. Tout mouvement qui nous mène de la solution courante à une solution de la liste tabou est appelé mouvement tabou. La méthode s'arrête soit après un nombre fixé d'itérations, soit après un nombre fixé d'étapes n'ayant pas amélioré la solution [3][5][7][14][21].

Voila une structure algorithmique générale de la recherche Tabou. Les symboles ***s*** et ***s'*** représente la solution courante et prochaine respectivement. ***T*** dénote la liste Tabou. ***N*** est l'ensemble des voisins de la solution courante ***s***.

Algorithme Recherche Tabou

- Créer une solution initiale s ;
- Initialiser la liste Tabou T ;
  - **Tant que** (critère d'arrêt n'est pas satisfait) **faire**
    Déterminer l'ensemble des voisins N de la solution courant s ;
    Sélectionner la meilleure solution non-tabou s' de N.
    Basculer vers la solution s' (s ⟵ s').
    Mettre à jour la liste T.
    Mettre à jour la meilleure solution trouvée (si nécessaire);
  Fin
- **Retourner** la meilleure solution trouvée.

5.2 Métaheuristiques à base de population :

5 .2.1 Introduction :

Les métaheuristiques à base de population ont été introduites afin d'améliorer la diversité des solutions proposées par les algorithmes d'optimisation. Parmi les métaheuristiques à base de population on distingue les algorithmes évolutionnaires basés sur la théorie de l'évolution de Darwin [Darwin 1859], l'intelligence en essaim basée sur le comportement des espèces vivants en colonies comme les fourmis ou les abeilles, et les systèmes immunitaires artificiels qui miment les systèmes immunitaires biologiques [19].Dans ce mémoire de master, nous définissons l'algorithme génétique et l'approche d'optimisation à base d'essaims.

Les algorithmes évolutionnaires sont des techniques de recherche inspirée par l'évolution biologique des espèces apparues à la fin des années 50 [28]. L'évolution d'une espèce est marquée par une suite de transformations, permettant d'améliorer l'adaptation de l'espèce à son milieu. Cette adaptation est réalisée grâce à la sélection naturelle et aux mécanismes de la reproduction. Les méthodes dites évolutionnaires, ont connu un intérêt limité du fait de leur important coût d'exécution. Elles manipulent un ensemble des solutions simultanément. Dans les années 1960 à 1970, des que les premiers calculateurs de puissance plus crédible ont commencé à être accessibles, de nombreuses tentatives de modélisation de l'évolution ont été entreprise. Parmi ces tendances on trouve les algorithmes génétiques [14].

5.2.2 Algorithme génétique (AG) :

Les premiers travaux sur les algorithmes génétiques (AG) ont commencé dans les années cinquante lorsque plusieurs biologistes américains ont simulé des structures biologiques sur ordinateur. Puis dans les années 1960, John Holland étudie les systèmes évolutifs et, en 1975, il a introduit le premier modèle formel des algorithmes génétiques Il expliqua comment ajouter de l'intelligence dans un programme informatique avec les croisements (échangeant le matériel génétique) et la mutation (source de la diversité génétique) [18].

a. Principe de base de l'AG:

Les algorithmes génétiques, sont des algorithmes d'optimisation s'appuyant sur des techniques dérivées de la génétique et des mécanismes d'évolution de la nature : croisement, mutation, sélection. L'analogie entre la théorie de l'évolution consiste à

considérer les solutions appartenant à l'espace de recherche du problème à optimiser comme des chromosomes et des individus soumis à l'évolution.
Le vocabulaire utilisé est le même que celui de la théorie de l'évolution et de la génétique, on emploie le terme individu (solution potentielle),
Chaque individu est un codage ou une *représentation* (binaire, entiers, …) pour une *solution* candidate d'un problème donné (de même qu'un chromosome est formé d'une chaine de *gène*s) Dans chaque *chromosome*, les gènes sont les variables du problème et leurs valeurs possibles sont appelées *allèles* (alleles). (Voir figure 1.7). La fonction de codage associe à chaque *phénotype* (la solution du problème réel) une représentation de génotype.
Le *génotype* est utilisé au cours de l'étape de *reproduction* de l'algorithme alors que le phénotype est nécessaire pour l'*évaluation* du coût d'un individu.

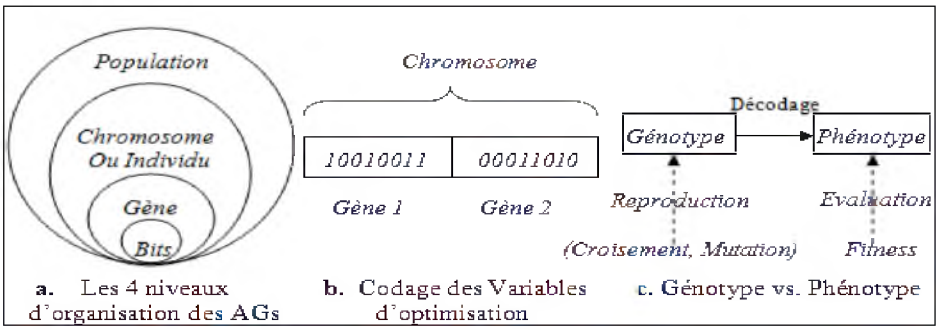

Figure 1.7 Les concepts principaux utilisés dans les algorithmes génétiques [22]

b. Processus de l'AG *:*

Un algorithme génétique recherche le ou les extrema d'une fonction définie sur un espace de données. Son mise en œuvre nécessite les opérations principales suivantes:

- ✓ **Le codage** :

La première étape est de définir et coder convenablement le problème. Cette phase détermine la structure de données qui sera utilisée pour coder le génotype des individus de la population. Le codage doit donc être adapté au problème traité. Plusieurs types de codages sont utilisés dans la littérature, les premiers résultats théoriques sur les algorithmes génétiques ont opté pour un codage binaire.

- ✓ **Génération de la population initiale :**

C'est-à-dire le choix des dispositifs de départ que nous allons faire évoluer. Ce choix de la population initiale d'individus conditionne fortement la rapidité de l'algorithme. Néanmoins, une initialisation aléatoire est plus simple à réaliser : les valeurs des gènes sont tirées au hasard selon une distribution uniforme.

- ✓ **Evaluation de la fonction d'adaptation (Fitness) :**

L'évaluation de la Fitness est généralement l'étape dans laquelle on mesure la performance de chaque individu. Pour pouvoir juger la qualité d'un individu et ainsi le comparer aux autres, il faut établir une mesure commune d'évaluation. Aucune règle n'existe pour définir cette fonction, son calcul peut ainsi être quelconque, que ce soit une simple équation ou une fonction affine. La manière la plus simple est de poser la fonction d'adaptation comme la formalisation du critère d'optimisation.

✓ **Phase de sélection** :

Permet d'identifier statistiquement les meilleurs individus d'une population et d'éliminer les mauvais, pendant le passage d'une génération à une autre, ce processus est basé sur la performance de l'individu. L'opérateur de sélection doit être conçu pour donner également une chance aux mauvais éléments, car ces éléments peuvent, par croisement ou mutation, engendrer une descendance pertinente par rapport au critère d'optimisation. Il existe différentes techniques de sélection :

- **Sélection uniforme :** On choisit au hasard n individus. Pas très efficace
- **Sélection par rang :** On choisit toujours les n meilleurs individus. Efficace, mais risque de provoquer une convergence trop rapide vers un optimum local
- **Sélection par tournois :** On choisit (uniformément ou non) des paires d'individus et on les fait "s'affronter" : le plus adapté sera choisi Chacun de ces types de sélection peut être enrichi d'un mécanisme élitiste : on garde toujours le meilleur individu (pour ne pas "régresser" dans notre recherche. . .)

✓ **Opérateur de croisement** :

L'opérateur de croisement favorise l'exploration de l'espace de recherche et enrichit la diversité de la population en manipulant la structure des chromosomes, le croisement fait avec deux parents et génère deux enfants, en espérant qu'un des deux enfants au moins héritera de bons gènes des deux parents et sera mieux adapté qu'eux (voir Figure1.9)

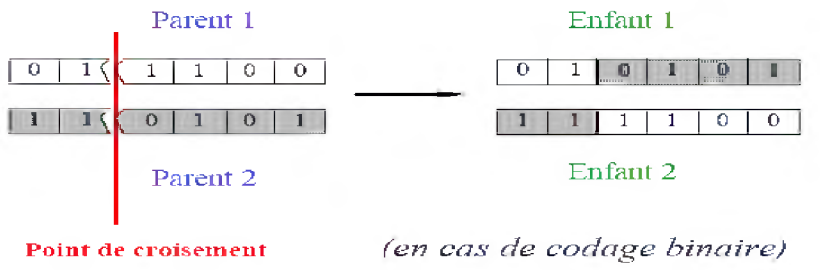

Figure 1.8 Représentation schématique d'un croisement dans le cas d'un codage binaire.

✓ **Opérateur de mutation** :

Cet opérateur altère un seul individu. L'opérateur de mutation est un processus où un changement mineur du code génétique appliqué à un individu pour introduire de la diversité et ainsi d'éviter de tomber dans des optimums locaux. Voir Figure 1.10

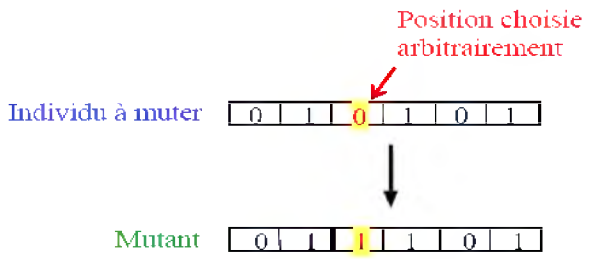

Figure 1.9 Représentation schématique de l'opérateur de mutation

✓ **Phase de remplacement :**

Elle consiste a choisir les membres de la nouvelle génération « seuls les individus adaptés sont supposés survivre »: on peut, par exemple, remplacer les plus mauvais

individus (au sens de la fonction objectif) de la population courante par les meilleurs individus produits (en nombre égal) [20] [21].

✓ **Critères d''arrêt :**

Une question délicate est : quand considère-t-on qu'on a terminé ? Critères d'arrêt possible :

- On a trouvé une solution assez bonne
- Budget (temps ou calcul) épuisé
- Nombre de générations maximal atteint
- Pas d'amélioration dans la qualité des meilleures solutions sur ***N*** générations
- Observation humaine ensemble

c. Limites de l'AG :

- La principale difficulté des algorithmes génétiques est qu'ils sont très sensibles au choix des paramètres : Taille de la population, Taux de renouvellement, Taux de mutation
- On n'est jamais sûr d'avoir trouvé la solution optimale !
- Pour éviter cela, on peut essayer de toujours garder une certaine variété de population.

5.2.3 Les essaims particulaires :

L'optimisation par essaim particulaire (OEP) est une méthode proposée en 1995 aux Etats Unis sous le nom de Particle Swarm Optimization (PSO) par Eberhart et Kennedy [57]. Cet algorithme s'inspire à l'origine du monde du vivant. Il s'appuie notamment sur les observations effectuées lors de simulations informatiques des comportements sociaux des animaux devant atteindre un objectif donné dans un espace de recherche commun et évoluant en essaim tels que le déplacement collectif d'un banc de poissons ou d'un groupe d'oiseaux.

a. Principe de fonctionnement :

L'OEP repose sur un ensemble de solutions. Chaque solution est appelée particule et le groupe devient un essaim. Chaque particule est modélisée par sa position dans l'espace de recherche et par sa vitesse de déplacement est dotée d'une mémoire pour conserver sa meilleure position visitée.

L'OEP commence par une répartition au hasard de l'essaim dans l'espace de recherche, ainsi, chaque particule ayant également une vitesse aléatoire. Ensuite, chaque particule ajuste sa position et sa vitesse pour le prochain déplacement en utilisant son expérience et les connaissances que ses voisines possèdent du milieu, donc à chaque pas du temps elle est capable d' :

- Evaluer la qualité de sa position et de garder en mémoire sa meilleure performance (sa meilleure position atteinte et sa qualité).
- Obtenir la meilleure performance trouvée par tout le groupe auquel elle appartient.

La capacité de communiquer avec les particules constituant son entourage. À partir de ces informations, la particule va suivre une tendance faite, d'une part, de sa volonté à retourner vers sa solution optimale, et d'autre part, de son mimétisme par rapport aux solutions trouvées dans son voisinage. À partir d'optimums locaux et empiriques, l'ensemble des particules va, normalement, converger vers la solution optimale globale.

b. Algorithme de base :

L'idée de base de cet algorithme est qu'un groupe d'individus dont chacun est peu intelligent puisse avoir une organisation globale complexe permettant aux particules de converger vers un minimum local grâce à des règles de déplacement simples, malgré qu'aucun de ces oiseaux ne connaît où se trouve exactement la nourriture.

Contrairement aux autres algorithmes d'origines discrets, le premier algorithme de cette métaheuristique a été conçu pour fonctionner pour des espaces en variables continues. À chaque itération de l'algorithme, chaque particule possède une valeur de fitness (distance par rapport à la solution) qui peut être évaluée afin de mettre à jour les meilleures positions connues et se déplace dans l'espace de recherche selon sa vélocité qui représente la vitesse utilisée pour guider le mouvement de la particule et sa position dans l'espace de recherche.

Soit x_i un vecteur de position de la i ème particule de l'essaim v_i un vecteur de vitesse de cette particule. D la dimension de ce problème. x_i et v_i sont des vecteurs à D éléments dont la jème est notée respectivement x_{ij} et v_{ij} Soit p_i un vecteur de dimension D qui correspond à la meilleure position atteinte par la particule i et p_{ij} sa coordonnée sur la dimension j. on note g le vecteur de dimension D qui correspond à la meilleure position connue de l'essaim.

Algorithme de l'Optimisation par Essaims Particulaires :

Initialiser aléatoirement un essaim.
Evaluer la fonction « objectif » pour chaque essaim
$x_i \leftarrow p_i$; i=1,…N (taille de l'essaim)
Calcul g
Tant que (le critère d'arrêt n'est pas vérifié)
  Mise-à-jour x_i et v_i selon les équations (01) et (02)
  Evaluer la fonction « objectif »
  mise-à-jour p_i
  mise-à-jour g
Fin tq

A l'itération t+1 le déplacement des particules est calculé à l'aide des équations :

- $x_{ij}(t+1)=x_{ij}(t)+v_{ij}(t+1)$……(2)
- $v_{ij}(t+1)=w.v_{ii}(t) +c1.r1.(p_{ij}-x_{ij}(t))+ c2.r2.(g_j-x_{ij}(t))$ ……(1)

W : coefficient de l'inertie ; c1, c2 : coefficients d'accélération ; r1, r2 ∈ [0,1] (générés aléatoirement).

La méthode d'optimisation par essaim particulaires partage beaucoup de similarité avec l'Algorithme Génétique dans le sens où les propriétés d'un individu sont influencées par les caractéristiques des autres.

6 Conclusion

Les métaheuristiques ont révélé leur grande efficacité pour fournir des solutions approchées de bonne qualité pour un grand nombre de problèmes d'optimisation classiques et d'applications réelles de grande taille. C'est pourquoi l'étude de ces méthodes est actuellement en plein de développement.

Dans ce chapitre nous avons présenté le cadre générale de ces approches. Dans un premier temps nous avons défini les concepts de base liés aux problèmes d'optimisation, l'approche heuristique et les métaheuristiques. En particulier, nous avons décrit les métaheuristiques à base de solution unique (dites de trajectoire) et ceux à base population de solutions.

Nous constatons que les métaheuristiques présentent une classe de méthodes approchées adaptables à une grande variété de problèmes d'optimisation combinatoire. Aussi, les métaheuristiques présentent un bon outil qui peut être appliquées pour la résolution des problèmes complexes telle que la sélection de variables .Dans le chapitre suivant, nous présentons un état de l'art de ce problème.

Chapitre 2 :

Problème De Sélection De Variables

1 Introduction

La sélection de variables est un sujet de recherche très actif et une étape de prétraitement qui joue un rôle important dans le domaine d'apprentissage artificiel, de la fouille de données (FDD), du traitement d'images, et de l'analyse de données en bio-informatique. Elle consiste à choisir un sous-ensemble de variables pertinents parmi un ensemble d'attributs de grande taille, en éliminant les variables redondantes, non pertinentes ou bruitées et qui ont peu ou pas d'influence sur l'information que l'on souhaite prédire.

Dans ce chapitre, nous présentons d'abord le problème de la sélection d'attributs pour situer le travail et l'intérêt de notre thèse de master. Dans un premier temps, nous abordons les difficultés de pertinence, non-pertinence et redondance autour desquelles s'articule la sélection d'attributs. Ensuite, nous présentons les différentes approches de sélection de variables. Enfin nous abordons quelques travaux existants pour la résolution de ce problème complexe.

2 Sélection de variables

2.1 Problématique :

La sélection de variables est une problématique complexe et d'une importance cruciale dans différents domaines cités précédemment. Dans le cadre de ce mémoire de master, nous traitons uniquement la sélection d'attributs réalisée pour la classification supervisée. Dans ce contexte, les principales motivations de la sélection d'attributs sont les suivantes [33] :

- Utiliser un sous-ensemble plus petit permet d'améliorer la classification si l'on élimine les attributs qui sont source de bruit.
- Des petits sous-ensembles d'attributs permettent une meilleure généralisation des données en évitant le sur-apprentissage.
- Une fois que les meilleurs attributs sont identifiés, les temps d'apprentissage et d'exécution sont réduits et en conséquence l'apprentissage est moins coûteux.

En présence de centaines, voire de milliers de variables, il y a beaucoup de chances pour que des variables soient corrélées et expriment des informations similaires, on dira alors qu'ils sont redondantes. D'un autre côté, les variables qui fournissent le plus d'information pour la classification seront dites pertinentes.

L'objectif de la sélection est donc de trouver dans une base volumineuse un sous-ensemble optimal de variables qui ait les propriétés suivantes : il doit être composé de variables pertinentes et il doit chercher à éviter les variables redondantes. De plus cet ensemble doit permettre de satisfaire au mieux l'objectif fixé c'est-à-dire la précision de l'apprentissage, sa rapidité ou bien encore la compréhensibilité du classifieur proposé [34] [36]. Nous croyons que les notions de dimension, pertinence et redondance jouent un rôle fondamental dans la sélection d'attributs. Afin de mettre en évidence l'aspect de la difficulté de cette tache, nous présentons d'abord ces éléments essentiels qui nécessitent généralement la compréhension de ce processus.

2.2 Difficulté de la sélection d'attributs :

2.2.1 Dimensionnalité :

La sélection de variables a pris toute son importance avec l'apparition et la multiplication des données de très grande dimension ces dernières années. Cette révolution Big Data a mené la communauté scientifique à se poser des questions sur les infrastructures et les architectures capable de traiter ces gros volumes de données variées, Grâce aux progrès technologiques, l'acquisition de données devient de plus en plus facile techniquement et des bases de données gigantesque sont collectées quasi-quotidiennement ,et de nombreuses solutions professionnelles et open source apparaissent sur le marché facilitant le traitement et l'analyse de ces données[31].
Plus spécifiquement, nous allons nous intéresser à un phénomène qui est observé lorsque la dimension de l'espace des variables (c'est-à-dire le nombre de variables) grandit si vite que les données qu'il inclut deviennent éparses et éloignées. Cette phénomène appelé « the curse of dimensionality ». Aussi, la complexité des algorithmes augmente lorsque la dimension D croit et certains attributs sont séparément pertinents, mais le gain est faible lorsqu'ils sont combinés [21].

Les méthodes statistiques usuelles auront tendance dans ces situations à donner des résultats faussées et biaisées : c'est le « fléau de la dimension » [37]. Donc Le but est de diminuer la dimensionnalité du problème "Less is more" [36] qui met en exergue la nécessité de supprimer l'ensemble des portions non pertinentes des données de manière préalable à tout traitement si on désire en extraire des informations utiles et compréhensibles nécessaires à l'algorithme d'apprentissage.

2.2.2 Pertinence d'attributs :

La qualité de la classification ne dépend pas du nombre d'informations à disposition mais de la pertinence de ces informations. Parmi les variables à disposition, il s'avère souvent que seules certaines d'entre elles contiennent la structure d'intérêt des observations : c'est les variables pertinentes [31]

Malgré les efforts déployés par les différents auteurs, La notion de pertinence, n'a pas de définition formelle qui est acceptée par tous [44].

Donc, Il existe plusieurs définitions de la pertinence d'un attribut dans la littérature. Celles-ci dépendent de la nature des données, de l'existence de bruit dans les données et de données dupliquées.

Definition1 : Gennari, Langley et Fisher en 1989[45] considèrent des données bruitées et pouvant être dupliquées. Ils définissent les attributs pertinents comme ceux dont les valeurs changent systématiquement en fonction de l'appartenance de la donnée à telle ou telle catégorie.

Definition2 : Almullin et Diettrich,en 1991 (tous les attributs sont booléens)
- Un attribut Fi est dit pertinent pour un concept C si Fi apparait dans chaque formule booléenne qui représente C, il est dit non pertinent sinon [3]
- Un attribut F_i est pertinent, s'il existe deux instances (x_1, y_1) et (x_2, y_2) appartenant à deux classes différentes ($y_1 \neq y_2$). Tel que la valeur F_i est différente pour les deux instances et les valeurs des autres attributs sont identiques. [19]

Definition3 : Koahavi et John [46] définissent les variables pertinentes comme celles dont les valeurs varient systématiquement avec les valeurs de classe. Autrement dit, une variable est pertinente si la connaissance de sa valeur change les probabilités sur les valeurs de la classe. Mais cette définition peut être précisée pour distinguer les variables fortement pertinentes et les variables faiblement pertinentes grâce aux définitions suivantes :

Une variable X_i est fortement pertinente si et seulement si :
$p(C\backslash F_i, S_i) \neq p(C\backslash S_i)$ ..... (2.1)
Une variable Xi est faiblement pertinente si et seulement si :
$p(C\backslash F_i, S_i) = p(C\backslash S_i)$ et s'il existe S_i' tel que $p(C\backslash F_i, s_i) \neq p(C\backslash S'_i)$ ..... (2.2)
Une variable Xi est non pertinente si et seulement si :
$P(C\backslash F_i, S_i) = p(C\backslash S_i)$ et quelque soit $S_i' \subset S_i$ tel que $p(C\backslash F_i, S'_i) = p(C\backslash S'_i)$ .....(2.3)

Les attributs fortement pertinents sont indispensables et ils devraient figurer dans tout sous ensemble optimal sélectionné, car leur absence devrait conduire à un défaut de reconnaissance de la fonction cible. La faible pertinence suggère que l'attribut n'est pas toujours important, mais il peut devenir nécessaire pour un sous ensemble optimal dans certaines conditions. La non pertinence se définit simplement par rapport à (2.1), (2.2) et indique qu'un attribut n'est pas du tout nécessaire dans un sous ensemble optimal d'attributs [3]. Pour savoir comment choisir quels attributs faiblement pertinents, il faut mettre en évidence la notion de redondance.

2.2.3 Redondance :

La redondance d'attributs **[3]** se comprend intuitivement et elle est généralement exprimée en termes de corrélation entre attributs. On peut dire que deux attributs sont redondants (entre eux) si leurs valeurs sont complètement corrélées. Cette définition ne se généralise pas directement pour un sous-ensemble de variables.

On trouve dans **[47]** une définition formelle de la redondance qui permet de concevoir une approche pour identifier et éliminer les attributs redondants. Cette formalisation repose sur la notion de couverture de Markov (markov Blanket) d'un attribut qui permet d'identifier les attributs non pertinents et redondants.
Soit F l'ensemble total d'attributs et C la classe. Soit Fi un attribut, et Mi un sous-ensemble d'attributs qui ne contient pas Fi, c'est-à-dire : $M_i \subseteq F$ et $F_i \notin M_i$
Mi est une couverture de Markov pour Fi si
$P(F - M_i - \{F_i\}, C|F_i, M_i) = P(F - M_i - \{F_i\}, C|M_i)$ .....(2.4)

La définition de couverture de Markov impose que Mi contient non seulement l'information que Fi apporte sur C mais aussi l'information qu'il apporte sur toutes les autres variables. Une autre définition de Koller et Sahami, 1996[47] a montré qu'un sous-ensemble d'attributs optimal peut être obtenu par une procédure d'élimination descendante, appelée filtrage par couverture de Markov (Markov blanket Filtering) et définie comme suit :

Soit G l'ensemble d'attributs courant (G = F au départ). A chaque étape de la procédure, s'il existe une couverture de Markov pour l'attribut Fi dans l'ensemble G courant, Fi est enlevé de G. on peut montrer que ce processus garantit qu'un attribut enlevé dans une étape précédente peut trouver une couverture de Markov dans une étape postérieure. Selon les définitions précédentes de la pertinence d'attributs, on peut également montrer que les attributs fortement pertinents ne peuvent trouver aucune couverture de Markov.Par contre, les attributs non pertinents doivent être enlevés de toute façon, et il n'est donc pas nécessaire de s'y intéresser dans la définition des attributs redondants.

2.3 Processus global de sélection de variables :

Idéalement, les méthodes de la sélection d'attributs recherchent à travers tous les sous ensembles de caractéristiques, et essaient de trouver les meilleurs M attributs à partir d'un ensemble N, tel que M<N et que la fonction d'évaluation (critère) choisie soit optimale sur le sous-ensemble de taille M choisi. Toutefois, cette procédure est exhaustive, elle peut être trop coûteuse et pratiquement exagérée, même pour une taille moyenne d'ensemble de caractéristiques. D'autres approches basées sur des méthodes de recherche heuristiques ou aléatoires tentent de réduire la complexité de calcul en compromettant les performances. Ces méthodes ont besoin d'un critère d'arrêt pour éviter la recherche exhaustive de sous-ensembles.

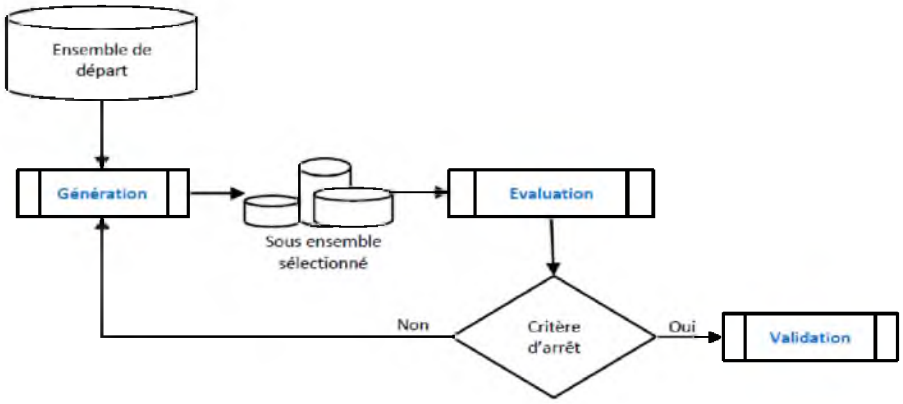

Figure 2.1 : processus de sélection de variables [36]

D'après Dash et Liu [48], Les différentes méthodes proposées dans la littérature pour la sélection de variables peuvent être décrites par un schéma général (figure 2.1) dans lequel on trouve les éléments clés suivants : (Une procédure de génération, Une fonction d'évaluation, Une condition d'arrêt, Un processus de validation pour vérifier si l'objectif souhaité est atteint). Voir figure 2.1

2.3.1. Procédure de génération :

La procédure de génération est une procédure de recherche permettant d'explorer l'espace de recherche pour construire les différentes combinaisons de caractéristiques. La génération de sous-ensemble est essentiellement un processus de recherche heuristique qui, à chaque étape, détermine un sous-ensemble candidat dans l'espace de recherche pour l'évaluation. Cette étape est caractérisée par une direction de recherche et une stratégie de recherche.

a. *Direction de recherche :* C'est la détermination du point (ou les points) de départ de la recherche. En effet, la sélection d'un point dans l'espace sous-ensemble de

caractéristiques, pour commencer la recherche et permettre le passage d'un état à un autre ou chaque état spécifie un sous-ensemble de variables .Elle peut être :

- Ascendante : (forward selection) Cette stratégie d'ajout de variables débute avec l'ensemble vide, puis, à chaque itération, la variable optimale suivant un certain critère est ajoutée. Le processus s'arrête quand il n'y a plus de variable à ajouter, ou quand un certain critère est satisfait.
- Descendante : (backward elimination) La stratégie de suppression de variables débute avec l'ensemble de toutes les variables, puis, à chaque itération, une variable est enlevée de l'ensemble. Cette variable est telle que sa suppression donne le meilleur sous-ensemble selon un critère particulier. Le processus s'arrête quand il n'y a plus de variable à supprimer, ou quand un certain critère est satisfait.
- Approche bidirectionnelle : Ces méthodes permettent de pallier au problème de l'irrévocabilité de la suppression ou de l'ajout d'une variable. En effet, l'importance d'une variable peut se voir modifiée au cours des différentes itérations du processus de sélection de variables. Ces méthodes autorisent l'ajout et la suppression d'une variable de l'ensemble des variables à n'importe quelle étape de la recherche (autre que la première) contrairement à l'ajout de variables (respectivement, suppression de variables) pour laquelle une fois qu'une variable a été ajoutée (respectivement, supprimée) il est impossible de la retirer (respectivement, réintégrer)[38].

b. *Stratégie de recherche* : C'est une procédure qui permet d'explorer l'espace des combinaisons des attributs. Pour un ensemble de données avec N caractéristiques, il existe 2^N sous-ensembles candidats. Par Conséquent, différentes stratégies ont été explorées:

- Recherche complète : Les approches regroupées dans cette catégorie effectuent une recherche complète du Sous-ensemble optimal par rapport à la fonction d'évaluation choisie. Cette méthode n'est pas forcément exhaustive. Différentes fonctions d'évaluation sont utilisées pour réduire l'espace de recherche sans perdre les chances de trouver le sous-ensemble optimal [41].
- Recherche heuristique : Cette catégorie regroupe les algorithmes itératifs pour lesquels chaque itération permet de sélectionner ou rejeter une ou plusieurs caractéristiques. Les algorithmes avec une génération séquentielle sont simples à implémenter et rapides dans la production des résultats [3]
- Recherche aléatoire : La procédure commence avec un sous ensemble sélectionné aléatoirement et procède de deux façons différentes. L'une consiste à continuer la génération des sous-ensembles avec la recherche séquentielle (type I) alors que l'autre consiste à générer le sous ensemble suivant d'une manière complètement aléatoire (type II). Ces méthodes recherchent des sous ensembles en effectuant un maximum d'itérations.

Plusieurs implémentations de génération aléatoire de sous-ensembles de variables sont présentées dans Press, et al en1992. Pour ces trois types de procédures de génération (complète, heuristique ou aléatoire), différent méthodes ont été développées et utilisées pour la sélection d'attributs. Liu et Yu [43] proposent de séparer les méthodes en fonction de la stratégie utilisée, où Les procédures complètes sont subdivisées en «exhaustive» et «non exhaustive», les procédures heuristiques sont subdivisées en «sélection forward», «sélection backward», «forward /backward combinés», et les catégories «d'instance-based». De même, les procédures de génération aléatoires sont regroupés en «type I» et «type II » (cité précédemment).

La figure 2.2 si dessous donne un aperçu général sur les méthodes de sélection de variables basées sur la stratégie de recherche

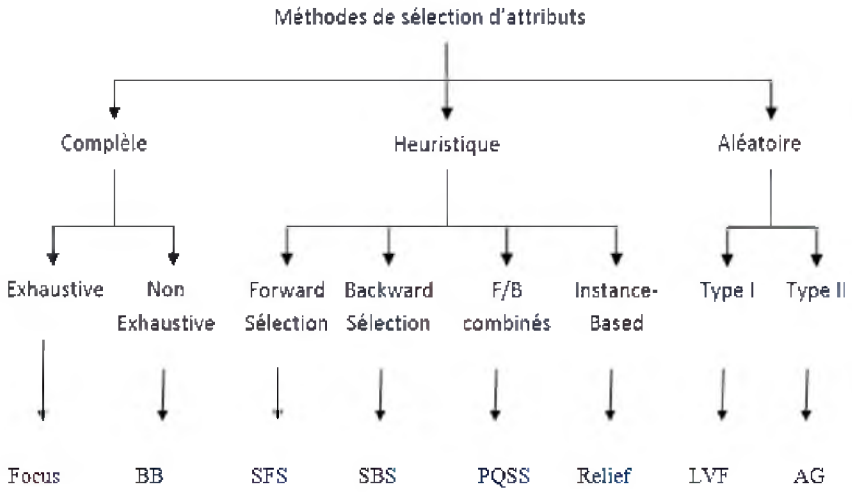

Figure2.2 Catégorisation des méthodes de sélection de caractéristiques [41].

2.3.2 Fonction d'évaluation :

Typiquement, une fonction d'évaluation tente de mesurer la capacité de discrimination d'une caractéristique ou d'un sous-ensemble de variables pour distinguer les différentes classes. . L'optimalité d'un sous-ensemble est relative à la fonction d'évaluation utilisée. Plusieurs critères d'évaluation ont été proposés, basés sur des hypothèses statistiques ou sur des heuristiques. Dash et Liu [48] considèrent que ces fonctions peuvent être regroupées en cinq catégories qui sont les suivantes :

a. Information : fonctions quantifiant l'information apportée par une variable sur la variable à prédire. La variable, ayant le gain d'information le plus élevé, est préférée aux autres variables. (Le gain d'information étant la différence entre l'incertitude a priori et l'incertitude a posteriori c'est-à-dire avant et après la sélection d'une caractéristique X).

b. Distance : fonction s'intéressant au pouvoir discriminant d'une variable. Elle évalue la séparabilité des classes en se basant sur les distributions de probabilités des classes. Une variable est préférée à une autre si elle induit une plus grande séparabilité.

c. Dépendance : également appelée mesure de corrélation ou mesure de similarité. Elle permet de mesurer la capacité d'une caractéristique à prédire la valeur d'une autre. Dans la sélection de caractéristiques pour un problème de classification, on cherche à quel point une caractéristique est associée à une classe. La caractéristique X est préférée à Y, si la corrélation d'une variable X avec une classe C est plus importante que la corrélation de la variable Y avec C.

d. Consistance : fonctions liées au biais des variables minimum. Ces méthodes recherchent le plus petit ensemble de variables qui satisfait un pourcentage d'inconsistance minimum défini par l'utilisateur.

e. Précision : ces méthodes utilisent le classifieur comme fonction d'évaluation. Le classifieur choisit, parmi tous les sous-ensembles de variables, celui qui est à l'origine de la meilleure précision prédictive.

2.3.3 Critère d'arrêt :

Les itérations du processus de sélection de caractéristiques continuent à s'exécuter jusqu'à ce qu'un critère d'arrêt soit atteint. La procédure de génération et la fonction d'évaluation peuvent influencer sur le choix d'un critère d'arrêt.
Les critères d'arrêt basés sur la procédure de génération incluent :

- Nombre de caractéristiques sélectionnées est égal à un nombre prédéfini.
- Nombre prédéfini d'itérations est atteint.
- L'ajout (ou suppression) d'une caractéristique ne produit pas un meilleur sous ensemble.
- Un sous-ensemble optimal de caractéristiques est obtenu à partir de certaines fonctions d'évaluation.

2.3.4 Procédure de validation :

La procédure de validation n'est pas une étape du processus de sélection de caractéristiques lui-même, mais une méthode de sélection d'attributs (en pratique) doit être validée. Elle consiste à tester la validité des sous-ensembles de caractéristiques sélectionnées avec la réalisation de différents tests, et la comparaison des résultats obtenus avec les résultats précédents, ou avec des résultats d'autres méthodes de sélection de caractéristiques en utilisant des données artificielles, réelles, ou les deux [32] [36].

3 Approches de sélection de variables

Dans cette section, nous présentons les différentes approches de sélection de variables (features selection). Il existe principalement, trois familles de techniques dans la littérature :
– **les méthodes par Filtre** (filter methods) : qui consiste à effectuer la sélection indépendamment de la classification pendant une étape de prétraitement.
– **les approches à adaptateur** (wrapper methods):approche à adaptateur qui utilise le système de classification comme une boite noire uniquement pour évaluer le pouvoir prédictif d'un groupe de variable.
– **les méthodes intégrées** (embedded methods):c'est la famille des méthodes intégrés utilise également le clasifieur mais en sélectionnant les variables durant la phase d'App [29][30].

3.1 Méthode par Filtre :

Le filtrage est un processus de prétraitement des données (l'étape de l'analyse des données), par filtrage des variables non pertinentes avant le passage à la phase de classification. Cette approche utilise les caractéristiques générales de l'ensemble de variables pour sélectionner certaines variables et en exclure d'autres. Elle se base sur la performance de la fonction d'évaluation calculée directement sur l'ensemble d'apprentissage comme : la distance, l'information, la dépendance, et la cohérence.

Les méthodes de type filtre sélectionnent les variables indépendamment du modèle utilisé pour la classification (figure 2.3). Un de ces avantages est d'être complètement indépendant du modèle de données que nous cherchons à construire. Elle

propose un sous ensemble de variables satisfaisant pour expliquer la structure des données qui se cachent et que le sous ensemble est indépendant de l'algorithme d'apprentissage choisi. Ces méthodes sont particulièrement efficaces et moins couteuses en temps de calcul puisqu'elles évitent les exécutions répétitives des algorithmes d'apprentissage sur différents sous ensemble de variables, et robustes face au sur-apprentissage (over-fitting). En revanche, l'inconvénient majeur de ces méthodes est qu'elles ne tiennent pas compte des interactions entre les variables et tendent à sélectionner des variables comportant de l'information redondante plutôt que complémentaire. [19] [29] [32] [36]

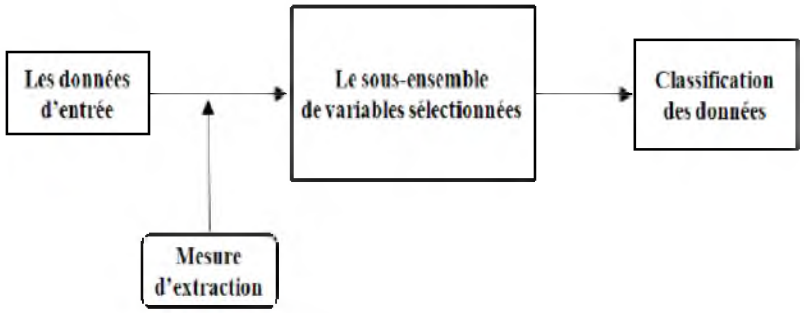

Figure 2.3 :Le principe génerale d'une méthode de sélection de type filter [30]

3.2 Approches à adaptateur (Wrapper) :

Les wrappers ont été introduits par Kohavi et John en 1994 ; Kohavi & John en1997 [36].Leurs principe est de générer des sous ensembles candidats et de les évaluer grâce à un algorithme de classification ce qui permet, contrairement aux approches filter, de prendre en compte les éventuelles interactions entre variables. Pour ces auteurs, les algorithmes de filtrage ne sont pas toujours efficaces car ils ignorent totalement l'influence de l'ensemble de variables sélectionnées sur les performances de l'algorithme de classification.

Pour résoudre ce problème, ils proposent une approche différente qui utilise le résultat de l'algorithme de classification comme fonction d'évaluation. L'algorithme de classification appliqué aux données prétraitées est utilisé comme un sous-programme et considéré comme une boite noire (l'apprentissage est effectué avec les variables sélectionnées et les performances sont estimées à partir de l'erreur de généralisation par cet ensemble de méthodes). Cette évaluation est faite par un calcul d'un score, par exemple un score d'un ensemble sera un compromis entre le nombre de variables éliminées et le taux de réussite de la classification sur un fichier de test.

L'approche Wrapper présente l'avantage d'être appliquée à un très grand nombre d'attributs, car elle est de complexité raisonnable. Elle ne tient compte que des informations présentées dans les données et elle est indépendante du processus de la classification. Aussi, elle génère un sous ensemble bien adaptés à l'algorithme de classification (Figure 2.4). Les taux de reconnaissance sont élevés car la sélection prend en compte le biais intrinsèque de l'algorithme de classification. Un autre avantage est sa simplicité conceptuelle ; nous n'avons pas besoin de comprendre comment l'induction est affectée par la sélection des variables, il suffit de générer et de tester. Cependant, la méthode Wrapper repose sur le choix d'un seuil comme un critère de pertinence choisi ou d'un nombre d'attributs à choisir (ceci n'est pas facile à réaliser). Aussi, les méthodes wrapper n'apportent pas vraiment de justification théorique à la sélection et

elles ne nous permettent pas de comprendre les relations de dépendances conditionnelles qu'il peut y avoir entre les variables. D'autre part, la procédure de sélection est spécifique à un algorithme de classification particulier et les sous ensembles trouvés ne sont pas forcément valides si nous changeons la méthode d'induction. Enfin, Les principaux inconvénients de ces méthodes sont le risque de sur-apprentissage lorsque le nombre d'observations est insuffisant ainsi que le temps de calcul qui devient important lorsque le nombre de variables est grand, les calculs devient de plus en plus très longs, voir irréalisables [3] [19] [29] [32] [36].

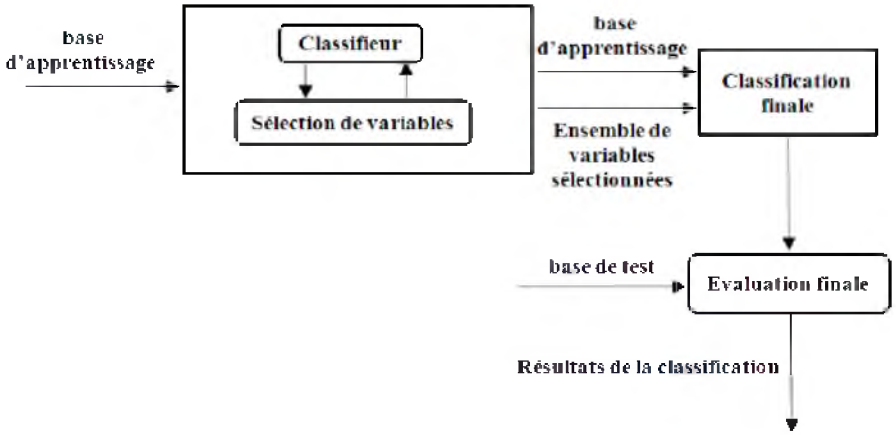

Figure 2.4 : Le principe générale d'une méthode de sélection de type wrapper[30]

La stratégie wrapper était supérieur à la stratégie filtre en terme de performance de classification. L'approche filtre est plus rapide que l'approche Wrapper en terme de génération de résultats. Cependant, cette dernière à l'avantage de fournir généralement des résultats plus pertinents pour la classification. La Figure 2.5 présente deux modèles généraux des approches Filtre et Wrapper pour la sélection de caractéristiques

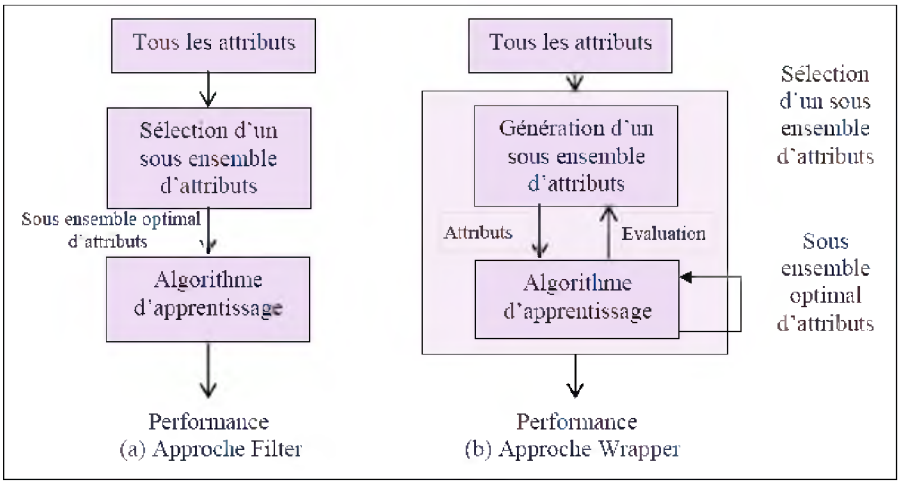

Figure 2.5 : comparaison entre les deux approches de la sélection d'attributs [3]

3.3 Approches intégrées (Embedded) :

Récemment, les méthodes embedded ont été proposées en classification pour diminuer le sur-apprentissage. Elles ont pour objectif de combiner les avantages des deux types de méthodes précédents. L'algorithme d'apprentissage utilise son propre algorithme de sélection de variables et nécessite donc de pouvoir caractériser a priori ce

que serait une bonne sélection, ce qui limite leur utilisation. Dans la plupart des problèmes de sélection de variables l'objectif est que les attributs sélectionnés soient pertinents [19].

Ces approches incorporent la sélection de variables lors du processus d'apprentissage, sans étape de validation, pour maximiser la qualité de l'ajustement et minimiser le nombre de variables. Cette méthode est proche de celle de wrapper car elle combine le processus d'exploration avec l'algorithme d'Apprentissage (voir figure 2.6) .Cependant, l'ajout des méthodes wrapper est que le classifieur sert non seulement à évaluer un sous ensemble candidats mais aussi à guider le mécanisme de sélection.Un exemple très connu est celui des arbres de décision, où les variables sélectionnées sont celles présentes au niveau de la division de chaque nœud. Selon Guyon et al, ces approches seraient bien plus avantageuses en terme de temps de calcul que les méthodes de type wrapper et seraient robustes face au problème de sur-apprentissage [36][38].

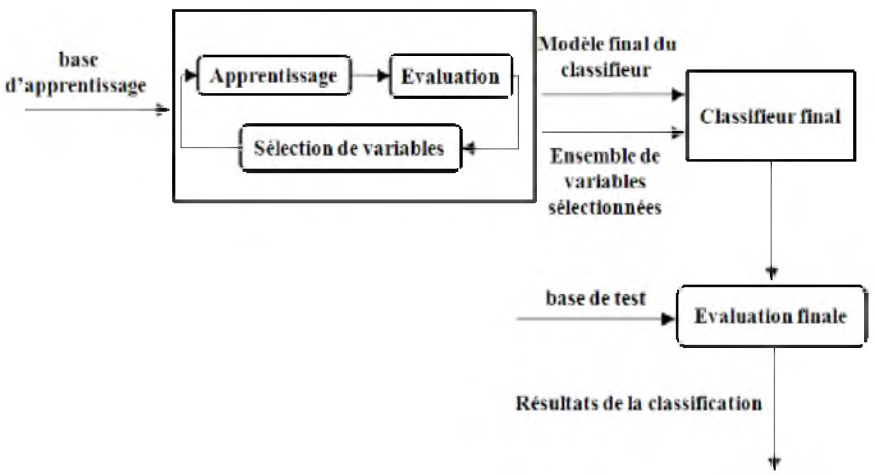

Figure 2.6 : Le principe générale d'une méthode de sélection de type embedded [30]

4 Métaheuristiques pour la sélection de variables

La sélection d'attributs est un sujet de recherche très actif depuis une dizaine d'années dans le domaine d'apprentissage artificiel. Pour cela, l'utilisation des métaheuristiques, en particulier, les méthodes inspirées de la biologie (ce qui nous intéresse dans cet thèse de master), ont attiré l'attention de la communauté de recherche depuis long temps. Dans cette section, nous présentons quelques méthodes de sélection de variables citées dans la littérature, et qui se basent sur les métaheuristiques telles que les algorithmes génétiques (AGs), les colonies de fourmis (ACO) et les essaims particulaires (PSO).

Le Tableau 2.1 présente quelques travaux de la littérature sur les applications d'algorithmes de sélection de variables utilisant les approches métaheuristiques. La première colonne contient des informations sur les auteurs et la référence du travail en question. La 2ème et la 3ème colonne indiquent la métaheuristique utilisée pour la sélection des variables et l'approche de sélection appliquée respectivement. La 4ème colonne présente le domaine d'application ou les bases de données utilisées pour les expérimentations. La dernière colonne présente les résultats de ces travaux en termes de nombre de caractéristiques sélectionnées, l'ensemble d'origine ou l'ensemble initial des

caractéristiques, et SV (Sélection de Variables) représente la taille du sous ensemble optimal sélectionné par la méthode considérée.

Référence	Métaheuristique Utilisée	Approche de SV	Domaine d'application (bases de données utilisé)	Résultats	
				Ensemble d'origine	SV
Kabir et al [49]	AG hybride	Filtre	Breast cancer	9	≈3.25
Zhang et al[50]	AG	Wrapper	Le cancer du sein	14	3
Kabir et al [51]	ACO	Wrapper + filtre	Thyroid	21	3
Wang et al [52]	PSO	Wrapper	Lung	56	4
Liu et al [53]	PSO	Wrapper	Image segmentation	30	5
Prasartvit et al [54]	ABC	Wrapper	Sonar	60	31
Chuang [55]	PSO	Wrapper	LeuKemia1	5327	19%
Allaoui Abdiya[3]	ACO /C4.5	Wrapper	-Heart-statlog -Sonnar	13 60	9 22
Cheng-lung Huang[30]	AG+SVM	Wrapper	Pima	8	4
Deng et al[56]	ACO	Wrapper	Le cancer du poumon	56	49

Tableau 2.1 : Applications des métaheuristiques à la sélection de variables.

Aussi, voici d'autres travaux de recherches qui montrent l'utilisation des métaheuristiques pour la résolution du problème de Sélection de variables (SV) :

- Une méthode utilisant l'Algorithme de Colonie de Fourmis est proposée par les auteurs **Deriche, en 2009**[39]. C'est une approche hybride qui donne l'importance à la performance d'attributs et à la recherche locale. Cette méthode combine l'approche Wrapper pour améliorer la performance d'attributs et l'approche Filter pour la recherche locale.
- D'après les auteurs **wang et al, en 2007**[40], l'optimisation par essaim particulaires (PSO) comparée avec l'Algorithme Génétique n'a pas besoin de paramètres complexes (croisement, mutation,…). La PSO est efficace pour la sélection d'attributs en se basant sur la théorie des ensembles.
- Benahmed [41] propose une méthode d'optimisation pour la sélection et la pondération des caractéristiques d'un système de reconnaissance de chiffres manuscrits isolés, basé sur les algorithmes génétiques dans le but d'optimiser un système de reconnaissance de l'écriture manuscrite.
- Les auteurs **Huang et Dun** en **2008** ont implémenté l'optimisation par essaim particulaires avec le classifieur Séparateur à Vaste Marge (SVM) via une architecture

distribuée, utilisant le web service pour réduire le temps de calcul dont le résultat est la sélection correcte d'attributs avec une précision élevée de la classification [3]

- L'auteur Wassem Shahzad [42] a proposé un algorithme hybride entre l'optimisation par colonie de fourmis et l'algorithme ID3 dont il a utilisé l'information de gain comme une valeur heuristique.

5 Conclusion

La sélection de variables est un aspect essentiel de l'apprentissage supervisé. Nous devons déterminer les variables pertinentes pour la prédiction des valeurs des variables pour différentes raisons : un modèle plus simple sera plus facile à comprendre et à interpréter ; nous aurons besoin de moins d'informations à recueillir pour la prédiction ; enfin, un modèle simple se révèle souvent plus robuste en généralisation.

Dans ce chapitre, nous avons décrit l'importance de la sélection d'attributs, présenté le processus et les différentes composantes nécessaires pour un algorithme de sélection de caractéristiques, et défini les alternatives possibles pour leurs mises en œuvre. Un certain nombre d'algorithmes ont été décrits en fonction de ces composantes (la fonction d'évaluation et la stratégie de recherche).Aussi, nous avons présenté les principaux travaux utilisant les métaheuristiques pour la résolution du problème de sélection de variables. Étant donné le succès des métaheuristiques pour la sélection de variables, nous proposons d'adopter l'algorithme génétique et le classifieur SVM dans le contexte de classification des données médicales. C'est l'objet du chapitre suivant.

Chapitre III :
Implémentation du modèle OP-VAR

1 Introduction

La sélection d'attributs est une technique permettant de construire des modèles d'apprentissage robustes et d'obtenir une bonne performance de classification.
L'utilisation des méthodes d'optimisation est donc un moyen efficace de traiter ces problèmes. En effet, pour aborder des problèmes ayant un grand nombre de variables, les métaheuristiques (recherche locales, algorithmes évolutionnaires, . . .) ont prouvé leur efficacité. Pour caractériser et acquérir ce concept théorique, la mise en œuvre en pratique est le moyen le plus idéal. Dans notre contexte de sélection de variables par optimisation combinatoire, nous proposons d'utiliser une recherche à base de population (Optimisation par les algorithmes génétiques) dans laquelle une solution est évaluée par l'utilisation d'un classifieur à base ' SVM '(Support Vector Machine) .A noter que le système développé est nommé Op-Var (Optimisation des variables), et est basé sur une approche enveloppante (wrapper).

2 Description du jeu de données

Les données utilisées dans cette étude sont issues de la base de données internationale UCI « University of California Irvin ». Cette collection qui a été largement utilisé par les chercheurs du monde entier comme une source primaire d'apprentissage automatique des ensembles de données. Dans ce papier, nous utilisons trois bases de données médicales. Ces données sont caractérisées par N exemples d'apprentissage (patients).Chaque exemple est représenté par un vecteur de caractéristiques (attributs) et associé à une classe label. Les principales caractéristiques de ces ensembles de bases sont représentées dans le tableau suivant :

Bases de données	Nbr d'attributs	Taille de la base	Cas positifs	Cas négatifs
Pima(Diabète)	**8**	**768**	**268**	**500**
Heart diseas (Stat-Log Project)	**13**	**270**	**120**	**150**
Hepatitis	**19**	**155**	**32**	**123**

Table 3.1 Les trois bases de données utilisées dans cette étude

2.1 Description de la base de données de diabète

La base de données Pima Indians Diabetes a été choisie du dépôt d'UCI [43]. Les auteurs ont réalisé 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 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 Pima sont décrits dans le tableau suivant :

N°	Les attributs	La moyenne	L'écart-type	Min	Max
1	nombre de grossesses.	3.8	3.4	0	17
2	concentration du glucose plasmatique.	120.9	32.0	0	199
3	tension artérielle diastolique, (mm Hg).	69.1	19.4	0	122
4	épaisseur de pli de peau du triceps, (mm).	20.5	16.0	0	99
5	dose d'insuline, (mu U/ml).	79.8	115.2	0	846
6	index de masse corporelle,(poids en kg/(taille m)2).	32.0	7.9	0	67.1
7	fonction de pedigree de diabète (l'hérédité).	0.5	0.3	0.078	2.42
8	âge (Année).	33.2	11.8	21	81

Table 3.2 Caractéristiques et paramètres de l'ensemble de données sur le diabète

2.2 Description de la base de données Heart Statlog

Le Système est conçu aussi sur la base de données heart statlog. Cet ensemble de données fait partie de la collection de bases de données à l'Université de Californie, Irvine recueillis par David Aha[59]. Le but de cet ensemble de données est de prédire la présence ou l'absence de maladie de cœur, étant donné les résultats des différents tests médicaux effectués sur un patient. Cette base de données contient 13variables en entrée et 270exemples de patients, et une seul sortie qui représente le diagnostic. Les champs se divisent pour certaine section et chaque section a une valeur. Voila un tableau qui représente les différentes composantes de cette base de données :

N°	paramètres	description	Valeur
1	Age	Age en année	continue
2	Sex	Male ou femelle	1=masculin 0=féminin
3	CP	Douleur thoracique	1=angine typique 2=angine atypique 3=douleur non angine 4=asymptomatique
4	Thestbps	La pression artérielle au repos	Valeur continue mm hg
5	chol	cholestérol	Valeur continue mm/dl
6	Restecg	L'électrocardiogramme ECG	0=normale 1=ayant ST-T anomalie d'onde (onde T inversion et/ou élévation du segment ST) 2=hypertrophie (montrant gauche probable ou certaine hypertrophie ventriculaire)
7	fbs	Glycémie	0=faux <=120 mg/dl 1=vrai >=120 mg/dl
8	thalach	Fréquence cardiaque au repos maximale	Valeur continue
9	exang	L'exercice	0=faux 1=vrai
10	oldpeak	Vieux pic (ST dépression induite par l'exercice par rapport au repos)	Valeur continue
11	slope	Pente du segment ST	1=pente ascendante 2=plat 3=downsloping
12	ca	nombre de navires principaux colorés par fluoroscopie	0-3 valeurs
13	thal	Scintigraphie au thallium	3=6=défaut normale fixe 7=défaut révisable

Table 3.3 Description des caractéristiques de la base de données Heart statlog

2.3 Description de la base de données Hepatitis

Cet ensemble de données a été donné par Jozef Stefan Institute, Yougoslavie. Le but de l'ensemble de données est de prédire la présence ou l'absence de maladie de l'hépatite à partir des différents résultats de tests médicaux d'un patient. Cette base de données contient 19 attributs. Il y a 13 binaires et 6 valeurs discrètes. Cet ensemble de données

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

comprend 155 échantillons provenant de deux classes différentes (32 cas "die", 123 cas "live").Les Attributs des symptômes qui sont obtenus à partir des patients sont donnés dans le tableau suivant [60] :

N°	Le Nom de l'attribut	L'intervalle de l'attribut
1	Age	7-78
2	Sex	male , female
3	Steroid	No, Yes
4	Antivirals	No, Yes
5	Fatigue	No, Yes
6	Malaise	No, Yes
7	Anorexia	No, Yes
8	Liver Big	No, Yes
9	Liver Firm	No, Yes
10	Spleen Palpable	No, Yes
11	Spiders	No, Yes
12	Ascites	No, Yes
13	Varices	No, Yes
14	Bilirubin	0.3–8
15	Alk Phosphate	26–295
16	SGOT	14–648
17	Albumin	2.1–6.4
18	Protime	0–100
19	Histology	No, Yes

Table3.4 les différents attributs de la base de données Trouble de foie (Hepatitis)

3 Environnement de développement

Nous nous sommes tournées vers l'outil Matlab. Matlab est l'abréviation de matrix laboratory, où l'élément de base de données est une matrice. Avec ses fonctions spécialisées, MatLab peut être aussi considéré comme un langage de programmation adapté pour les problèmes scientifiques. C'est un interpréteur : les instructions sont interprétées et exécutées ligne par ligne. Il fonctionne dans plusieurs environnements Matlab est organisé en boîte à outils (toolbox) spécialisés [http://www.mathworks. com /products/neuralnet/description6.html]. Les toolboxes sont réellement des boîtes à outils comportant une collection de fonctions relatives à plusieurs domaines scientifiques et techniques.

4 Critères d'évaluation :

Nous avons choisi d'évaluer les performances en termes de taux d'erreur du classifieur construit à l'aide de ces trois jeux de données médicales. Dans la pratique l'évaluation de taux d'erreur est faite par une validation croisée (K-FOLD=5). La

performance est mesurée par le taux de bonne classification moyennée sur les ensembles de test, et par le calcul de pourcentage de sensibilité (SE),la spécificité (SP).
– **Sensibilité (SE%) :** [SE = 100 * VP / (VP + FN)] la sensibilité (Se) du test est sa Capacité de donner un résultat positif quand la maladie est présente.
– **Spécificité (SP %) :** [SP = 100* VN / (VN + FP)] la spécificité du test est cette capacité de donner un résultat négatif quand la maladie est absente.
– **Taux de classification (Tc%) :** [TC = 100 * (VP + VN) / (VN + VP + FN +FP)] est le pourcentage des exemples correctement classées.
Avec :
– VP : malade classé malade.
– VN : non malade classé non malade.
– FP : non malade classé malade.
– FN : malade classé non malade.

5 Le model Op-Var:

Les sections suivantes donnent une description complète de classifieur utilisé et les étapes de manière plus détaillée de notre approche proposée (Op-Var).

5.1 Description du classifieur SVM :

Nous proposons une nouvelle technique de sélection de variables basée sur une approche enveloppante « wrapper » à travers un algorithme génétique couplé à un classifieur. Notre choix s'est porté sur le classifieur SVM pour ses performances et sa robustesse, ou chaque candidat est évalué grâce au classifieur. Les Séparateurs à Vaste Marge ou Support Vector Machines (SVM) ont été proposés en 1995 par V. Vapnik dans son livre « The nature of statistical learning theory » [58]. Elles sont des techniques largement répandues en apprentissage statistique, elles ont eu beaucoup de succès dans quasiment tous les domaines où elles ont été appliquées.

Elles reposent sur deux idées clés : la notion de marge maximale et la notion de fonction noyau. Cette méthode est donc une alternative récente pour la classification.
Elle repose sur l'existence d'un classifieur linéaire dans un espace approprié.
Étant donné que c'est un problème de classification à deux classes, cette méthode fait appel à un jeu de données d'apprentissage pour apprendre les paramètres du modèle. Elle est basée sur l'utilisation de fonctions dites noyau (kernel) qui permettent une séparation optimale des données.Numériquement, toutes les équations s'obtiennent en fonction de certains produits scalaires utilisant le noyau et certains points de la base de données (ce sont les Support Vectors).

5.1.1 Hyperplans séparateurs et discriminants dans un problème à deux classes :

Pour deux classes d'exemples données, le but de SVM est de trouver un classificateur qui va séparer les données et maximiser la distance entre ces deux classes. Avec SVM, ce classificateur est un classificateur linéaire appelé hyperplan. Dans le schéma 3.1, on détermine un hyperplan qui sépare les deux ensembles de points.

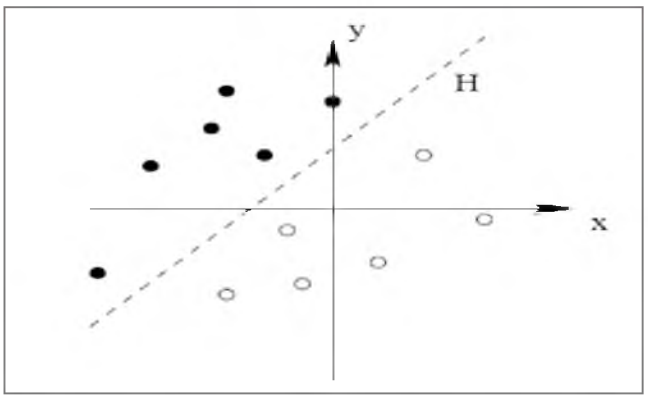

Figure 3.1 un hyperplan séparant un ensemble de donnée en deux classes

Les points les plus proches, qui seuls sont utilisés pour la détermination de l'hyperplan, sont appelés vecteurs de support. Il est évident qu'il existe une multitude d'hyperplan valide mais la propriété remarquable des SVM est que cet hyperplan doit être optimal. Nous allons donc en plus chercher parmi les hyperplans valides, celui qui passe ≪ au milieu ≫ des points des deux classes d'exemples. Intuitivement, cela revient à chercher l'hyperplan le ≪ plus sur ≫. Formellement, cela revient à chercher un hyperplan dont la distance minimale aux données d'apprentissage est maximale. On appelle cette distance ≪ marge ≫ entre l'hyperplan et les données. L'hyperplan séparateur optimal est celui qui maximise la marge. Comme on cherche à maximiser cette marge, on parlera de séparateurs à vaste marge.

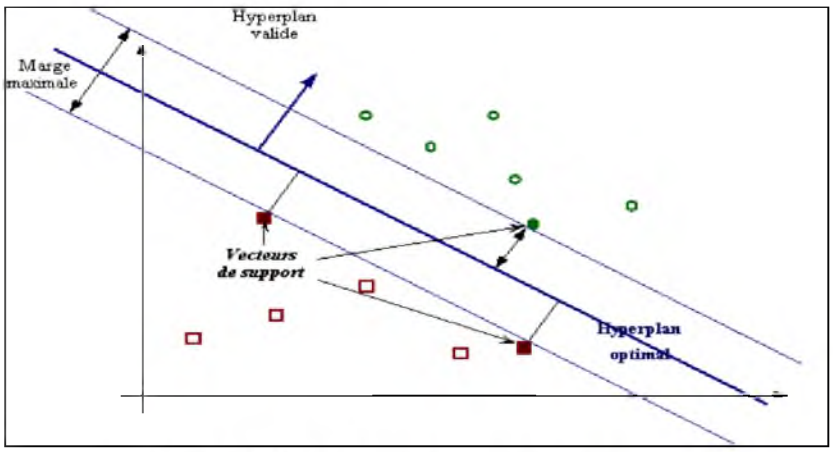

Figure 3.2 Séparateur à vaste marge

Parmi les modèles des SVM, on constate les cas linéairement séparables et les cas non linéairement séparable. Les premiers sont les plus simples de SVM car ils permettent de trouver facilement le classificateur linéaire. Dans la plupart des problèmes réels il n'y a pas de séparation linéaire possible entre les données, le classificateur de marge maximale ne peut pas être utilisé car il fonctionne seulement si les classes de données d'apprentissage sont linéairement séparables.

5.2 Architecture du modèle:

La procédure générale de notre approche Op-Var qui constitue une démarche de sélection de variables peut être caractérisée par un processus itératif à deux étapes fondamentales pour réduire graduellement l'espace de recherche et sélectionner un sous ensemble pertinent d'attributs. (Voir figure 3.3)

I) la première étape effectue une conception d'un chromosome qui consiste le sous-ensemble d'attributs.

II) la deuxième étape évalue la qualité de chaque sous-ensemble pour en choisir le meilleur, à partir de la conception d'une fonction de remise en forme (fitness function) appropriée qui combine les objectifs visés de notre étude (taux de classification élevé avec un sous ensembles d'attributs pertinents réduits).

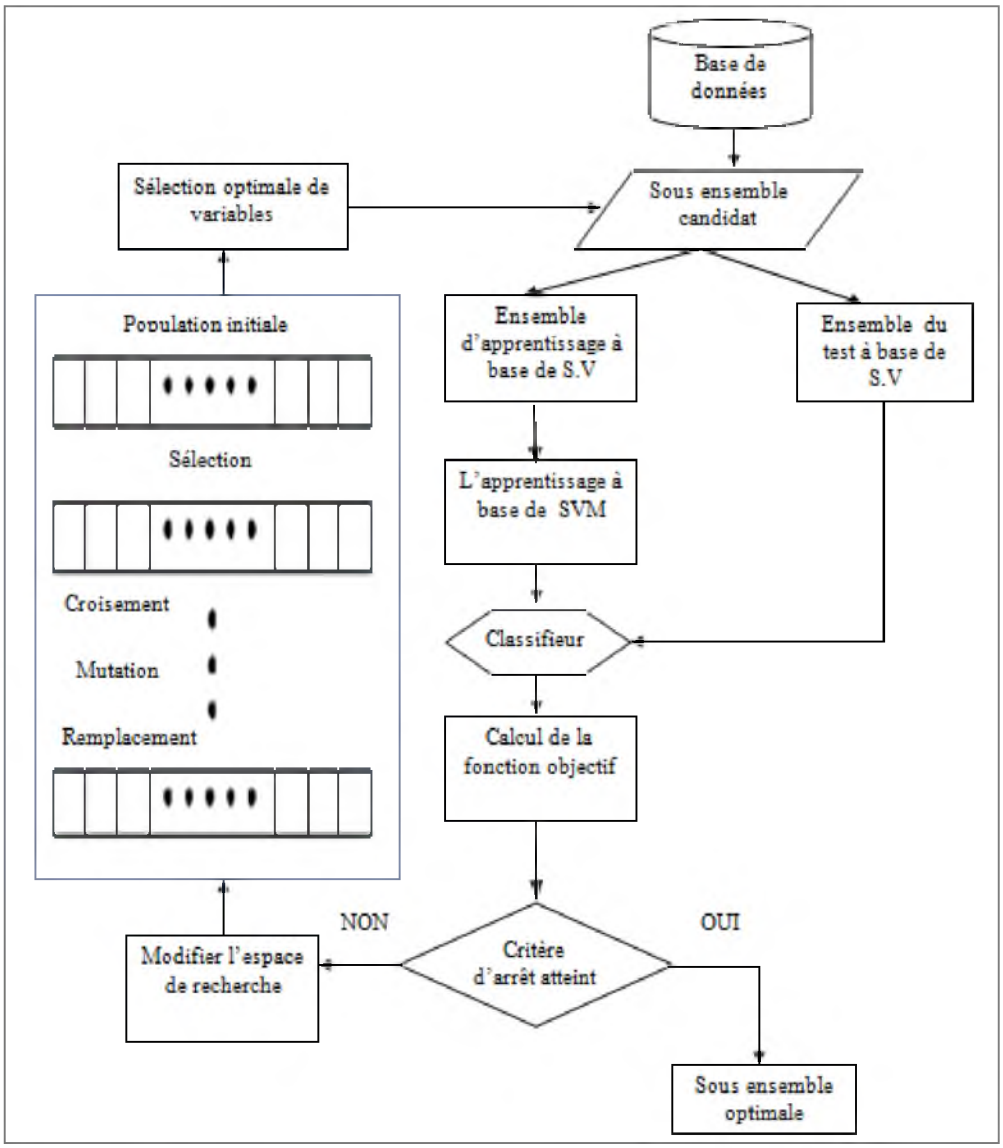

Figure 3.3 : le processus général du modèle OpVar

5.2.1 Conception du chromosome :

Les chromosomes de la population initiale sont codés en mode binaire, chaque allèle (bit) du chromosome représente un attribut de la base de données .Les chaînes de bits représentant le génotype qui devrait être transformé en phénotype. Si un allèle est à «1», cela signifie que cet attribut est pertinent et donc il est conservé dans le sous-ensemble sélectionné de la nouvelle base et s'il est à «0» il indique que l'attribut n'est pas pertinent et donc il n'est pas inclus dans le sous-ensemble sélectionné. Chaque chromosome représente donc un sous-ensemble d'attributs. La population initiale de l'AG est générée aléatoirement selon une répartition uniforme et la longueur du chromosome dépend de la taille du sous-ensemble d'attributs.

5.2.2 Conception de la fonction de remise en forme :

Une fonction de remise en forme est nécessaire dans l'algorithme génétique ; elle sert à évaluer si une personne est «apte» pour survivre. Dans le modèle OpVar, nous utilisons deux critères, qui sont la précision de la classification et le nombre d'entités sélectionnées, pour concevoir la fonction de remise en forme. Le principe est que les individus avec une précision élevée de classification et un petit nombre de caractéristiques sélectionnées a une grande valeur de remise en forme, et donc une forte probabilité d'être passé ses attributs à la prochaine génération. La fonction unique objective de remise en forme qui combine les deux objectifs en un seul a été conçue pour résoudre les différents critères du problème.

5.2.3 Étapes de base de la méthode GA-SVM :

Les étapes de base de notre procédé GA-SVM sont les suivantes :

1) Créer une population initiale de certaine taille, à savoir un groupe d'individus avec des chromosomes différents. Les chromosomes de la population initiale ont été créés de manière aléatoire. La taille de la population initiale doit être déterminée correctement par l'utilisateur d'inclure autant de solutions possibles que possible.

2) Calculer la valeur de remise en forme de chaque individu dans la première population et de les classer en fonction de leur aptitude. Pour calculer la valeur de remise en forme d'un individu ou d'un chromosome, les génotypes sont d'abord convertis en phénotypes, Ces valeurs convertis sont ensuite utilisés en entrée du classificateur SVM pour effectuer la classification; Après, la précision de la classification est évaluée en fonction du test de jeux de données; enfin, la valeur de remise en forme de l'individu est calculé en fonction de la précision de la classification et de nombre des attributs sélectionnées.

Pour évaluer chacun de ces sous-ensembles il faudra lancer plusieurs fois le classifieur utilisé afin de déduire la mesure de performance. Pour cela nous avons utilisé un mécanisme de validation croisée.

3) Sélectionner un certain nombre d'individus ayant une valeur élevée de remise en forme comme «l'élitisme» de la population et de les conserver dans la prochaine génération. De cette manière, les individus avec une grande valeur de remise en forme sont conservés dans la population et le principe de «Survie du plus fort» de l'algorithme génétique est transmis.

4) Vérifiez si les conditions d'arrêt sont satisfaites. Si l'évolution est arrête et le résultat optimal représenté par le meilleur individu est retourné. Dans le cas contraire,

l'évolution continue et la génération suivante soit produite. Les conditions d'arrêt peut être soit un seuil prédéfini de remise en forme ou le nombre de génération a évolué.
5) Si la population continue d'évoluer, la prochaine génération est produite en suivant le processus génétique. Dans cette étape, le système cherche de meilleures solutions par des opérations génétiques, y compris la sélection, croisement, mutation, et le remplacement. En premier lieu, un certain nombre d'individus est choisis au hasard pour concurrencer le droit d'accouplement.
6) Répéter les opérations de l'étape (2) à l'étape (4).

5.3 Choix des paramètres :

Le point critique de notre système est bien le réglage des paramètres que ce soit pour l'algorithme génétique ou pour les SVM Et comme il n'existe pas de règles universelles de réglages donc seuls les résultats expérimentaux peuvent donner une idée du comportement de notre système (algorithme génétique + SVM).
Pour cela il nous a semblé nécessaire d'expérimenter de nombreuses combinaisons pour obtenir de bons résultats. Les paramètres modifiables de notre système sont les suivants :

Paramètres	Caractéristiques
Points de départ	Sous-ensembles d'attributs aléatoires
Espace de recherche	(2^N-1) Sous ensembles d'attributs
Nature de la population	Ensemble des sous ensemble d'attributs
Taille de la population	Fixée par expérience
Codage du chromosome	Binaire
Fonction d'évaluation	SVM
Nombre de génération maximale	Fixée par expérience
Fitness	Taux d'erreur
Type de sélection	Sélection par tournoi
Probabilité de croisement	0.6
Stratégie d'évaluation	Méthode enveloppante à base SVM
Critère d'arrêt	Nombre de génération / taux de classification à 100%
Validation des Résultats	Surveillance du changement des performances par rapport aux changements des ensembles d'attributs

Tableau 3.5 : Paramètre de l'algorithme génétique

Ce choix se justifie, par les expériences que nous avons effectuées.

5.4 Approche GA-SVM dans la littérature :

Il existe plusieurs travaux dans la littérature dans différents domaine issu de l'intelligence artificielle qui ont utilisé cette hybridation entre l'algorithme génétique et le support à vecteur machine pour la sélection d'un meilleur sous ensemble d'attributs, parmi ces derniers on peut citer :

❖ Li Zhuo a, Jing Zheng [62] utilisent une approche wrapper entre l'Algorithme génétique et SVM pour la sélection et la classification des images hyper spectrales, cette méthode a été utilisé pour optimiser à la fois le sous-ensemble de caractéristiques, à savoir la bande sous-ensemble, des données spectrales et les paramètres du noyau SVM simultanément.

Les résultats montré que la méthode GA-SVM pourrait réduire de manière significative la le coût de calcul, tout en améliorant la précision de la classification qui est passée de 88,81% à 92,51%.

❖ Cheng-Lung Huang , Chieh-Jen Wang [63] ont présenté une nouvelle technique de classification de motif qui ont été largement utilisés dans de nombreux domaines d'application. L'objectif de cette recherche est d'optimiser simultanément les paramètres de svm et de sélectionner un sous-ensemble d'attributs sans dégrader la précision de classification de svm. donc ils ont proposé une approche basée sur l'algorithme génétique pour résoudre ce genre de problème. Ils ont essayé plusieurs ensembles de données issu du l'UCI en utilisant l'approche proposée à base de GA et de l'algorithme de Grid (une méthode traditionnelle d'exécution pour améliorer les paramètres de recherches). Par rapport à l'algorithme de Grid, l'approche proposée améliore significativement la précision de la classification et a moins d'attributs d'entrée pour les machines à vecteurs de support (SVM).

❖ Edmundo Bonilla Huerta, B´eatrice Duval, et Jin-Kao Hao [61] ont intéressé à la sélection génétique et la classification de l'ADN ; données de micro puce afin de distinguer les échantillons de tumeurs de celles normales. A cet effet, ils ont proposé un modèle hybride qui utilise plusieurs techniques complémentaires : une logique floue, un algorithme génétique (GA) associé à un vecteur de support Machine (SVM) et une technique de sélection de gènes à base d'archives. Cette approche a plusieurs caractéristiques particulières. Tout d'abord, pour faire face à la difficulté liée à des données de grande dimension, on introduit un outil de pré-traitement à base logique floue qui permet de réduire largement la dimensionnalité des données en regroupant des gènes similaires.

❖ Mohd Saberi Mohamad, Safaai Deris, Safie Mat Yatim, et Muhammad Razib Othman[65]ont implémenté une méthode de sélection des attributs efficaces qui cherche et sélectionne les variables informatives à partir de données de petite ou grande dimension en maximisant la précision de la classification. Dans ce travail, les auteurs ont appliqué un algorithme génétique pour rechercher et identifier les potentiels des caractéristiques informatives pour la classification et ensuite utiliser la précision de la

classification du support machine vecteur pour déterminer l'aptitude dans l'algorithme génétique. Les résultats expérimentaux avec des ensembles de données de référence montrent l'utilité de l'approche proposée pour les données de petite et haute dimension.

❖ Les auteurs K.C. Tan, E.J. Teoh , Q. Yua,, K.C. Goh [67] ont présenté une nouvelle approche hybride comprenant deux algorithmes d'apprentissage pour effectuer la sélection d'attribut : l'algorithmes génétiques (GA) et les machines à vecteurs de support (SVM) basée sur une approche de wrapper. Plus précisément, les composants de GA recherchent le meilleur attribut défini en appliquant les principes d'un processus évolutif. Le SVM classifie ensuite les motifs dans les ensembles de données réduits, correspondant aux sous-ensembles d'attributs représentés par les chromosomes de GA. La Proposition hybride de GA-SVM est ensuite validée en utilisant des ensembles de données obtenues à partir de l'UCI. Les résultats de simulation montrent que l'hybride GA-SVM produit une bonne précision de classification, une plus grande consistance qui est comparable à d'autres algorithmes établis et son potentiel pour être un bon classificateur pour l'avenir de l'exploration de données.

5.5 Aperçu sur l'interface Op-Var :

Les figures suivantes montrent le prototype proposé de notre système (OpVar).

La première fenêtre qui s'affiche si on exécute l'application est présentée dans la figure 3.4

Figure 3.4 : interface principal du modèle Op-Var

1)- Barre de menu : contient les outils suivant :(de gauche à adroite respectivement)
– File : permet de charger la base de données.
–Classification : permet de choisir manuellement le modèle de classification soit sans sélection(SVM) soit avec sélection GA-SVM). Voir figure 3.5

Figure 3.5 : interface pour la création du modèle de classification

– Historique : permet d'afficher les différents résultats obtenus par les deux modèles de classification (cités précédemment).
La figure3.6 : affiche la fenêtre de création de Modèle SVM, elle montre les champs des paramètres utilisés pour ce classifieur (Type de fonction noyau).

- Un bouton "Lancer" pour lancer l'exécution de ce modèle et afficher les résultats obtenus dans le tableau des résultats.
- Un bouton "Sauvegarder" pour enregistrer chaque résultat dans la fenêtre de l'Historique.
- Un bouton '' Nouveau app'' pour vider le tableau et de réinitialiser un nouveau apprentissage en changeant, soit les paramètres, soit la base de donnée.

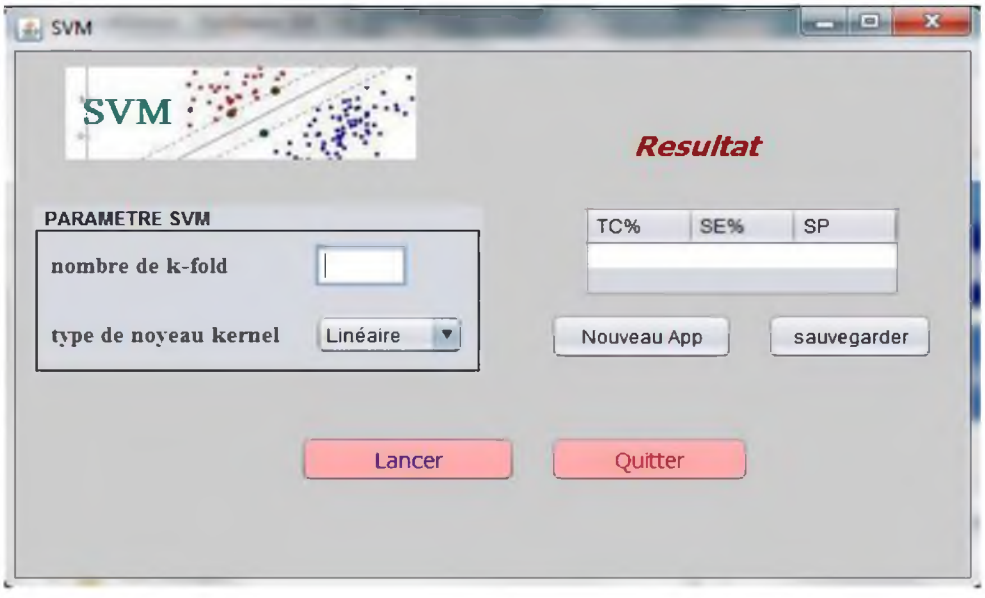

Figure 3.6 : fenêtre de la création du modèle SVM

La figure 3.7 : affiche la fenêtre de création du Modèle GA-SVM. Elle montre les différents
champs des paramètres de l'AG ainsi que le paramètre (fonction noyau) de SVM.

Figure 3.7 : fenêtre du création du modèle GA-SVM

La figure 3.8, permet de visualiser les différents résultats obtenus par les deux méthodes utilisées afin de pouvoir faire une comparaison entre eux en termes de performance de classification.

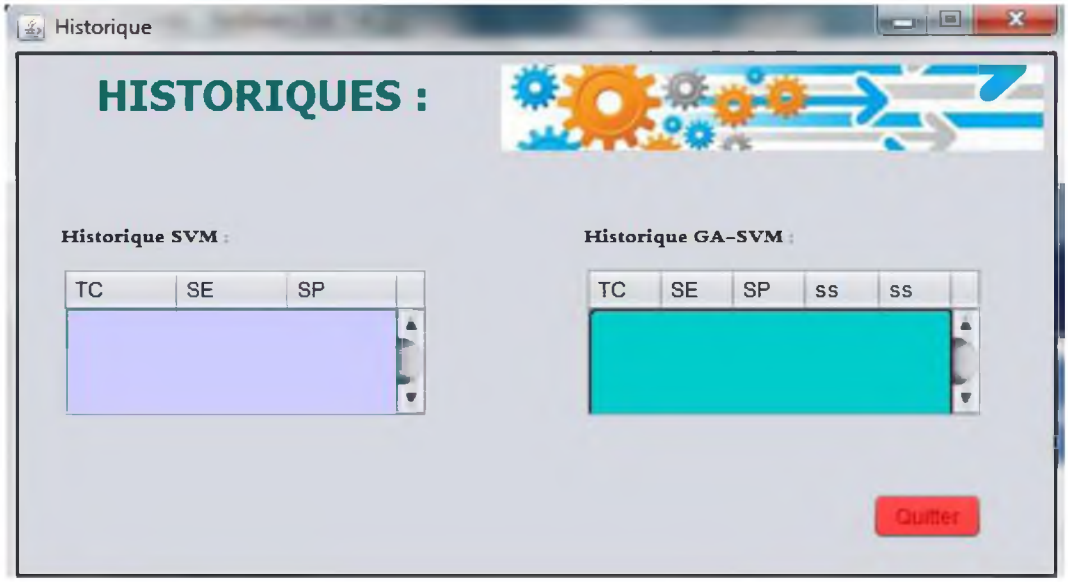

Figure 3.8 : Fenêtre de Visualisation des résultats.

6 Validation Expérimentale

Afin d'évaluer les performances de notre modèle proposé (Op-Var), nous avons utilisé trois jeux de données médicales (cités précédemment), qui sont utilisés dans de nombreux travaux concernant la fouille de données. Ces jeux constituent en quelque

sorte des jeux de tests (benchmark) qui permettent de comparer les méthodes proposées depuis quelques années dans le domaine de classification des données médicales.

6.1 Analyse des résultats

Nous avons effectué plusieurs expérimentations sur l'ensemble de données utilisées, en modifiant les valeurs de chacun des paramètres de l'algorithme génétique, et celle du classifieur SVM. Les résultats de ces expériences sont présentés dans les tableaux en dessous :

- **Pima**

Exp	Taille population	Nbr génér-ation	Nbr d'attributs sélectionnés	Attributs sélectionnés	Erreur	TC %	Se %	Sp %
#1	16	8	5	2- 3- 6-7-8	0.2239	77.61	90.72	90.72
#2	50	20	5	1-2-3-6-7	50	50	50	50
#3	70	40	6	1-2-3-4-6-7	0.1053	**89.47**	92.86	80.00
#4	30	10	5	2-3-6-7-8	0.1699	83.01	93.02	62.33
#5	10	70	6	1-2-3-4-6-8	0.1569	84.31	95.19	61.22
#6	10	7	6	2-3-5-6-7	0.1895	81.05	94.68	59.32
#7	8	5	4	2-4-6-8	0.2026	79.74	85.71	63.41

Tableau 3.6 Résultats obtenus pour la base Pima

D'après le tableau 3.6 on peut observer que :

❖ Le changement de nombre de génération et de taille de population influe sur les résultats obtenus soit en termes de Taux d'erreurs ou de nombre d'attributs sélectionnés.

❖ L'augmentation de valeurs de ces deux paramètres améliore les performances de système.

❖ Le Tc obtenu par les différentes expérimentations sur la même base est atteint pour des sous ensembles d'attributs différents.

❖ Le taux de bonne classification (Tc) varie aléatoirement en fonction du nombre d'attributs sélectionné.

Pour toute les expérimentations on voie que le modèle Op-Var sélectionne presque les 5 variables suivants (2 : Glucose ,3 : tension artérielle diastolique, 6 : index de masse corporelle, 7 : l'hérédité, 8 : age) comme des variables qui ont le pouvoir discriminant entre les deux classes.

En revenant aux nombres d'attributs de la base de donnée Pima(8 caractéristiques) et à la nature de la maladie du diabète nous pouvons confirmer que cette méthode a effectivement trouvé les variables les plus pertinentes ,car le changement de la concentration de glucose est le paramètres le plus utilisé pour le diagnostic de cette maladie ,et si nous revenons aux causes de diabète ,nous remarquons que la majorité des personnes diabétiques souffrent de problèmes de surpoids et que le facteur génétique est le responsable de la plupart des cas atteint la maladie.

Et pour le taux de classification acquise durant l'expérimentation il n'est pas assez bon et cela dû au fait que le vecteur caractéristiques de la base Pima contient que 8 attributs ce qui explique que tous les variables sont presque pertinents et donc la sélection dans ce cas vas nous perdre en terme de précision. Ces résultats montrent que le modèle améliore sa capacité de détection des cas positifs avec presque 92% de succès ce qui a génère une grande amélioration de taux de reconnaissance et une diminution de sa spécificité par rapports à la sensibilité ce qui veut dire que le système a fait une mauvaise détection des cas négatives. Donc beaucoup de patients non diabétiques ont été reconnus comme diabétiques.

- **Heartstatlog**

Exp	Taille population	Générat-ions	Nbr d'att sélectio-nnés	Nbr d'Attributs sélectionnées	Erreur	TC %	Se %	Sp %
#1	100	50	8	1-3-5-8-9-10-12-13	0.0556	94.44	100	87.50
#2	10	30	8	2-3-4-7-9-10-12-13	0.1407	85.93	86.67	83.33
#3	16	5	6	3-7-9-10-12-13	0.1444	85.56	90	83.33
#4	80	70	9	1-2-3-4-7-8-10-12-13	0.0370	**96.3**	97.06	90
#5	60	20	8	1-2-3-5-7-9-12-13	0.0741	92.59	90.63	95.45
#6	20	90	9	1-2-3-4-7-8-10-12-13	0.0741	92.59	93.94	90.48
#7	30	10	7	5-7-8-9-11-12-13	0.0556	94.44	92.31	96.43

Tableau 3.7 Résultats obtenus pour la base Heartstatlog

D'après les expérimentations effectuées sur cette base on peut observer que pour cet ensemble de donnée les paramètres d'algorithme génétique influents sur les résultats ; et pour tous les tests le modèle nous donnera une diminution des caractéristiques avec un taux d'erreur assez petit.
A partir des résultats obtenus, on peut dire que si on diminue le nombre de génération le modèle converge vite mais avec un taux de précision pas performant(E3).
Par contre si on augmente la taille de population et le nombre de génération, la capacité de système augmente (haute performance) mais il est un peu couteux en temps de calcul (E4).

Les variables les plus sélectionnés presque dans toutes les expérimentations sont : 1 : Age, 2 : le sexe, 3 : Douleur thoracique, 4 : La pression artérielle au repos ,7 : Glycémie, 8 : Fréquence cardiaque au repos maximale, 9 : L'exercice, 10 : Vieux pic, 12: nombre de navires principaux colorés par fluoroscopie, 13 : Scintigraphie au thallium.
Dans le coté médicale ces variables sont suffisantes et les plus significatifs d'après les médecins pour savoir si un malade a atteint une maladie du cœur ou non. Pour cette expérimentation La sensibilité de système est très élevé 97% ce qui veut dire que le système a fait un bon apprentissage pour les données positives. Donc si un patient est malade notre modèle le détecte avec beaucoup de succès. Par contre la spécificité est un peu faible 90% ce qui veut dire que 10% des patients non malades ont été reconnus comme malade.

- **Hepatitis :**

D'après les expérimentations effectuées sur la base de donnée hepatitis, on peut remarquer que toujours la taille de population et le nombre de générations effectués jouent un rôle très important. Avec des valeurs minimales de ces deux paramètres, on a arrivé a des bonnes résultats en terme d'attributs sélectionnés et de taux de classification (E6 et E7). On voit que pour plusieurs tests, on a trouvé une spécificité qui est égale à 100%(E1, E3, E4, E6) ce qui veut dire que le système a fait un bon apprentissage pour les données négatives.

Et pour une sensibilité qui varie autour du 95%. Et pour le meilleur résultat(E7) on a trouvé une sensibilité égale à 100% ce qui veut dire que le système a fait un bon apprentissage pour les données positives. L'approche GA-SVM fonctionne bien pour cet ensemble de données par rapport aux autres jeux de données. Dans tous les tests, le système atteint l'optimum dans un temps très réduit.

Exp	Taille population	générations	Nbr d'attsélectionnés	Nbr d'Attsélectionnées	Erreur	TC %	Se %	Sp %
#1	70	10	10	1-2-3-6-10-12-14-15-17-18	0.0645	93.55	92.59	100
#2	10	70	9	2-3-5-6-8-9-11-15-17	0.1290	87.1	88 .46	80.00
#3	20	30	13	1-2-4-5-6-7-8-11-14-15-16-18-19	0.0645	93.55	92.31	100
#4	40	20	10	2-5-6-7-11-13-14-15-17-18	0.0968	90.32	85.00	100
#5	20	30	12	2-5-6-7-9-10-11-14-15-17-18-19	0.1161	88.39	88.89	50
#6	8	5	10	4-5-6-8-9-10-11-14-17-19	0.0323	96.77	96.15	100
#7	15	8	12	1-3-4-5-6-8-9-12-14-17-18-19	0.0323	**96.77**	100	80

Tableau 3.8 Résultats obtenus pour la base Hepatitis

Synthèse :

A partir des Tableaux [6-7-8], nous pouvons constater que notre système Op-Var a produit des résultats significatifs en termes de réduction du nombre des caractéristiques sélectionnées et d'amélioration des performances de classification avec toutes les bases de données utilisées.

- La complexité de l'algorithme de sélection change selon la valeur des paramètres utilisés. Si le nombre de générations est très élevé, cet algorithme est très gourmand en temps de calculs, « Sa complexité est très élevée »
- La taille de la population est l'un des choix les plus importants rencontrés au cours des expérimentations effectuées. Si la taille de la population est trop petite, l'algorithme génétique peut converger trop vite. Par contre, si elle est trop grande, l'algorithme génétique peut gaspiller des ressources de calcul et le temps d'attente pour une amélioration pourrait être trop long. Donc, un choix correct de la population influe sur la vitesse de convergence et orientera la zone des meilleurs résultats dans l'espace de solution.

- L'ensemble d'attributs sélectionnés est spécifique à l'algorithme d'induction utilisé. Donc nous ne pouvons pas garantir que l'ensemble trouvé donne des bons résultats par rapport à d'autres algorithmes, quelque soit le lejeux de données utilisées.

6.2 Etude comparative :

6.2.1-Comparaison des résultats avec et sans sélection :

Dans cette section, nous comparons les résultats obtenus par le classifieur SVM et les meilleurs résultats obtenues par notre modèle Op-Var pour les trois bases de données.
Dans le but de tester et de prouver l'efficacité de l'approche proposée, la comparaison est basée sur le nombre réduit des attributs par rapport à l'ensemble d'attributs d'origine ,en se basant sur le taux de classification .Les résultats expérimentaux sont présentés dans le Tableau 3.9.

Jeu de données	**SVM**		**Op-Var**	
	Nbr d'attributs originales	Taux de classification **%**	Nbr d'attributs réduits	Taux de classification%
Pima	8	76.43	6	89.47
Heart Statlog	13	84.07	9	96.3
Hepatitis	19	78.71	12	96.77

Tableau 3.9 Résultats expérimentaux obtenus par Op-Var et SVM

L'analyse de tableaux montre que la sélection des variables à base de l'AG améliore les performances du classifieur en termes de taux de classification. Aussi, les sous-ensembles des attributs sont réduits pour toutes les bases de données utilisées dans nos expérimentations. Cela implique que la sélection permet effectivement de réduire les attributs inutiles et le bruit pour améliorer le taux de classification.

6.2.2-comparaison des résultats avec les méthodes Relief et Rank-Features :

Pour mieux montrer l'intérêt de notre approche dans la sélection d'attributs, et vu que les Wrappers donnent généralement des meilleurs taux de réussites que les filtrantes, nous essayons de faire une petite comparaison entre nos résultats avec ceux des méthodes filtrantes, en se basant sur un même nombre d'attributs sélectionnés pour chaque base de donnée.
Sachant que les algorithmes filtrants utilisent le principe d'ordonnancement des attributs, ce type d'algorithme produisent un score relatif à chaque attribut « i » pour garder ceux qui présentent les scores les plus élevés (les K attributs les mieux classés).
Pour cette étude, nous avons choisi les deux méthodes Relief et Rank-feature.(voir Tableau 3.10).Après avoir les résultats des différentes méthodes, nous pouvons observer que :

- Le taux de classification varie d'une méthode à l'autre et cela dû à la nature des sous ensemble sélectionné qui se différent d'une méthode à autre.
- En comparant entre les sous-ensembles sélectionnés par les trois techniques, nous observons que presque la majorité des attributs se répètent.

- Notre modèle a réussi de trouver le meilleur sous ensemble qui a donné un Taux de précision performant.
- L'approche wrapper offre un avantage principal du fait qu'elle permet d'explorer au maximum tout le biais de l'algorithme SVM mais l'exécution sera lente puisque les wrappers doivent appliquer le classificateur pour évaluer chaque sous-ensemble d'attributs.

Dans la suite, nous proposons de faire une comparaison avec des travaux importants dans le domaine de la sélection et de la classification des données.

	Les methodes	Taux de classification %	Ordonnancement des attributs	Attributs selectionnés
Pima	Relief_svm	76.72	2.8.6.1.7.3.5.4	5
	Rankfeatures _svm	82.89	2.6.8.1.7	5
	Op var	83.01	2.3.6.7.8	5
Heart statlog	Relief_svm	85.19	12.3.11.13.10.8 4.7.1.6. 9 .5 .2	9
	Rankfeatures _svm	87.04	13.12.3.8.9.10.11.2.1	9
	Op var	96.3	1.2.3.4.7.8.10.12.13	9
Hepatitis	Relief_svm	74.19	12.13.11.6.10.14.15.7.19.8.9.18. 16.1.17.3.2.5.4	10
	Rankfeatures _svm	80.65	17.5.19.11.6.2.12.18.14.13	10
	Op var	96.77	4.5.6.8.9.10.11.14.17.19	10

Tableau 3.10 Comparaison des résultats obtenus par Op-var avec les aproches Relief et Rankfeature

6.2.3 Comparaison avec les résultats de la littérature :

Afin de pouvoir situer les performances de notre modèle proposé, nous avons réalisé une étude comparative entre les résultats obtenus par l'approche Op-Var et celles des travaux déjà réalisés dans ce domaine (Tab : 11-12-13).

A noter que le symbole (–) signifie que ce critère n'est pas traité dans l'article consulté.

Référence	Méthode de sélection	Type de sélection	Taux de classification %	Nbr d'att sélectio-nnés	Attributs sélectionnées
Cheng.Lung Huang[63]	GA.SVM	Wrapper	81.5	4	_
K.C. Tan et E.J. Teoh [67]	GA.SVM	hybride	78.64	4	2.6.7.8
Alaoui Abdiya [3]	ACO/C4.5	Wrapper	75	5	_
	GA/C4.5	Wrapper	75.3	5	_
Sarojini Balakrishnan[69]	FCBF+SVM	Filter	77.99	4	_
Notre Approche	Opvar(GA.SVM)	Wrapper	**89.47**	6	1.2.3.4.6.7

Tableau 3.11 : Synthèse des travaux utilisant la sélection d'attributs pour la base PIMA

référence	Méthode de sélection	Type de la méthode	Taux de classification%	Nbr d'attributs sélectio-nnés	Attributs sélectionnées
K.C. Tan et E.J. Teoh [67]	GA.SVM	hybride	84.07	8	1.3.4.8.9.11.12.13
Alaoui Abdiya [3]	ACO/C4.5	Wrapper	75	5	_
	GA/C4.5	Wrapper	75.3	5	_
Sarojini Balakrishnan[69]	FCBF+SVM	Filter	77.99	4	_
M.A et Asha Karegowda et A.S [71]	GA+ Naïve Bayes	Wrapper	85.87	11	_
Notre Approche	Opvar(GA.SVM)	Wrapper	**96.3**	9	1.2.3.4.7.8.10.12.13

Tableau 3.12 Synthèse des travaux utilisant la sélection d'attributs pour la base Heart Statlog

Référence	Méthode de sélection	Type de la méthode	Taux de classification%	Nbr d'attributs sélectionnés	Attributs sélectionnées
K.C. Tan et E.J. Teoh [67]	GA.SVM	hybride	86.12	15	1,2,6,7,8,9,10 ,11,12,14,15, 16,17,18,19
MENGHOR Kamilia [32]	ACO /Naïve Bayes	Wrapper	61.64	2	5, 7
	PSO /Naïve Bayes	Wrapper	72.4	7	2, 5, 9, 12, 15, 17, 19
S. Anto, S. Chandramathi [70]	GASA.SVM	_	87	10	2,3,4,5,7,8,10 ,11,12,13
Notre Approche	OpVar	Wrapper	96.77	12	1.3.4.5.6.8.9. 12.14.17.18.1 9

Tableau 3.13 Synthèse des travaux utilisant la sélection d'attributs pour la base Hepatitis

Après plusieurs recherche sur la sélection d'attributs pour les bases de données utilisées dans cette étude, nous constatons qu'il existe plusieurs travaux dans ce domaine, même des chercheurs qui ont utilisé la même méthode mais nous remarquons que pour les trois jeux de données, nous avons obtenue des résultats mieux que les autres, cela du au fait de :

- Type de l'approche utilisé (Wrapper)
- Meilleur matériel génétique (réglage des paramètres de configuration de système).

Cependant, cette méthode est très fastidieuse avec un degré d'incertitude (un choix aléatoire des paramètres optimisés par l'expérimentation). Les valeurs de certains paramètres peuvent être plus adaptables à différents types d'ensembles de données, ce qui entraîne une meilleure précision pour certains ensembles de données et pire pour les autres. En outre, il n'y a aucune règle général pour régler les paramètres. En comparant nos résultats avec la méthode de Cheng.Lung Huang [63] (qui a également utilisé la même approche wrapper GA-SVM), nous pouvons conclure que nos paramètres de l'AG sont plus adaptables avec les bases Pima et Heart statlog par rapport aux paramètres (cités ci-dessous) utilisés dans leur étude.

Taille de population =100 ; Probabilité de croisement=0.7 ; Probabilité de mutation=0.02 ;

Approche de Sélection=Roulette ; Approche de remplacement = Elitism.

En comparant nos résultats avec la méthode de K.C. Tan et E.J. Teoh [67] appliquée sur les trois bases de données on constate que l'approche wrapper a un grand avantage pour l'amélioration de capacité de classification. Les résultats de cette analyse comparative

avec d'autres types de méthodes proposées pour la sélection et la classification supervisé ,pour les mêmes jeux de données ,nous a permis de savoir à quel point notre approche est compétitive. Il ressort que l'approche Op-Var est capable de fournir des sous-ensembles optimaux avec une meilleure performance.

7 Conclusion

Le choix des descripteurs pertinents est l'un des défis major du diagnostic automatique médical. Dans ce chapitre, nous nous intéressons à traiter ce problème en utilisant l'approche enveloppante à base de l'algorithme génétique et du classifieur SVM. Dans ce chapitre nous avons présenté et détaillé notre modèle proposé (Op-Var), qui vise à sélectionner le meilleur sous ensemble d'attributs pertinents à partir d'un ensemble de données volumineux. Pour démontrer l'efficacité du système proposé, une série d'expérimentations ont été soulevées sur trois ensembles de données obtenues de l'UCI, avec leurs tests de validation correspondants. D'après les résultats expérimentaux, nous avons pu remarquer que l'approche proposée peut atteindre un taux d'erreur de prédiction considérablement réduit et produisent un petit sous ensemble de caractéristiques. En comparant les résultats obtenus avec d'autres méthodes de la littérature, nous pouvons arriver à la conclusion que le système Op-Var proposé constitue un outil puissant et fiable pour la classification des données médicales.

Conclusion Générale

Dans leurs activités, les ingénieurs et les décideurs sont confrontés à des problèmes de complexité grandissante, en vue de maximiser les bénéfices, minimiser les pertes… Ces problèmes surgissent dans des domaines très divers, comme la conception et l'implémentation des systèmes d'aide à la décision, les réseaux informatiques, le traitement d'images, en robotique, en électronique…

Dans ces dernières années, le volume de toutes sortes de données, croît de plus en plus. Avec cette croissance de données disponibles, il est nécessaire de développer des algorithmes qui peuvent extraire des informations significatives à partir de ce vaste volume de données.

La sélection des variables est l'une des solutions a proposée, pour résoudre ce type de problème. C'est un domaine de recherche qui donne lieu à de nombreuses études et à de nouvelles approches. La sélection des variables (SV) représente aussiune tâche importante dans le processus de fouille de données (Data mining), qui peut être vu comme un problème d'optimisation difficile.

Le travail que nous avons présenté dans ce mémoire est issu de cette problématique. L'objectif est de proposer une contribution concernant la sélection des variables à base des métaheuristiques, dont le but est de mieux situer les descripteurs les plus discriminants et qui permettent d'améliorer les performances du modèle développé (taux de précision élevé).

Le modèle proposé nommé OP-VAR est basé sur une approche « wrapper » qui utilise une stratégie de recherche génétique. Les AG explore chaque sous-ensembles candidats, et permet de les évalué par la suite, grâce à un classifieur à base SVM. Le taux de classification qui indique si le sous-ensemble sélectionné permet une bonne discrimination des classes. Cette information est donc la fonction d'aptitude retenue dans l'AG.

Nous avons étudié les différentes configurations de l'algorithme génétique afin d'en définir les meilleurs paramètres adaptés à notre problématique. Aussi, nous avons réalisé de nombreuses expérimentations afin d'évaluer l'approche OP-VAR, en utilisant trois jeux de données issu de l'UCI (Irvine Machine Learning Repository). Les résultats obtenus sont très encourageants, que ce soit au niveau des variables sélectionnées ou de la précision de classification.

La comparaison de notre modèle Op-Var avec d'autres méthodes de sélection de type filtrante et même d'autre méthodes déjà réalisés dans ce domaine, a mis en évidence que celle-ci rivalise très bien avec les méthodes de référence, du point de vue du taux de classification et du nombre d'attributs sélectionnés. Cette étude confirme encore une fois que la sélection permet effectivement de réduire les données inutiles pour améliorer la performance du classifieur.

En perspective, plusieurs voies de recherches peuvent être suivies pour une continuité de ce travail:

- Utiliser d'autres algorithmes de classification tels que les réseaux de neurones, le plus proche voisin, les arbres de décision, …
- Utiliser d'autres métaheuristiques telles que l'optimisation par essaim particulaires, optimisation par colonie d'abeille…etc.

- Assurer l'interprétabilité des résultats du modèle en intégrant le concept de la logique floue.
- Tester l'approche proposée sur d'autres type d'apprentissage (non supervisé, semi supervisé)
- Utiliser la méthode Op-Var sur des jeux de données multi-labels.
- Appliquer l'approche sur des bases de données volumineuse (base de données biologiques)

ANNEXES

Problème NP- Difficile (NP-Hard) :

Les problèmes NP-hard sont des problèmes d'optimisation pour le quel, leur problèmes de décision associé sont NP-complet. La majorité des problèmes d'optimisation du monde réel sont NP-difficile .Ils nécessitent un temps exponentiel pour les résoudre. Les métaheuristiques constituent une alternative importante pour résoudre cette classe de problème**. [21]** Exemple de problèmes NP-Hard : problème de voyageur de commerce.

Les colonies de fourmis

La méthode de la colonie de fourmis simule le comportement de ces insectes qui, lorsqu'on pose un obstacle sur leur trajet, trouvent toujours le chemin le plus court pour contourner.

Leur technique repose sur la pose de marqueurs chimiques, les phéromones, déposés sur Les trajets parcourus. Cela peut paraître surprenant au premier abord mais un chemin plus court reçoit plus de phéromones qu'un chemin plus long.

Cette métaheuristique a été introduite pour la première fois en 1992 et a été appliquée du voyageur de commerce. Gambardella et alont appliqué cette métaheuristique pour le VRP en 2003[10].

Le pseudo code de la colonie de fourmis est représenté dans l'algorithme **:**

1. **Initialiser** les traces
2. **Tant qu'un** critère d'arrêt n'est pas satisfait
3. **Répéter** en parallèle pour des p fourmis
4. Construire une nouvelle solution à l'aide des informations contenues dans les traces
5. une fonction d'évaluation partielle
6. Evaluer la qualité de la solution
7. Mettre à jour les traces

Revue des différentes méthodes de sélection d'attributs :

Branch and Bound (B&B) : cet algorithme fournit une solution optimale mais à une condition : le critère d'évaluation des sous-ensembles doit être monotone. Cette contrainte limite bien sûr le choix de la mesure de distance. Généralement celles qui sont utilisées sont la distance de Mahalanobis, la distance de Bhattacharya, le critère de Fisher, la fonction discriminante et la divergence. Cette approche couvrant une grande partie de l'espace des solutions, elle nécessite un grand nombre d'opérations. Elle apporte tout de même une réduction par rapport à une approche exhaustive. Cependant, lorsque le nombre de variables devient important (> 30), il n'est tout de même pas raisonnable d'utiliser cette approche**.**

L'algorithme *FOCUS* : fait également partie des méthodes appelées complètes. Contrairement à *B&B* qui utilise une mesure de distance, ce dernier algorithme utilise un critère d'évaluation des variables basé sur la cohérence. L'inconvénient de cet algorithme est qu'il ne fonctionne pas correctement lorsque les données sont bruitées.
De plus il ne peut être utilisé que pour des problèmes de classification à deux classes. Il existe des variantes de cet algorithme ne permettant pas cependant d'abolir ces restrictions.

L'algorithme *Relief* : La procédure de sélection est réalisée par l'intermédiaire d'une pondération des différents attributs. Premièrement l'utilisateur doit fixer le nombre d'échantillons que l'algorithme choisira aléatoirement dans le corpus d'apprentissage. Pour chacun d'eux il recherche au sein de ce sous-ensemble l'échantillon de la même classe le plus proche et celui d'une classe différente le plus proche également. Les poids des attributs sont mis à jour en fonction de ces valeurs. À la fin du processus les attributs sélectionnés sont ceux qui ont une pondération supérieure à un seuil donné. Ce dernier peut être déterminé automatiquement. L'algorithme *Relief* permet une sélection efficace lors de la présence de variables corrélées. Cependant il ne détecte pas la présence d'attributs redondants. Une restriction importante est qu'il ne fonctionne que dans le cas de la présence de deux classes. Konnenko propose une extension aux problèmes multi-classes. Cette version de l'algorithme est appelée *Relief-F*.

SFS (Sequential Forward Selection):

SFS est pour la recherche en avant. Il consiste à ajouter les attributs un à un à partir d'un ensemble vide d'attributs, en prenant à chaque étape celui qui permet de maximiser un critère J. Lorsque le nombre d'attributs attendu est plutôt faible, SFS s'avère nettement moins couteux [3].

SBS (Sequential Backward Selection):

SBS est pour la recherche en arrière. Cet algorithme part de l'ensemble complet d'attributs, et les supprime un à un, à chaque étape. Il supprime celui dont l'absence permet de maximiser un critère J. La génération des successeurs qui est effectuée à chaque itération ne requiert que l'application d'un seul opérateur de base, l'algorithme SBS est performant et conduit vite à des optima locaux. Mais les premières itérations de SBS sont très couteuses. SBS permet d'évaluer l'influence de chaque attribut sur la classe en présence des autres attributs.
L'idée est de choisir le nœud le plus promoteur qui n'a pas été étendu [3].

LVF (Las Vigas Algorithm):

C'est un algorithme probabiliste, pour la sélection d'attributs. LVF fait des choix probabilistes de sous ensembles dans la recherche de l'ensemble optimal, il garde le plus petit sous ensemble d'attributs généré aléatoirement dont le taux d'inconsistance satisfait un seuil. Il est rapide pour faire décroitre le nombre d'attributs.LVF est aveugle et génère des sous ensembles non intéressants, ceci est du au problème de dégradation et il est de plus en plus lent au fur et à mesure qu'il s'approche d'une solution optimale.LVF trouve un sous ensemble d'attributs même s'il y a du bruit dans les données. De plus, l'utilisateur obtient rapidement un bon sous ensemble. **[3].**

Type de sélection pour les GA :

a) La loterie biaisée ou roulette wheel :

Cette méthode est la plus connue et la plus utilisée.
Avec cette méthode chaque individu a une chance d'être sélectionné proportionnelle à sa performance, donc plus les individus sont adaptés au problème, plus ils ont de chances d'être sélectionnés.
Pour utiliser l'image de la "roue du forain", chaque individu se voit attribué un secteur dont l'angle est proportionnel à son adaptation, sa "fitness".
On fait tourner la roue et quand elle cesse de tourner on sélectionne l'individu correspondant au secteur désigné par une sorte de "curseur", curseur qui pointe sur un secteur particulier de celle-ci après qu'elle se soit arrêté de tourner [18].

b) La méthode élitiste :

Cette méthode consiste à sélectionner les n individus dont on a besoin pour la nouvelle génération P' en prenant les n meilleurs individus de la population P après l'avoir triée de manière décroissante selon la fitness de ses individus.
Il est inutile de préciser que cette méthode est encore pire que celle de la loterie biaisée dans le sens où elle amènera à une convergence prématurée encore plus rapidement et surtout de manière encore plus sûre que la méthode de sélection de la loterie biaisée ; en effet, la pression de la sélection est trop forte, la variance nulle et la diversité inexistante, du moins le peu de diversité qu'il pourrait y avoir ne résultera pas de la sélection mais plutôt du croisement et des mutations.
Là aussi il faut opter pour une autre méthode de sélection [18].

c) La sélection par tournois :

Cette méthode est celle avec laquelle on obtient les résultats les plus satisfaisants.
Le principe de cette méthode est le suivant : on effectue un tirage avec remise de deux individus de P, et on les fait "combattre". Celui qui a la fitness la plus élevée l'emporte avec une probabilité p comprise entre 0.5 et 1. On répète ce processus n fois de manière à obtenir les n individus de P' qui serviront de parents.
La variance de cette méthode est élevée et le fait d'augmenter ou de diminuer la valeur de p permet respectivement de diminuer ou d'augmenter la pression de la sélection [18].

Bibliographie

[1] Pr Patrick siarry,conception de métaheuristiques pour l'optimisation dynamique. Application a l'analyse de séquences d'image IRM. Université paris –est. Décembre 2011.

[2]V.gardeux., conception d'heuristiques d'optimisation pour les problèmes de grande dimension. Application a l'analyse de données de puces à ADN, ph.d. Thesis, université de Paris-est Créteil, 2011.

[3] Alaoui abdiya, application des techniques des métaheuristiques pour l'optimisation de la tache de la classification de la fouille de données, université des sciences et de la technologie d'Oran Mohamed boudiaf, faculté des sciences, département d'informatique, thèse soutenu en 2011/2012.

[4] Jin-Kao ha, Philippe galinier, Michel Habib, revue d'intelligence artificielle : métaheuristiques pour l'optimisation combinatoire et l'affectation sous contraintes no. 1999.

[5] Benmaazouz maar, Khouani amine, optimisation paramétrique d'un classifieur neuronale par métaheuristique : application données médicales, université Abou bekr belkaid, Tlemcen, thèse soutenu le 15 juin 2015.

[6] Ilham boussaid. Perfectionnement de métaheuristiques pour l'optimisation continue. Other. Universite paris-est; universite des sciences et de la technologie houari Boumediene (Alger), 2013. French. <Nnt: 2013pest1075>. <Tel-00952774>.

[7] Johann Dreom;patrick siarry,metaheuristiques pour l'optimisation difficile ,imprimé en France, ISBN/2-212-11368-4,Juillet 2003

[8] I.h. Osman and g. Laporte, metaheuristics: a bibliography. Annals of operations research 63, 513-623, 1996.

[9] l. Jourdan, "metaheuristiques coopératives : du déterministe au stochastique. Modeling And simulation," m.s. Thesis, universite des sciences et technologie de Lille - Lille i, 2010.

[10] Kimouche Abdelkader, métaheuristique pour la résolution de problèmes de transport : application transport des patients, universite hadj lakhdar Batna ,17.12.2012 .

[11] S. Vob, s. Martello, i.h. Osman and c. Roucairol (eds), meta-heuristics - advances and trends in local search paradigms for optimization. Kluwer academic publishers, Dordrecht, the Netherlands, (1999).

[12] Glover. Future paths for integer programming and links to artificial intelligence Computers and operations research, 13:533–549, 1986.

[1 3] P. Hansen. The steepest ascent mildest descent heuristic for combinatorial programming In congress on numerical methods in combinatorial optimization, 1986. Capri, Italy.

[14] Mehdi souier, métaheuristiques pour la manipulation de routages Alternatifs en temps réel dans un job shop, universite Abou bakr belkaid Faculté des sciences de l'ingénieur département d'automatique, 2011.

[15] Kirkpatrick, s., gelatt, c. D. And vecchi, m. P.optimization by Simulated annealing, science, volume 220, 4598, p .671-680, 1981.

[16] Laetitia Jourdan. Métaheuristiques pour l'extraction de connaissances: application a la Génomique. Autre [cs.oh]. Universite des sciences et technologie de Lille - Lille i, 2003.Français. <tel-00007983>.

[17] Benkaddour Halima, aribi ramzi, métaheuristiques parallèles pour la résolution des problèmes difficiles, universite kasdi mer bah Ouargla, Le: 06/2013.

[18] Souquet amedee radet François-Gérard, algorithmes génétiques, te de fin d'année tutorat de Mr Philippe audebaud, le 21/06/2004.

[19] Julie Hamon. Optimisation combinatoire pour la sélection de variables en régression en grande dimension : application en génétique animale. Applications [stat.ap]. Universite des sciences et technologie de Lille - Lille i, 2013. Français. <tel-00920205>.

[20] Berkani Abdelhakim, métaheuristique hybride réseaux de neurones artificiels-pso du recuit simule pour la commande d'un procède industriel non-linéaire, universite de Batna faculté de technologie, 2012/2013.

[21] Mlle. Fatima bekaddour, amélioration des performances des classifieurs a base de métaheuristiques, universite Abou-bekr belkaid Tlemcen faculté des sciences département informatique, mai 2014.

[22] El-ghazali talbi, Metaheuristics: from design to implementation. ISBN: 978-0-470-27858-1 .624 pages. July 2009. Wiley [23] Blum c, roli a. Metaheuristics in combinatorial optimization: overview and conceptual comparison. Acm computing surveys 35(3): 268–308, 2003.

[23] Blum c, roli a, Metaheuristics in combinatorial optimization: overview and conceptual comparison. Acm computing surveys 35(3): 268–308 .2003.

[24] Bendahmane amine – master2 rfia le recuit simule ,25 octobre 2011.
[25] Dreo, j. (2004), adaptation de la méthode des colonies de fourmis pour l'optimisation en variables continues. Application en genie biomedical, these de doctorat 2004, universityé paris12.

[26] W. Metropolis, a. Roenbluth, m. Rosenbluth, a. Teller, e.teller, equation of the state calculations by fast computing machines. Journal of chemical physics 21 : 1087-1092, 1953.

[27] Mouhamadou lamine samb, fode Camara, samba ndiaye Yahya slimani, Mohamed amir esseghir, approche de sélection d'attributs pour la classification basée sur l'algorithme rfe-svm, special issue cari'12,2014.

[28] Fraser, a. S. (1957). Simulation of genetic systems by automatic digital computers, Australian journal of biological sciences, 10:484-491.

[29] Nesma settouti, Amel hafa. Approche filtre pour la sélection des gènes pertinents des données biopuces du cancer du colon. 2013. <hal-00843080>.

[30]Bekhti Mohamed anes, la sélection de variables neuronales pour la reconnaissance du diabète, universite Abou bakr belkaid.tlemcen, 2012.

[31] Caroline meynet, sélection de variables pour la classification non supervisée en grande dimension, tel-00752613, version 1 - 16 nov. 2012.

[32] Melle menghour kamilia, approches bio-inspirées pour la sélection d'attributs, universite badji mokhtar-Annaba, année 2014-2015

[33] A. Jain and d. Zongker. Feature selection: evaluation, application, and small sample performance. Ieee trans pattern anal. Mach. Intell., 19(2):153–157, 1997.

[34] Jose crispin Hernandez. Algorithmes métaheuristiques hybrides pour la sélection de gènes et la classification de données de biopuces. Computer science. Universite d'Angers, 2008. French. <tel-00447684>.

[35] Christine tuleau, sélection de variables pour la discrimination en grande dimension et classification de données fonctionnelles, université paris xi u.f.r. Scientifique d'Orsay, 2005.

[36] Ali el akadi , contribution a la sélection des variables pertinentes en classification supervisée :application a la sélection des gènes pour les puces a ADN et des caractéristiques faciales université Mohammed v – agdal faculté des sciences rabat, 31/03/2012

[37]https://quantmetry-blog.com/2015/09/29/le-fleau-de-la-dimension-techniques-de-selection-de-variables.

[38] Mahdjane karima, detection d'anomalies sur les données biologiques par svm, universite mouloud Mammeri de tizi ouzou, 14/10/2012.

[39] M. Deriche, "feature selection using ant colony optimization," in 6th international multi-conference on systems, signals and devices", pp. 1-4, 2009.

[40] X. Wang, j. Yang, xi. Teng, w. Xia et Richard Jensen: feature selection based on rough sets and particle swarm optimization. Journal pattern recognition letters, 2007.

[41] Benahmed n, "optimisation de réseaux de neurones pour la reconnaissance De chiffres manuscrits isoles : sélection et pondération des primitives par Algorithmes génétiques", mémoire de maitrise, universite du Québec, Canada, 2002.

[42] W. Shahzad: classification and associative classification rule discovery using ant colony optimization. These, Pakistan. 2010.

[43] Liu h., yu l. "toward integrating feature selection algorithms for classification and clustering", IEEE transactions on knowledge and data engineering, vol. 17, no 4, pp 491–502. 2005.

[44] D. Bell et h. Wang: formalism for relevance and its application in feature subset selection. Machine learning, pages 175–195, 2000.

[45] J. H. Gennari, p et d fisher: models of incremental concept formation. Artificial intelligence, pages 11-61, 1989.

[46] Kohavi, r., & john, g. (1997). Wrappers for feature selection. Artificial intelligence, 97(1-2), 273-324.

[47] D. Koller et m. Sahami: toward optimal feature selection. 13th international conference on machines learning, pages 1–15, 1996.

[48] Dash, m., & liu, h. (1997). Feature selection for classification. Intelligent data analysis, 1(3).

[49] Kabir m.m., shahjahan m., murase k. "involving new local search in hybrid genetic algorithm for feature selection", in: leung, c.s., lee, m. and chan, j.h. (eds.): "iconip", part ii, vol. 5864, pp 150–158. 2009.

[50] Zhang p, verma b, kumar k., "neural vs. Statistical classifier in conjunction with genetic algorithm feature selection in digital mammography". In proc. Congress on evolutionary computation (cec- 2003), vol 2, pp 1206 – 1213, 8-12 déc 2003.

[51] Kabir m.m., shahjahan m., murase k, "a new hybrid ant colony optimization algorithm for feature selection", expert systems with applications, vol. 39, pp. 3747–3763. 2012

[52] Wang, x., yang, j., peng, n. And teng, x. "finding minimal rough set reducts with particle swarm optimization". Rough sets, fuzzy sets, data mining, and granular computing, lncs vol. 3641, pp 451-460, 2005.

[53] Liu y., qin z., xu z., he x. "feature selection with particle swarms". In Zhang j., he j.-h. Et y. Fu, (eds): "computational and information Science", lncs, vol. 3314, pp 425–430. 2004.

[54]Prasartvit t., kaewkamnerdpong b., et achalakul t. "dimensional Reduction based on artificial bee colony for classification problems". D. - S. Huang et al. (eds.): icic 2011, lnbi 6840, pp. 168–175, 2012.

[55] Chuang, l.y., chang, h.w., tu c.j., yang c.h. "improved binary pso for Feature selection using gene expression data", computational biology and Chemistry, vol. 32, no. 1, pp 29–38. 2008.

[56] Deng t., yang c., zhang y., Wang x. “an improved ant colony Optimization applied to attributes reduction”, in cao b., zhang c. Et li t. (eds.): “fuzzy info. And engineering”, advances in soft computing vol.
54, springer, pp. 1–6. 2009.

[57] Eberhart, r. C. And kennedy, j. (1995). New optimizer using particle swarm theory, proceedings of the 6th international symposium on micro machine and human science, pp. 39–43, Nagoya, Japan.

[58] V. N. Vapnik, “the nature of statistical learning theory”. New York, ny, USA: springer-verlag New York, inc., 1995.

[59] http://archive.ics.uci.edu/ml/datasets/heart+disease

[60] C. Fernandez-Lozano, C. Canto, M. Gestal, j. M. Andrade-garda, j. R. Rabuñal, j. Dorado, and a. Pazos, Hybrid model based on genetic algorithms and svm applied to variable selection within fruit juice classification, the scientific world journal 2013

[61] Edmundo bonilla huerta, Beatrice duval, and jin-kao hao, a hybrid ga/svm approach for gene selection and classification of microarray data, leria, university d’Angers, 2 boulevard Lavoisier, 49045 angers, france,2006.

[62] Li zhuo , jing zheng , fang Wang , xia li , bin ai , junping qian , a genetic algorithm based wrapper feature selection method for classification of hyper spectral images using support vector machine, the international archives of the photogrammetry, remote sensing and spatial information sciences. Vol. Xxxvii. Part b7. Beijing 2008.

[63] Cheng-lung huang , Chieh-jen Wang , a ga-based feature selection and parameters optimization for support vector machines, expert systems with applications 31 (2006) 231–240.

[64] Khyati k. Gandhi, prof. Nilesh b. Prajapati, study of diabetes prediction using feature Selection and classification, international journal of engineering research & technology (ijert), vol. 3 issue 2, February – 2014.

[65] Mohd saberi Mohammad, safaai deris, safie mat yatim, Muhammad razib Othman, Feature selection method using genetic algorithm for the classification of small and high dimension data, first international symposium on information and communications technologies. October 7-8, 2004. Putrajaya, Malaysia.

[66] Aishwarya and anto , a medical expert system based on genetic algorithm and extreme learning machine for diabetes disease diagnosis, international journal of science, engineering and technology research (ijsetr), volume 3, issue 5, may 2014.

[67] K.c. Tan ,e.g. Teoh , q. Yu, k.c. Goh , a hybrid evolutionary algorithm for attribute selection in data mining, expert systems with applications 36 (2009) 8616–8630

[68] K.c. Tan, e.g. Teoh, q. Yua, k.c. Goh, a hybrid evolutionary algorithm for attribute selection in data mining, 2008 published by Elsevier ltd.

[69] R.n sarojini balakrishnan,'' features selection using fcbf in type ii diabetes databases'', international conference on it to celebrate s.charmonman's 72nd Birthday, Thailand, pp.50-58, 2009.

[70] S. Anto, s. Chandramathi, an expert system based on svm and hybrid ga-sa optimization for hepatitis diagnosis, international journal of computer engineering in research trends volume 2, issue 7, July 2015, pp 437-443.

[71] M.a.jayaram, asha gowda karegowda, a.s. Manjunath,feature subset selection problem using wrapper approach in supervised learning, 2010 international journal of computer applications (0975 – 8887)volume 1 – no. 7.

ملخص :

يعد اختيار المتغيرات، موضوع بحث نشط جدا، و يشمل هذا عدة مجالات مختلفة مثل التعلم الآلي، استخراج البيانات وتحليل المعطيات في المعلوماتية الحيوية. إن البحث عن مجموعات فرعية للسمات الأساسية يعتبر إشكالية معقدة يمكن حلها باستعمال نظام تحسين يعتمد على meta-heuristic

في رسالة الماستر هذه نقترح نموذج ملفوف لاختيار المتغيرات ذات الجودة العالية في التعلم الموجه اسمه Op-Var، والذي يعتمد على الجمع بين تقنية من تقنيات meta-heuristic و هي الخوارزمية الجينية بالإضافة إلى مصنف المعلومات (Svm). تم تقييم أداء نظامنا على قواعد بيانات طبية معروفة مثل (السكري، أمراض القلب، والتهاب الكبد). تشير النتائج المتحصل عليها إلى أن النموذج المقترح ينبأ بنتائج جيدة.

الكلمات المفتاحية : اختيار المتغيرات ،تحسين ،تصنيف، الخوارزمية الجينية،SVM, metaheuristic.

Résumé

La sélection des variables est un sujet de recherche très actif dans différents domaines tel que l'apprentissage artificiel, la fouille de données et l'analyse de données en bio-informatique.Cette recherche d'un sous ensemble d'attributs pertinents est un problème d'optimisation qui peut être résolu par les méta-heuristiques.Dans le cadre de ce mémoire de master, nous proposons une approche enveloppante pour la sélection des variables pertinentes en classification supervisée. Le système développé, nommé Op-Var (OPtimisation des VARiables), est basé sur la combinaison d'une métaheuristique à base d'algorithme génétique et d'un classifieur SVM (Support Vector Machine). Les performances de notre système ont été évaluées sur trois jeux de données de l'UCI (Pima, heartstatlog,hepatitis). Les résultats obtenus indiquent que le modèle proposé est très prometteux.

Mots clés :Sélection des variables, Métaheuristique, Algorithme génétique, SVM, Problème d'optimisation, classification.

Abstract

Feature selection is still a very active research topic, and has been widely applied to many fields such as machine learning, data mining and data analysis in bio-informatic. The research of a subset of relevant attributes is an optimization problem that can be solved using metaheuristics. In this paper, we propose a wrapper approach for features selection problem resolution, in the context of supervised classification. The developed system, named Op-Var (Variables Optimization), is based on the combination between genetic algorithm metaheuristic and the SVM (Support Vector Machine) classifier. The performance of our system was evaluated on three UCI datasets (diabetes, heartstatlog, hepatitis). The results indicate that the proposed model is very promising.

Key Words: feature selection, metaheuristic, genetic algorithm, SVM, optimization, classification.