MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE ABDELHAMID IBN BADIS – MOSTAGANEM

Faculté des Sciences Exactes et d'Informatique

Département de Mathématiques et informatique

Filière : Informatique

MEMOIRE DE FIN D'ETUDES

Pour l'Obtention du Diplôme de Master en Informatique

Option : **Ingénierie des Systèmes d'Information**

Présenté par :

« **Messaoudene Souad** »

« **Saad Azzouz Touatia** »

THÈME:

Efficient Algorithm To Improve Feature Selection Accuracy

Soutenu le : 03/07/2022

Devant le jury composé de :

Dr M. MOUSSA	Université de Mostaganem	Président
Dr F. HASSAIN	Université de Mostaganem	Examinateur
Dr B. MEROUFEL	Université de Mostaganem	Encadreur

Année Universitaire **2021/2022**

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 bioinformatique. 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 développons un algorithme de sélection de caractéristiques efficace qui permet d'obtenir une précision de prédiction élevée en apprentissage automatique. Tâches pour résoudre ce problème. Notre stratégie est une sélection de filtre qui peut être appliquée sur un ensemble de données numériques et elle fonctionne dans les deux cas : supervisé et non supervisé

Mots-clés: Sélection des variables, Méta-heuristique, sélection de filter.

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-informatics. The research of a subset of relevant attributes is an optimization problem that can be solved using metaheuristics.

As part of this master's thesis, we develop an efficient feature selection algorithm that achieves high prediction accuracy in machine learning. Tasks to solve this problem. Our strategy is a filter selection that can be applied on a numerical dataset and it works in both cases: supervised and unsupervised

Keywords: feature selection, Metaheuristics, filter selection.

Liste des figures

Liste des Tableaus

Liste des algorithmes

Liste des abréviations

Abréviation	Expression Complète	Page
IA	Intelligence Artificial	6
ML	Machine Learning	7
K-NN	K-plus proches Voisins	8
AD	Arbres de Décision	9
TALN	Traitement Automatique de Langage Naturels	15
SFS	Sequential Forward Selection	24
SBS	Sequential Backward Selection	24
K-PPV	k-plus proches voisins	27
FsbO	Feature Selection based on Originality	29
OS	Originality Score	36
HTML	Hyper Text Markup Language	40
ANS	Apprentissage non supervisée	43
AS	Apprentissage supervisée	45

Table des matières

Introduction Générale

L’apprentissage automatique fait référence au développement, à l'analyse et à l'implémentation de méthodes qui permettent à une machine d'évoluer, dans la résolution d'une catégorie de problèmes, grâce à un processus d'apprentissage. Le but de l'apprentissage automatique, et en particulier la classification, est de résoudre automatiquement des problèmes complexes par la prise de décision sur la base des échantillons de ces problèmes.

Un système de classification permet d'extraire les points communs d'un ensemble d'objets en formant des classes qui partagent des caractéristiques similaires. La complexité de cette tache s'est fortement développée ces deux dernières décennies lorsque les masses de données disponibles ont vu leur volume exploser .En effet, non seulement le nombre des échantillons dans les bases de données a fortement augmenté, mais également la taille de leur description. la représentation de ces échantillons permet de convertir les données réelles (mesures physiques, réponse à un stimulus,...) dans un format propre à leur utilisation, a fin de ressortier des descripteurs pour représenter ces données. Cependant, les descripteurs fournissent souvent des données de grandes dimensions et la classification de telles données est un problème difficile.

La réduction de dimensionnalité via l'extraction et la sélection d'attributs est une étape fondamentale dans le traitement des données qui peut influer considérablement sur la performance du système de classification.

La sélection d’attributs est un sujet de recherche très actif et une étape de traitement qui permet de trouver les attributs pertinents, les plus intéressants et les plus importants afin de résoudre un problème donné.

L'objectif ultime de la sélection des attributs est de réduire la quantité de données en minimisant le nombre des attributs utilisé. Ainsi elle présente divers avantages : elle facilite l'acquisition des données, leur gestion et elle réduit le temps d'apprentissage. D'autre part, elle permet de mieux comprendre les résultats obtenus par un système basé sur les attributs choisis en pointant le lien entre le sous-ensemble d'attributs et le résultat attendu.

Dans ce mémoire, notre but est de présenter en détail l'importance de la sélection des attributs et les différentes étapes du processus de sélection. Nous présentons aussi les différents algorithmes de sélection des attributs existant dans la littérature.

Dans ce contexte, ce rapport est organisé en quatre chapitres

- Le premier chapitre : est consacré à l’apprentissage automatique qui présente le contexte de notre travail.
- Le deuxième chapitre : consiste à présenter en détaille la sélection des attributs et ses différentes techniques.
- Le troisième chapitre : consiste à présenter notre approche FsbO (Feature

Selection based on Originality).

- Le quatrième chapitre : est consacré à l'implémentation et présentation des Résultats.

Enfin, nous terminerons par une conclusion générale.

Chapitre 01
Apprentissage Automatique

1.1. Introduction

L'apprentissage automatique est une forme de l'intelligence artificielle, il existe plusieurs méthodes en apprentissage automatiques que ce soit pour la régression ou la classification. Pour bien choisir une méthode il faut comprendre les fondements de ces méthodes existantes et de ce qui permet de les distinguer afin de déterminer les modèles qui traiteraient au mieux un problème particulier.

Dans ce premier chapitre nous adressons le domaine de l'apprentissage automatique et nous décrivons brièvement les algorithmes d'apprentissage.

1.2. Définition de l'Apprentissage automatique

L'apprentissage automatique, également appelé apprentissage machine ou apprentissage artificiel et en anglais machine Learning, est une forme d'intelligence artificielle (IA) qui permet à un système d'apprendre à partir des données et non à l'aide d'une programmation explicite. Cependant, l'apprentissage automatique n'est pas un processus simple. Au fur et à mesure que les algorithmes ingèrent les données de formation, il devient possible de créer des modèles plus précis basés sur ces données. Un modèle de machine Learning est le résultat généré lorsque vous entraînez votre algorithme d'apprentissage automatique avec des données. Après la formation, lorsque vous fournissez des données en entrée à un modèle, vous recevez un résultat en sortie. Par exemple, un algorithme prédictif crée un modèle prédictif. Ensuite, lorsque vous fournissez des données au modèle prédictif, vous recevez une prévision qui est déterminée par les données qui ont servi à former le modèle [1].

Exemple 1 supposons que l'on dispose d'une collection d'articles de journaux. Comment identifier des groupes d'articles portant sur un même sujet ?

Exemple 2 supposons que l'on dispose d'un certain nombre d'images représentant des chiens, et d'autre représentants des chats. Comment classer automatiquement une nouvelle image dans une des catégories « chien » ou « chat » ?

Exemple 3 supposons que l'on dispose d'une base de données regroupant les caractéristiques des logements dans une ville : superficie, quartier, étage, prix, année de construction, nombre d'occupants, montant des frais de chauffage. Comment prédire la

facture de chauffage à partir des autres caractéristiques pour un logement qui n'appartiendrait pas à cette base ?

Trois grandes approches relèvent de l'apprentissage automatique : l'apprentissage supervisé, l'apprentissage non supervisé, et l'apprentissage par renforcement. Bien entendu, cette classification est sujette à discussion, l'apprentissage semi-supervisé ou l'apprentissage faiblement supervisé (par exemple) apparaissant aux interfaces de ces approches. Dans l'exemple 1, on cherche à regrouper les articles portant sur un même sujet, sans disposer d'exemples d'articles dont on sait a priori qu'ils portent sur ce sujet, et sans connaître à l'avance les sujets à identifier. On parlera donc de problème d'apprentissage non supervisé. Dans les exemples 2 et 3, on cherche à prédire une caractéristique qui est soit une catégorie (exemple 2), soit un montant de facture (exemple 3), à partir d'exemples pour lesquels on connaît la valeur de cette caractéristique. Il s'agit de problèmes d'apprentissage supervisé. Avant de détailler apprentissage supervisé ou non supervisé, concentrons-nous sur la notion de données [1].

1.3. Les types d'apprentissage

L'apprentissage automatique (Machine Learning) est utilisé en intelligence artificielle et en science et analyse des données (Analytics and Data Science). Il existe différents types d'apprentissage automatique et chaque type a de plusieurs algorithmes [11] selon la figure 1:

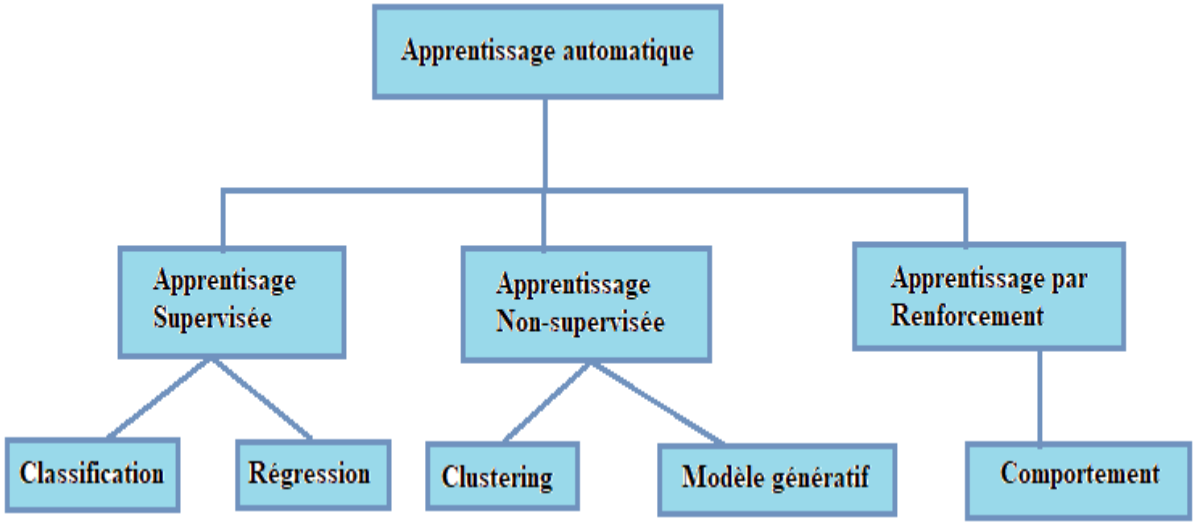


Figure 1 - les types d'apprentissage [11]

1.3.1. Apprentissage supervisée

L'apprentissage supervisé consiste à utiliser un ensemble de données pour prédire des événements futurs statistiquement probables, c'est-à-dire qu'il forme un modèle de prédiction à partir des évènements déjà prédits auparavant. Les modèles de ML peuvent être utilisées dans des applications de « prédiction » ou de « classification ». On distingue deux types de problèmes d'apprentissage supervisé [3] :

1.3.1.1. Classification

Les méthodes de classification s'appliquent lorsque l'ensemble des valeurs résultats est discret. Ceci revient à attribuer une classe (aussi appelée étiquette ou label) pour chaque valeur d'entrée. Les techniques de classification peuvent être basées sur des hypothèses probabilistes (exemple, naïf bayésien), des notions de proximité (exemple, k plus proches voisins) ou des recherches dans des espaces d'hypothèses (exemple, arbres de décision) [3].

Le choix de la technique convenable est important ; il faut pouvoir choisir la méthode la plus adaptée qui sera capable de séparer au mieux les données d'apprentissage.

1.3.1.1.1. Naïf Bayésien

La classification naïve bayé sienne repose sur l'hypothèse que toutes les caractéristiques sont conditionnellement indépendantes les unes des autres. Cette méthode est basée sur le théorème de Bayes qui calcule la probabilité d'un événement à l'aide de la connaissance au préalable des conditions connexes. Ce théorème a été découvert par un statisticien anglais, Thomas Bayes, au 18ème siècle mais il n'a jamais publié son travail. Après son décès, ses notes ont été éditées et publiées par le mathématicien Richard Price [4]. Le théorème est donné par la formule suivante (voir formule (F1)) [4]:

$$P(A\backslash B) = \frac{P(B\backslash A)\,P(A)}{P(B)} \qquad \text{(F1)}$$

- A et B sont des événements.
- $P(A)$ est la probabilité d'observer l'événement A
- $P\,(B)$ est la probabilité d'observer l'événement B.
- $P(A\backslash B)$ est la probabilité conditionnelle d'observer A, sachant qu'un autre événement B de probabilité non nulle s'est réalisé.

1.3.1.1.2. k-plus proches voisins

Parmi les algorithmes d'apprentissage automatique les plus basiques, le k-plus proche voisin, souvent abrégé en K-NN où k est un entier positif. En Data Science, cet algorithme est largement utilisé pour les problèmes de classification des données. Mais avant de se lancer dans cette méthode, il faut savoir que les calculs peuvent s'avérer très couteux en temps de calcul, ainsi, les données doivent être prétraitées. Cette méthode peut être également utilisée dans les problèmes de régression.

Prenons par exemple le problème de classification suivante (dans figure 2): dans le diagramme ci-dessous, il y a des objets ronds verts et des objets carrés bleus. Ceux-ci appartiennent à deux classes différentes : la classe des ronds et la classe des carrés. Lorsqu'un nouvel objet est inséré dans l'espace - dans ce cas, un cercle rouge - nous voulons que l'algorithme d'apprentissage automatique classe le cercle dans une certaine classe [5]

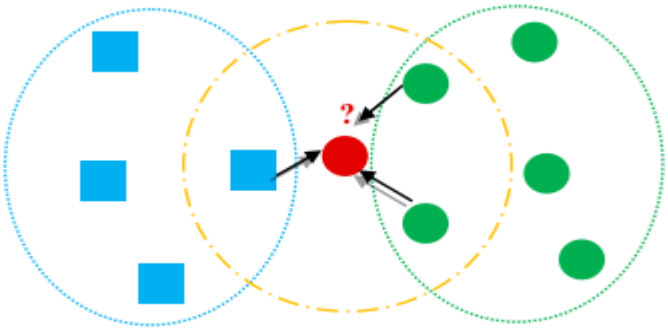


Figure 2 – exemple de classification par la méthode des k-NN [5].

Si on choisit k = 3, l'algorithme cherche les trois plus proches voisins du cercle rouge pour pouvoir le classer soit dans la classe des cercles, soit dans la classe des carrés. Dans ce cas, les trois plus proches voisins du cercle rouge sont un carré et deux cercles. Par conséquent, l'algorithme classera la sphère dans la classe des cercles.

Dans la méthode de k-NN, le résultat est l'appartenance à une classe. L'algorithme stocke tous les cas disponibles et classe toute nouvel objet en vérifiant ses k-plus proches voisins. Ensuite, l'objet est attribué à la classe avec laquelle il a le plus en commun [5].

1.3.1.1.3. Arbres de décision

Les arbres de décision (AD) sont un outil de classification très utilisé. Son principe repose sur la construction d'un arbre de taille limitée [3]. Considérons l'exemple simple représenté ci- dessous dans la figure 3 :

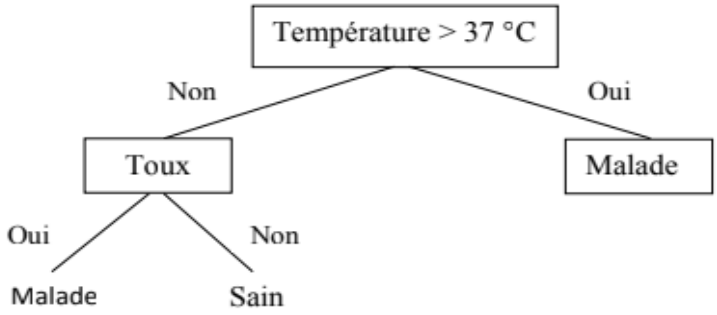


Figure 3 – schéma d'un arbre de décision [3]

Dans cet exemple, le sommet « température > 37 °C » représente la racine. Les feuilles correspondent aux classes ou décisions (appelées aussi nœuds terminaux), ici « malade » ou « sain ». De plus, on appelle « Température > 37 °c » et « toux » les attributs ou les variables (appelés aussi nœuds intermédiaires) [3].

Les AD jouent un rôle très important dans ML. Ils sont capables de gérer les variables continues et discrètes et fournissent par la suite une indication claire pour la prédiction ou la classification sans effectuer beaucoup de calcul. Les AD sont si simples à comprendre et à interpréter, qu'ils permettent une meilleure approximation quelle que soit la complexité des données. L'arbre le plus simple est souvent le meilleur, à condition que tous les autres arbres possibles produisent les mêmes résultats. Leur construction se réalise en divisant l'arbre du sommet vers les feuilles (du haut vers le bas) en choisissant à chaque étape une variable de

séparation sur un nœud d'où les critères de segmentation. Pour cela, les algorithmes les plus utilisés sont « la classification et arbre de régression » et « le Dichotomiseur itératif 3 » [4].

1.3.1.2. La régression

Les méthodes de régression s'appliquent lorsque le résultat que l'on cherche à estimer est une valeur continue. En ML, la régression est un outil important de l'apprentissage supervisé pour la modélisation et l'analyse des données. Elle est notamment utilisée en statistiques et en économie.

1.3.1.2.1. Régression linéaire

On appelle modèle de régression tout modèle capable à établir une relation linéaire entre une variable, dite expliquée ou dépendante, et une ou plusieurs variables, dites explicatives ou variables indépendantes. Le but principal c'est d'ajuster une meilleure droite représentée par une équation linéaire $Y = f(X) + \varepsilon$ afin de prédire $\hat{Y} = \hat{f}(X)$ pour une valeur de X quelconque [6].

− Y représente les variables dépendantes.

−X représente les variables indépendantes.

−ε représente le terme d'erreur ou perturbation.

La régression linéaire est principalement divisée en deux catégories : la régression linéaire simple et la régression linéaire multiple. La régression linéaire simple est caractérisée par une variable indépendante. Par contre, la régression linéaire multiple est caractérisée par au moins de deux variables indépendantes.

1.3.1.2.2. Régression logistique

La régression logistique est une technique permettant d'ajuster une surface de régression à des données lorsque la variable dépendante est dichotomique. Il s'agit en fait de connaitre les facteurs associés à un phénomène en élaborant un modèle de prédiction [7]

La régression logistique peut être binaire ou multinomiale. La régression logistique binaire traite des situations dans lesquelles le résultat observé pour une variable dépendante ne peut avoir que deux types possibles, par exemple « malade » ou « sain », ces deux possibilités sont étiquetées par « 0 » et « 1 ». La régression logistique multinomiale concerne les situations dans lesquelles le résultat peut avoir trois types possibles ou plus qui ne sont pas ordonnés, par exemple « maladie A » par rapport à « maladie B » par rapport à « maladie C » [7].

La régression logistique cherche à :

- modéliser la probabilité qu'un événement se produise en fonction des valeurs des variables indépendantes, qui peuvent être catégoriques ou numériques.
- estimer la probabilité qu'un événement se produise pour une observation choisie au hasard par rapport à la probabilité que l'événement ne se produise pas.
- prédire l'effet d'une série de variables sur une variable à réponse binaire.
- classer les observations en estimant la probabilité qu'une observation soit dans une catégorie particulière.

1.3.2. Apprentissage non supervisée

Dans l'apprentissage non supervisé il n'y a pas de valeurs de sortie, il s'agit de trouver des structures cachées à partir d'un ensemble de données qui doit être regroupé d'où le terme «clustering ». Le but de ce type d'apprentissage est de séparer les données en groupes ou en catégories.

1.3.2.1. Clustering

Le clustering est une technique d'apprentissage automatique non supervisé, utilisé pour le regroupement des données non étiquetées dans de nombreux domaines. Si on dispose d'un nombre fini de points de données et on cherche à les classer dans des groupes de sorte que chaque groupe contient des points de données ayant des propriétés et/ou caractéristiques similaires. Le problème principal qui se pose dans ces algorithmes c'est le choix des propriétés à prendre en compte au cours du regroupement [15]. L'un des algorithmes de clustering les plus utilisés est le « K-Means ».

1. K-Means

C'est l'algorithme de classification le plus connu. Son principe est simple, facile à comprendre et à implémenter dans un code. Tout d'abord, on sélectionne un certain nombre de groupes puis, aléatoirement, on initialise le centre associé à chaque groupe. Il est préférable de commencer par analyser globalement les données présentes et essayer d'identifier des groupes distincts afin de mieux déterminer le nombre de classes à utiliser.

Chaque point est classé en calculant la distance entre ce point et le centre de chaque groupe. Par conséquent, le point sera déplacé vers le groupe dont le centre est le plus proche. Pour chaque classe, un nouveau centre est calculé comme étant la moyenne de tous les points de ce groupe. Après chaque itération, les centres se déplacent lentement et la distance totale entre chaque point et le centre qui lui est attribuée devient de plus en plus petite. Les étapes ci-dessus seront répétées pour un nombre défini d'itérations ou jusqu'à ce que les centres du groupe ne changent plus. K-Means rassure la convergence vers un optimum local. Cependant, cela ne doit pas nécessairement être la meilleure solution globale (optimum global), pour cette raison, on peut initialiser plusieurs fois les centres du groupe de manière aléatoire, puis sélectionner le cycle qui donne les meilleurs résultats [8] voir la figure 4 :

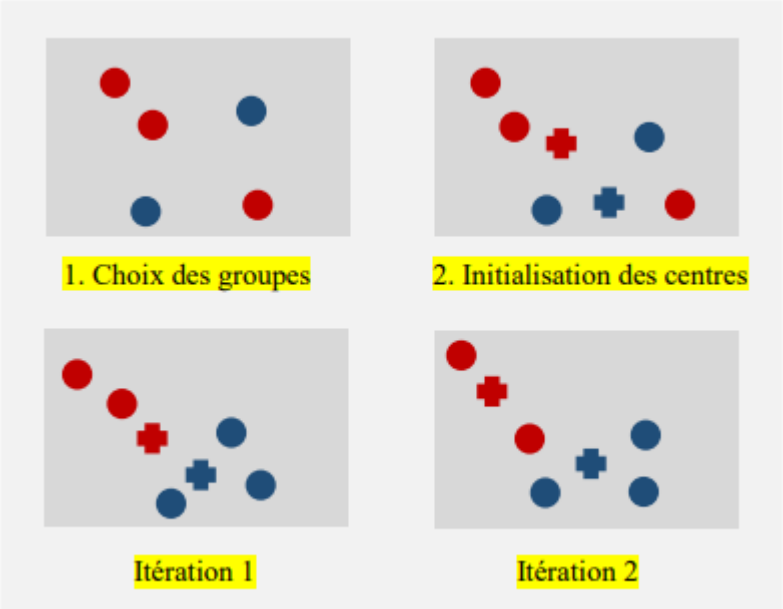


Figure 4 – Exemple de classification par la technique des K-Means avec K=2 [8].

1.3.2.2. Réduction de dimension

La réduction de dimension consiste à récupérer des données d'un espace de grandes dimensions, et à les remplacer par des données dans un espace plus restreint. En machine Learning, les grandes dimensions nuisent à l'efficacité des systèmes d'apprentissage automatique. On parle de fléau de la dimension, avec la production de résultats en trop grand nombre, difficile à associer ou comparer, sans compter le temps nécessaire pour traiter ces quantités de données [14].

La réduction de dimension permet, en réduisant le nombre de variables dans les données d'apprentissage, de mieux visualiser les données obtenues pour d'effectuer des comparaisons et analyses plus fiables.

La réduction de la dimension permet une amélioration de la machine Learning, en construisant des modèles plus simples, où les variables inutiles ont été écartées. Le paramétrage se révèle de fait plus efficace, en limitant les erreurs qui pourraient survenir avec des caractéristiques de départ non pertinentes [14].

Afin de réduire la dimension, il existe différentes méthodes, comme :

1. La sélection des attributs

La sélection des attributs est une technique de réduction de dimension très importante dans l'apprentissage car elle peut être utilisée pour améliorer les performances d'un modèle d'apprentissage ou bien pour faire l'apprentissage lui-même. Dans le chapitre suivant, nous avons focalisé sur cette technique (sélection des attributs) et nous avons donné les différents concepts et stratégies de sélection.

1.3.3. Apprentissage par renforcement

L'apprentissage par renforcement permet d'analyser et d'optimiser le comportement d'un agent en fonction du retour d'informations de l'environnement. Les machines essaient différentes situations pour déterminer les actions les plus avantageuses, plutôt que de simplement recevoir des instructions sur les actions à entreprendre [2].

Ce qui distingue l'apprentissage par renforcement des autres techniques, ce sont l'apprentissage par essais et erreurs et la récompense différée. L'apprentissage par renforcement est un modèle d'apprentissage comportemental.

L'algorithme reçoit des informations grâce à l'analyse des données, de sorte que l'utilisateur est orienté vers le meilleur résultat. L'apprentissage par renforcement diffère des autres types d'apprentissage supervisé, car le système n'est pas entraîné à partir d'un ensemble de données : il apprend par essais et erreurs [2].

Par conséquent, une série de décisions a pour effet de « renforcer » le processus, car celui-ci convient le mieux pour résoudre le problème.

L'apprentissage par renforcement est utilisé par exemple pour les voitures autonomes. Entraîner une voiture autonome constitue un processus extrêmement complexe en raison des nombreux obstacles possibles. Si toutes les voitures étaient autonomes, les essais et les erreurs seraient plus faciles à surmonter. Dans le monde réel, cependant, les facteurs humains sont souvent imprévisibles.

Même dans une situation aussi complexe, l'algorithme peut être optimisé au fil du temps pour trouver des moyens de s'adapter à l'état où les actions sont récompensées.

Pour comprendre l'apprentissage par renforcement, pensons au dressage d'un animal afin qu'il agisse de telle ou telle façon en fonction des récompenses qu'on lui donne. Si un chien reçoit une friandise chaque fois que son maître lui demande de s'asseoir, cette action deviendra alors chez lui automatique.

Dans l'apprentissage par renforcement on a plusieurs algorithmes, parmi ces algorithmes:

1.3.3.1. Markov Décision Processus

Les processus décisionnels de Markov sont définis comme des processus stochastiques contrôlés satisfaisant la propriété de Markov, assignant des récompenses aux transitions d'états [9].

1.3.3.2. Brute force

Une attaque par force brute (brute force Attac) consiste à tester, l'une après l'autre, chaque combinaison possible d'un mot de passe ou d'une clé pour un identifiant donné affin se connecter au service ciblé.

Il s'agit d'une méthode ancienne et répandue chez les pirates. Le temps nécessaire à celle-ci dépend du nombre de possibilités, de la vitesse que met l'attaquant pour tester chaque combinaison et des défenses qui lui sont opposées [10].

1.4. Les étapes d'apprentissage automatique

Il existe 5 étapes de base pour effectuer une tâche d'apprentissage automatique, voir la figure 5:

1. Collecte de données: que ce soit les données brutes d'Excel, l'accès, les fichiers texte, etc., cette étape (collecte des données passées) constitue le fondement de l'apprentissage futur. Plus la variété, la densité et le volume des données pertinentes sont élevées, meilleures sont les perspectives d'apprentissage de la machine.

2. Préparation des données: tout processus analytique repose sur la qualité des données utilisée. Il faut passer du temps à déterminer la qualité des données, puis à prendre des mesures pour résoudre les problèmes tels que les données manquantes et le traitement des valeurs aberrantes. L'analyse exploratoire est peut-être une méthode pour étudier les nuances des données dans les détails, augmentant ainsi le contenu nutritionnel des données.
3. Formation d'un modèle: cette étape consiste à choisir l'algorithme approprié et la représentation des données sous la forme du modèle. Les données nettoyées sont divisées en deux parties - train et test (proportion en fonction des prérequis); la première partie (donnée d'entraînement) est utilisée pour développer le modèle. La deuxième partie (données de test), sert de référence.
4. Évaluation du modèle: pour tester la précision, la deuxième partie des données (données de retenue / test) est utilisée. Cette étape détermine la précision du choix de l'algorithme en fonction du résultat. Un meilleur test pour vérifier l'exactitude du modèle est de voir ses performances sur des données qui n'ont pas du tout été utilisées pendant la construction du modèle.
5. Amélioration des performances: cette étape peut impliquer de choisir un modèle complètement différent ou d'introduire plus de variables pour augmenter l'efficacité. C'est pourquoi un temps considérable doit être consacré à la collecte et à la préparation des données [1].

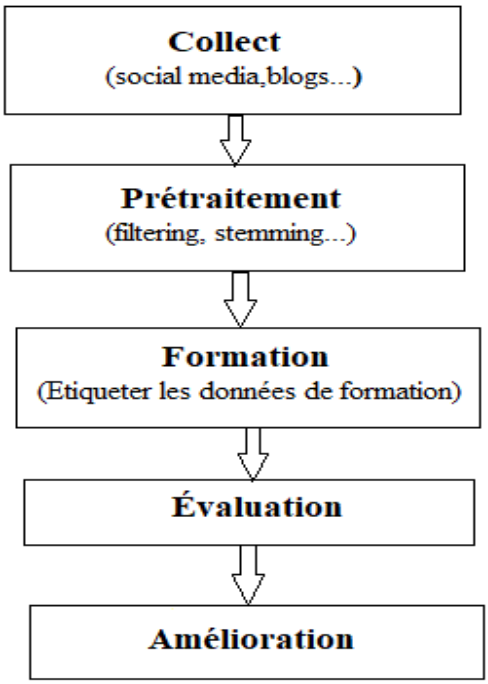


Figure 5 – Les étapes d'apprentissage automatique.

1.5. Limites de l'apprentissage automatique

Malgré la puissance d'apprentissage automatique et sa large utilisation dans les différents domaines et disciplines, il souffre de quelques limites talque :

- Pour des tâches complexes, on a besoin d'une grande quantité de données
- Dans le cas de l'apprentissage supervisé, l'annotation de données est une tâche fastidieuse; qui prend beaucoup de temps.
- Le traitement automatique de langages naturels (TALN) reste un défi.
- Les données d'entrainement sont souvent biaisées [26].

1.6. Conclusion

Dans ce chapitre nous avons entamé une étude globale de l'apprentissage automatique, ses définitions et ses types. Nous avons aussi essayé de donner quelques algorithmes et méthodes d'apprentissage. La sélection des attributs est une technique d'apprentissage puissant et populaire dans kla littérature, elle peut être utilisée pour l'apprentissage ou même pour le pre-processing de dataset. Notre deuxième chapitre est consacré à cette technique.

Chapitre 02
Sélection d'Attributs

2.1. Introduction

La sélection d'attributs est un sujet de recherche très actif depuis une dizaine d'années dans les domaines de l'apprentissage artificiel, de la fouille de données, du traitement d'images, et de l'analyse de données en bio-informatique. Dans tous ces domaines, les applications nécessitent de traiter des données décrites par un très grand nombre d'attributs. Ainsi, on peut avoir à traiter des pages web décrites par plusieurs milliers de descripteurs, des images décrites par plusieurs millions de pixels ou des données en bio-informatique donnant les niveaux d'expression de plusieurs milliers de gènes.

Nous introduisons dans ce chapitre par une définition de la sélection des caractéristiques en donnant ses différents avantages. Nous donnons par la suite le processus général de la sélection et les différents types des méthodes de sélection des attributs.

On donne aussi une revue des méthodes de sélection des caractéristiques et les domaines d'application de ces méthodes. On termine ce chapitre par une conclusion.

2.2. Définition de La sélection d'attribut

La sélection d'attributs, de caractéristiques où feature sélection en anglais, est un problème difficile qui a été étudie depuis les années 70 [16].

La sélection de caractéristiques est un processus de recherche ou une technique utilisée pour choisir les caractéristiques, les variables ou les mesures les plus intéressantes, pertinentes ou informatives d'un système donné, dans le but de réaliser la tâche pour laquelle il a été conçu voir figure 6.

Dans le domaine d'apprentissage automatique ou, plus précisément, celui de la classification, certaines caractéristiques non pertinentes et/ou redondantes, existant généralement dans les données d'apprentissage, non seulement rendent l'apprentissage plus difficile, mais dégradent aussi les performances de généralisation des modèles d'apprentissage.

Dans la littérature, les auteurs donnent des définitions différentes à la sélection de caractéristiques, Unler et Murat [22] définissent la sélection de caractéristiques pour un problème de classification de la manière suivante:

soit G une base de données avec R échantillons et K dimensions (attributs ou caractéristiques), représentée sous forme d'une matrice G = R * K. L'objectif de la tâche de sélection de sous-ensemble de caractéristiques est d'obtenir k dimensions, de l'ensemble total de l'espace de caractéristiques ou k <K [16].

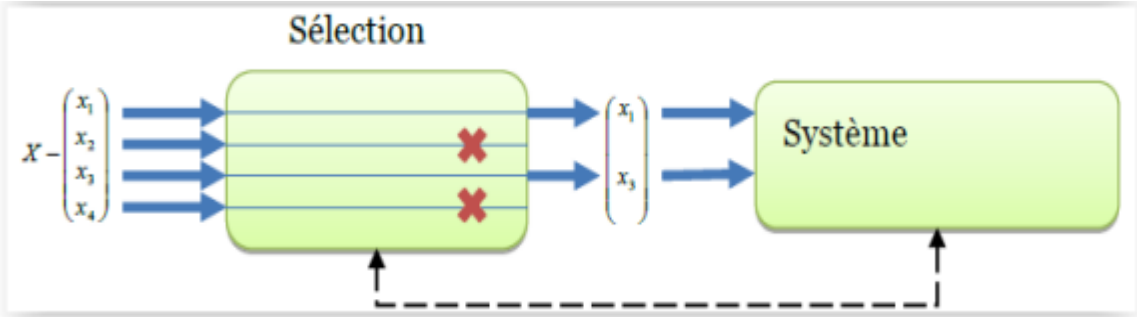


Figure 6 – Principe de sélection d'attribut [16].

En conséquence, la sélection des attributs présente les avantages suivants **:**

1. Elle réduit le nombre des attributs en cherchant à enlever les attributs non pertinents, redondants, non appropriés et bruitées. Conformément à la définition, ce processus sélectionne les attributs en se basant sur certains critères pour éliminer tous les facteurs qui ne sont pas en rapport avec le problème traité, et garder efficacement les attributs importants.
2. La sélection des attributs peut augmenter la précision et améliorer les performances du classificateur. Après l'opération de sélection d'attributs, un grand nombre de données non pertinentes est supprimé. Seuls les attributs les plus importants sont choisis et gardés, ce qui rend le modèle de classification obtenue beaucoup plus simple afin d'améliorer sa capacité de résolution du problème, et sa précision de classification.
3. La sélection d'attributs réduit le temps de calcul. En effet, après la sélection des attributs, la complexité de calcul est réduite, ce qui augmente la vitesse d'exécution de l'algorithme et la vitesse d'apprentissage.
4. Les attributs conservés correspondent aux variables liées aux phénomènes d'intérêt et permettent d'en faire une interprétation plus simple.

2.3. Processus de sélection d'attributs

Une procédure générale pour élaborer une méthode de sélection d'attributs est illustrée par la figure 7. On distingue 4 étapes pour la sélection des attributs en commençant par l'ensemble initial des attributs : la génération du sous-ensemble, l'évaluation du sous-ensemble, le critère d'arrêt, et la validation des résultats.

- **La génération du sous-ensemble** est une stratégie de recherche utilisée pour déterminer des sous-ensembles d'attributs candidats pour l'évaluation.
- **L'évaluation du sous-ensemble** : un certain critère d'évaluation est estimé pour mesurer la qualité du sous-ensemble candidat. Ensuite il est comparé avec le meilleur sous-ensemble précédent pour déterminer si ce sous-ensemble est convenable ou non. Si

le nouveau sous-ensemble candidat est meilleur, il remplace le précédent meilleur. En répétant ce processus le sous-ensemble associé à la meilleure valeur du critère est sélectionné.

- **Critère d'arrêt:** il est nécessaire que chaque sous-ensemble d'attributs après l'évaluation soit comparé au critère d'arrêt pour vérifier si les attributs du sous-ensemble actuel ont atteint un niveau prédéfini. Si les exigences sont vérifiées, la sélection d'attributs s'arrête et le sous-ensemble courant est considéré comme le résultat final; sinon le processus de recherche continue.
- **La validation:** le sous-ensemble choisi doit généralement être validée par différents tests avec des données du monde réel ou non réel [12].

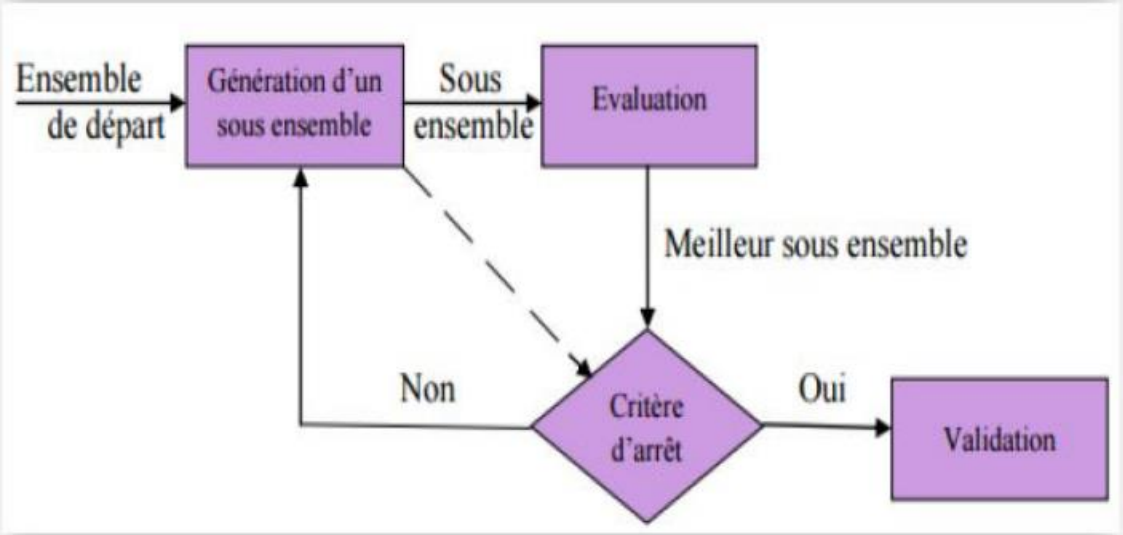


Figure 7 – Processus de sélection des attributs [12].

2.3.1. La 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-ensembles 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 [14].

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 sélection) 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 élimination) 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 le 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) [15].

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 [14].

- **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.

- **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), différentes méthodes ont été développées et utilisées pour la sélection d'attributs. Liu et Yu [17] proposent de séparer les méthodes en fonction de la stratégie utilisée, où Les procédures complètes sont subdivisées en «exhaustives» et «non exhaustives», 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» l'et «type II » (cité précédemment).

La figure 8 ci-dessous donne un aperçu général sur les méthodes de sélection de variables basées sur la stratégie de recherche

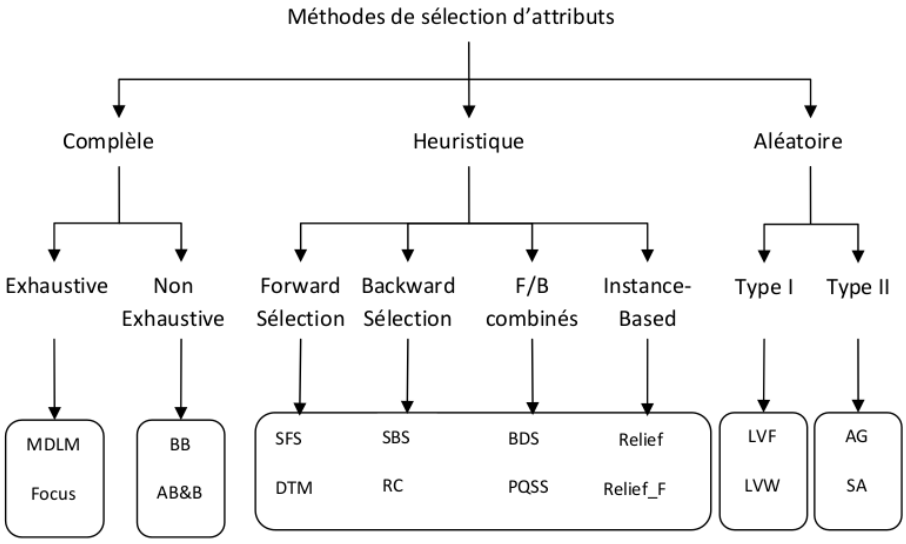


Figure 8 – Catégorisation des méthodes de sélection de caractéristiques [14].

2.3.2. La procédure d'évaluation

L'évaluation constitue une partie importante de la sélection d'attributs. On peut distinguer trois catégories pour l'évaluation dans les algorithmes de sélection: "filter", "wrapper" et "embedded" :

- **Filter méthodes:** cette technique est généralement utilisée pour la sélection d'attributs. Elle évalue la pertinence des attributs en examinant seulement leurs propriétés intrinsèques. Cette technique est considérée comme une étape de prétraitement (filtrage) avant le processus d'apprentissage [12], cela signifie que l'évaluation de cette technique est indépendante du classificateur. Dans la majorité des cas, un score de pertinence d'attributs est calculé, et les attributs à faible score sont supprimés. Le meilleur sous-ensemble d'attributs obtenus par cette technique est présenté en entrée de l'algorithme de classification [14]. La procédure du modèle "filter" est illustrée par la figure 9.

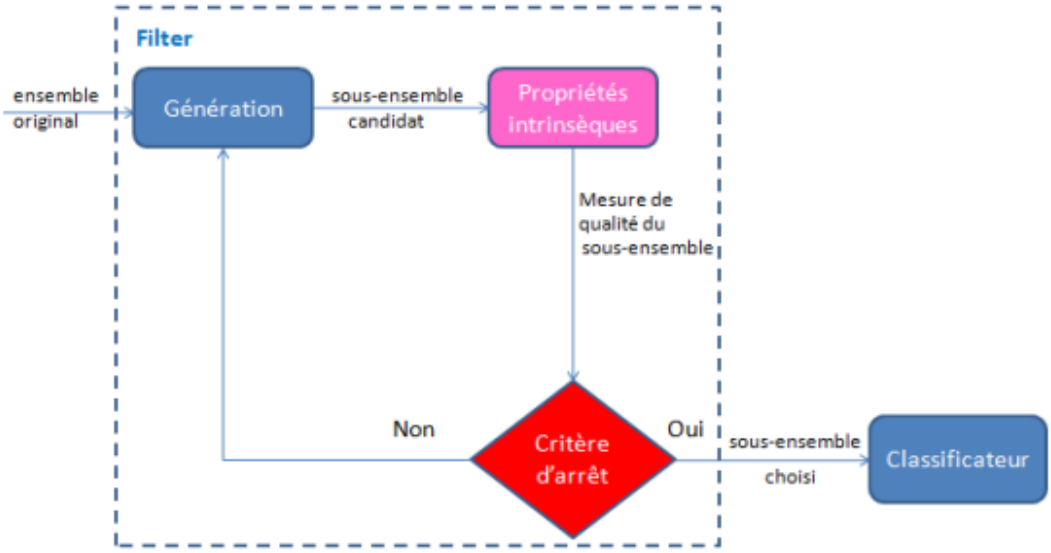


Figure 9 – Sélection d'attributs à base de filtre [15].

- **Wrapper method :** le principal inconvénient de la technique "filter" provient du fait qu'elle ignore l'influence des attributs sélectionnés sur la performance du classificateur. Kohavi et John proposent la technique wrapper pour résoudre ce problème [14]. Cette technique évalue un sous-ensemble d'attributs en utilisant l'algorithme de classification. Cette technique produit une précision plus élevée puisque les attributs sélectionnés correspondent bien aux algorithmes d'apprentissage. Mais cette méthode présente l'inconvénient d'avoir un coût de calcul plus élevé que dans le cas des méthodes filtre dû à l'appel de l'algorithme de classification pour chaque sous-ensemble considéré. En plus, le sous-ensemble sélectionné dépend de l'algorithme de classification, ainsi si on

change l'algorithme de classification, il faut recommencer la sélection ! La procédure du modèle "wrapper" est illustrée par la figure 10[12].

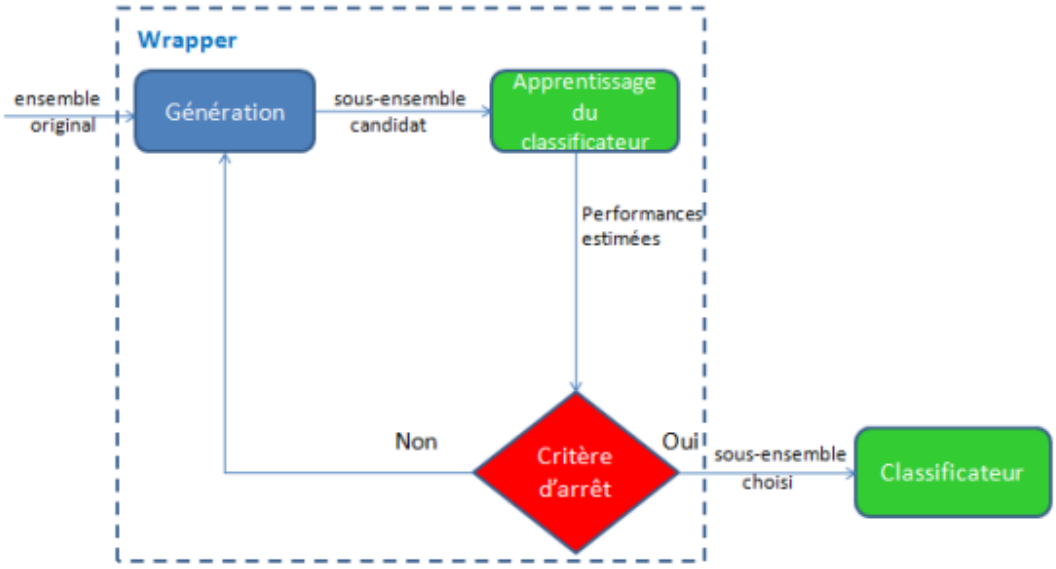


Figure 10 – Sélection d'attributs à base Wrapper [15].

- **Embedded method:** cette technique accomplit la sélection simultanément avec la procédure de classification. Le sous-ensemble optimal d'attributs est déterminé lors de l'apprentissage du classificateur. Tout comme les techniques "wrapper", les techniques "Embedded" sont spécifiques à un algorithme d'apprentissage donné [14]. L'avantage principal de cette technique est qu'elle est plus rapide que la technique "Wrapper". La procédure du modèle "embedded" est illustrée par la figure 11.

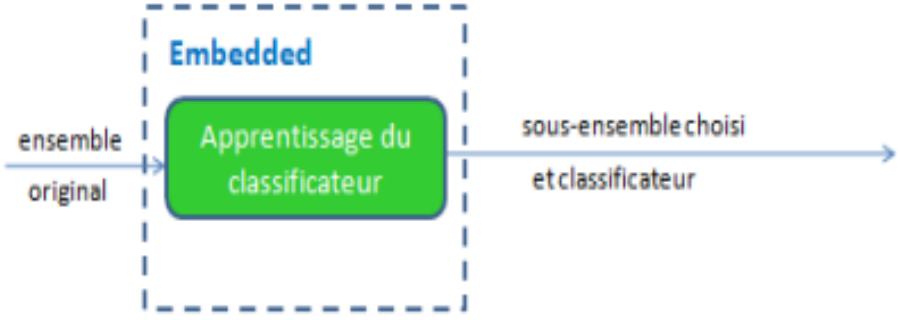


Figure 11 – Sélection d'attributs à base Embedded [14].

Ce tableau présente les avantages et les inconvénients de ces trois approches tableau : Avantages et inconvénients des méthodes filtres, wrapper et embedded :

Tableau 1 – Avantages et Inconvénients de trois approches

Méthode	Avantage	Inconvénients
Filter	• Exécution rapide. • coût de calcul faible.	• Aucune interaction avec le classificateur
Wrapper	• Interaction avec le classificateur • Bonne performance de classification	• coût de calcul élevé • exécution lente
Embedded	• Interaction avec le classificateur • Bonne performance de classification	• coût de calcul élevé mais plus faible que Wrapper. • exécution lente mais plus rapide que Wrapper. • pas adapté à tous les types de classificateurs.

2.3.3. Le critère d'arrêt

Les itérations du processus de sélection de caractéristiques continuent à s'exécuter jusqu'à ce qu'un critère d'arrêt soit rempli. La procédure de génération et la fonction d'évaluation peuvent influencer le choix d'un critère d'arrêt [13]. Les critères d'arrêt basés sur la procédure de génération incluent :

- un seuil est atteint, tel que le nombre minimal d'attributs ou le nombre maximal d'itérations.
- Il n'y a plus d'amélioration de précision, autrement dit, lorsqu'il n'y a pas de possibilité de trouver un sous-ensemble meilleur que le sous-ensemble actuel.

Certains critères d'arrêt couramment utilisés sont basés sur l'ordre d'attributs classés selon un score de pertinence. Ceux qui ont les scores les plus élevés seront sélectionnés et utilisés par un classificateur (méthode "filtre").

2.3.4. La procédure de validation

Dash et Liu [17] proposent d'ajouter une quatrième composante à un algorithme de sélection de caractéristiques : une procédure de validation. Deux alternatives sont proposées en fonction de la nature des données utilisées lors de cette procédure : artificielles ou réelles. Généralement une base de données synthétique est construite dans le but de tester un concept ou une application particulière. De ce fait les variables pertinentes sont connues et identifiées.

La validation d'un algorithme sera alors directe puisqu'il suffit de vérifier si le sous-ensemble retenu contient bien les variables pertinentes

Dans le cas de données réelles, les variables pertinentes ne sont généralement pas connues, la procédure consiste alors à évaluer la précision de la classification obtenue avec le sous-ensemble de variables sélectionnées par l'intermédiaire d'un classifieur (classifieur de Bayes,. . .). Cette dernière peut alors être comparée à d'autres approches ou à celle obtenue par des techniques classiques [13] [15].

2.4. Revue des méthodes de sélection des caractéristiques

Il existe beaucoup de travaux de recherche dans la littérature sur les méthodes de sélection des caractéristiques, on présente dans ce qui suit une sélection de ces méthodes :

2.4.1. Sequential Forward Selection (SFS)

Cette méthode apparaît en 1963 et c'est une approche de recherche heuristique. Elle commence par un ensemble vide et on ajoute successivement une caractéristique jusqu'au ce que le critère d'arrêt soit satisfait. Cette méthode était utilisée pour réduire la taille des données et améliorer les résultats de classification. L'algorithme (Algorithme 1) de cette méthode est comme suit [12] :

Entrées:
$F = \{f_1, f_2, .., f_N\}$
M : taille de l'ensemble final
Sorties: $E = \{f_{s1}, f_{s2}, .., f_{sM}\}$
$E = \varnothing$
Pour $i = 1$ à M **Faire**
  Pour $j = 1$ à $|F|$ **Faire**
    Évaluer $f_j \cup E$
  Fin Pour
  $f_{max} =$ meilleure f_j
  $E = E \cup f_{max}$, $F = F \setminus f_{max}$
Fin Pour
Retourner E

Algorithme 1 – Algorithme de la méthode SFS [12]

2.4.2. Sequential Backward Selection (SBS)

Cette méthode date de 1971. Le principe général de cette méthode est de commencer par l'ensemble entier de toutes les caractéristiques et faire la suppression successive des caractéristiques. Cette technique est plus performante que la précédente (SFS) mais son problème majeur réside dans le temps de calcul [12].

L'algorithme (Algorithme 2) de la méthode SBS est comme suit:

Entrées:
$F = \{f_1, f_2, .., f_N\}$
M : taille de l'ensemble final
Sorties: $E = \{f_{s1}, f_{s2}, .., f_{sM}\}$
$E = F$
Pour $i = 1$ à N-M **Faire**
Pour $j = 1$ à $|E|$ **Faire**
Évaluer $E \setminus f_j$
Fin Pour
f_{min} = la plus mauvaise f_j
$E = E \setminus f_{min}$
Fin Pour
Retourner E

Algorithme 2 – L'algorithme de la méthode SBS [12]

2.4.3. FOCUS

Une méthode conçue en 1991, repose sur une recherche exhaustive sur l'ensemble initial des caractéristiques [21].

L'algorithme FOCUS (Algorithme 3 présenté ci-dessous) commence par générer et évaluer tous les sous-ensembles de taille T (initialement un), puis tous les couples de caractéristiques, les triplets et ainsi de suite jusqu'à ce que le critère d'arrêt soit satisfait [13].

Entrées: Une base d'apprentissage $A = \{X_1, X_2, .., X_M\}$ où $X_i = \{x_{i1}, x_{i2}, .., x_{iN}\}$
T : Taille maximale de l'ensemble final et un seuil ϵ
Sorties: S : ensemble final des caractéristiques
$S = \varnothing$
Pour $i = 1$ à T **Faire**
Pour chaque sous-ensemble (S_1) de taille (i) **Faire**
$Cons = Inconsistance(A, S_1)$
Si $Cons < \epsilon$ **alors**
$S = S_1$
Retourner S
Fin Si
Fin Pour
Fin Pour

Algorithme 3 – L'algorithme de la méthode FOCUS [21]

L'inconvénient majeur qui se présente est que sa méthode d'évaluation est sensible au bruit et aussi, le temps de calcul deviendra grand lorsque la taille des caractéristiques est grande. Une autre version de FOCUS, FOCUS 2 a été proposée en 1992.

2.4.4. Relief

Une méthode proposée en 1992, son principe est de calculer une mesure globale de la pertinence des caractéristiques en accumulant la différence des distances entre des exemples d'apprentissage choisis aléatoirement et leurs plus proches voisins de la même classe et de l'autre classe [21].

Cette méthode a beaucoup d'avantages, elle est simple, facile à mise en œuvre et elle est précise même sur les données bruitées. Mais son inconvénient est que sa technique aléatoire ne peut pas garantir la cohérence des résultats lorsqu'on applique plusieurs fois la méthode sur les mêmes données. Et aussi, elle ne prend pas en considération la corrélation éventuelle entre les caractéristiques [13].

L'algorithme 4 représente l'algorithme de la méthode Relief :

Entrées: Une base d'apprentissage $A = \{X_1, X_2, .., X_M\}$ où chaque exemple $X_i = \{x_{i1}, x_{i2}, .., x_{iN}\}$
Nombre d'itérations T
Sorties: W[N] : vecteur de poids des caractéristiques (f_i), $-1 \leq W[i] \leq 1$
$\forall i, W[i] = 0;$
Pour $t = 1$ à T **Faire**
Choisir aléatoirement un exemple X_k
Chercher deux plus proches voisins (un dans sa classe (X_a) et un deuxième dans l'autr classe (X_b))
Pour $i = 1$ à N **Faire**
$W[i] = W[i] + \frac{|x_{ki} - x_{bi}|}{M \times T} - \frac{|x_{ki} - x_{ai}|}{M \times T}$
Fin Pour
Fin Pour
Retourner W

Algorithme 4 – L'algorithme de la méthode Relief [21]

Une version déterministe de Relief était proposée sous le nom de Relief D'en 1994, cette version évitera le caractère aléatoire de Relief. D'autres variantes de Relief étaient proposées en 1996 et en 2002.

2.4.5. Les algorithmes génétiques

Ces algorithmes ont été proposés pour éviter les optima-locaux. Il faut mentionner que les algorithmes génétiques sont les meilleurs dans les techniques de recherche. Plusieurs chercheurs proposent l'hybridation de l'algorithme génétique avec d'autres méthodes [19] [20].

IL y a eu une proposition d'une nouvelle méthode hybride pour la sélection des caractéristiques, elle utilise les deux algorithmes Branch and Bound [18] et les algorithmes génétiques pour résoudre le problème de la sélection. Une autre approche de sélection des

caractéristiques à base d'un algorithme génétique et avec l'utilisation de K-ppv comme étant une fonction fitness.

Il y a eu aussi l'hybridation de l'algorithme génétique avec l'algorithme « Stepwise Greedy ».

2.5. Domaines d'application de la sélection des caractéristiques

Les principaux domaines d'application de la sélection des caractéristiques :

- **La fouille de données (Data Mining) :** la sélection des caractéristiques est une étape importante dans le processus de la fouille de données, elle permet de réduire la taille des données par l'élimination des données non importantes et redondantes.
- **L'apprentissage automatique :** ce domaine repose sur la construction d'un système automatique à partir des connaissances existantes. Le problème de cette grande masse de données est la présence des données bruitées et non pertinentes. La sélection des caractéristiques est une étape cruciale pour le processus de l'apprentissage automatique dans le but d'améliorer la performance du système.
- **Catégorisation de textes :** un problème central pour la catégorisation de textes est la grande dimension de l'espace de représentation. Par exemple, avec la représentation en sac de mots, chacun des mots d'un corpus est un descripteur potentiel ; ou pour un corpus de taille raisonnable, ce nombre peut être de plusieurs dizaines de milliers. Pour beaucoup d'algorithmes d'apprentissage, il faut sélectionner un sous-ensemble de ces descripteurs pour éviter le problème du coût de traitement ainsi que le problème de faible fréquence de certains termes.
- **Reconnaissance de formes :** de nombreux problèmes de reconnaissance de formes dans des domaines aussi divers que l'interprétation des scènes dans une image, de textes ou de microarrays génétiques engendrent la manipulation de grand nombre de variables. Ce grand nombre des variables nuits sur la performance du système d'apprentissage, par exemple : les variables peu informatives d'un signal agissent comme un bruit. Il s'agit donc de réduire le nombre des variables par l'élimination de celles non importantes.
- **La fouille des images :** la quantité des images acquises dans différents domaines, comme les images médicales, images satellitaires, et autres engendre une très grande accumulation. L'interprétation et la manipulation de cette masse d'images nécessitent tout d'abord un processus de prétraitement et de nettoyage.

2.6. Conclusion

La sélection d'attributs constitue un problème majeur dans plusieurs domaines. Par conséquent, il a été un sujet d'intérêt pour de nombreux chercheurs.

Nous avons introduit dans ce chapitre le processus de sélection des caractéristiques qui est considéré comme une étape de prétraitement pour les systèmes d'apprentissage automatique. Nous avons donné des notions de base fréquemment utilisées par les méthodes de sélection des caractéristiques, ainsi que le processus général de cette méthode. Suivi par une revue des méthodes existant dans la littérature.

Chapitre 03
Approche Proposé

3.1 Introduction

Nous introduisons dans ce chapitre par les limites de techniques de sélection d'attributs

Ensuite on va détaille notre approche de sélection d'attributs qui fonctionne dans les deux cas : supervisé et non supervisée et qui va prendre en considération le critère d'originalité d'attribut. On termine ce chapitre par une conclusion.

3.2 Les limites de techniques de sélection d'attributs :

Durant de nos lectures, nous avons remarqué que la majorité des travaux [25] ne traitent pas l'originalité des attributs, ils utilisent des méthodes statistiques (facteur de corrélation, Z-score par exemple), méthodes heuristiques [Cout dans [24]), méthodes méta heuristique (algorithme génétique dans [23]) et même des stratégies d'apprentissage (cas de stratégie wrapper) pour sélectionner l'attribut sans prendre en considération l'historique et l'originalité de cet attribut. L'originalité d'un attribut reflète les changements qui a subi cet attribut durant la phase de pré-processing de dataset et elle a un grand impact sur la qualité de sélection des attributs.

Nous avons proposé une nouvelle approche qui permet d'améliorer les résultats de la méthode de filtrage. Une stratégie heuristique de sélection des attributs qui fonctionne dans les deux cas : supervisé et non supervisée et qui va prendre en considération le critère d'originalité d'attribut. Notre stratégie est nommée FsbO (Feature Sélection based ont Originality),

3.3 Approche proposée FsbO

Nous avons proposé une méthode de sélection de variables de type (Filter), C'est une méthode simple et considérée comme étape de prétraitement pour la classification.

Dans notre approche on va choisir un sous-ensemble de variables pertinents parmi un ensemble d'attributs de grande taille et éliminer les variables redondantes qui possèdent un score fort.

Notre approche est basée sur la valeur de score qui sera calculé à base de taux des données perdues et de taux des outliers. En cas d'apprentissage supervisé, on ajoute la corrélation pour améliorer le résulta. Les détailles de cette approches sont a dans les prochains paragraphes,

3.3.1 Schéma de FsbO :

La méthode de FsbO est un filtrage, dans un but de sélectionner un sous-ensemble optimal d'attributs à partir de tout l'ensemble basant sur sa pureté (originalité).

L'idée est la suivant : si un attribut a subit des changements ou des modifications (causés généralement par des outliers ou bien des données nulles) durant la phase de pré-processing, la possibilité que cet attribut présente vraiment le dataset sera réduite et dans ce cas il est préférable de ne pas sélectionner cet attribut dans la phase de filtrage,

.

La figure 12 ci-dessous représente le schéma général de l'approche proposée :

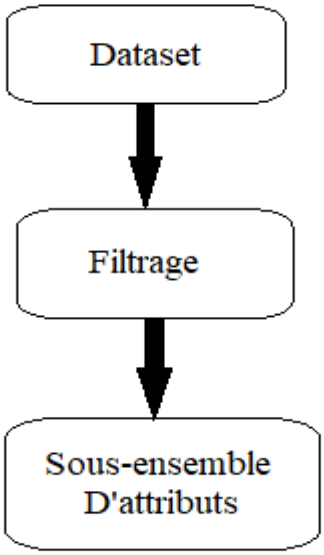
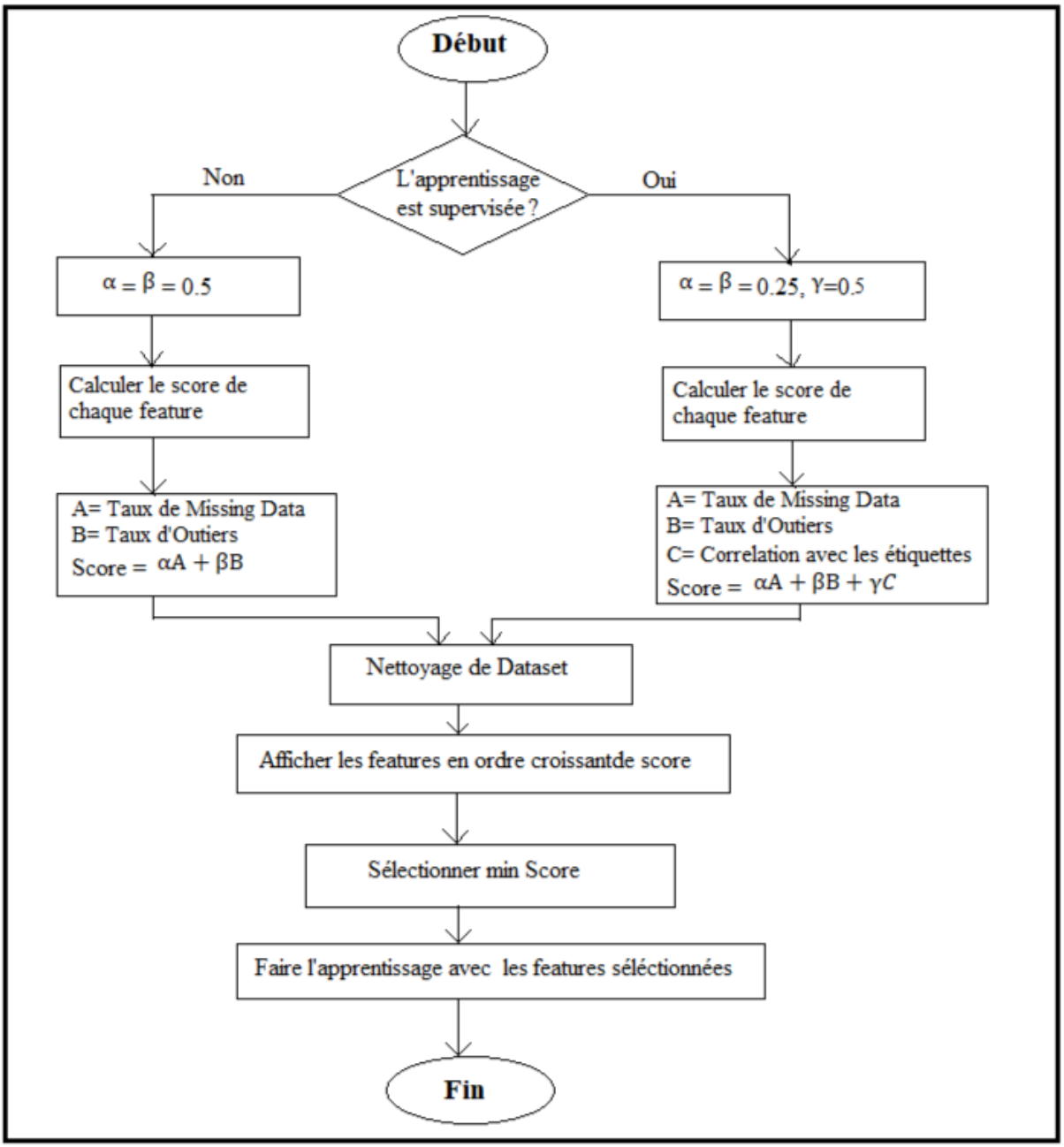


Figure 12 – Schéma général de l'approche proposée FsbO

Le principe général de l'approche proposée est d'utiliser les résultats de filtrage pour sélectionner un sous-ensemble qui améliore le résultat de classification.

3.3.2 Organigramme de notre approche FsbO:



Organigramme de FsbO

3.3.3 Méthode de filtrage :

Les méthodes de filtrage sont des techniques de classement des caractéristiques qui évaluent la pertinence des caractéristiques en regardant les propriétés intrinsèques des données indépendamment de l'algorithme de classification [27]. Un critère de classement approprié est utilisé pour classer les variables et un seuil est utilisé pour éliminer la variable en dessous du seuil [27]. En utilisant la méthode de filtrage, la sélection des caractéristiques est effectuée une fois et elle peut être utilisée avec différents classificateurs [28].

Dans les méthodes de classement des caractéristiques (Autre nomination des méthodes de filtrage), chaque caractéristique est classée selon une métrique de sélection, telles que le gain d'information et le rapport de gain (gain ratio), etc. Les caractéristiques les mieux classées sont sélectionnées comme caractéristiques pertinentes selon une valeur de seuil prédéfinie [29].

3.3.4 Schéma général de la méthode de filtrage

Les méthodes de filtrage sont indépendantes de l'algorithme de classification. Elles sélectionnent les caractéristiques selon leurs pertinences. La figure 12 représente le processus de la méthode de filtrage [29].

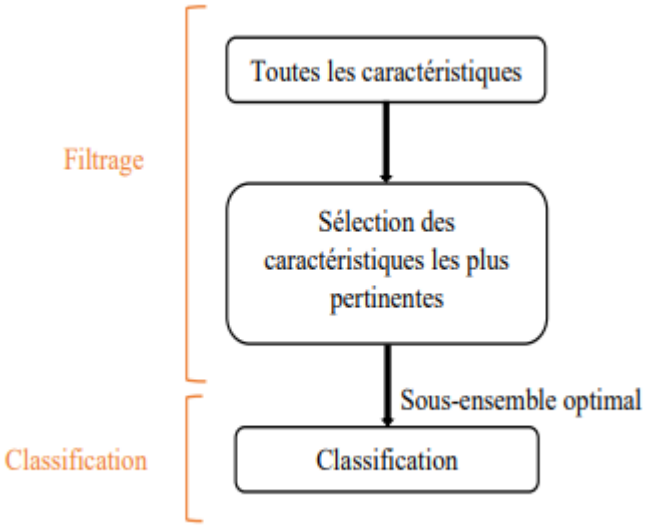


Figure 13 – Processus de la méthode de filtrage [29].

Comme la montre la figure ci-dessus, le processus de filtrage sélectionne un sous-ensemble de caractéristiques à partir de l'ensemble total, sans intervention de l'algorithme de classification.

La sélection se fait par le calcul de la pertinence des attributs en utilisant les mesures de pertinence comme la mesure d'information, la mesure de corrélation.

3.3.5 Avantages et limites des méthodes de filtrage

Les méthodes de filtrage sont les plus simples parmi les méthodes de sélection des caractéristiques. Elles ne sont pas coûteuses puisqu'elles n'introduisent aucun algorithme d'apprentissage dans leur processus. Mais, l'inconvénient major de ces méthodes, c'est qu'elles ne détectent pas les redondances et les similarités entre les caractéristiques. De ce fait, le sous-ensemble généré peut contenir des caractéristiques redondantes [30].

3.4 Les algorithmes utilisés dans notre approche

Dans notre approche, nous avons utilisé trois paramètres pour calculer le score d'originalité : la corrélation, le taux des données perdus et le taux des outliers,

3.4.1 La corrélation

La corrélation est une mesure qui décrit la force et la direction d'une relation entre deux variables. Il est couramment utilisé dans les statistiques, l'économie et les sciences sociales pour les budgets, les plans d'entreprise, etc. [31].

La méthode utilisée pour étudier le degré de corrélation entre les variables s'appelle l'analyse de corrélation. Voici quelques exemples de corrélation forte :

- le nombre de calories que vous mangez et votre poids (corrélation positive)
- La température extérieure et vos factures de chauffage (corrélation négative)

Et voici les exemples de données qui ont une corrélation faible ou nulle :

- le nom de votre chat et son plat préféré
- La couleur de vos yeux et votre taille

Une chose essentielle à comprendre à propos de la corrélation est qu'elle montre seulement à quel point deux variables sont étroitement liées. La corrélation, cependant, n'implique pas la causalité. Le fait que les changements dans une variable soient associés aux changements dans l'autre variable ne signifie pas qu'une variable entraîne en réalité le changement de l'autre [31].

3.4.1.1 Coefficient de corrélation

Le coefficient de corrélation est la mesure spécifique qui quantifie la force de la relation linéaire entre deux variables d'une analyse de corrélation. Le coefficient est noté r dans un rapport de corrélation [31].

Le coefficient de corrélation de la série (Xi, Yi) représenté par la formule suivante (voir formule F2):

$$r = \frac{\sum[(x_i - \bar{x})(y_i - \bar{y})]}{\sqrt{\sum(x_i - \bar{x})^2 * \sum(y_i - \bar{y})^2}} \quad \text{(F2)}$$

r: coefficient de corrélation

x_i: Valeur de X

$\bar{x}$: Moyenne de la variable X

y_i : Valeur de Y

$\bar{y}$: Moyenne de la variable Y

$\sum(x_i - \bar{x})^2$: Somme des écarts au carré pour X

$\sum(y_i - \bar{y})^2$: Somme des écarts au carré pour Y

Un coefficient de 1 signifie une relation positive parfaite - à mesure qu'une variable augmente, l'autre augmente proportionnellement.

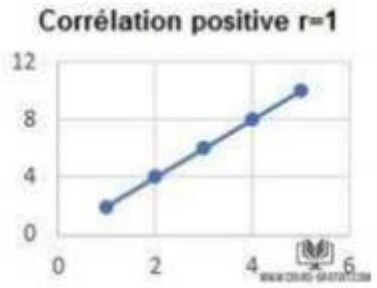


Figure 14 – Coefficient de corrélation proche de 1

Un coefficient de -1 signifie une relation négative parfaite - à mesure qu'une variable augmente, l'autre diminue proportionnellement

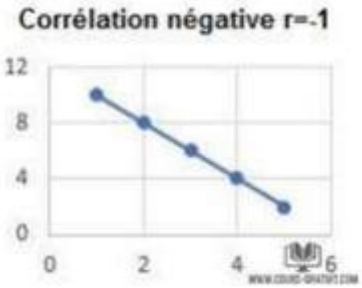


Figure 15 – Coefficient de corrélation proche de –1

Un coefficient de 0 signifie qu'il n'y a pas de relation entre deux variables - les points de données sont dispersés sur tout le graphique.

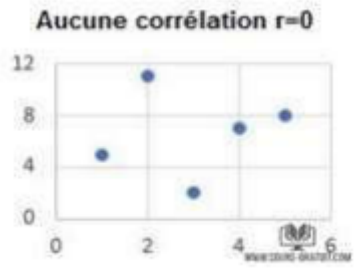


Figure 16 – Coefficient de corrélation proche de 0

3.4.1.2 Matrice de corrélation

Une matrice de corrélation est simplement un tableau qui affiche les coefficients de corrélation pour différentes variables. La matrice décrit la corrélation entre toutes les paires de valeurs possibles dans un tableau. C'est un outil puissant pour résumer un grand ensemble de données et pour identifier et visualiser des modèles dans les données fournies [31].

Une matrice de corrélation est constituée de lignes et de colonnes montrant les variables. Chaque cellule d'une table contient le coefficient de corrélation.

Dans l'exemple ci-dessus, nous sommes intéressés à connaître la corrélation entre la variable dépendante (nombre d'appareils de chauffage vendus) et deux variables indépendantes (température mensuelle moyenne et coûts publicitaires). Nous ne regardons donc que les chiffres à l'intersection de ces lignes et colonnes, qui sont mis en évidence dans la capture d'écran ci-dessous (voir figure 17) :

	Température C°	*Coûts publicitaire*	*vente app chauffage*
Température C°	1		
Coûts publicitaire	-0,94008875	1	
vente app chauffage	-0,97237731	0,957827719	1

Figure 17 – Tableau de matrice de corrélation [31].

- Le coefficient négatif de -0,97 (arrondi à 2 décimales) montre une forte corrélation inverse entre la température mensuelle et les ventes d'appareils de chauffage - à mesure que la température augmente, moins d'appareils de chauffage sont vendus.
- Le coefficient positif de 0,95 (arrondi à la deuxième décimale) indique un lien direct étroit entre le budget publicitaire et les ventes - plus vous dépensez d'argent en publicité, plus les ventes sont élevées.

3.4.2 Score pour chaque attribut

Pendant la sélection automatique d'attributs, un score est calculé pour chaque attribut, et seuls les attributs qui ont les meilleurs scores sont sélectionnés pour le modèle.

Pour donne le score de chaque attribut, nous avons utilisé Missing data et Outliers.

- **Missing Data (valeurs manquantes)**

Les données manquantes sont un problème qui se produit dans presque tous les jeux de données réels. La définition de manquer valeurs est que certaines informations sur les variables sont manquantes. En général, le problème avec valeurs manquantes est que les analyses ne peuvent pas être rendues correctes sur la base des données et des conclusions tirées d'un ensemble de données avec des valeurs manquantes peuvent ne pas être véridiques [32].

- **Outilers (valeurs aberrantes)**

Une valeur aberrante, en mathématiques, statistiques et technologies de l'information, est un point de données spécifiques qui se situe en dehors de la plage de probabilité pour un ensemble de données. En d'autres termes, la valeur aberrante est distincte des autres points de données environnantes d'une manière particulière. L'analyse des valeurs aberrantes est extrêmement utile dans divers types d'analyses et de recherches, dont certaines sont liées aux technologies et aux systèmes informatiques [33].

Pour calculer le score nous utilisons 2 formules selon la stratégie d'apprentissage utilisée :

3.4.2.1 Apprentissage Supervisée

Dans l'apprentissage supervisé nous avons utilisé cette formule (voir formule F3) pour calculer le score OS (Originality Score):

$$\mathbf{OS} = \boldsymbol{\alpha A + \beta B + \gamma C} \qquad \text{(F3)}$$

Où :

- A: le taux de missing data (valeurs manquantes)
- B : le taux d'outliers (valeurs aberrantes)
- C : le taux de corrélation avec les étiquettes
- α , β ,γ: Sont des facteurs d'importance Avec: $\alpha + \beta + \gamma = 1$

Exemple :

Tableau 2 – Score pour l'apprentissage supervisé

Nombre de Feature	Taux de MD	Taux d'Outliers	Taux de corelation	Score
F1	5%	3%	0.1%	0.03
F2	20%	30%	0.5%	0.17
F2	30%	25%	0.9%	0.19
F3	22%	30%	0.8%	0.18
F4	6%	8%	-0.1%	0.07
F5	9%	2%	-0.2%	0.09
F6	30%	15%	0.7%	0.16
F7	2%	7%	-0.3%	0.05
F8	20%	15%	0.4%	0.15
F9	22%	33%	0.9%	0.19

3.4.2.2 Apprentissage Non supervisé

Dans l'apprentissage non supervisé nous avons utilisé cette formule (voir formule F4) pour calculer le score OS (Originality Score) :

$$OS = \alpha A + \beta B \qquad (F4)$$

Où :

- A: le taux de missing data (valeurs manquantes).
- B : le taux d'outliers (valeurs aberrantes).
- α , β : sont des facteurs d'importance Avec: $\alpha + \beta = 1$

les étiquètes

Exemple :

Tableau 3 – Score pour l'apprentissage non supervisé

Nombre de Feature	Taux de MD	Taux d'Outliers	Score
F1	5%	3%	0.04
F2	20%	30%	0.25
F2	30%	25%	0.27
F3	22%	30%	0.26
F4	6%	8%	0.07
F5	9%	2%	0.05
F6	30%	15%	0.22
F7	2%	7%	0.04
F8	20%	15%	0.17
F9	0%	0%	0

3.4.3 Sélection k attributs :

La sélection d'attributs permet à la fois d'éviter d'avoir trop de données qui présentent peu d'intérêt ou de n'avoir pas assez de données utiles. Notre objectif, en utilisant la sélection d'attributs, est d'identifier le nombre minimum de colonnes de la source de données qui sont importantes pour le modèle à créer.

Après avoir calculé le score de chaque attribut, nous l'avons ordonné d'ordre croissant pour sélectionner k attribue parmi un ensemble des attributs qui contient un minimum de score.

Exemple :

Tableau 4 – liste de features

Nombre de Feature	La valeur de score
F1	0.18
F2	0.16
F2	0.15
F3	0.13
F4	0.12
F5	0.10
F6	0.09
F7	0.08
F8	0.07
F9	0.0

Dans cet exemple,

Le score F1 est de 0,18, ce qui est plus fort que les autres features, Il contient donc de nombreuses valeurs manquantes et valeurs aberrantes (Missing Data et Outliers forts).

Le score de F9 est 0.0, Il ne contient pas donc des valeurs manquantes et valeurs aberrantes (Missing Data et Outliers).

3.5 Conclusion

Dans ce chapitre, nous avons présenté notre approche que nous avons proposée pour la sélection d'attributs, avec les algorithmes qui nous avons utilisé.

Notre stratégie est une sélection de filtres qui peut être appliquée sur un ensemble de données numériques et elle utilise deux paramètres pour rejeter ou accepter une caractéristique, contrairement aux stratégies classiques.

Chapitre 04
Implémentation

4.1 Introduction

Après une étude approfondie de sélection d'attribut, nous avons donné le processus général de la sélection et les différents types des méthodes de la sélection d'attribut. Et après la présentation de notre approche. Nous sommes lancées dans le volet technique qui concerne l'implémentation de notre approche.

4.2 Outils et méthodologie

Actuellement, plusieurs grandes plates-formes existent sur le marché. Elles sont globalement constituées de deux composantes : le langage de programmation et la base de données. On donnera une liste non exhaustive de différentes composantes utilisées :

4.2.1 Langage de programmation

Langage de programmation utilisé en machine Learning et en data science, le langage Python s'impose également dans d'autres secteurs d'activité grâce à sa simplicité et sa compatibilité.

Le langage Python est un langage de programmation open source multiplate-formes et orientée objet. Grâce à des bibliothèques spécialisées, Python s'utilise pour de nombreuses situations comme le développement logiciel, l'analyse de données, ou la gestion d'infrastructures. Il n'est donc pas, comme le langage HTML par exemple, uniquement dédié à la programmation web [34].

Langage de programmation interprété, Python permet l'exécution du code sur n'importe quel ordinateur. Utilisable aussi bien par des programmeurs débutants qu'experts, Python permet de créer des programmes de manière simple et rapide.

Python est principalement utilisé pour le Scripting et l'automatisation de tâches simples mais fastidieuses, c'est-à-dire l'interaction avec les navigateurs web. Mais Python est aussi utilisé pour :

- programmer des applications ;
- générer du code ;
- créer des services web ;
- faire du méta programmation.

Langage principalement utilisé pour la machine Learning et la data science, Python a fortement contribué à l'essor du Big data. Grâce à ses nombreuses bibliothèques telles Panda, Bokeh, Numpy, Scipy, Scrapy, Matpotlib, Scikit-Learn ou encore TensorFlow, Python offre

une grande flexibilité dans les tâches à effectuer et une grande compatibilité quelle que soit la plateforme utilisée [34].

4.2.2 la base de données

Nous avons utilisé dataset IRIS pour notre implémentation,

L'ensemble de données IRIS a été généré en 1936 par le statisticien et biologiste britannique Ronald Fisher. Il contient 150 échantillons au total, comprenant 50 échantillons de 3 espèces différentes d'Iris (Iris Setosa, Iris Versicolor et Iris Virginica). Pour chaque échantillon, les mesures des fleurs sont enregistrées pour la longueur des sépales, la largeur des sépales, la longueur des pétales et la largeur des pétales.

L'ensemble de données Iris peut être utilisé par un modèle d'apprentissage automatique pour illustrer la classification (une méthode utilisée pour déterminer le type d'un objet par comparaison avec des objets similaires qui ont déjà été catégorisés). Une fois formé sur des données connues, le modèle d'apprentissage automatique peut effectuer une classification prédictive en comparant un objet de tests à la sortie de ses données de formation [35].

4.3 Descriptions et les étapes de notre implémentation

Étape 0 :

Lire dataset.

Étape 1 : faire la corrélation entre les attributs**.**

a. Calculer le coefficient de corrélation (voir la figure 18).

```
PetalLengthCm  SepalWidthCm    -0.420516
Species        SepalWidthCm    -0.419446
PetalWidthCm   SepalWidthCm    -0.356544
SepalWidthCm   SepalLengthCm   -0.109369
Species        SepalLengthCm    0.782561
PetalWidthCm   SepalLengthCm    0.817954
PetalLengthCm  SepalLengthCm    0.871754
Species        PetalLengthCm    0.949043
               PetalWidthCm     0.956464
PetalWidthCm   PetalLengthCm    0.962757
dtype: float64
```

Figure 18 – Coefficient de corrélation

b. Afficher la matrice de corrélation (voir la figure 19).

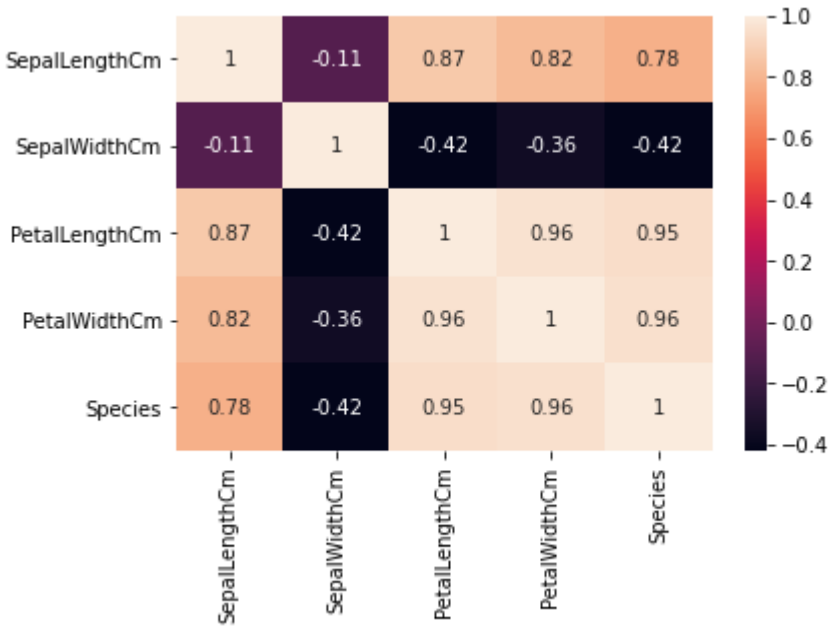


Figure 19 – Matrice de corrélation

Etape 2 :

Sélection d'attributs pour apprentissage non supervisée (k-mens) :

a. Éliminer les attributs qui sont fortement corrélés

Nous comparons la corrélation entre les caractéristiques et supprimons l'une des deux caractéristiques qui ont une corrélation supérieure à 0,8

b. Faire l'apprentissage k-means

On sélectionne 3 groupes puis, aléatoirement, on initialise le centre associé à chaque groupe. Il est préférable de commencer par analyser globalement les données présentes et essayer d'identifier des groupes distincts afin de mieux déterminer le nombre de classes à utiliser.

c. Calculer l'accuracy de k-means (voir la figure 20).

```
Accuracy score: 0.51
```

Figure 20 – Accuracy de k-means

d. Insérer valeur manquante (missing data)

Nous avons inséré (20%, 40%,64%) de missing data dans le dataset, et nous avons fixé outliers de valeur 0.

e. Calculer le score pour chaque attribut de méthode FbsO (voir la figure 21).

```
[['SepalWidthCm', 0.09], ['SepalLengthCm', 0.05], ['PetalLengthCm', 0.04], ['Pe
talWidthCm', 0.02], ['Species', 0.0]]
```

Figure 21 – le score pour chaque attribut de méthode FbsO

f. Nettoyage de dataset

Nous avons nettoyé dataset par le remplacement des valeurs manquantes

g. La corrélation

Après le nettoyage de dataset nous avons fait la corrélation et calculer le coefficient de corrélation pour chaque attribut (voir la figure 22).

```
Species        SepalWidthCm    -0.365577
PetalLengthCm  SepalWidthCm    -0.360616
PetalWidthCm   SepalWidthCm    -0.327942
SepalWidthCm   SepalLengthCm   -0.078695
Species        SepalLengthCm    0.742154
PetalWidthCm   SepalLengthCm    0.759765
PetalLengthCm  SepalLengthCm    0.785818
PetalWidthCm   PetalLengthCm    0.898135
Species        PetalLengthCm    0.914728
               PetalWidthCm     0.930906
dtype: float64
```

Figure 22 – la corrélation après le nettoyage

h. Calculer l'accuracy avec l'apprentissage k-means (corrélation) (voir la figure 23).

	Missing data	Outliers	Acuraccy
0	0.00	0	0.51
1	0.20	0	0.26
2	0.40	0	0.11
3	0.64	0	0.31

Figure 23 – Liste d'accuracy de méthode corrélation en fonction de missing data pour ANS

i. Sélection d'attribut avec notre approche FbsO.

j. Calculer l'accuracy de notre approche avec l'apprentissage k-means (voir la figure 24).

	Missing data	Outliers	Acuraccy
0	0.00	0	0.51
1	0.20	0	0.26
2	0.40	0	0.36
3	0.64	0	0.83

Figure 24 – Liste d'accuracy de FsbO en fonction de missing data pour ANS

k. Insérer outliers

Nous avons inséré (10%, 20%,28%) d'outliers dans le dataset est Calculer le score de chaque attribut pour une méthode non supervisée

l. Nettoyage de dataset

Nous avons nettoyé le dataset par la suppression des outliers

m. Calculer l'accuracy avec l'apprentissage k-means (corrélation) (voir la figure 25).

	Missing data	Outliers	Acuraccy
0	0	0.00	0.51
1	0	0.10	0.50
2	0	0.20	0.27
3	0	0.28	0.09

Figure 25 – Liste d'accuracy de méthode corrélation en fonction des outliers pour ANS

n. Sélection d'attribut avec notre approche FbsO.

o. Calculer l'accuracy de notre approche avec l'apprentissage k-means (voir la figure 26).

	Missing data	Outliers	Acuraccy
0	0	0.00	0.51
1	0	0.10	0.96
2	0	0.20	0.31
3	0	0.28	0.26

Figure 26 – Liste d'accuracy de FsbO en fonction des outliers pour ANS

Etape 3 :

Sélection d'attributs pour l'apprentissage supervisée (k-NN)

Nous suivons les mêmes étapes que l'apprentissage non supervisé, après avoir saisi les missing data et outliers, nous avons calculé l'accuracy avec deux méthodes de sélection d'attributs (sélection classique et FsbO), nous trouvons :

Pour sélection classique (voir les figures 27 et 28) :

	Missing data	Outliers	Acuraccy
0	0.0	0	0.960000
1	0.2	0	0.923333
2	0.5	0	0.820000
3	0.7	0	0.786667

Figure 27 – Liste d'accuracy de méthode corrélation en fonction de missing data pour AS

	Missing data	Outliers	Acuraccy
0	0	0.00	0.960000
1	0	0.10	0.916667
2	0	0.20	0.908333
3	0	0.28	0.892857

Figure 28 – Liste d'accuracy de méthode corrélation en fonction des outliers pour AS

Pour FsbO (voir les figures 29 et 30) :

	Missing data	Outliers	Acuraccy
0	0.0	0	0.960000
1	0.2	0	0.946667
2	0.5	0	0.920000
3	0.7	0	0.913333

Figure 29 – Liste d'accuracy de FsbO en fonction de missing data pour AS

	Missing data	Outliers	Acuraccy
0	0	0.00	0.960000
1	0	0.10	0.924242
2	0	0.20	0.925620
3	0	0.28	0.937500

Figure 30 – Liste d'accuracy de FsbO en fonction des outliers pour AS

4.4 Comparaison entre les deux méthodes

4.4.1 Apprentissage non supervisée

Ce graphique représente « Accuracy de méthode corrélation et accuracy de méthode FsbO » en fonction du « Missing data » de l'apprentissage non supervisée (voir la figure 31).

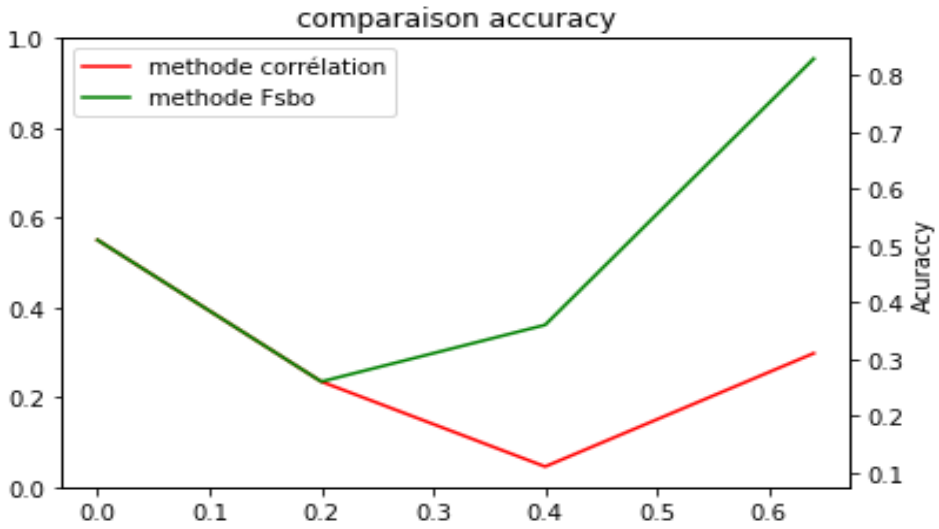


Figure 31 – Graphe du l'Accuracy en fonction de Missing data pour ANS

- La ligne rouge représente « Accuracy de méthode corrélation » en fonction du « Missing data », de 0.0 à 0.4 : on observe une diminution du Accuracy de [0.51 à 0.11] en fonction de Missing data. de 0.4 à0.6 : on observe une augmentation du Accuracy de [0.11 à 0.31] en fonction de Missing data.
- La ligne verte représentée « Accuracy de méthode FsbO » en fonction du « Missing data », de 0.0 à 0.2 : on observe une diminution rapide du Accuracy de [0.51 à 0.26] en fonction de « Missing data ». de 0.2 à 0.6 : on observe une augmentation du Accuracy de [0.26 à 0.83] en fonction de Missing data.

L'Accuracy des deux méthodes diminue des mêmes valeurs jusqu'aux les valeurs 0.2 de Missing data. De 0.2 à 0.6, on remarque que les valeurs de l'Accuracy de FsbO augmentées par rapport aux valeurs de l'Accuracy de corrélation.

Ce graphique représente « Accuracy de méthode corrélation et accuracy de méthode FsbO » en fonction du « Outliers » de l'apprentissage non supervisée (voir la figure 32).

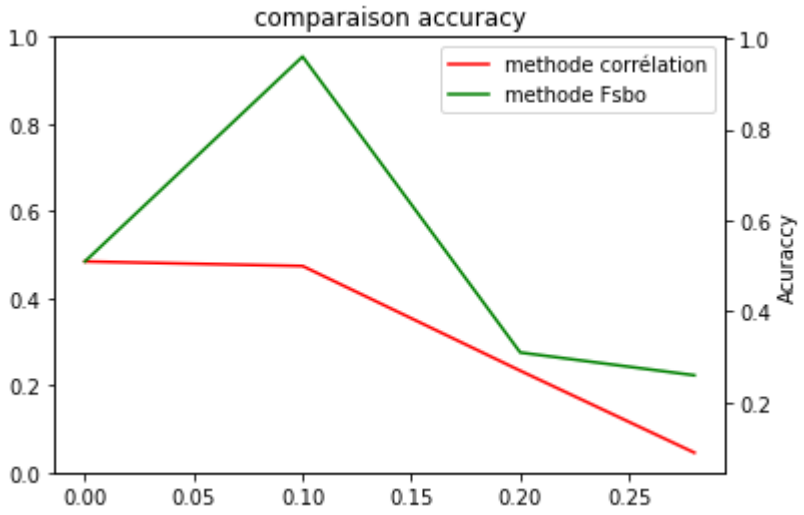


Figure 32 – Graphe de l'Accuracy en fonction des Outliers pour ANS

- La ligne rouge représente « Accuracy de méthode corrélation » en fonction du « Outliers», de 0.0 à 0.10 : on observe une constance de l'Accuracy de [0.51 à 0.50] en fonction des Outliers. de 0.10 à0.28 : on observe une diminution rapide de l'Accuracy de [0.50 à 0.09] en fonction des Outliers.
- La ligne verte représentée « Accuracy de méthode FsbO » en fonction du « Missing data », De 0.0 à 0.10 : on observe une augmentation rapide de l'Accuracy de [0.51 à 0.96] en fonction des « Outliers». de 0.10 à 0.28 : on observe une diminution de l'Accuracy de [0.96 à 0.20] en fonction d'Outliers.

Donc, les valeurs de l'Accuracy de FsbO augmentées par rapport aux valeurs de l'Accuracy de corrélation.

4.4.2 Apprentissage supervisée

Ce graphique représente « Accuracy de méthode corrélation et accuracy de méthode FsbO » en fonction du « Missing data » de l'apprentissage supervisée (voir la figure 33).

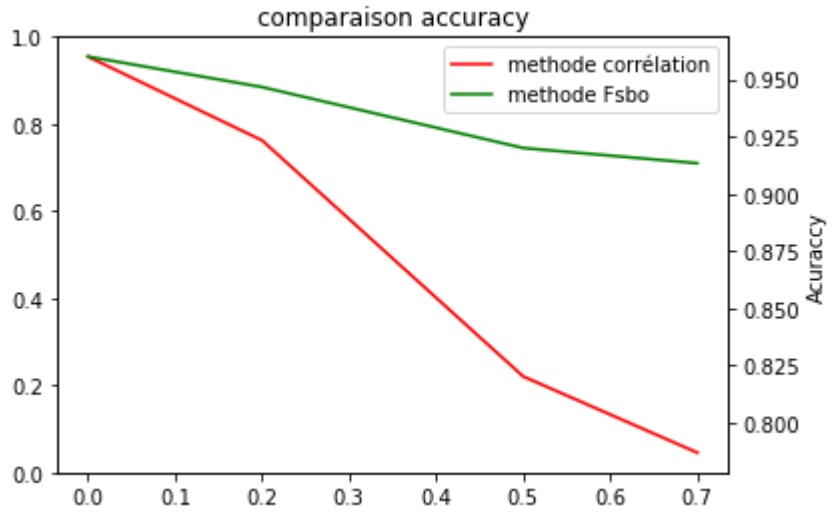


Figure 33 – Graphe de l'Accuracy en fonction de Missing data pour AS

- La ligne rouge représente « Accuracy de méthode corrélation » en fonction du « Missing data », de 0.0 à 0.7 : on observe une diminution rapide du Accuracy de [0.96 à 0.78] en fonction de Missing data.
- La ligne verte représentée « Accuracy de méthode FsbO » en fonction du « Missing data », de 0.0 à 0.7 : on observe une diminution progressivement du accuracy de [0.96 à 0.91] en fonction de « Missing data ».

Donc, Les valeurs de l'accuracy de FsbO augmentées par rapport aux valeurs de l'accuracy de corrélation.

Ce graphique représente « Accuracy de méthode corrélation et accuracy de méthode FsbO » en fonction du « Outliers » de l'apprentissage supervisée (voir la figure 34).

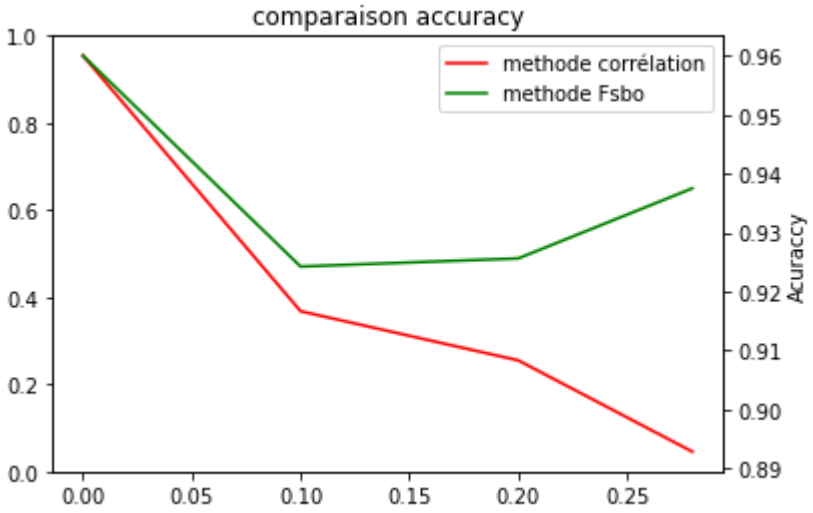


Figure 34 – Graphe de l'accuracy en fonction des Outilers pour AS

- La ligne rouge représente « Accuracy de méthode corrélation » en fonction du « Outliers», de 0.0 à 0.10 : on observe une diminution rapide de l'accuracy de [0.96 à 0.91] en fonction d'Outliers. de 0.10 à 0.28 : on observe une diminution progressivement rapide de l'accuracy de [0.91 à 0.89] en fonction des Outliers.
- La ligne verte représentée « Accuracy de méthode FsbO » en fonction des «Outliers », de 0.0 à 0.10 : on observe une diminution rapide de l'accuracy de [0.96 à 0.92] en fonction de « Outliers». de 0.10 à 0.28 : on observe une augmentation progressivement de l'accuracy de [0.92 à 0.93] en fonction d'Outliers.

Dans ce graphe, les valeurs de l'accuracy de FsbO augmentées par rapport aux valeurs de l'accuracy de corrélation.

Dans le résultat, le score de sélection d'attributs classiques est faible par rapport au score de notre approche FbsO, parce que la sélection d'attributs classiques ne traite pas l'originalité des attributs. Et notre approche FsbO fonctionne dans les deux cas : supervisé est non supervisée, elle prendre en considération le critère d'originalité d'attribut. Elle est basée sur la valeur de score qui sera calculé à base de taux des données perdues et de taux des outliers.

4.5 Conclusion

Dans ce dernier chapitre, nous avons présenté la partie implémentation de notre projet, nous avons aussi parlé des différents langages et logiciels utilisés pour sa réalisation, et nous avons décrit les graphes les plus importants de notre approche.

Conclusion Générale

La sélection des fonctionnalités est une étape très importante dans la construction de modèles de machine Learning. Cela peut accélérer le temps de formation, rendre nos modèles plus simples, plus faciles à déboguer et réduire le temps de mise sur le marché des produits d'apprentissage automatique.

Ce mémoire est divisé en quatre parties. Dans la première partie Nous avons défini l'apprentissage automatique et ses grandes approches.

Et dans la deuxième partie, Nous avons présenté une étude complète et détaillée de la sélection des attributs. Nous avons expliqué les différentes étapes du processus de sélection des attributs et les différentes stratégies de recherches et les différents critères d'évaluation qui peuvent être utilisées dans les algorithmes de sélection des attributs.

Ensuite la troisième partie, nous avons détaillé notre approche de sélection d'attributs qui fonctionne dans les deux cas : supervisé et non supervisé

Dans La quatrième partie, elle consiste à la phase de l'implémentation et la présentation des résultats.

La majorité des travaux de sélection des features à base de filtrage ne considèrent pas l'historique de l'attribut sélectionné. Sachant qu'un attribut peut subir beaucoup de modifications durant la phase de pré-processing talque : traitement des données perdu, suppression des outliers, codage (surtout dans le cas des features de type catégories), il est possible que ce feature soit incapable de présenter vraiment le dataset ou bien d'être utilisé pour prédire des nouveaux résultats du modèle. A partie de là, nous avant proposer le paramètres d 'originalité qui assure que les feautre sélectionné est original et peut-être utilisé pour la modélisation.

Vu le temps limité de notre projet, nous n'avons pas terminé nos objectifs, dans nos perspectives ; nous devons essayer de tester notre approche sur plusieurs datasets et comparer les résultats avec des stratégies de filtrage plus performantes.

Bibliographie

[1] Christopher M. Bishop, «Pattern recognition and machine Learning », edition Springer, 2006.

[2] CPA New Brunswick, « Introduction à l'apprentissage automatique», monographie de CPA nouveau, volume 50,14 out 2019.

[3] Mokhtar Taffar. « Initialisation à l'apprentissage automatique », Support de Cours pour étudiants en Master en Intelligence Artificielle, Université de Jijel, année 2014.

[4] Gavin Hackeling. « Mastering Machine Learning with scikit-learn Second Edition », edition Hackeling, Join 2017

[5] Metomo JOSEPH BERTRAND RAPHAËL. « Machine Learning : Introduction à l'apprentissage automatique ». SUPINFO International Université, 10 Octobre 2017.

[6] Julien Ah-Pine. « Apprentissage automatique ». Volume 90. 2019-2020.

[7] Julie Desjardins, « L'analyse de régression logistique », Tutorial in Quantitative Methods for Psychology, Vol. 1(1), p. 35-41., 2005.

[8] Likas A, vlassis M e Verbeek J, « he global k-means clustering algorithm, Pattern Recognition», volume 36, pp.451-461.,2003 .

[9] Wieland Eckert, « Using Marcov Decision Process for learning dialogue strategies», on

Volume: 1, 2016

[10] Leon Bošnjak, J. Sres, Bosnjak Brumen, «Brute-force and dictionary attack on hashed real-world passwords», volume 6, 1 may 2018

[11] Fabien Moutarde, «les IntelligenceS ArtificielleS pour l'Industrie : quel type pour quelle innovation ? », volume 6,10 févr. 2019.

[12] H. Chouaib, "Sélection de caractéristiques: méthodes et applications," Thèse de doctorat Université Paris Descartes, 2011

[13] BENHAMMADA Sadek, "Etude comparative de méthodes de sélection de caractéristiques en apprentissage automatique. Proposition d'une variante ", Thèse de doctorat Université de Constantine, 2006-2007.

[14] SOUIER Imane et YOUBI Fatiha, «SELECTION DES VARIABLES A BASE DES METAHEURISTIQUES», Thèse de Master Université de Tlemcen, 2015-2016.

[15] Mahdjane Karima, « détection d'anomalies sur les données biologiques par SVM », thèse de master université mouloud Mammeri de Tizi-Ouzou, 14/10/2012.

[16] Melle MENGHOUR Kamilia, « Approches Bio-inspirées pour la Sélection d'Attributs », these de doctorat, Université Badji Mokhtar-Annaba, 2014-2015.

[17] 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.

[18] P. M. Narendra and K. Fukunaga, "A branch and bound algorithm for feature subset selection," IEEE Transactions on Computers, pp. 917-922, 1977

[19] 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

[20] Kabir m.m., shahjahan m., murase k. "involving new local search in hybrid genetic algorithm for feature selection", vol. 5864, pp 150-158. 2009.

[21] K. Kira and L. A. Rendell, "The feature selection problem: Traditional methods and a new algorithm," in Aaai, 1992, pp. 129-134.

[22] Unler A., Murat A., «A discrete particle swarm optimization method for feature selection in binary classification problems », European Journal of Operational Research, Vol. 206, issue 3, pp 528–539, 2010.

[23] Sonkamble, B. « Effective Feature Selection Using Ensemble Techniques and Genetic Algorithm, 6th International Congress on Information and Communication Technology», ICICT 2021; 236:367-375, 2022

[24] Long, X., Qian, W., Wang, Y. et al. « Cost-sensitive feature selection on multi-label data via neighborhood granularity and label enhancement ». Appl Intell 51, 2210–2232 (2021).

[25] Asim, Syed & Shah, Ali & Shabbir, Hafiz & Rehman, Saif ur & Waqas, «Muhammad. A Comparative Study of Feature Selection Approaches: 2016-2020 ». International Journal of Scientific and Engineering Research. 11. 469, 2020

[26] https://projeduc.github.io/intro_apprentissage_automatique/introduction.html?/

[27] M. W. Mwadulo, "A Review on Feature Selection Methods For Classification Tasks," international Journal of Computer Applications Technology and Research, vol. 5, pp. 395-402, 2016.

[28] A. G. Karegowda, A. Manjunath, and M. Jayaram, "Comparative study of attribute selection using gain ratio and correlation based feature selection," International Journal of Information Technology and Knowledge Management, vol. 2, pp. 271-277, 2010.

[29] D. Asir, S. Appavu, and E. Jebamalar, "Literature Review on Feature Selection Methods for High-Dimensional Data," International Journal of Computer Applications, vol. 136, pp. 9-17, 2016.

[30] Elong Nadjla, "La sélection des caractéristiques appropriées en vue d'une classification des images médicales ", Thèse de doctorat Université de Constantine, 2019-2020.

[31] Adel SIDI-YAKHLEF, "Coefficient de corrélation", Université de Tlemcen, 17 pages.

[32] Petit, R. J. et Rubin, D.B. « Analyse statistique avec données manquantes». John Wiley & Fils, Hoboken, Vol. 793, 2019.

[33] https://fr.theastrologypage.com/outlier-detection

[34] https://www.futura-sciences.com/tech/definitions/informatique-python-19349/

[35] https://scikitlearn.org/stable/auto_examples/datasets/plot_iris_dataset.html