République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université BATNA 2

Faculté de Technologie

Département Génie Industriel

Mémoire présenté en vue de l'obtention du diplôme de

Magister en Génie Industriel

Option : Génie des Systèmes Industriels

Par

Djamil Rezki

Ingénieur d'Etat en Informatique

THEME

Système intelligent d'aide à la décision pour le pilotage d'un processus de forage pétrolier

Travail effectué au sein du Laboratoire d'Automatique et Productique LAP (U. BATNA 2)
Directeur de Mémoire **: Pr .L.Hayet Mouss**

Membres du jury :

Pr .K. Nadia Mouss	Université de Batna 2	Professeur	Président
Pr .L. Hayet Mouss	Université de Batna 2	Professeur	Rapporteur
Pr. Abdelkamal Tari	Université de Bejaïa	Professeur	Examinateur
Dr. Hayet Melakhsou	Université de Batna 2	Maitre de conférences	Examinateur

L'année universitaire : 2015/2016

Dédicaces

Je dédie ce modeste travail à :

mes parents

ma femme et mes enfants

mes frères et mes sœurs

toute la famille

mes collègues

Djamil

Remerciements

J'adresse à ma directrice de Mémoire , Professeur L. Hayet Mouss, mes sincères remerciements pour son aide, sa patience et ses encouragements. Je la remercie pour les qualités scientifiques et pédagogiques de son encadrement et pour sa disponibilité. Je souhaite qu'elle reçoit à travers ces lignes toutes les marques de ma reconnaissance.

Je tiens à remercier vivement Le Président du Jury, Professeur Kinza Nadia Mouss, de m'avoir fait l'honneur d'accepter de présider le jury de soutenance de ce mémoire .

Je remercie également les membres de jury, Professeur Abdelkamal Tari et Docteur Hayet Melakhssou d'avoir accepté de faire partie du jury d'évaluation de ce mémoire ainsi que pour le temps consacré à l'étude de celui-ci.

Je présente aussi mes remerciements aux enseignants de département génie industriel Batna : Pr M. Djamel Mouss, Dr Samia Aitouche, Mr Abdelghafour Kanit, Mr Tarek Maaref, Mr Rafik Bensaadi et Melle Hanane Zermane.

Mes remerciements sont adressés Mr Rafik Boudour enseignant à l'IAP Boumerdes pour ses conseils et orientations, je remercie Mr Abdelkamal Zeghib ingénieur outil de forage ENSP pour son aide pendant toute la période de mon mémoire.

Enfin, mes remerciements sont adressés à toutes les personnes ayant contribuées de près ou de loin, à la réalisation de ce travail.

Résumé

L'optimisation efficace de ROP (vitesse de forage) est un élément clé de la réussite du processus de forage pétrolier. En raison de la complexité de pénétration et l'hétérogénéité de la formation, l'approche traditionnelle comme les équations de ROP et l'analyse de régression sont confinés par leurs limitations dans la prédiction de la vitesse de progression de forage. Les méthodes intelligentes comme les forêts aléatoires et nelder-mead simplex deviennent de puissants outils pour obtenir les paramètres optimisés avec l'accumulation des données de la géologie et des journaux de forage. Ce travail de recherche présente une approche de prédiction et optimisation de ROP basée sur l'algorithme des forêts aléatoires et l'algorithme heuristique nelder-mead simplex. L'idée principale est, d'abord, de construire le modèle de prédiction de ROP à partir de données historiques de puits utilisant l'algorithme des forêts aléatoires , puis optimiser les paramètres mécaniques (RPM, WOB) en appliquant l'algorithme nelder-mead simplex. Au cours du processus de modélisation, l'algorithme des forêts aléatoire est réglé par le choix des meilleurs paramètres pour avoir un taux de corrélation maximum en utilisant un méta-classifier dans l'outil WEKA. Nous avons utilisés des données des puits foré dans le champ de hassi terfa situé dans le sud de l'Algérie. Les résultats de l'expérience montrent que l'approche proposée est capable d'utiliser efficacement les données d'ingénierie pour fournir prédiction efficace de ROP et d'optimiser les paramètres de forage de puits, l'application développée est un système d'aide à la décision qui permet aux ingénieurs de forage de prédire et optimiser la vitesse de progression de forage au moyen de la sélection de la meilleure combinaison des paramètres mécaniques (WOB, RPM).

Mots clés : forage pétrolier, prédiction du ROP, les forêts aléatoires, optimisation du ROP, Nelder-Mead simplex, système d'aide à la décision.

Absract

Effective optimization ROP (drilling speed) is key to success oil drilling process. Due to the complexity and heterogeneity of penetration of the formation, the traditional way as ROP equations and regression analysis are confined by their limitations in drilling prediction. The intelligent methods such as random forest and Nelder-Mead simplex become powerful tools for optimized settings with the accumulation of geological data and drilling logs. This research presents an approach for predicting and optimizing ROP based on random forests algorithm and Nelder-Mead simplex heuristic algorithm. The main idea is, first, to build the ROP prediction model from the well of historical data using the algorithm of random forests, and then optimize the mechanical parameters (RPM, WOB) by applying the algorithm nelder- Mead simplex. During the modeling process, random forests algorithm is set by choosing the best settings for maximum correlation coefficient using a meta-classification in the weka tool. We used data from the well drilled in the Hassi Terfa field in southern Algeria. The results of the experiment show that the proposed approach is able to effectively use the engineering data to provide effective prediction ROP and optimize drilling parameters, the developed application is a decision support system that allows drilling engineers to predict and optimize the drilling progression speed by selecting the best combination of mechanical parameters (WOB, RPM).

Key words : oil well drilling, ROP prediction, random forests, ROP optimization ROP, Nelder-mead simplex, support system decision.

ملخص :

التحسين الفعال لسرعة الحفر هو مفتاح نجاح عملية التنقيب عن النفط . نظر لتعقيد التغلغل، وعدم تجانس الطبقات الجيولوجية ، الطرق التقليدية التي تعتمد على استعمال معادلات رياضية لسرعة الحفر والتحليل بالانحدار أثبتت محدوديتها في التنبؤ بسرعة الحفر. الطرق التي تعتمد على الذكاء الاصطناعي مثل الغابات العشوائية تصبح أدوات فعالة للتنبؤ وتحسين سرعة الحفر بالاعتماد على البيانات المستخرجة من سجلات الحفر، يقدم البحث مقاربة للتبوء بسرعة الحفر باستعمال خوارزمية الغابات العشوائية وتحسينها باستعمال خوارزمية التبسيط نالدر-ميد. وقد استخدمنا بيانات من سجلات الحفر من أبار في حقل نفطي في جنوب الجزائر –حاسي الطرفة – وقد أثبتنا أن النموذج المقترح قادر التنبؤ بسرعة الحفر وتحسينها بطريقة فعالة ، البرنامج المنشأ هو نظام دعم القرارات التي تسمح مهندسين الحفر للتنبؤ وتحسين سرعة تطور الحفر عن طريق اختيار أفضل توليفة من المعلمات الميكانيكية (WOB، RPM) .

الكلمات الدلالية : الحفر النفطي، التنبؤ بسرعة الحفر، تحسين سرعة الحفر ، الغابات العشوائية، تبسيط نالدر –ميد ، نظام دعم القرارات.

Liste des figures

Chapitre 1 : Contexte et problématique

Chapitre 2 : Généralités sur le forage pétrolier

Chapitre 3 : Les forêts aléatoires

Chapitre 4 : l'algorithme Nelder-Mead Simplex

Chapitre 5 : Implémentation et expérimentation

Liste des tableaux

Table des matières

Liste des abréviations

ROP : la vitesse de progression de forage

WOB : le poids sur l'outil

RPM : la vitesse de rotation

AV : Viscosité apparente

SPM : le débit de la boue mesuré en coups par minute

RNA : Réseau de neurone artificiel

AHP : Le processus de hiérarchie analytique

IA : intelligence artificielle

K2 : algorithme d'apprentissage de la structure graphe orienté acyclique en réseau bayésien

IC : algorithme d'apprentissage de la structure graphe orienté acyclique en réseau bayésien

K-means : k moyenne algorithme de clustering

BHA : Assemblage de fond

PDC : Poly cristallin de diamant

EM : télémétrie électromagnétique

MWD : les mesures pendant le forage

LWD : digraphie pendant le forage

SBM : Boue synthétique

SVM : Support Vector Machine

CART : l'arbre de décision pour classification et régression

WEKA : outil Learning open source

GNU : License publique générale

UCS : la résistance à la compression uni-axiale

MLP : Architecture multicouche d'un réseau de neurones

BFGS : est une méthode permettant de résoudre un problème d'optimisation non linéaire sans contraintes

SMOreg : qui implémente les supports vecteurs machines pour régression dont l'apprentissage est effectué en utilisant les noyaux polynomiaux ou RBF.

RBF : la fonction de base radiale

CV : la validation croisée

FIS : système d'inférence floue.

BYM : Modèle de Bourgoyne et Young

EoC : Ensemble des classifieurs.

GPP : Méthode des gradients conjugués pré conditionnés

GRNN : la régression généralisée d'un réseau de neurone

Introduction générale

L'optimisation des paramètres de forage est considérée comme un défi unique pour les compagnies pétrolières, le but d'optimisation d'un forage est d'atteindre l'objectif avec un coût minimum. Les technologies de communication et d'informatique(TIC) sont parmi les disciplines les plus importants qui peuvent contribuer à l'optimisation de forage. Une grande quantité de données serait acheminé à travers différents endroits sur la planète dans les manières fiables et efficaces. Les technologies informatiques de pointe sont maintenant capables de stocker de grandes quantités de données, et de résoudre des problèmes complexes.

Depuis les premiers forages les compagnies pétrolières ont toujours cherché à réduire les coûts de forage principalement en augmentant la vitesse de progression de forage(ROP). Dans l'industrie du forage, le premier puits foré dans un nouveau champ aura généralement le coût le plus élevé. Avec l'augmentation de familiarité dans la région, l'optimisation de forage pourrait être mise en œuvre en réduisant les coûts de chaque puits foré à la suite jusqu'à atteindre un point où il n'y a aucune amélioration plus significative.

Un travail important pour optimiser efficacement ROP est la modélisation et la prévision. Malheureusement, la plupart des méthodes de prévision de ROP existants reposent essentiellement sur les expérimentations physiques sur terrain. Il est difficile de pratiquer ces méthodes sur site, il est donc nécessaire de trouver une méthode de prédiction de ROP commode et relativement précise.

Les principales variables de forage considérées comme ayant un effet sur la vitesse de progression de forage (ROP) ne sont pas entièrement comprises et sont complexes à modéliser. Pour cette raison le modèle mathématique de haute précision pour la vitesse de forage rotatif n'a pas été réalisé jusqu'à présent. Il existe de nombreux modèles mathématiques proposés qui ont tenté de combiner les relations connues de paramètres de forage. Les modèles proposés tentent d'optimiser le fonctionnement du forage au moyen de la sélection du meilleur poids sur l'outil et la vitesse de rotation pour atteindre le coût minimum, malheureusement ces modèles mathématiques ont montré des limitations surtout en précision dans la prédiction de vitesse de forage. Les méthodes intelligentes comme les forêts aléatoires et Nelder-Mead simplex deviennent de puissants outils pour obtenir les paramètres optimisés avec l'accumulation des données de la géologie et des journaux de forage.

L'objectif de cette étude est de développer un modèle de prédiction et optimisation de la vitesse de forage en utilisant des techniques intelligentes, les forêts aléatoires pour la

prédiction du ROP en suite l'algorithme du simplexe heuristique Nelder-Mead pour l'optimisation du ROP au moyen de la sélection du meilleurs poids sur l'outil (WOB) et vitesse de rotation (RPM).

Les données historiques des puits forés dans le champ de Hassi-Terfa situé au Sud de l'Algérie, ont été utilisées pour valider notre modèle.

Le modèle développé peut servir comme outil d'aide à la décision pour les ingénieurs de forage dans le but de bien prédire et optimiser la vitesse de progression de forage.

Ce mémoire est structuré autour de cinq chapitres :

Le chapitre« 1 » présente un état l'art des travaux réalisés sur la prédiction et l'optimisation de la vitesse de progression de forage pétrolier. Ces travaux basés sur des modèles mathématiques et des techniques de l'intelligence artificielle, suivi par une critique constructive et une description de l'approche proposée pour prédire et optimiser la vitesse d'avancement.

Le deuxième chapitre présente des généralités sur le forage pétrolier. Il commence par un historique sur le forage rotary, une description de forage et enfin la vitesse de progression avec les paramètres qui l'affectent.

Le chapitre « 3 » est dédié à notre algorithme de prédiction « les forêts aléatoires » qui fait partie des méthodes d'apprentissage ensemblistes qui ont pour objectifs de rendre un prédicteur moyen en un prédicteur très efficace à travers la combinaison de plusieurs classifieurs sur des ensembles de base de données d'apprentissage et les agréger pour produire un classifieur plus efficace.

Le chapitre « 4 » est consacré à l'algorithme simplex heuristique nelder-mead très utilisé pour résoudre les problèmes d'optimisation pour les fonctions non linéaires et non dérivables. Il se caractérise par la simplicité de mise en œuvre et la vitesse de convergence.

Le chapitre « 5 » est réservé à l'implémentation de notre approche. Il commence une par comparaison entre trois algorithmes learning machine pour la régression :les forêts aléatoires , les réseaux de neurones artificiels et les SVM en utilisant l'outil learning machine open source WEKA, après le choix de l'algorithmes des forêts aléatoires une optimisation des paramètres est effectuée dans le but d'augmenter la précision de l'algorithme , ensuite l'utilisation de l'algorithme Nelder-Mead pour optimiser la vitesse d'avancement au moyen de la sélection du meilleur poids sur l'outil et vitesse de rotation, le langage java a été utilisé sous l'environnement NETBEANS 8.0 pour implémenter notre application.

Nous avons terminé notre travail par une conclusion avec quelques perspectives.

Chapitre 1 : Contexte et problématique

Résumé

Dans ce chapitre nous allons présenter un l'état de l'art sur les travaux réalisés dans le cadre de la prédiction et l'optimisation des paramètres du forage pétrolier.

Le but d'optimisation d'un forage est d'atteindre l'objectif avec un coût minimal, la vitesse de progression de forage (ROP) joue un rôle primordial dans la détermination de coût forage, la plupart des méthodes de prévision du ROP se reposent sur l'expérience humaine, ces expériences ont montré des limitations.

Les principales variables de forage considérées comme ayant un effet sur la vitesse de progression de forage (ROP) ne sont pas entièrement comprises et sont complexes à modéliser, ce qui rend très difficile l'obtention d'une relation mathématique entre ces variables, plusieurs modèles mathématiques ont été développés pour répondre à ce problème parmi eux le modèle Bougoyne et Young(BYM) le plus utilisé dans la pratique, malheureusement ces modèles n'étaient pas efficaces au vu du manque de précision. Dans cette vision, les méthodes intelligentes et d'apprentissage automatique deviennent un choix stratégique pour améliorer la prédiction du forage, plusieurs travaux ont été réalisés axés sur les RNA, la logique floue et autres techniques. Notre approche proposée se repose sur un algorithme d'apprentissage automatique « les forêts aléatoires » qui fait partie des méthodes ensemblistes et qui a pour objectif de rendre un prédicteur individuellement faible en prédicteur efficace et performant. Aussi, un algorithme de simplex heuristique Nelder-Mead qui permet de d'optimiser une fonction objectif non linéaire et non dérivable a été développé.

1.1 Introduction

Dans ce chapitre nous allons présenter un l'état de l'art sur les travaux réalisés dans le cadre de la prédiction et l'optimisation des paramètres du forage pétrolier.

L'efficacité des coûts dans les projets de forage pétrolier devient un aspect très important de nos jours. Les efforts visant à prédire les effets des paramètres de forage et d'optimiser un tel coût ont été largement effectuée dans de nombreuses études. Ces études visent à augmenter les performances et de réduire la probabilité de rencontrer des problèmes. Dans la plupart des cas, le coût d'un forage est réduit par augmentation de la vitesse de progression de forage. Ceci est principalement fait en maximisant la vitesse de progression de forage (ROP). Cette dernière dépend de nombreux autres paramètres de forage.

La prédiction de ROP aide à sélectionner les meilleurs paramètres d'entrée pour obtenir la vitesse de progression la plus élevée de forage avec le moindre coût. Ainsi, il a été l'objet de nombreux travaux de recherche. Les recherches se poursuivent pour trouver des résultats plus précis.

D'autre part, les applications des méthodes d'intelligence artificielles (IA) dans l'ingénierie pétrolière ont récemment émergé comme de puissants outils conduisant à une nouvelle génération d'outils d'aide à l'analyse pour les praticiens, les scientifiques et ingénieurs travaillant dans les domaines de l'industrie pétrolière [26].

Actuellement, les techniques de calcul et de modélisation disponibles pour la prédiction ROP mettent en œuvre des modèles de régression multiple, la recherche opérationnelle, les réseaux de neurones artificiels et la simulation. Les paramètres qui affectent ROP sont difficiles à modéliser. Des paramètres d'entrée différents sont utilisés dans les études. Le poids sur l'outil (WOB) et la vitesse de rotation par minute (RPM) sont les principaux paramètres qui sont utilisés dans la littérature la plus citée [26].

Notre travail de recherche consiste à mettre en place un modèle de prédiction ROP en utilisant une autre technique de l'IA appelée les forêts aléatoires, elle présente certains avantages par rapport aux techniques présentées dans la littérature, une optimisation du ROP avec l'algorithme heuristique Nedler-Mead (downhill) simplex qui fait partie des méthodes de recherche directe pour minimiser les fonctions non linéaires sans dérivés.

L'application réalisée joue le rôle d' un système d'aide à la décision, elle permet aux ingénieurs et opérateurs de forage de prédire et optimiser la ROP en se basant sur la meilleure combinaison des paramètres mécaniques WOB et RPM.

1.2 Contexte d'étude

L'importance considérable de l'énergie pétrolière au plan économique justifie la rude concurrence actuelle et la recherche constante d'innovation. Dans ce secteur d'activité, qui nécessite l'optimisation les procédures de prospection, d'extraction et de transport de cette ressource pour minimiser les coûts. Par ailleurs, la croissance des besoins en énergie pétrolière en raison des évolutions sociales, démographiques et des progressions technologiques conduit à mener des investigations poussées afin de satisfaire cette demande.

De plus, l'augmentation de la capacité de production, nécessite des moyens performants et fiables. Le système de forage rotary est à la base du processus d'extraction du pétrole. Il est crucial de souligner que la réduction du temps de forage ainsi que la préservation des équipements sont conditionnées par une conduite appropriée.

Le but de l'optimisation d'un forage est de parvenir à l'objectif avec un prix de revient minimum, la plus grande partie de ces coûts sont proportionnels au temps de forage qui est à son tour lié aux travaux d'avancement, donc aux différents facteurs qui conditionnent la vitesse de progression de forage. Ces différents facteurs sont appelés : les paramètres de forage.

La connaissance et l'utilisation des paramètres de forage optimaux permettent de minimiser le temps de forage, donc le coût, ce qui est le premier but de forage, atteindre l'objectif à un prix minimal.

1.3 Positionnement de problème

Les opérations de forage sont les processus les plus coûteux dans l'industrie pétrolière. Les entreprises sont toujours intéressées à trouver des moyens pour le forage le plus sûr, ainsi que le plus économique. Ainsi, l'optimisation de forage devient une question cruciale pour les entreprises de forage.

L'objectif fondamental de l'optimisation de forage est d'atteindre le plus grand degré d'efficacité possible dans des conditions spécifiées, en essayant d'obtenir le résultat le plus élevé ou le plus bas d'une fonction objectif. Ainsi, en général, la technique d'optimisation implique la formulation de la fonction objectif, l'identification des variables contrôlables, dépendantes et indépendantes, des limitations techniques et technologiques ou de contraintes.

L'optimisation de forage est habituellement effectuée en utilisant des modèles pour l'estimation de ROP ainsi que le coût par mètre.

Cette étude vise à développer un outil intelligent d'aide à la décision pour prédire la vitesse de progression de forage (ROP) et permettre l'optimisation des paramètres de forage pour avoir un meilleur avancement en se basant sur les techniques de l' IA.

Le classifieur des forêts aléatoires est utilisé pour la prédiction, son avantage réside dans la capacité d'estimer la ROP en fonction des paramètres de forage indépendants avec une haute précision.

Après la prédiction avec la précision souhaitée, nous cherchons à optimiser la ROP au moyen de la sélection des meilleures combinaison des paramètres WOB et RPM en utilisant l'algorithme Nelder-Mead Simplex qui prend en charge l'optimisation d'une fonction objectif continue non dérivable, il se caractérise par la simplicité et la vitesse d'exécution.

Notre application est un système d'aide à la décision qui permet aux ingénieurs et opérateur de forage de prédire avec une grande précision la ROP, ensuite son optimisation.

1.4 Etat de l'art

La prédiction de vitesse de progression de forage est cruciale pour l'amélioration de la performance de forage. Cependant, un grand nombre de facteurs et événements imprévus influencent cette vitesse et ont font un processus complexe et stochastique. Par conséquent, la vitesse de progression de forage est restée un paramètre difficile à prédire au cours des dernières décennies.

Des nombreuses études ont été mises en œuvre pour la prédire Aussi les modèles mathématiques et les techniques de l'intelligence artificielle ont été utilisés pour résoudre ce problème.

Il est à signaler que la vitesse de progression de forage (ROP) est affectée par de nombreux paramètres tels que, le poids sur l'outil(WOB), la vitesse de rotation (RPM), le type d'outil, les propriétés de la boue et les caractéristiques de la formation [1]. Malheureusement, il n'existe pas de relation mathématique explicite entre la vitesse de forage et les différents facteurs de forage. C'est en raison de grand nombre de paramètres de forage qui influent sur la vitesse de progression de forage. En outre, la relation de ces facteurs entre les uns et les autres et la vitesse de progression de forage est non linéaire et complexe [2].

Cependant, les experts ont mis en avant quelques suggestions pour résoudre ce problème. Ils ont réussi à modéliser les effets des différents paramètres de forage comportant une vitesse de forage comme des fonctions mathématiques. Une de ces méthodes est le modèle de Bourgoyne et Young (BYM) largement utilisé en pratique [3](illustré dans la Figure 1.1).

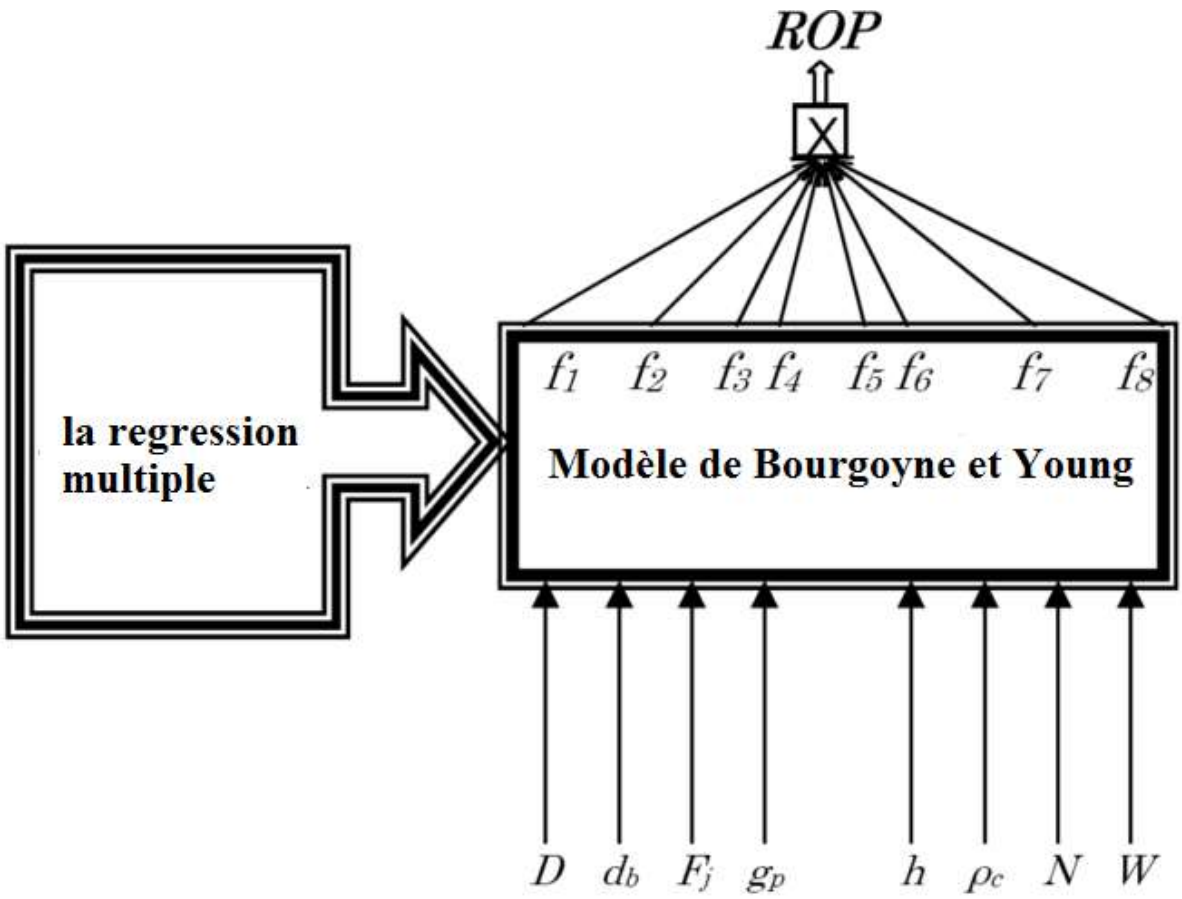

Figure 1.1 : Modèle de Bourgoyne et Young (BYM) [3]

Bourgoyne et Young [3] ont introduit des modèles simplifiés, qui cartographient les variables de forage importantes dans sa vitesse. Ce modèle se base sur huit fonctions, la relation des fonctions mentionnées et la vitesse de forage peuvent être beaucoup plus complexe dans la pratique.

Il est important de noter qu'il y a des paramètres inconnus ou des coefficients dans ce modèle, qui doivent être déterminés sur la base des expériences de forage antérieurs dans le domaine. La méthode de détermination de ces coefficients a un impact significatif sur la précision du modèle. Les concepteurs de modèle BYM ont suggéré une méthode de régression multiple pour déterminer les coefficients inconnus [3]. Cependant, l'application de la méthode de régression multiple ne garantit pas d'atteindre des coefficients et des fonctions physiques significatives.

Pour atteindre des résultats significatifs, l'ajustement des données par les moindres carrés non linéaires avec la méthode de région de confiance ont contribuées à la résolution de ce

problème [4]. Cette méthode minimise la somme de la fonction des erreurs carré. La méthode est basée sur la méthode de Newton-reflet intérieur.

Dans chacune des itérations, la solution approchée d'un grand système linéaire est estimée en utilisant la méthode des gradients conjugués pré conditionnés (GPP) [5,6]. Cette technique permet de déterminer les limites inférieures et supérieures pour les résultats et les limite à être dans les fourchettes raisonnables [6]. Cependant, cette technique ne donne pas une précision raisonnable.

Moradi et al [7] ont introduit un nouveau modèle de forage en utilisant Soft Computing. Bien que cette méthode a amélioré légèrement la précision, elle ne fournit pas d'informations sur la forabilité des différentes formations du champ. En d'autres termes, cette approche fonctionne comme une boîte noire, qui reçoit des entrées et calcule la vitesse de forage en sortie (illustré dans la figure 1.2).

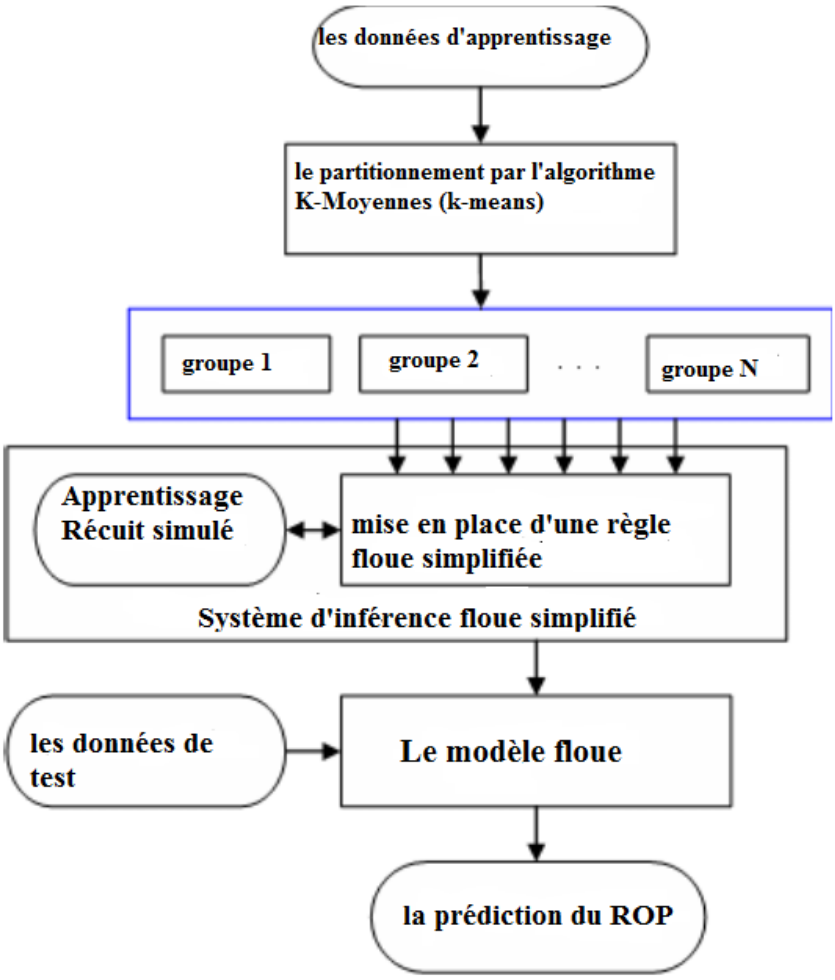

Figure 1.2 : Schéma de la plateforme de prédiction [7]

Bahari et al [8] ont utilisé les algorithmes génétiques pour fournir les coefficients physiquement significatifs au modèle de Bourgoyne et Yong BYM, Dans ce procédé, GRNN (general regression neural network) est utilisé pour découvrir une relation non linéaire et complexe entre la vitesse de forage précitée et huit fonctions de modèle BYM. Cette méthode non seulement augmente la précision de la prédiction considérablement, mais elle fournit également des informations de forage requis du champ comme la forabilité, l'application la méthode proposée à un champ de gaz iranien visualise l'efficacité de cette méthode dans la prédiction de vitesse de progression de forage.

Dans une étude de prédiction des paramètres de forage au Koweït [9] , les réseaux neuronaux artificiels sont utilisés avec succès pour prédire les paramètres de forage . Trois modèles ont été mis au point pour prédire respectivement le type d'outil, la vitesse de progression (ROP), et le coût par pied. Les trois modèles de forage ont été testées avec des données provenant des champs situés au Koweït. Les résultats montrent que le type d'outil, ROP, et le coût par pied peuvent être estimés de manière efficace pour le nouveau puits à forer. Cela peut être prédit avec les modèles développés par les réseaux de neurones. La prédiction de paramètres de forage avec la méthode développée permet de diminuer le pourcentage d'essais et d'erreur résultant des économies de coûts.

Une application des réseaux de neurones artificiels a été utilisée pour l'estimation de la vitesse de progression de forage (ROP) parmi les paramètres de forage obtenus à partir de l'un des champs de pétrole du sud Iranien. La méthodologie permet au personnel de l'industrie de forage d'estimer le ROP non seulement durant la procédure de la planification mais aussi en cours de forage. Les résultats de simulation montrent que l'approche des réseaux de neurones artificiels est supérieure aux méthodes classiques dans précision de la prédiction de vitesse de progression de forage [10].

Dans une étude qui traite l'optimisation des coûts de forage dans des environnements de haute complexité et les risques tels que ceux liés à la région de mer du Brésil. Dans cette recherche, les techniques suivantes sont étudiées: une approche d'inférence bayésienne pour cibler le processus de déclenchement et la combinaison ultérieure de modèles; et un système d'inférence dynamique en évolution Neuro-Flou (DENFIS). L'utilisation d'un réseau non hiérarchique (naïf Bayes) pour classifier les ROP n'était pas suffisant pour un bon classement des valeurs d'entrée. Ce genre de comportement montre la complexité du domaine: il n'est pas possible de dire que les variables elles sont mutuellement indépendants. Les aautres raisons possibles de la faible qualité de la classification sont directement liés à la qualité des données et de la division des classes pour chaque variable qui compose les nœuds du modèle d'entrée.

Tous ces facteurs rendent cette architecture de réseau hautement complexe et sa détermination n'est pas une tâche simple. L'amélioration cette approche nécessite l'utilisation des algorithmes d'apprentissage bayésiens avancés tels que K2, IC afin de trouver topologie efficace du réseau. Le DENFIS est capable de bien prédire l'opération de forage, la réalisation d'une erreur faible, mais les règles créées sont extrêmement compliqués pour l'utilisation. C'est un problème introduit par le système d'inférence floue (FIS) de type (Takagi-Sugeno) et pourrait être résolu en utilisant une approche qui s'appuie sur le système d'inférence floue de type Mandami FIS [11].

J. Ning [12] a proposé une méthode de prédiction appelée le modèle combiné de AHP (Analytic Hierarchy Process) et réseaux neuronaux rétro propagateurs (Back Propagation Neuron Network) basé sur l'exploration de données. Le modèle a été construit en utilisant UCS (la résistance à la compression uni axiale), le dimètre de l'outil, le type d'outil, le coefficient de forabilité, les heures brutes forées, WOB (poids sur l'outil), RPM (vitesse de rotation), la densité de la boue de forage et l'AV (de viscosité apparente) comme paramètres d'entrée. Il a utilisé AHP pour quantifier le poids de chaque paramètre parce que chaque paramètre a une influence différente sur la ROP , puis le réseau neuronal pour prédire le taux de forage. Ce modèle permet d'améliorer la vitesse de convergence et la fiabilité des résultats. La validité de cette méthode a été démontrée avec les données d'un champ existant dans le nord-ouest de la Chine. En outre, le modèle permet d'évaluer l'état d'un puits après le forage et de fournir une aide à la décision dans l'optimisation des paramètres de forage, le choix de trépan et la conception de forage (illustré dans la Figure 1.3).

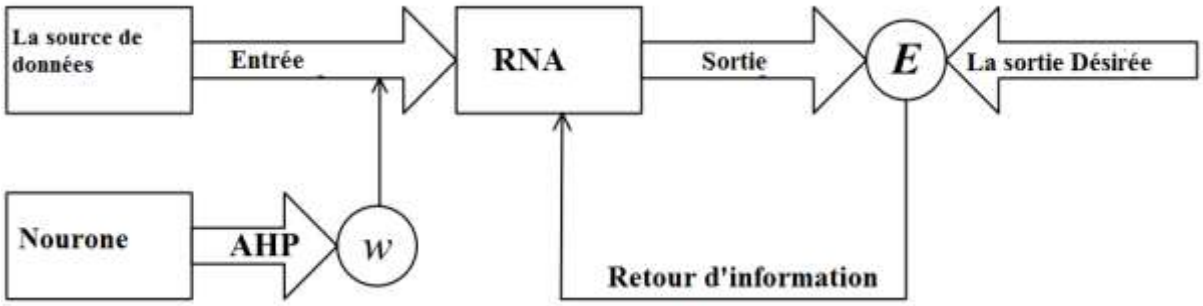

Figure 1.3 : Schéma de la méthodologie AHP-BPNN [12]

Altmis[13] a utilisé les réseaux de neurones artificiels pour prédire les paramètres de forage. Elle a utilisé un ensemble de données générées par un simulateur de forage avancé de grand diamètre. Les paramètres utilisés pour l'apprentissage du réseau neuronal sont la rotation par minute (RPM), le temps de forage, le type de l'outil, le poids sur l'outil(WOB), l'abrasivité de la formation, la forabilité de la formation, l'usure des roulements de l'outil, l'usure des dents

de l'outil, et le débit de la boue mesuré en coups par minute (SPM). Certaines des données d'Atlmis ont été obtenues à partir des champs dans les Etats-Unis, mais seulement RPM, le temps de forage, le type de l'outil, le poids sur l'outil (WOB), le couple de rotation (Torque), ROP, et les paramètres de SPM ont été inclus pour la prédiction.

Wojtanowicz et Kuru [14] ont présenté une nouvelle méthodologie dans l'optimisation de forage en utilisant une programmation dynamique (la dynamique stratégie de forage). La stratégie dynamique de forage est une nouvelle méthodologie de planification et de contrôle de processus de forage. Elle combine la théorie de contrôle unique de l'outil avec un programme de forage multi-outils optimal pour un puits. Dans l'étude de simulation, la stratégie dynamique de forage a été comparée à l'optimisation de forage conventionnel et pratique de terrains typiques. La méthode est apparue plus rentable pour les trépans PDC coûteux et durables grâce à une meilleure utilisation de leur performance et de réduction du nombre d'outils nécessaires pour un puits.

Bilal Esmael [15] a proposé une approche pour classifier les opérations de forage automatiquement en utilisant des techniques d'apprentissage machine. Cette approche prend en entrée les données des capteurs dans un intervalle de temps spécifique, et prédit l'opération de forage. L'approche est simple mais efficace, où pour chaque capteur de données (canal) une liste des caractéristiques statistiques est extraite, puis propose des algorithmes de sélection qui seront utilisés pour sélectionner les fonctions les plus informatives, et enfin, un classificateur est formé basé sur ces caractéristiques. Dans cette approche, de nombreux algorithmes fonction de pondération et de sélection ont été testés pour trouver des mesures statistiques pour distinguer clairement entre les nombreuses opérations de forage différentes. En outre, de nombreuses techniques de classification ont été utilisées pour trouver la meilleure solution en termes de précision et de vitesse. L'évaluation expérimentale avec des données réelles, à partir de quatre scénarios de forage différentes, montre que cette approche a la capacité d'extraire et de sélectionner les meilleurs éléments et de construire des classificateurs précis. La performance des classificateurs a été évaluée en utilisant la méthode de validation croisée.

Dans cette étude, un modèle basé sur les réseaux de neurones artificiels (RNA) a été conçu pour prédire le ROP en utilisant des données de terrain recueillies réels dans un champ pétrolifère Iranien (Ahwaz champ pétrolifère). Le modèle a réussi à prédire ROP . Pour obtenir les paramètres de fonctionnement qui conduisent à un maximum ROP, l'équation mathématique correspondante d'un modèle de réseaux de neurones a été mis en œuvre dans une procédure utilisant un algorithme génétique, qui est l'une des méthodes les plus fiables de

l'optimisation, et à différentes profondeurs les paramètres conduisant à un maximum ROP étaient obtenu. Ce modèle et ses résultats peuvent être utilisés dans les formations « Pabdeh » et « Gurpi » dans tous les champs pétroliers Iraniens et formations schisteuses similaires au Moyen-Orient tels que les formations Irak, Jaddia et Aaliji correspondant à la formation et à la formation Pabdeh Shiranish correspondant à Gurpi[16].

Dans une autre étude, les différentes approches de prédiction de vitesse de progression ont été testés pour trouver le modèle le plus précis et enquêter sur les conditions pour que chaque modèle fonctionne bien. Une approche a été développée pour prédire la vitesse de progression en utilisant la logique floue, et ce modèle a été utilisé en comparaison. Les résultats illustrent bien que lorsqu'ayant une grande quantité de données, les réseaux de neurones artificiels fonctionnent beaucoup mieux que d'autres approches dans la prédiction de la vitesse de progression. En outre, il a été constaté que les équations mathématiques présentées sont des outils prédictifs faibles bien qu'ils sont simples à utiliser [17].

R. Arabjamaloei et al [18] ont proposé une approche pour optimiser la trajectoire du puits pour atteindre la vitesse de progression (ROP) maximale ainsi que la stabilité maximale possible du puits. Pour cela, un modèle qui prédit la ROP dans un puits directionnel a été développé en utilisant des réseaux neuronaux artificiels (RNA) à partir des 15 paramètres d'entrée. Dans la modélisation, en plus de l'azimut et l'angle de la trajectoire du puits, des paramètres d'opération de forage et des contraintes principales de la région ont été inclus en tant qu'entrées. Le processus d'optimisation a été ensuite réalisé pour atteindre la vitesse maximale de pénétration de proposer l'azimut et l'angle de trajectoire correspondante. Enfin, la trajectoire prévue et a été vérifiée pour examiner la stabilité du puits. Comme le résultat final d'une trajectoire de puits qui fournit le taux maximum de pénétration ainsi que la meilleure stabilité de puits a été conçu et proposé. Ce travail examine également les propriétés des différentes formations existantes dans le chemin de puits, contrôle la direction de frapper la cible souhaitée à la profondeur spécifiée.

S. Edalatkhaha et al [19] ont proposé une approche. deux modèles ont été développés en utilisant les réseaux de neurones artificiels. Le premier modèle permet la sélection de l'outil de forage appropriée en fonction de la vitesse de progression (ROP) désirée doit être obtenu par l'application des paramètres de forage spécifiques. Le deuxième modèle utilise des paramètres de forage appropriés obtenus à partir d'une procédure d'optimisation pour sélectionner l'outil de forage qui fournit la vitesse de progression maximale peut être atteinte. Les algorithmes génétiques sont appliqués pour aider à l'optimisation des outils et ses

paramètres de forage connexes. Avec les ensembles de données fournies, ces modèles prédits avec succès les types des outils et les paramètres optimaux de forage.

Notre proposition consiste à utiliser l'algorithme des forêts aléatoires , qui fait partie des méthodes d'ensembles qui permettent de construire une collection de prédicteurs et à agréger l'ensemble de leurs prédictions. L'utilisation des méthodes à ensemble garantie la précision et l'efficacité, ensuite l'algorithme Nelder-Mead simplex pour optimiser les paramètres de forage.

Nous constatons que pour les travaux réalisés précédemment utilisant des modèles mathématiques ne donnent pas une bonne précision. D'autres utilisant les techniques de l'IA comme les réseaux de neurones artificiels(RNA) ont donnés de meilleurs résultats. Néanmoins ces derniers restent moins performants par rapport l'algorithme adopté à savoir « les forêts aléatoires ». Cette performance réside aussi bien dans la vitesse que de la précision. Suite aux tests que nous avons effectués dans l'environnement Learning machine WEKA sur des données de forage obtenus du champs de Hassi Terfa situé au Sud de l'Algérie, nous avons également constaté que l'algorithme des forêts aléatoires est plus précis et plus rapide par rapport aux autres techniques comme les SVM, la régression linéaire, les réseaux de neurones et les arbres de décision classiques, pour l'optimisation du ROP nous avons utilisé l'algorithme heuristique Nelder-Mead simplex qui est très utilisé dans le domaine de la simulation et les fonctions bruitées. Il est simple, efficace et bien adapté à ce genre de problème. Pour toutes ces raisons nous avons opté pour l'algorithme des forêts aléatoires pour la prédiction et l'agorithme Nelder-Mead Simplex pour l'optimisation.

1.5 Conclusion

Un aspect important de l'industrie pétrolière est la prédiction de la vitesse de progression (ROP) . De nombreuses études ont été mises en œuvre à prédire. Principalement, les modèles mathématiques et modèles de réseaux neuronaux artificiels ont été utilisés, dans ce chapitre l'objectif est de présenter un état de l'art des travaux réalisés sur la prédiction et l'optimisation des paramètres de forage pétrolier avec une critique constructive de ces travaux suivie par notre approche et ses avantages argumentée par des tests sur des données réelles venues d'un champ au sud de l'Algérie.

Le chapitre suivant présente des généralités sur le processus de forage pétrolier qui joue un rôle très important dans l'industrie pétrolière.

Chapitre 2 : Généralités sur le forage pétrolier

Résumé

Après avoir présenté l'état de l'art des travaux réalisés sur la prédiction des paramètres de forage de pétrolier, nous passons maintenant à la présentation de quelques généralités sur le forage pétrolier.

Le forage est un processus très important dans l'industrie pétrolière. Il est effectué à l'aide des machine appelées appareils de forage. Ces appareils se composent de plusieurs systèmes : Système de suspension , système rotary, système de circulation de boue, système de production d'énergie et système de contrôle du puits.

Le forage est un processus graduel combinant plusieurs phases. Lors de chaque phase, l'assemblage de fond est remonté à la surface pour la maintenance du processus, pour le tubage et la cimentation de la partie forée.

Plusieurs méthodes sont utilisée pour transmettre les données forage parmi elles : la télémétrie par modulation, la télémétrie électromagnétique , les trains de tiges câblés et les outils dits accessibles(récupérables).

Il existe deux types de paramètres de forage :

1)les paramètres mécaniques : le poids sur l'outil, la vitesse de rotation et le couple de torsion.

2)Les paramètres hydrauliques : le type de boue, le débit de pression hydraulique et la densité de la boue.

La vitesse de forage joue un rôle très important dans l'opération de forage, elle est influencée par plusieurs facteurs qui dépendent essentiellement de : la formation, l'outil et la boue.

2.1 Introduction

Après avoir présenté l'état art des travaux réalisés sur la prédiction des paramètres de forage pétrolier, nous passons maintenant à la présentation de quelques généralités sur le forage pétrolier.

Le forage est l'opération de désagrégation mécanique des roches en vue de pénétrer progressivement dans le sous-sol et d'atteindre l'aquifère situé à une certaine profondeur .

Le forage est une activité importante dans la recherche et l'exploitation des hydrocarbures. Il complète la prospection géologique et géophysique; Il précède la mise en production des hydrocarbures.

Le processus de forage est effectué à l'aide des machines, appelées appareil de forage, qui consiste en une combinaison de nombreux systèmes. Ce sont les masses-tiges, vissés sur le fond de l'assemblage de tube de forage au-dessus du trépan, qui fournissent le poids nécessaire, et éviter le flambage des tiges de forage au-dessus. Les masses-tiges de forage, ainsi que des tiges de forage et l'outil tous composent la chaîne de forage, qui est mise en rotation par la table rotative et le Kelly. Les éléments constitutifs du train de tiges sont creux dans l'axe, de sorte que le fluide de forage peut circuler vers le bas de l'outil. Un joint tournant étanche aux fluides, la tête d'injection, est situé au sommet de la Kelly et fournit une connexion entre la conduite de refoulement de la pompe à boue et à l'intérieur du train de tiges. Un système de levage est nécessaire pour supporter le poids de la chaine de forage, descendu dans le trou et tiré. Ceci est la fonction du derrick, le crochet et les travaux de tirage. L'appareil de forage est doté avec des installations pour traiter le fluide de forage quand il revient à la surface, une zone de stockage pour les produits tubulaires, des abris et des bureaux sur place(illustré La figure 2.2).

En outre, quand un puits est foré, il est régulièrement tubé. il est bordée de tuyaux en acier, ou boîtier, qui est descendu dans le trou sous son propre poids en diamètres plus en plus petits que le trou obtient plus profondément.

Afin de forer un puits, trois facteurs doivent être mis en place simultanément; i) une certaine charge doit être appliqué sur le trépan, ii) l'outil doit être mis en rotation, et iii) un fluide de forage doit être distribué à l'intérieur du puits de forage.

Le forage pétrolier nécessite deux constituant majeur i) la main-d'œuvre ii) le système matériel. La main-d'œuvre comprend un groupe d'ingénierie de forage et un groupe de commandes de l'appareil de forage. La première offre un soutien technique pour les opérations de forage optimales, y compris la sélection des appareils de forage, le programme

de conception la boue, le programme de tubage et de cimentation, le programme hydraulique, programme des outils , le programme de la chaine de forage et le programme de contrôle de puits.

Après le démarrage de forage, les opérations quotidiennes sont gérées par un groupe d'opérateurs de forage dirigé par un chef de chantier. Les systèmes matériels qui composent un appareil de forage rotatif sont i) un système de production d'énergie, le système ii) de levage, iii) le système de circulation de fluide de forage, iv) le système rotatif, v) le système de contrôle des éruptions de puits, et vi) le système de suivi de l'acquisition des données de forage.

2.2 L'histoire de forage

En 1889, à Titusville(Pennsylvanie), le pétrole jaillissait pour la première fois sur le sol des Etats-Unis, d'un puits foré à 69 ft (environ de 21 m). Le Colonel Drake venait d'entrer dans l'histoire de l'exploration pétrolière. Mais, même si cet évènement a marqué le début industriel du forage pétrolier, il ne faut pas oublier les très nombreux puits forés bien avant pour la production d'eau, de saumure et déjà du naphte utilisé pour le calfatage, l'éclairage ou la médecine.

Tous ces forages anciens, y compris celui de Drake était foré par le battage (illustré dans la Figure 2.1).

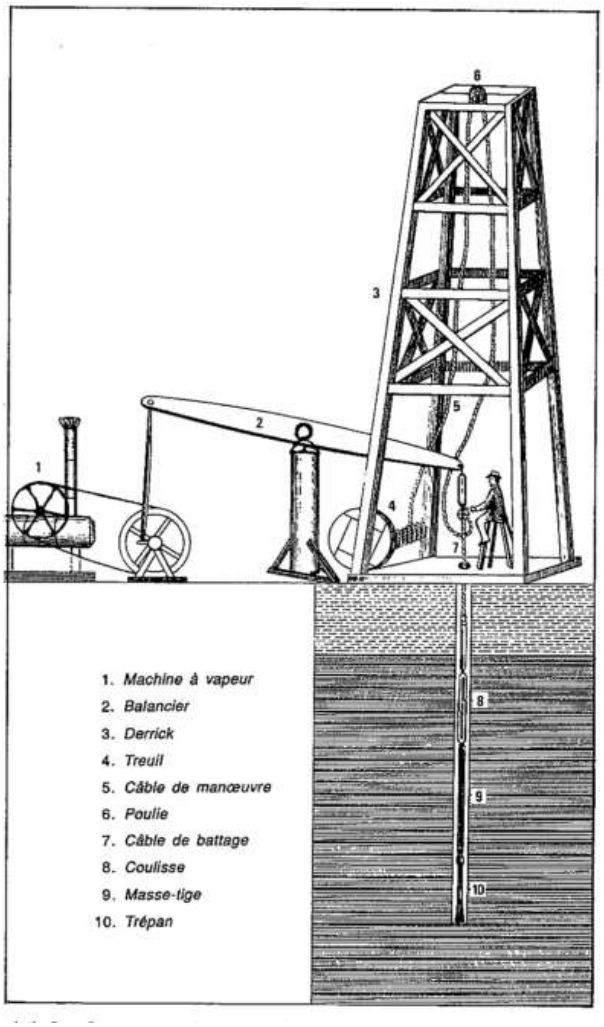

Figure 2.1 : Le forage par Battage

Dans cette technique un outil massif comparable au ciseau des sculpteurs, fixé au bout d'une tige lourde (masse-tige), elle-même suspendue à un balancier, tombait sous son propre poids et débitait la roche en éclats. Le balancier animé par l'action humaine, ou animale dans les temps anciens. Mais quel que soit le mode d'entrainement, il faillait périodiquement débarrasser le fond du trou des déblais. Le puits était alors rempli d'eau et la boue résultant du mélange de l'eau et débris de roche était vidée à l'aide d'un outil cylindrique muni d'un fond en forme de clapet, ouvert à la descente et fermé lors de la remontée au treuil. Le plus profond forage par battage atteignit 2250 en 1918.

C'est au début XX siècle que Antony Lucas démontra au monde entier l'efficacité du forage Rotary par la découverte du champ de Spindeltop (Texas) en utilisant la combinaison d'un outil rotatif et l'injection continue de boue. Depuis ce jour, cette technique est universellement utilisée et a profité des améliorations apportées par le progrès technique.

2.3 Le principe du forage rotary [20]

La méthode de forage Rotary consiste à utiliser un trépan, sur lequel on applique une force procurée par un poids tout en l'entrainant en rotation avec l'injection permanente de la boue de forage pour évacuer les débris.

La sonde de forage Rotary est l'appareillage nécessaire à la réalisation des trois fonctions suivantes (illustré dans la figure 2.2) :

- Poids sur l'outil,
- Rotation de l'outil,
- Injection d'un fluide.

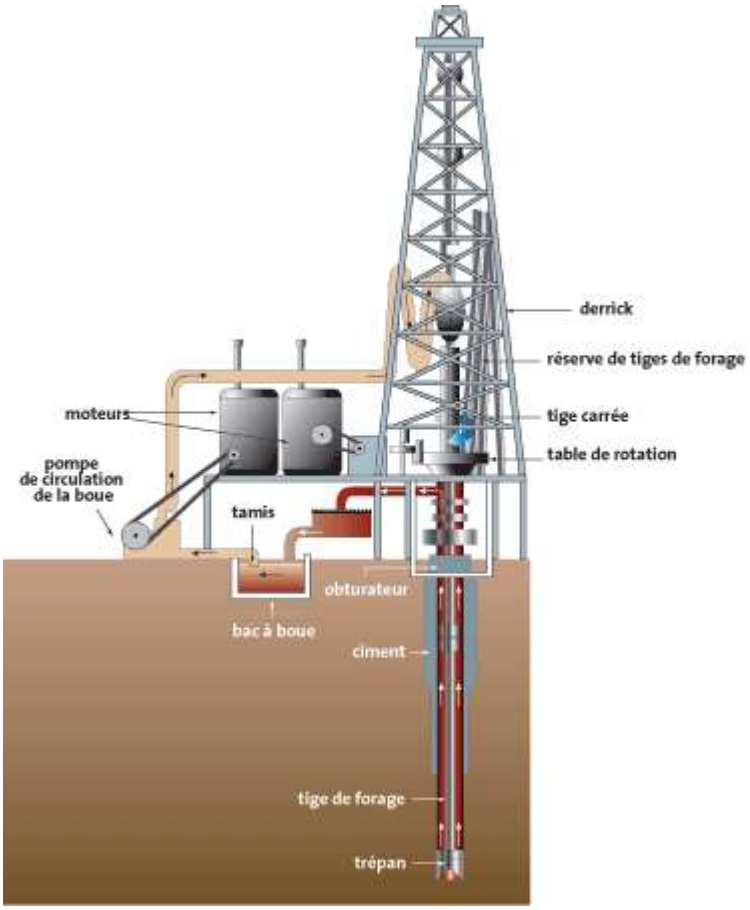

Figure 2.2 : Sonde de Forage Rotary (Source : "http://geothermie.tpe.free.fr/partie2.htm")

2.4 Description d'une installation de forage [21]

L'installation de forage inclus les systèmes suivants :

- **Système de suspension**

  Il est constitué d'un derrick pouvant atteindre 80m et d'un treuil motorisé situé au sol. Il sert à faire descendre et remonter l'équipement de forage. Il permet également de

fixer le poids appliqué au trépan en retenant partiellement le poids de l'ensemble de la garniture.

- **Le système de rotary**

Il est composé de toutes les parties qui permettent la transmission de la rotation à l'outil , citons la table de rotation et sa motorisation, la tige d'entrainement ainsi que le train de tiges et la tête d'injection.

- **Le système de circulation de boue**

Il assure la circulation de la boue de forage et il est associé à une station de pompage servant au traitement du fluide de forage. La boue est en effet un mélange d'eau, d'argile et d'additifs. La circulation de boue contribue à la lubrification des pièces en mouvement, à leur refroidissement et l'évacuation vers la surface des débris. La boue a aussi une grande utilité pendant l'opération de forage car son analyse fournit des informations sur la nature géologique des milieux traversés.

- **Le système de production de l'énergie**

L'énergie est produite par des moteurs à courant continu. Elle est transmise sous forme électrique ou mécanique vers les différents systèmes de l'installation.

- **Le système de contrôle du puits**

Il sert à détecter et gérer les apparitions soudaine des fluides de forage sous pression, ces irruptions connues sous le nom kick, peuvent être extrêmement violentes.

2.5 Description de la garniture

Elle correspond à la partie opérative dans le puits. Elle effectue plusieurs tâches dont la transmission de l'énergie nécessaire à la désagrégation de la roche, le guide et le contrôle de la trajectoire du puits, la transmission de la force poussée(W) ainsi que la circulation du fluide. Elle est constituée essentiellement des masses tiges(drill collars) et des train de tiges (drill pipes).

- **Trains de tiges**

Ils sont constitués de tuyaux en acier enchevêtrés les uns aux autres et pouvant s'étaler à des milliers de mètres. Ils transmettent le couple au trépan et servent de support aux masses tiges.

- **Masses tiges (drill collars)**

Les masses tiges sont des tubes en acier se situant au-dessus des trains de tiges. Elles contribuent à la création du poids agissant sur le trépan et sont soumises à plusieurs

contraintes engendrées par le diamètre du trépan, la production des pertes de charge minimales, la résistance au flambage et la rigidité.

- **Assemblage de fonds (Bottom Hole Assembly)**

L'assemblage de fond, Bottom Hole Assembly (BHA), correspond à la partie inférieure de la garniture de forage et renferme les trains de tiges, les stabilisateurs ainsi que le trépan. Sa longueur fluctue entre 100 et 300 mètres et dépend de la pression envisagée.

- **Stabilisateurs**

Ils se situent dans la garniture de forage et plus particulièrement dans les masses tiges et facilitent le contrôle de la trajectoire du trépan.

2.6 Déroulement d'une opération de forage [22]

Le forage est un processus graduel combinant plusieurs phases. Lors de chaque phase, l'assemblage de fond est remonté à la surface pour la maintenance du processus, pour le tubage et la cimentation de la partie forée. Le tubage consiste à déployer des tubes en acier dans le puits (Casing). Dans certaines situations, le tubage peut être enroulé (Coil tubing) et correspond au déploiement progressif d'un tube simultanément au forage. Dans toutes les situations le tubage permet de consolider les parois du puits au cours du forage, et de préparer les éléments nécessaires à la production une fois les réservoirs contenant les hydrocarbures atteints.

La cimentation consiste à cimenter l'annulaire à la fin de chaque phase de casing. Ce processus correspond à l'installation d'un anneau de ciment favorisant l'obtention d'un lien étanche et résistant entre le corps du tube et les parois du puits. Pendant le forage la partie basse du puits n'est pas couverte (Open Hole) tandis que sa partie supérieure est tubée.

2.7 La boue de forage

Est un mélange d'eau ou d'huile, argile, d'additifs chimiques et de la baryte. Elle permet l'évacuation des déblais et contribue à la compréhension de la nature géologique des milieux traversés. Le fluide doit être compatible avec les roches à forer pour garantir le bon déroulement du forage.

La boue de forage permet de

- ✓ maintenir les déblais en suspension après arrêt de la circulation,
- ✓ de maintenir les parois du trou grâce à la pression exercée latéralement,
- ✓ de retenir sous pression les fluides contenus dans la roche et donc d'empêcher la venue de fluides à l'intérieur des puits.

- ✓ Elles permettent aussi de refroidir l'outil de forage.

2.8 Les outils de forage (trépans) [21,22]

Ils sont conçus pour forer une certaine gamme de roches, il existe plusieurs types de trépans et ils sont choisis en fonction de puits à réaliser. La mécanique de l'outil influence directement la vitesse de progression (ROP). On distingue deux catégories : Les trépans tricônes et les trépans monobloc de type PDC (Polycrystaline Diamond Compact). Les trépans tricônes sont principalement composés d'acier ou de carbure de tungstène. Tandis que les monoblocs de type PDC sont composés de diamants, ou de diamants synthétiques. La grande différence entre ces deux familles de trépans réside dans leurs façons d'arracher la roche(illustré dans la Figure 2.3).

Figure 2.3 : Trépan Tricône et trépan PDC

- **Trépans tricônes**

Les trépans tricônes disposent de trois cônes rotatifs (molettes libres) qui embarquent des plaquettes de coupe conçues en fonction de la roche à forer. Ils peuvent être en acier, en carbure de tungstène ou en diamant. L'arrachage de la roche s'effectue lorsque les cônes effectuent des rotations autour du trépan. Le principal mode de destruction de la roche, par les taillants fixés sur les molettes, est le poinçonnement. Il se déroule par la pénétration verticale du taillant dans la roche sous l'effet d'un effort normal créant un champ de contraintes au voisinage du taillant. Lorsque les limites à la rupture sont atteintes un déblai se produit. Ce type de trépan est particulièrement adapté lorsque les roches à forer présentent une forte dureté.

- **Trépans PDC**

Ces trépans se composent du diamant naturel ou synthétique brasé sur du carbure de tungstène. Ils détruisent la roche par cisaillement, ce qui exige moins d'énergie que la rupture de la formation basée sur la compression. L'emplacement des pastilles dans ce type de trépan est primordial pour son optimisation et présente une influence considérable sur la vitesse de pénétration de la garniture, sur l'équilibre du trépan ainsi que l'évacuation des déblais. Les PDC sont stables pour les vitesses de rotation élevées mais instables pour les vitesses faibles.

2.9 Méthodes de transmission des données

Il plusieurs méthodes de transmissions des données forage parmi eux citons la télémétrie par modulation (mud pulse telemetry), la télémétrie électromagnétique (EM tool), les trains de tiges câblés (wired drill pipe) ainsi que les outils dits accessibles (retrievable tools).

- **La télémétrie par modulation de boue**

Cette méthode est la plus utilisée par les systèmes (MWD). Son fonctionnement nécessite l'exploitation d'une valve modulant le débit de la boue de forage. La modulation de boue s'effectue en fonction de la donnée à transmettre et crée une fluctuation de pression représentant l'information à délivrer. Ces fluctuations se propagent dans le fluide de forage vers la surface où elles sont recueillies par des capteurs de pression. Enfin, elles sont traitées par des ordinateurs pour reconstruire l'information transmise.

- **Les trains de tiges câblés**

Cette méthode utilise des câbles électriques placés dans les différents modules des trains de tiges et transportent un signal électrique à la surface. Elle se caractérise par sa capacité à présenter un taux de transmission de données important par rapport aux autres systèmes de transmission télémétriques.

- **Mesures pendant le forage (Measurement While Drilling, MWD)**

Les MWD sont transportés dans les puits en étant soit intégrés dans l'assemblage de fond soit embarqués dans les masses tiges. Ils délivrent les mesures relatives aux natures des roches, aux pressions dans le puits, aux températures, aux vibrations, aux chocs, aux couples etc...Quelques mesures peuvent être enregistrées dans les systèmes MWD et les autres sont transférées à la surface en utilisant le système télémétrique modulé par la boue ou d'autres sources de transmissions de données. Les mesures suivantes peuvent être fournies par le système MWD :

- La vitesse de rotation du train de tige (trépan),

- les types et sévérités des vibrations,
- la température dans le puits,
- le couple (Torque) et le poids sur l'outil,
- le débit du fluide de forage.

Les dispositifs MWD peuvent intégrer ou être lier avec des dispositifs de diagraphie (diagraphes) pendant le forage (Logging While Drilling, LWD), pour fournir des mesures décrivant les propriétés de la formation (la porosité, l'inclinaison, la résonance, magnétique, la pression de la formation etc…)

2.10 Les paramètres de forage [21,22]

Les paramètres de forage sont les différents facteurs mécaniques et hydrauliques pouvant agir sur la vitesse de progression (Rate of penetration, ROP) ainsi que sur le comportement directionnel.

Le ROP correspond à la profondeur en mètres forés par heure. L'optimisation du ROP est très importante dans le processus de forage car c'est directement lié au temps passé dans l'installation du forage.

Il y a deux types de paramètres :

2.10.1 Les paramètres mécaniques [22]

- **le poids sur l'outil (Weight On Bit,W)**

Ce paramètre désigne la force appliquée par la garniture sur le trépan suivant son axe de révolution. La valeur du poids dépend de la dimension et du type de trépan, de sa vitesse de rotation et du type de formation à forer. Une partie de ce poids provient de l'hydraulique créée par l'injection du fluide qui transite par les trains de tiges.

- **La vitesse de Rotation (RPM)**

Le choix de la vitesse de rotation dépend de celui (WOB). En surface, elle peut être précisément contrôlée mais elle peut être différente de la vitesse de rotation du trépan. Elle peut varier entre 50 et 1000 tr/min.

- **Le couple exercé sur l'outil (Torque On Bit)**

Ce paramètre correspond au couple transmis par la garniture au trépan suivant son axe de révolution. Il représente les effets combinés du couple réactif et des forces de frottement non linéaires sur la longueur du BHA.

2.10.2 Les paramètres hydrauliques [22]

- **Type de boue**

Le type de boue est choisi en fonction des performances recherchées et désigne les propriétés physico-chimiques du fluide de forage. Trois types de boues sont souvent employés : la boue à base d'eau (Water Based Mud, WBM), la boue à base d'huile (Oil Based Mud, OBM) et la boue synthétique (Synthetic Based Mud, SBM). Une boue synthétique est constituée d'un mélange d'eau et d'additifs chimiques.

- **Débit et pression hydraulique**

Le débit et la pression hydraulique représentent les variables physiques qui doivent favoriser une bonne évacuation des déblais et éviter des problèmes d'encrassement du trépan ou du puits.

- **Densité de la boue**

L'obtention des informations relatives au puits et particulièrement le contrôle de la pression dans le puits s'effectue à travers la densité de la boue. La boue de forage ramène à la surface les déblais, mais aussi du gaz contenu dans les roches. Cela fournit des indications sur la nature des fluides se situant dans le réservoir et représente un élément important dans le pilotage de la garniture.

2.11 Les facteurs influant la vitesse de progression (ROP)

Les facteurs qui influencent la ROP peuvent être en deux groupes principaux : les facteurs contrôlables et les facteurs environnementaux. Les facteurs contrôlables peuvent être modifiés plus facilement que les variables environnementaux. En raison des conditions économiques et géologiques, la variation des facteurs environnementaux est difficile ou coûteux. Le nombre de facteurs allusion à la complexité de l'interaction trépan/roche. puisque les propriétés de la boue, comme le type, la densité, etc., sont tous dépend du type de formation, la pression de formation, etc., alors les propriétés de la boue sont inclus dans «facteurs environnementaux»[23].

Tableau 2.1 : Les facteurs influant la ROP [23]

Facteurs contrôlables	**Facteurs environnementaux**
Etat des usure des dents de trépan	La profondeur
Conception de trépan	Les propriétés de la formation
Le poids sur l'outil (WOB)	Le type de la boue
La vitesse de rotation	La densité de la boue
Débit de la boue	Autres propriétés la boue
Hydraulique de l'outil	Pression renversée de la boue
La taille des buses de l'outil	Pression de la boue au fond du trou
Moteur / géométrie de la turbine	La taille de trépan

a. L'influence des paramètres mécaniques sur La ROP [24]

Pour les outils diamants, il est important de respecter les paramètres indiqués par le fabricant car les efforts s'appliquant sur l'outil ont été pris en compte pour le réaliser.

En règle générale ne pas dépasser 500 Kg de poids sur l'outil par taillant PDC de ½" (13 mm) actif (c'est à dire participant activement à la destruction de la roche, ne pas compter les taillants de grande taille servant à maintenir le diamètre) ; ceci fait approximativement 12 tonnes sur un outil 6" comportant 9 lames et 25 PDC actifs de 13 mm.

Pour les outils à diamant naturel, le poids est un paramètre important. Par contre les outils PDC, la vitesse de rotation est le paramètre le plus important, c'est pour cela qu'il est important de ne pas appliquer tout de suite des poids trop importants quand l'outil est encore neuf et donc très agressif et fragile.

Influence des formations sur les paramètres utilisés :

- **Roches abrasives :** privilégier le poids et réduire le ROP (augmentation de la profondeur de coupe diminuer le nombre de révolutions par mètre et diminuer d'autant le trajet total que les taillants auront à faire dans la roche abrasive).
- **Roches compactes :** et/ou plastiques : garder un poids suffisant mais augmenter le ROP (zone de proportionnalité) ; dans les roches non abrasives (carbonates, argiles, évaporites), la vitesse n'endommage pas notamment la structure de coupe mais influence très favorablement les coûts de forage.

L'objectif de l'optimisation de forage est de maximiser le ROP toute en minimisant l'usure des dents, comme le forage avance rapidement il va accélérer l'usure des dents qui aura un effet négatif sur le ROP jusqu'à l'arrêt de forage quand les dents de l'outil sont complètement usés,

pour cette raison, des recherches ont été menées pour développer une méthode pour prédire l'usure de l'outil pendant le forage.

b. Influence des paramètres hydrauliques [24]

Pour refroidir correctement les diamants et éviter de les 'brûler', il est important d'avoir un débit de circulation important.

Le respect d'un débit de circulation élevé est prépondérant pour ce type d'outil tout d'abord pour refroidir les taillants et ensuite pour nettoyer le front de taille.

b.1 - Effet du nettoyage du front de taille sur le ROP

Comme pour les outils à molettes, l'évacuation des déblais de front de taille a une très grande influence sur le ROP.

Avec les PDC [1, 2], dans les formations tendres, il est préférable d'augmenter le débit plutôt que la vitesse du jet de boue à la sortie des dûses. Avec une boue à l'huile ou dans certains types d'argiles, on prendra des valeurs plus faibles. Dans des formations plus dures, de la boue à l'eau, une puissance de 3 à 5 HP/in2.

La notion de puissance hydraulique par in2 de trou est commode à utiliser mais ne recouvre pas de réalité particulière ; un bon design hydraulique (radiale) avec si possible une duse par lame ou une duse pour deux lames et des lames suffisamment hautes (pour éviter un colmatage prématuré aux reprises de fond ou au changement brusques de formations), doit être préférée. En effet, un petit outil sera toujours mieux nettoyé qu'un gros du fait de la décroissance rapide des vitesses de fluides dès que l'on s'éloigne notablement des duses (tant verticalement que rapidement).[24]

Par ailleurs, l'utilisation de duses dont le diamètre est à 12/32 doit être évitée à cause du risque de bouchage de telles duses par les débris divers qui se trouvent dans la boue en forage réel (bacs sales ou graviers dans les tiges, cuttings (déblais) pénétrant en circulation inverse lors des ajouts de tiges…).

Il faut savoir que le ou les passages d'eau colmatés lors du forage (souvent suite à une obstruction des duses) ne se débouchent jamais et finissent par arrêter l'outil complètement par compaction des déblais dans les passages d'eau correspondants ; Ceux-ci arrivent alors rigoureusement au raz de la face de l'outil, empêchant toute pénétration des taillants.

- **Effet des caractéristiques de la boue sur le ROP**

Les effets de la densité, de la filtration, de la viscosité et de la teneur en solides sur le ROP sont similaires à ceux des outils tricônes.

Pour les outils à diamants naturels comme pour les outils à molettes, la boue à l'huile a un effet plutôt néfaste sur le ROP, du fait du mode de destruction de la roche (la roche tend à passer sous les diamants rendus plus glissants plutôt que d'être entraînée par eux...).

Par contre, la boue à l'huile a tendance à améliorer les performances des PDC et des TSP (lorsque ces derniers travaillent à la façon des PDC en mode cisaillant).

Cependant, dans le cas précis des outils PDC, le processus de destruction de la roche par cisaillement est moins sensible à la pression différentielle, car les fluides passant plus rapidement sous la surface du fond par l'arrière des PDC permettant l'équilibrage des pressions de part et d'autre du déblais (la fissure de décollement du déblais se fait dans le sens de déplacement du taillant contrairement au cas du picot où le déblais progresse en sens inverse du picot et être maintenu). [24]

La boue à l'eau a tendance à favoriser le bourrage des PDC dans certains types d'argiles.

2.12 Conclusion

Le forage est la clé de toute prospection pétrolière. Cette étape représente le principal et l'essentiel du coût total d'une installation(environ les 2/3). Ce coût dépend bien entendu de la localisation et de la profondeur du terrain. L'exploration offshore (en mer) coûte bien plus (plusieurs fois) que la prospection on shore.

Malgré les progrès des méthodes d'explorations géologiques, la découverte, surtout de gros gisements, reste un événement rare. Dans le monde, on compte en moyenne une découverte pour dix forages effectués ; mais il faut 100 forages pour découvrir un gisement de 10 millions de tonnes par an.

Les techniques modernes de forages permettent de forer en déviation à partir d'un seul point, cela limite les dimensions des installations de surface en concentrant les puits (limite la déforestation ou la taille des plates-formes offshore). Les puits peuvent simplement être déviés ou réellement horizontaux voire en U (U-shape). Optimisant ainsi la surface d'échange entre le puits et la roche réservoir, les puits horizontaux peuvent avoir des productivités cinq à dix fois supérieures aux puits verticaux.

Pour assurer le bon déroulement d'une opération de forage il faut bien surveiller et régler ces paramètres, la vitesse de progression est un paramètre majeur, il dépend de plusieurs facteurs, sa prédiction nous permet de bien régler nos paramètres pour avoir un bon avancement, pour avoir une prédiction on doit utiliser des outils learning machine puissants, parmi ces outils nous avons choisi l'algorithme des forêt aléatoires qui fera l'objet de chapitre suivant.

Chapitre 3 : Les forêts aléatoires

Résumé

Le chapitre 3 présente les méthodes ensembles et les forêts aléatoires.

Le principe général des méthodes d'ensemble est de construire une collection de prédicteurs, pour ensuite agréger l'ensemble dans leurs prédictions. Dans un cadre de régression, agréger la prédiction de q prédicteurs revient par exemple à en faire la moyenne. Ainsi de la classification revient par exemple à faire un vote majoritaire parmi les classes fournies par les prédicteurs, dans le but de plus de précision et efficacité au prédicteur produit, la combinaison des prédicteur peut être séquentielle, parallèle ou hybride.

Les méthodes ensembles se reposent sur plusieurs méthodes pour l'induction comme : Bagging, Boosting et Random Subspace,et Randomizing output.

Les arbres de décision sont des méthodes graphiques largement répandues dans les domaines statistiques, dans l'analyse des décisions et dans l'apprentissage automatique, elle se caractérisent par la simplicité et la lisibilité.

CART est une méthode statistique introduite par Breiman qui construit des prédicteurs par arbre binaire en régression et en classification aussi.

Les forêt aléatoires font partie de la famille des méthodes ensemblistes qui prennent l'arbre de décision comme prédicteur individuel généralement l'algorithme CART, elles se basent sur les méthodes de Bagging, Randomizing Outputs et Random Subspace en excusant le boosting.

3.1.Introduction

Nous avons décrit dans la chapitre 2 le forage pétrolier, nous passons au 3ème chapitre consacré à l'étude la méthode des forêts aléatoires introduite par Breiman en 2001. Ils sont une méthode statistique non paramétrique qui s'avère être très performante dans de nombreuse application, aussi bien pour des problème de régression que de classification supervisée.

La méthode des forêt aléatoires fait partie des méthodes ensemble qui permettent de combiner plusieurs classifieurs et les agréger pour avoir classifieur plus performant .

L'efficacité d'un ensemble de classifieurs se base sur la combinaison complémentaire ou divers qui sont relativement bon afin de nous produire un prédicteur plus pour améliorer la prédiction.

Les forêts aléatoires se reposent sur la combinaison de plusieurs arbres de décision CART et la randomisation au bagging.

3.2. Apprentissage statistique

Le cadre mathématique de l'apprentissage statistique est le suivant : Soit

$$\mathcal{L}_n = \{(X_1, Y_1), \ldots\ldots, (X_n, Y_n)\}$$

un échantillon d'apprentissage, c'est-à-dire une suite de vecteurs aléatoires indépendants et identiquement distribués, de même loi qu'un vecteur aléatoire (X, Y). Le vecteur (X, Y) est indépendant de $\mathcal{L}_n$ et sa loi est inconnue. L'entier naturel n désigne le nombre d'observations de l'échantillon d'apprentissage.

Le but de l'apprentissage statistique est d'**apprendre** la loi inconnue de (X, Y), au travers de l'échantillon d'apprentissage dont on dispose. Notons $\mathcal{X}$ et $\mathcal{Y}$ les espaces mesurables dans lesquels vivent respectivement les variables aléatoires X et Y. X est vue comme la variable d'entrée et Y est comme celle de sortie. Le but est d'apprendre le lien entre l'entée et la sortie.

Etant donnée une variable $x \in \mathcal{X}$ (x est une variable de test qui n'appartient pas à l'échantillon d'apprentissage), la méthode statistique doit être capable de « prédire » la sortie $\hat{y} \in \mathcal{Y}$ correspondante. La prédiction $\hat{y}$ doit être la plus proche possible de la vraie sortie y associée à x. [26]

Il existe deux principaux cadres en apprentissage statistique : la régression et la classification.

3.2.1 Régression

Le cadre de régression est celui ou la sortie y est continue (une variable réelle), le modèle statistique alors s'écrit sous la forme suivante :

$$Y = s(X) + \varepsilon$$

La fonction $s: \mathcal{X} \to \mathbb{R}$est la fonction inconnue que nous cherchons à estimer. ε est une variable aléatoire réelle. Elle est appelée la variable de bruit : les mesures Y_i dont nous disposons dans l'échantillon $\mathcal{L}_n$ sont des observations de $s(X_i)$ bruitées par des variables aléatoires $\mathcal{E}_i$.[26]

3.2.2 Classification

En classification (supervisé), la réponse Y est discrète et désigne la classe à laquelle l'entrée X est associée. Ici, $\mathcal{Y} = \{1, \dots \dots, L\}$, où L désigne le nombre de classes.

En classification on cherche à estimer la probabilité a posteriori définie pour un $x \in \mathcal{X}$ fixé par : $\forall\, c \in \{1, \dots, L\}\ \ P(Y = c|X = x)$.

C'est-à-dire les probabilités pour Y d'appartenir à chacune des classes, conditionnellement à X.[26]

3.3. Les méthodes d'ensemble

Le principe générale des méthodes d'ensemble est de construire une collection de prédicteurs, pour ensuite agréger l'ensemble dans leurs prédictions. Dans un cadre de régression, agréger la prédiction de q prédicteurs revient par exemple à en faire la moyenne. Dans un cadre de classification revient par exemple à faire un vote majoritaire parmi les classes fournies par les prédicteurs. [27]

L'objectif de ces méthodes est de construire un collection de prédicteurs qui vérifie ces deux points :

- Chaque prédicteur individuel doit être relativement bon.
- Les prédicteurs individuels doivent être différents les uns des autres.

L'utilisation des méthodes d'ensemble garantie :

- La précision : de meilleurs classifieurs peuvent être obtenus en combinant les prédictions de plusieurs classifieurs (même faiblement efficace)
- L'efficacité : un problème complexe peut être décomposé en multiples sous problèmes plus simple à résoudre.

3.3.1 Pourquoi combiner plusieurs classifieurs

On peut vouloir combiner plusieurs classifieurs pour les raisons suivantes [28] :

- Ils permettent de fiabiliser les décisions en s'appuyant sur l'avis de plusieurs experts au lieu d'un seul.

- Ils élargissent l'ensemble des solutions possibles en proposant des modèles plus complexes que l'on ne pourrait pas obtenir avec des classifieurs uniques.
- Ils élargissent l'ensemble des solutions possibles en proposant des modèles plus complexes que l'on ne pourrait pas obtenir avec des classifieurs uniques.
- Ils permettent d'éviter les optima locaux.
- Ils permettent de traiter des problèmes trop complexes pour être appréhendés dans leur globalité.
- Ils sont plus génériques que les classifieurs uniques, et peuvent appréhender plus efficacement un plus grand nombre de problèmes.

3.3.2 Combinaison de classifieurs

3.3.2.1 Architectures de combinaison

Il existe plusieurs schémas de combinaison très différent les uns des autres et qui ont chacun un intérêt différent [28].

On peut distinguer trois grands schémas de combinaison de classifieurs adoptant des architectures différentes :

- **Combinaison Séquentielle ou Série :** ce schéma de combinaison organise les classifieurs élémentaires en niveaux successifs de décision, de sorte que chacun d'eux prenne en compte la prédiction du classifieur placé en amont. Les classes candidates sont ainsi progressivement évincées jusqu'à ce qu'il ne reste qu'une décision possible(illustré dans la figure 3.1).

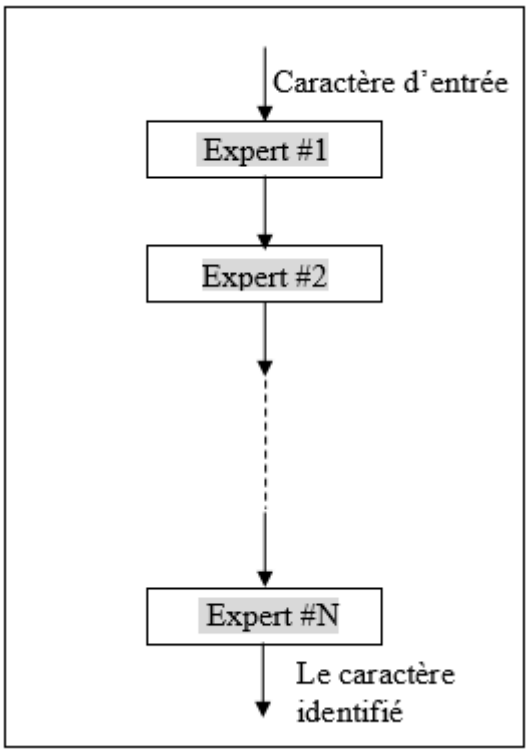

Figure 3.1 : Combinaison séquentielle des classifieurs [28]

- **Combinaison parallèle :** dans ce schéma, les classifieurs élémentaires opèrent indépendamment les uns des autres et prennent leurs décisions sans tenir compte du reste du comité. Les décisions individuelles sont ensuite fusionnées à l'aide d'un opérateur de combinaison (voir la figure 3.2).

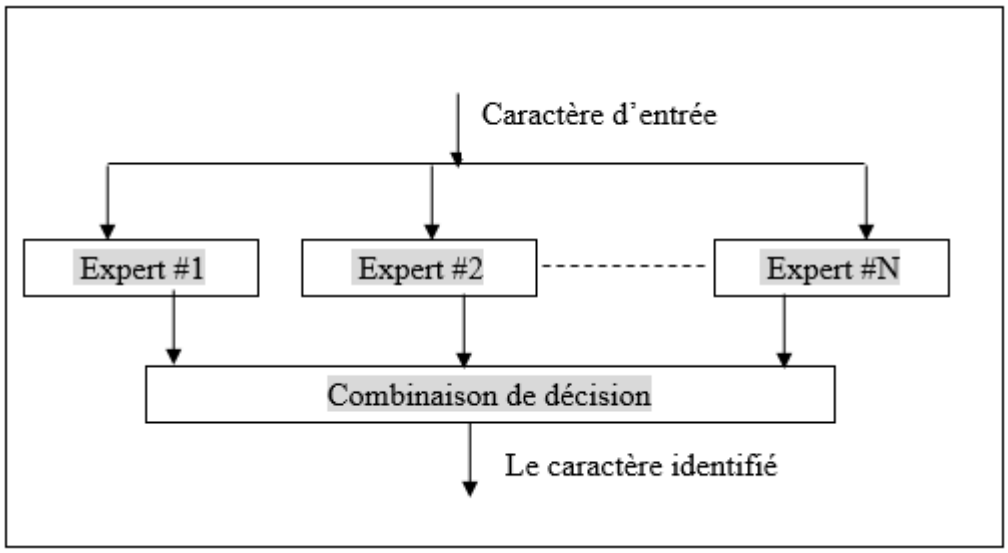

Figure 3.2 : Combinaison parallèle de classifieurs [28]

- **Combinaison hybride :** Les méthodes appartenant à cette catégorie sont généralement conçues pour des applications spécifiques, comme c'est le cas par exemple de la méthode proposée par Kim et al pour la reconnaissance de mots cursifs anglais extraits de chèques bancaires. Dans cette méthode, deux niveaux de classification sont mises en œuvre ; le premier pouvant traiter parallèlement deux espaces de description concurrents, et le deuxième ayant pour rôle de fusionner les décisions du niveau précédent (illustré dans la figure 3.3).

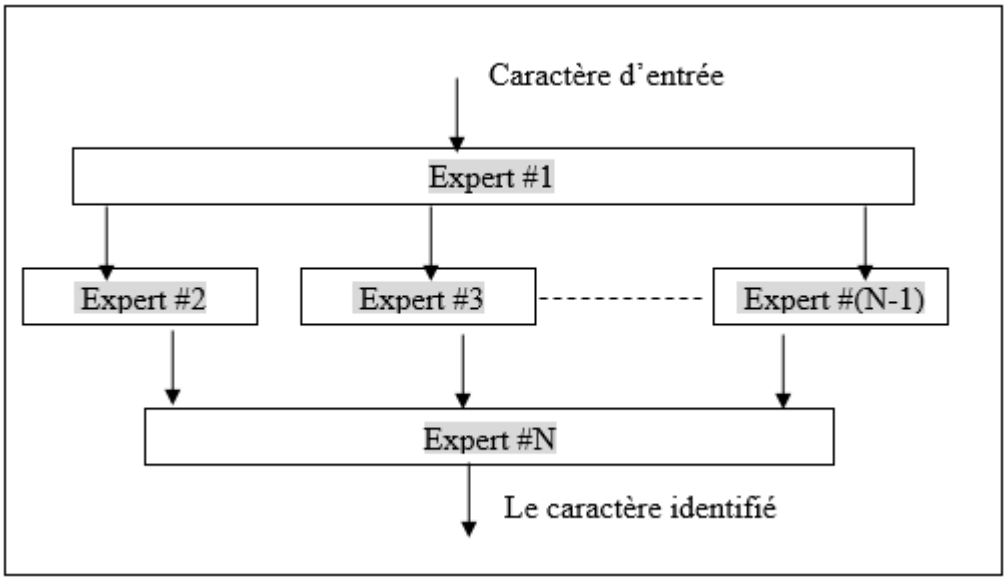

Figure 3.3 : Combinaison Hybride de classifieurs [28]

Parmi ces trois approches, celle qui suscite le plus grand intérêt de la communauté scientifique est la combinaison parallèle de classifieurs.

3.3.2.2 Induction d'Ensembles de classifieurs [28]

L'induction d'EoC consiste à générer, à partir d'un unique algorithme d'apprentissage, un ensemble de classifieurs capables de donner des prédictions différentes sur un même ensemble de données à classer. L'enjeu de l'induction d'EoC est donc de définir un moyen de créer ces différences parmi le comité de classifieurs, pourtant tous générés avec le même algorithme d'apprentissage. On peut distinguer trois approches par niveau :

- **Le niveau "donnée" :** cette catégorie regroupe les méthodes d'induction d'ensembles qui sont basées sur la manipulation des données d'apprentissage, et plus particulièrement de leur distribution, pour induire des classifieurs élémentaires différents, Il s'agit principalement de générer des sous-ensembles de données différents pour l'apprentissage de chacun des classifieurs individuels.

- **Le niveau "caractéristique"** : il s'agit ici des méthodes qui manipulent cette fois les espaces de description des données. Classiquement, comme avec les méthodes du niveau "donnée", les classifieurs individuels seront appris sur des sous-espaces de description différents.
- **Le niveau "classifieur" :** les méthodes de ce niveau s'intéressent plus particulièrement quant à elles à l'algorithme d'induction des classifieurs. L'idée est généralement d'utiliser des paramétrages différents chaque fois qu'un classifieur élémentaire est induit avec l'algorithme d'apprentissage.

En résumé, en recoupant ces deux taxonomies prédominantes dans les travaux concernant l'induction d'ensembles de classifieurs, on peut distinguer quatre principales stratégies, non exclusives, pour créer des EoC :

• Manipuler la distribution des données d'apprentissage.

• Manipuler les espaces de description.

• Manipuler les étiquettes de classes.

• Manipuler les paramètres de l'algorithme d'apprentissage.

3.3.2.3 Principes méthodes d'induction d'ensembles de classifieurs

Bagging

Le bagging est une méthode d'ensemble introduite par Breiman en 1996. Le mot Bagging est la contraction des mots Boostrap et Aggregating.

Le bootstrap est un principe de ré-échantillonnage statistique traditionnellement utilisé pour l'estimation de grandeurs ou de propriétés statistiques. Il permet, en plus de fiabiliser les estimations statistiques, de fournir plus d'indications sur ces estimations. En statistiques, lorsque l'on souhaite approximer la distribution d'une population de données, on utilise généralement une distribution empirique de données observées. L'idée du bootstrap est d'utiliser non plus une unique distribution empirique, mais plusieurs ensembles de données ré-échantillonnées à partir de l'ensemble des données observées et ce à l'aide d'un tirage aléatoire avec remise.[27]

Supposons que l'on dispose d'un ensemble$T = \{x_1, x_2, x_3, \dots, x_N\}$ de Ndonnées observées de notre population, et que l'on s'intéresse à un statistique notée $S(T)$.

Le bootstrap va consister à former Léchantillons $T_K^* = \left(x_1^*, x_2^*, x_3^*, \dots, x_{N'}^*\right)$pour$k = 1, \dots, L$ où chaque T_K^*est constitué par tirage aléatoire avec remise de N' données dans T.Ces L échantillons sont usuellement appelés les échantillons bootstrap.

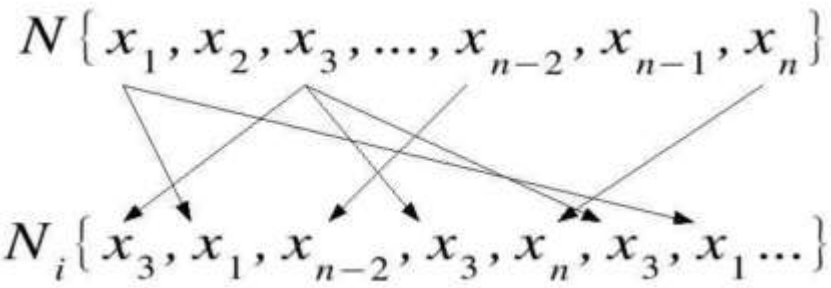

Figure 3.4 : Illustration d'un tirage aléatoire avec remise pour la formation d'un échantillon [28]

À partir de ce principe de ré-échantillonnage, Breiman en 1996 introduit la méthode de Bagging. Il s'agit simplement de considérer que la statistique que l'on cherche à étudier est un algorithme d'apprentissage noté $h(x)$et d'appliquer alors le principe de bootstrap tel que nous venons de l'expliquer. Ainsi chaque classifieur élémentaire $h_k(x)$ de l'ensemble sera entrainé sur un des L échantillons bootstrap de sorte qu'ils soient tous entrainés sur un ensemble d'apprentissage différent. La figure5 illustre le procédé de Bagging appliqué à un ensemble d'arbres de décision.

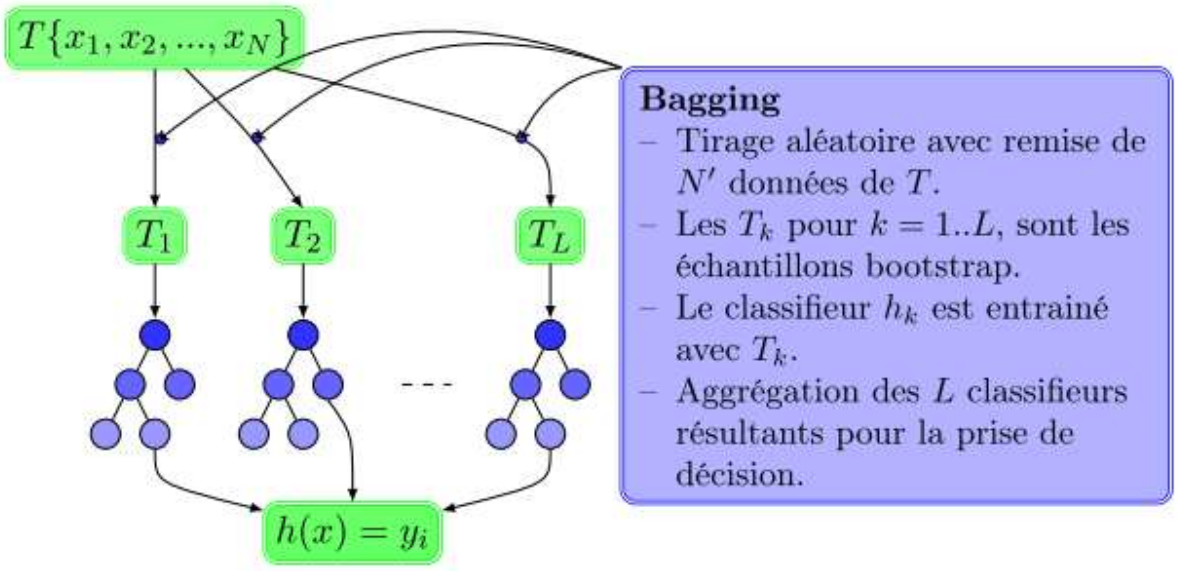

Figure 3.5 : Illustration du principe de Bagging pour un ensemble d'arbres de décision [28].

Une étude plus approfondie sur la technique "bagging" a été mise en place par (Skurichina et al) [31] . Cette étude a montré que, généralement, le "bagging" permet d'améliorer la performance des classificateurs instables. L'instabilité d'un algorithme de classification traduit le fait qu'une légère modification des données d'apprentissage entraine des différences importantes au niveau de l'estimation des frontières de décision.[31]

Le bagging peut réduire l'instabilité des classifieurs comme les arbres de décision ainsi que les réseaux de neurones afin d'améliorer leur performances par ailleurs le Bagging ne permet pas d'améliorer les performances des classifieurs stables, comme par exemple les performances d'un k-Plus Proche Voisins.

La principale force du bagging est donc de réduire l'instabilité pour augmenter les performances en généralisation. Mais il y a un autre point qui fait la force du bagging, ce sont les mesures out-of-bag.

Mesures Out-Of-Bag

Est l'ensemble des individus qui ne sont pas sélectionnés dans les échantillons bootstrap. Ce paramètre introduit par la méthode bootstrap permet l'évaluation interne du classifieur et l'estimation de l'importance des variables pour la sélection des variables.

Soit T une base d'apprentissage de N individus, N' est le nombre de données tirées aléatoirement pour chaque ensemble bootstrap. Dans le cadre du Bagging, N' est systématiquement fixé à N, mais peut aussi être inférieur. En revanche il n'est pas conseillé de le fixer avec une valeur très supérieure à N. La raison en est qu'avec le bagging, la diversité est introduite dans l'ensemble de classifieurs par les différences créées dans chaque échantillon bootstrap, qui produit des différences dans les prédictions des classifieurs élémentaires — c'est d'autant plus le cas pour les classifieurs très instables.[28]

Un résultat mathématique intéressant lorsque $N = N'$ est que chaque échantillon bootstrap ne contient asymptotiquement — i.e. pour N très grand — que 63,2% des données d'apprentissage de T. En effet, si l'on effectue un tirage aléatoire avec remise des données, il n'est pas impossible de tirer plusieurs fois la même donnée pour le même échantillon bootstrap. La conséquence est que pour $N = N'$, toutes les données ne seront pas forcément présentes dans tous les échantillons. Et pour une valeur de N relativement grande, on peut démontrer qu'un peu plus d'un tiers des données n'apparaissent pas dans un échantillon bootstrap donné .

Les données out-of-bag fournissent alors un bon moyen d'obtenir des mesures sur les classifieurs élémentaires, comme par exemple une estimation de leurs performances en généralisation.

Randomizing Outputs

Breiman introduit la méthode Randomizing Outputs, le principe de cette méthode consiste à construire des échantillons indépendants dans lesquels on altère les sorties de l'échantillon d'apprentissage. La modification que subissent les sorties est obtenue en rajoutant une

variable de bruit à chaque Y_i et $\mathcal{L}_n$. On obtient alors une collection d'échantillons à « sorties randomisées », puis on applique une règle de base sur chacun et on agrège enfin des prédicteurs obtenus. [26]

L'idée de Randomzing Outputs est, encore, en appliquant une règle de base sur des échantillons à sorties randomisées, on obtient une collection de prédicteurs différents les uns et les autres.

Random Subspaces

Cette méthode est similaire dans l'idée au bagging , elle joue sur les caractéristiques pas sur les données. Le principe de base est d'entrainer chaque classifieur élémentaire sur un sous-espace aléatoire de l'espace de description. Tous les sous-espaces aléatoires ont les mêmes dimension P, avec $P < M$ où Mest la dimension de l'espace original.[28]

Par exemple , pour l'induction d'un classifieur h_k , on doit :

1. Effectuer un tirage aléatoire sans remise de P caractéristiques parmi les caractéristiques disponibles.
2. Projeter toutes les données d'apprentissage dans ce nouveau sous-espace de caractéristiques.
3. entrainer le classifieur h_k sur ces projections des données d'apprentissage.

La figure 3.6 illustre l'application de cette procédure à l'induction de forêts de décision.

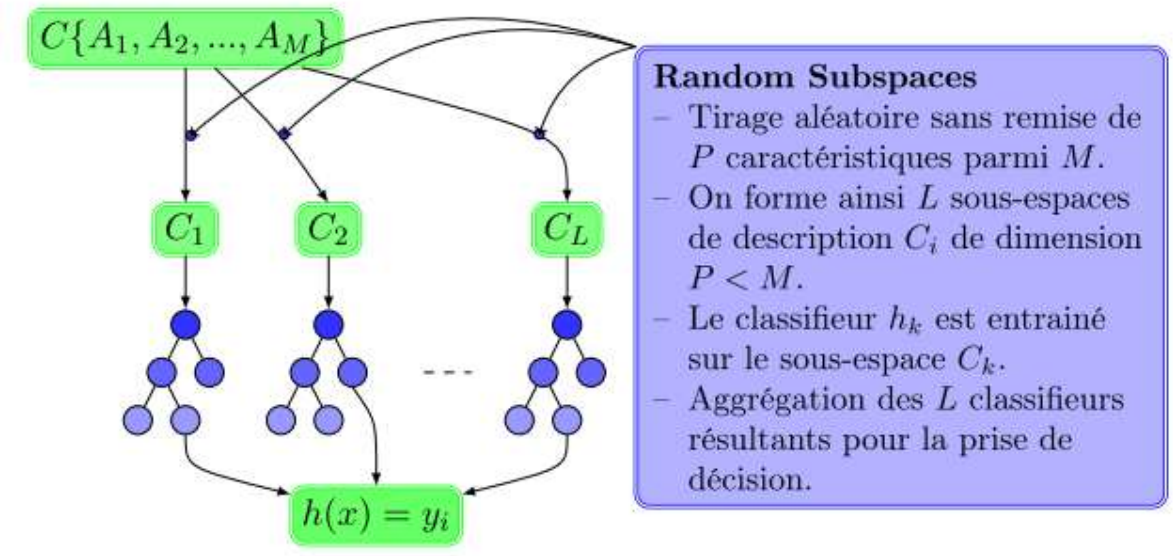

Figure 3.6: Illustration du principe de Random Subspaces pour un ensemble d'arbres de décision. [28]

Dans [28], Ho montre à propos du paramètre P que les meilleurs résultats sont généralement obtenus pour $P \approx M / 2$ caractéristiques. Elle a de plus mis en évidence que cette méthode

était particulièrement efficace quand l'espace de description présente une certaine redondance d'information dispersée sur l'ensemble des caractéristiques, plutôt que concentrée sur un sous-ensemble d'entre elles.

Cette procédure présente un avantage sur d'autres méthodes induction des forêts de décision : elle permet de réduire la dimension de l'espace de description des données d'apprentissage. Or, l'induction automatique d'un arbre de décision nécessite des parcours répétés de cet espace. Le coût computationnel est par conséquent considérablement réduit lorsque cette méthode est utilisée, en comparaison avec d'autres méthodes d'induction de forêts de décision qui ne modifient pas l'algorithme d'induction d'arbres de décision.

Lorsque l'espace de description présente de fortes redondances d'information, apprendre un classifieur sur un sous-ensemble des caractéristiques peut s'avérer plus efficace que de le faire sur l'ensemble. Combiner des classifieurs de ce type permet bien souvent d'améliorer les performances que l'on pourrait obtenir avec un classifieur unique, s'agissant même de classifieurs stables.

Error Correcting Output Codes

Cette méthode joue sur la manipulation des étiquettes des données d'apprentissage. Le principe est de générer des sous-problèmes aléatoires en regroupant de différentes façons les c classes d'un problème multi-classes en deux sous-ensembles aléatoires, notés A_i et B_i. Les données d'apprentissage sont alors ré-étiquetées avec l'une ou l'autre de ces deux superclasses et utilisées pour l'apprentissage d'un classifieur. Cette procédure est ensuite répétée pour tous les classifieurs élémentaires de l'ensemble de sorte qu'ils s'intéressent tous à des problèmes différents, concernant des groupes de classes différents. En phase de prédiction, chaque donnée à prédire est classée par tous les classifieurs, chacun d'entre eux attribuant un vote à l'ensemble des classes appartenant au sous-groupe prédit. Après que tous les classifieurs ont attribué leur vote, ceux-ci sont sommés pour chaque classe, et celle réunissant le plus grand nombre de votes est choisie comme prédiction finale.[28]

Boosting

Introduit par Freud and Schapire(1996), le Boosting est une des méthodes d'ensembles les plus performantes à ce jour. Le principe de base est de spécialiser progressivement les classifieurs de l'ensemble de façon itérative, et de combiner ensuite chacun des classifieurs obtenus à chaque itération. Typiquement, il s'agit à une itération quelconque de concentrer l'apprentissage du classifieur actuel sur les erreurs des classifieurs obtenus aux itérations précédentes.

Le Boosting se base sur l'utilisation des classifieurs élémentaire « faibles » (weak classifier), on appelle un apprenant faible un algorithme qui fournit des classifieurs faibles, capables de reconnaître deux classes au moins aussi bien que le hasard ne le ferait. C'est à dire que les classifieurs produits par ce type d'algorithme doivent en moyenne obtenir un taux de bonnes prédictions supérieur ou égal à 50%, l'objectif est de transformer un apprenant faible en un apprenant fort.

Le premier algorithme de boosting a été créé par Schapire en 1990, c'est un algorithme récursif et construit un système multi-classifieurs à l'aide de trois niveaux de combinaison (illustré dans figure 3.7). Le point clé de cet algorithme est que l'apprentissage de chaque classifieur faible est basé sur une distribution différente des données d'apprentissage, générée à l'aide des prédictions des classifieurs précédemment induits.[28]

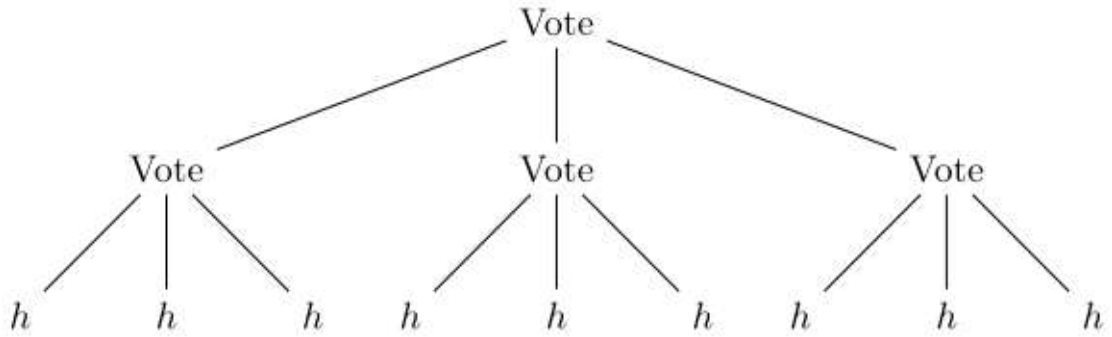

Figure 3.7: Illustration du premier algorithme de boosting proposé par Schapire, Chaque nœud "h" est un classifieur faible, et chaque nœud "Vote" est un opérateur de vote à la pluralité.[28]

Après Schapire, Freund introduit à son tour son algorithme de boosting , reprenant le principe global de l'algorithme de Schapire mais de façon non plus récursive mais itérative. En 1996 ensuite, les deux auteurs présentent l'algorithme AdaBoost [28]. La première version de cet algorithme appelé AdaBoost.M1 est décrite par l'algorithme 1.

Algorithme 1 AdaBoost.M1

Entrée : $\mathcal{L}$ un apprenant faible.

Entrée : L le nombre de classifieurs de l'ensemble final.

Entrée : T un ensemble de N données d'apprentissage.

1 : $D_1(x_i) = \frac{1}{N}, i = 1, \dots, N$ *Initialisation des poids (équiprobabilité)*

2 : ***pour*** $t = 1, \dots, L$ ***faire***

3 : $h_t =: \mathcal{L}(D_t)$ *apprentissage de* h_t

4 : $\widehat{\varepsilon_t} = \sum_{i:h(x_i)\neq y_i} D_t(x_i)$ *calcul de l'erreur pondéré de* h_t

5 : ***si*** $\widehat{\varepsilon_t} > \frac{1}{2}$ ***alors***

6 : Stopper la boucle

7 : $\beta_t = \frac{\widehat{\varepsilon_t}}{(1-\widehat{\varepsilon_t})}$ *calcul de coefficient de pondération* h_t

8 : ***pour*** $i = 1, \dots, N$ ***faire***

9 : ***si*** $h_t(x_i) = y_i$ ***alors***

10 : $D_{t+1}(x_i) = \frac{D_t(x_i)}{Z_t} \times \beta_t$

11 : ***sinon***

12 : $D_{t+1}(x_i) = \frac{D_t(x_i)}{Z_t}$

13 : $h_c(x) = \arg max_{y\in Y} \sum_{t=1}^{L} \log\frac{1}{\beta_t} \times 1_{h_t(x)=y}$ *Vote Pondéré*

La définition de l'apprenabilité faible impose que l'erreur d'un apprenant soit "légèrement meilleure que le hasard", ce qui peut se traduire par $\widehat{\varepsilon_t} > \frac{1}{2}$, où $\widehat{\varepsilon_t}$ représente l'erreur en apprentissage dans le cas de problème à deux classes. Cependant, dans le cas de problèmes à plus de deux classes, être meilleur que le hasard ne signifie pas forcément que l'erreur de l'apprenant est tout juste inférieure à $\frac{1}{2}$.

Pour un problème à c classes, prédire une classe au hasard fournit une espérance mathématique de l'erreur égale à $1 - \frac{1}{c}$. La contrainte $\widehat{\varepsilon_t} < \frac{1}{2}$ peut alors s'avérer particulièrement difficile à respecter avec certains problèmes multi-classes. C'est là le principal inconvénient de ce premier algorithme : il impose $\widehat{\varepsilon_t} < \frac{1}{2}$.

Freund et Schapire ont proposé une deuxième version pour remédier le problème cité précédemment, il se comporte la même manière qu'Adaboost.M1 dans le cas de problème à deux classes, mais qui corrige ces défauts dans le cas de problèmes à plus de 2 classes.

3.4. Les arbres de décision

Les arbres de décision sont des méthodes graphiques largement répandues dans les domaines statistiques, dans l'analyse des décision et dans l'apprentissage automatique, elle se caractérisent par la simplicité et la lisibilité. [27]

Les arbres de décision constituent un moyen de représenter un ensemble des règles qui sous-tendent des données par une structure hiérarchique, en partitionnant de façon récursive sa population.[28]

Les arbres de décision sont généralement définis comme des décisions ou des événements successifs représenté chronologiquement de gauche à droite (illustré dans la figure 3.8) .

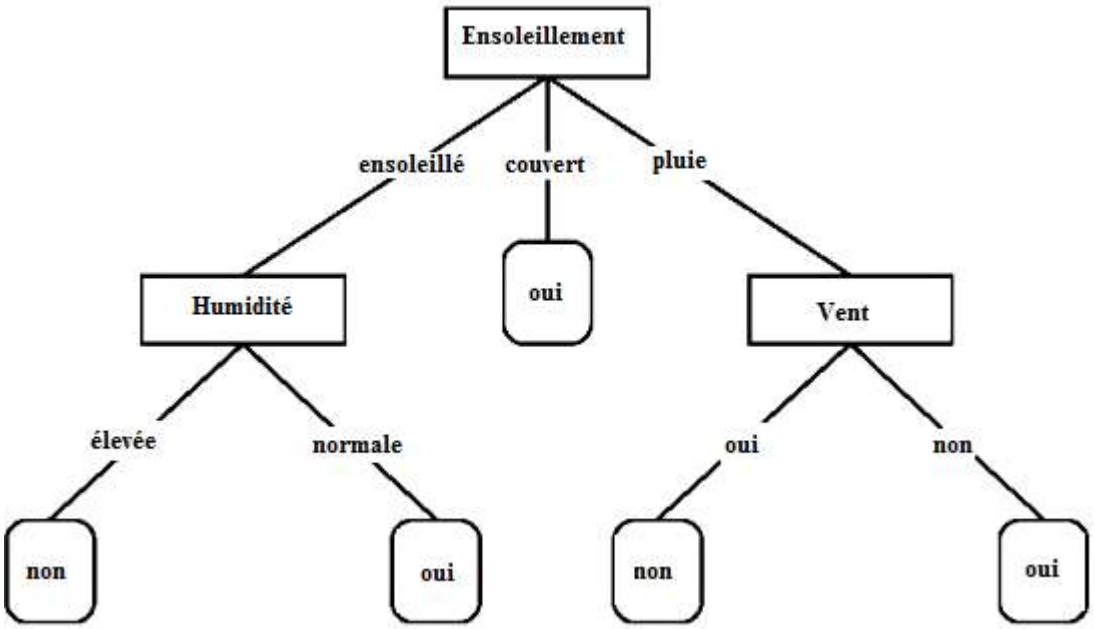

Figure 3.8 : Exemple d'un arbre de décision : pour la prédiction de jouer à l'extérieur en fonction de la météo (« Oui » ou « Non »)[28]

3.4.1 La construction de l'arbre de décision

Cette phase sert essentiellement à construire la structure hiérarchique de l'arbre, nous expliquons les étapes de construction à travers l'exemple cité précédemment « Weather » qui largement utilisé dans la littérature pour illustrer les problématiques et les méthodes de construction des arbres de décision.

Tableau 3.1 : Description de l'exemple "Weather" [28]

Numéro	Ensoleillement	Température	Humidité	Vent	Jouer
1	Soleil	Chaude	Elevée	Oui	Oui
2	Soleil	Chaude	Elevée	Oui	Non
3	Soleil	Chaude	Elevée	Non	Non
4	Soleil	Douce	Elevée	Non	Non
5	Soleil	Froide	Normale	Non	Oui
6	Couvert	Froide	Normale	Oui	Oui
7	Couvert	Froide	Normale	Non	Oui
8	Couvert	Douce	Elevée	Oui	Oui
9	Couvert	Froide	Normale	Non	Oui
10	Pluie	Douce	Normale	Oui	Non
11	Pluie	Douce	Normale	Oui	Non
12	Pluie	Douce	Elevée	Non	Oui
13	Pluie	Chaude	Normale	Non	Oui
14	Pluie	Douce	Elevée	Non	Oui

Dans cet exemple nous disposons de 14 observations, pour lesquelles nous cherchons à expliquer le comportement « jouer à l'extérieur », à partir d'un ensemble de prévisions météorologiques.

Supposons que l'ensemble des observations constitue la population de la « racine » de l'arbre on retrouve effectivement dans la figure 3.8 la répartition du tableau 1 avec 9 observations « oui » et 5 observations « non ».

A chaque étape de construction on ajoute des nouveaux nœuds à la structure et fait « grandir » l'arbre de décision, chaque nœud sert à diviser la population en plusieurs sous-groupes homogènes en utilisant des règles de répartition. [28]

Dans un nœud donné, la règle de partitionnement est déterminée de la façon suivante :

1) on choisit l'une des variables d'entrée correspondants aux attributs de chaque observation.
2) Suivant les valeurs de cette variable au sein de la population du nœud, on partitionne la population à l'aide des tests sur la valeur de la variable.
3) Chaque observation est répartie dans les sous-ensembles, en utilisant les tests de partitionnement.

Dans notre exemple, le premier partitionnement réalisé, utilise la variable « Ensoleillement » pour définir la règle de partitionnement, trois modalités sont alors possibles : « Soleil », « Couvert » et « Pluie », de ce fait trois nouveaux nœud seront automatiquement crées, chaque nœud correspondant à un sous-groupe en relation avec ces modalités.

Ce partitionnement intermédiaire donne naissance aux nouvelles branches, partant du nœud courant vers autant de nouveaux nœuds qu'il y a de partitions créées. On reprend alors le même processus avec tous ces nouveaux nœuds jusqu'à ce qu'il ne reste plus que des feuilles au bout de chaque branche.

L'appellation **feuille** indique ici un nœud terminal. Il s'agit d'un nœud qui ne possède pas de fils et dont la population est considérée comme homogène pour décider de ne pas la partitionner à nouveau, on appelle une feuille **pure** une feuille dont la population d'observations n'appartient qu'à une seule classe. La notion de pureté et impureté est utilisée généralement comme critère d'arrêt de croissance d'un arbre de décision, des fois on fait recours à un critère d'arrêt supplémentaire, il peut être en effet nécessaire d'interrompre la croissance d'un arbre pour qu'il deviendrait plus complexe avant de n'obtenir que des feuilles pures. Il existent des techniques d'élagage permettant de réduire la taille et la complexité d'un arbre après l'atteinte de sa taille maximale.[28]

L'élagage consiste à supprimer *a posteriori* les branches de l'arbre jugées inutiles. Ces techniques sont généralement mises en place dans les algorithmes d'induction pour améliorer les capacités en généralisation des arbres de décision. L'utilisation des techniques d'élagage permet d'éviter le problème de sur-apprentissage de l'arbre de décision qui est très mauvais en généralisation.

Dans notre exemple « Weather », aucun test d'arrêt supplémentaire n'a été utilisé. L'arbre a grandi jusqu'à atteindre sa taille maximale.

En bref, la construction d'un arbre de décision passe par les étapes suivantes [28] :

1- Choix d'une variable de partitionnement parmi les attributs qui décrivent les données d'apprentissage.
2- Choix d'une ou de plusieurs valeurs de coupure de cette variable pour définir la partition.
3- Recommencer les étapes 1 et 2 avec chacun des nœuds fils qui ne remplissent pas les critères pour devenir des feuilles (Si tous les nœuds sont des feuilles pures, la branche courante a atteint sa taille maximale).
4- Elaguer si besoin l'arbre.

5- Affecter à chaque feuille une conclusion.

3.4.2 Evaluation des règles de partitionnement

Le choix de la variable de partitionnement sur un nœud est réalisé par le test de toutes les variables potentielles et le choix du celle qui optimise un critère de qualité donné. Ce critère utilisé caractérise la pureté lors du passage du nœud à partitionner vers les feuilles produites par ce partitionnement. Il existe plusieurs critères statistiques, citons l'entropie de Shanon, le coefficient de Gini et les mesures statistiques de type KHI2 [28].

3.4.3 L'algorithme CART (Classification and Regression trees)

C'est l'algorithme utilisé généralement dans les forêts aléatoires, CART est une méthode statistique introduite par Breiman en 1984 qui construit des prédicteurs par arbre en régression et en classification aussi. L'algorithme CART partitionne récursivement l'espace d'entrée X d'une façon dyadique, afin de déterminer une sous-partition optimale pour la prédiction. [26]

A chaque étape, on partitionne une partie de l'espace en deux sous-parties en construisons un arbre binaire pour chaque nœud, les nœuds de l'arbre sont associés aux éléments de la partition(illustré dans la Figure 3.9).

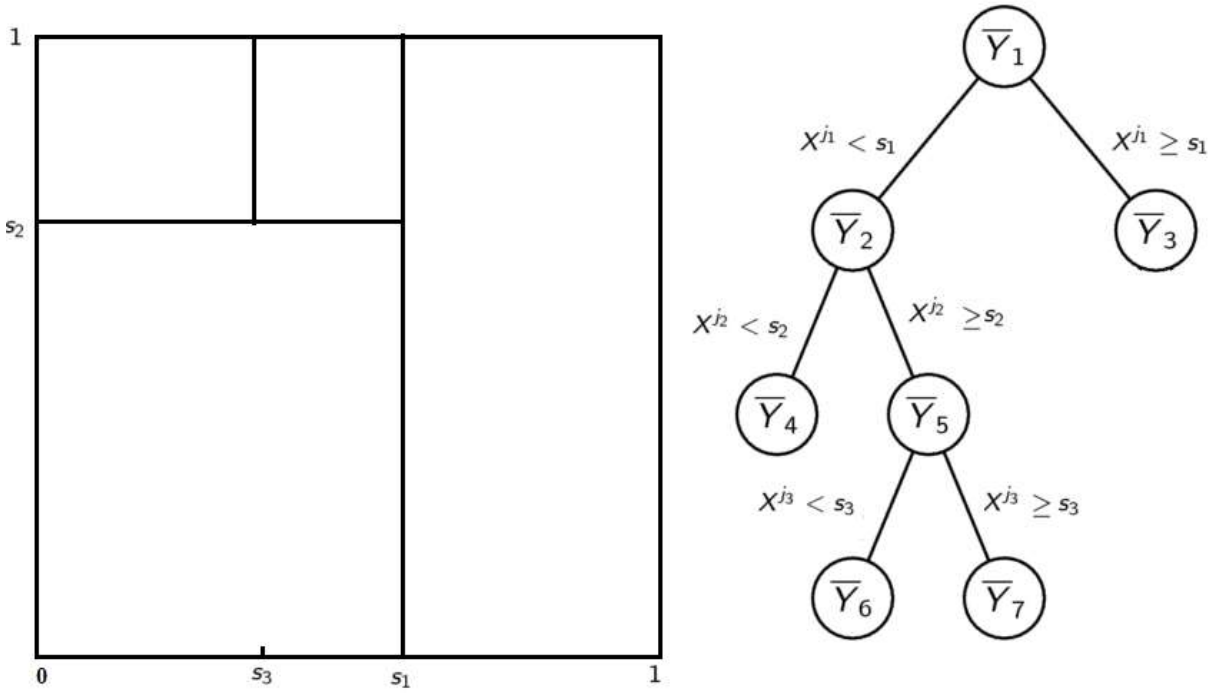

Figure 3.9 : une partition dyadique du carré unité et son arbre CART associé [26]

Pendant le découpage on cherche toujours les meilleurs découpe, c.-à-d. le couple (j, d) qui minimise certain fonction de coût (j : j ème variable, d est une valeur réelle) :

- En régression, on cherche à minimiser la variance des nœuds fils, la variance d'un nœud t est définie par $\sum_{i:X_i \in t}(Y_i - \overline{Y_t})^2$, où $\overline{Y_t}$ est la moyenne des Y_i observations présentes dans le nœud t.
- En classification ou L est le nombres de classes, on cherche à minimiser l'indice de Gini des nœuds fils. L'indice de Gini d'un nœud t est défini par $\sum_{c=1}^{L} \hat{p}_t^c\,(1 - \hat{p}_t^c)$, où $\hat{p}_t^c$est la proportion d'observation de la classe c dans le nœud t.

La deuxième étape consiste à chercher le meilleur arbre élagué de l'arbre maximal pour éviter le sur-apprentissage.

3.5 Les forêts aléatoires

Les forêts aléatoires ont été introduits par Leo Breiman en 2001, pour améliorer l'agrégation des arbres de décision par l'ajout de la « Randomisation » afin de rendre plus « indépendants » les arbres d'agrégation à travers le hasard dans le choix des variables qui interviennent dans les modèles.

Les forêt aléatoires font partie de la famille méthodes ensemblistes qui prennent l'arbre de décision comme prédicteur individuel, elles se basent sur les méthodes de Bagging, Randomizing Outputs et Random Subspace en excusant le boosting. [26]

Les éléments clés qui définissent les forêts aléatoires sont : [28]

1. Les forêts aléatoires sont basées sur des ensembles d'arbres de décision.
2. Une certaine "quantité" d'aléatoire est introduite dans le processus d'induction.

La construction d'un arbre ici est effectuée de la façon suivante, pour découper un nœud on tire aléatoirement un nombre m de variables, et on cherche la meilleure coupure uniquement pour les m variables sélectionnées sans élaguer l'arbre obtenu. L'agrégation des arbres obtenus (moyenne en régression, vote majoritaire en classification) pour créer le prédicteur RandomForest . [26]

3.5.1 Algorithmes d'induction des forêts aléatoires

On trouve plusieurs algorithmes d'induction des forêts aléatoires, le premier qui été introduit est « Random Forest Random Input », cet algorithme peut être vu comme une variante de Bagging seul la différence réside dans la construction des arbres individuels.

Le tirage à chaque nœud de m variables se fait sans remise, et uniformément parmi toutes les variables, m est fixé au début de construction de la forêt et identiques pour les arbres(m<=p). Pour Random forest RI, il y a deux sources d'aléas pour générer la collection des prédicteurs individuels : l'aléa du au Boostrap et l'aléa du au choix des variables pour découper chaque nœud d'un arbre, le Random Forest RI améliore les performances du Bagging par l'ajout d'un aléa supplémentaire pour construire les arbres, pour rendre les arbres plus différents afin d'obtenir un prédicteur agrégé meilleur.[26]

3.5.2 Random Feature Selection (Random Tree)

Il a été introduit principalement par Amit et Geman en 1996 dans le cadre de proposition d'une méthode d'arbres aléatoires pour la reconnaissance d'écriture manuscrite [28]. Ensuite cette idée a été reprise par Breiman avec le nom **Random Feature Selection.** L'idée consiste à introduire l'aléatoire dans le choix des règles de découpage à chaque nœud des arbres, de telle façon que chaque règle ne soit plus choisie à partir de l'ensemble des caractéristiques disponibles, mais à partir d'un sous-ensemble de ces caractéristiques.

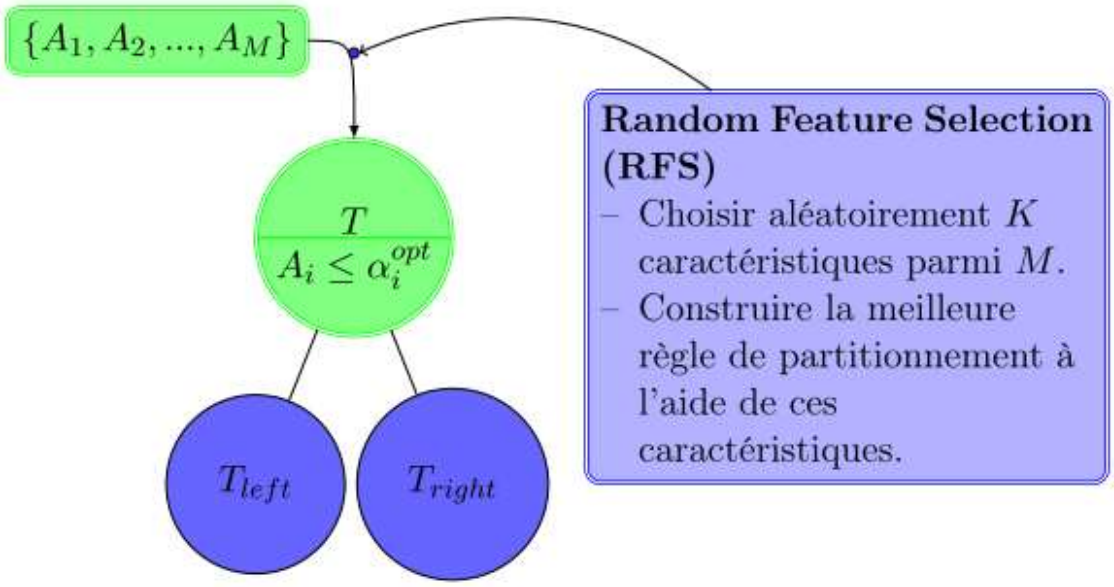

Figure 3.10 : Principe de l'algorithme Random Feature Selection [28]

3.5.3 Forest RI (Random Forests - Random Input)

Il a été introduit par Breiman en 2001, il est le plus utilisé dans les logiciels d'apprentissage automatique sous le nom « Random Forest », il se base sur l'utilisation de deux principes de randomisation : le principe de Bagging et le principe de Random Feature Selection. [28]

3.5.4 Paramètres de l'algorithme

Il existent deux paramètres principaux pour cet algorithme :

- Le nombre de variables choisies aléatoirement à chacun des nœuds des arbres. Il peut varier entre 1 et p (p nombre de variables), par défaut : $\sqrt{p}$ pour classification, $\frac{p}{3}$ pour régression.
- Le nombre d'arbres de la forêt (Ntree), sa valeurs par défaut est 100.

Nous pouvons également régler d'autres aspect de l'algorithme : le nombre minimum d'observations(node size) en dessous on ne découpe pas un nœud ou la façon d'obtenir les échantillons Bootstrap (sans remise ou avec remise et le nombre d'observations tirées). [27]

Algorithme ForestRI [28]

Entrée : T l'ensemble d'apprentissage

Entrée : L le nombre d'arbres dans la forêt

Entrée : K le nombre de caractéristiques à sélectionner aléatoirement à chaque nœud

Sortie : foret l'ensemble des arbres qui composent la forêt construite

1 : ***pour*** l de 1 à L ***faire***

2 : $T_l \leftarrow$ ensemble bootstrap, dont les données sont tiré aléatoirement (avec remise) de T

3 : arbre ← un arbre vide, i.e. composé de sa racine uniquement

4 : arbre.racine ←RndTree(arbre.racine, : T_l,K)

5: foret ← foret ∪ arbre

6: ***retour*** foret;

Algorithme RndTree

Entrée : n le nœud courant

Entrée : T l'ensemble des données associées au nœud n

Entrée : K le nombre de caractéristiques à sélectionner aléatoirement à chaque nœud

Sortie : n le même nœud, modifié par la procédure

1 : ***si*** n n'est pas une feuille ***alors***

2 : $C \leftarrow K$ caractéristiques choisies aléatoirement sans remise

3 : ***pour tout*** $A \in C$***faire***

4 :***Procédure CART*** pour la création et l'évaluation (critère de Gini pour classification , variance pour la régression) du partitionnement produit par A, en fonction de T

5 : *partition*←*partition* qui optimise (le critère de Gini pour la classification, la variance pour la régression)

6 : n. *Ajouter Fils(partition)*

7 : ***pour tout*** *fils*∈ *noeudFils* ***faire***

8 : *RndTree(fils,fils.donnee,* K*)*

9 : **retour** n

3.6 Conclusion

Dans ce chapitre nous avons introduit le concept de combinaison de plusieurs classifieurs et plus particulièrement celui des ensembles des classifieurs, après avoir montré l'intérêt de telles méthodes, nous avons tout d'abord présenté les différentes architectures de combinaison que l'on rencontre dans la multitude de travaux sur les systèmes multi-classifieurs.

Ensuite nous avons présenté l'algorithme des forêts aléatoires qui est une méthode d'ensemble pour la classification et la régression, qui opère en construisant une multitude d'arbre de décision pendant l'apprentissage afin de construire un classifieur plus performant, les forêts aléatoires corrige les erreurs des arbres de décisions sur les ensembles d'apprentissage.

Nous passons au chapitre suivant qui décrit l'algorithme Heuristique Nelder-Mead simplex utilisé pour optimiser la vitesse de progression de forage.

Chapitre 4 : l'algorithme Nelder-Mead Simplex

Résumé

Nous présentons dans ce chapitre l'algorithme de l'optimisation Nelder-Mead simplex ou le simplex de descente.

L'algorithme heuristique de simplex Nelder-Mead ou simplex de descente (Downhill simplex) publié par Jhon Nelder et Mead en 1965, est l'un des algorithmes les plus connus pour l'optimisation sans contrainte multidimensionnelle avec les fonction objectifs non dérivables. Il est totalement différent de la méthode de simplex de Dantzig utilisée dans la programmation linéaire.

Il est largement utilisé pour résoudre l'estimation des paramètres et des problèmes statistiques similaires, où les valeurs de la fonction sont incertaines ou soumises au bruit.

L'algorithme de simplex de Nelder-Mead est basé sur le principe suivant :

- construire la simplex initial S.
- Répétez les étapes suivantes jusqu'à ce que le test d'arrêt est satisfait:
  - calculer les informations de test d'arrêt;
  - si le test d'arrêt est pas satisfait, transformer la simplex de travail.
- Retour au meilleur sommet du simplex courant S et la valeur de fonction associée.

L'algorithme Nelder-Mead simplex est généralement plus rapide par rapport aux autres algorithmes dans le cas où la fonction objectif a besoin de plusieurs évaluations dans chaque itération.

L'algorithme Nelder-Mead simplex peut être hybridé avec les méthodes heuristiques comme le recuit simulé ou les algorithmes génétiques dans le cas où la fonction objectif possède plusieurs minima locaux, pour converger vers la solution.

4.1 Introduction

Dans le chapitre précédent nous avons présenté l’algorithme des forêts aléatoires utilisé pour prédire le ROP, nous présentons dans ce chapitre l’algorithme de l’optimisation Nelder-Mead (DownHill) simplex ou le simplex de descente.

L'algorithme Nelder-Mead simplex, publié par Jhon Nelder et Mead en 1965[32], est l'un des algorithmes les plus connus pour l’optimisation sans contraintes multidimensionnelles avec les fonctions objectifs non dérivable.

Cette méthode ne doit pas être confondue avec la méthode simplex de Dantzig pour la programmation linéaire, ce qui est complètement différente, car cette méthode est heuristique. L'algorithme de base est assez simple à comprendre et très facile à utiliser. Pour ces raisons, il est très populaire dans de nombreux domaines de la science et de la technologie, en particulier dans de le domaine de la chimie et de la médecine.

Le procédé ne nécessite pas d'information dérivée, ce qui le rend approprié pour des problèmes avec des fonctions non dérivables. Il est largement utilisé pour résoudre les problèmes d’estimation des paramètres et celui des statistiques similaires, où les valeurs de la fonction sont incertaines ou soumises au bruit. Il peut également être utilisé pour des problèmes avec les fonctions discontinues, qui se produisent fréquemment dans les statistiques et les mathématiques expérimentales.

4.2 Historique et origine

Les méthodes de recherche directe sont apparu dans les années 1950 et au début des années 1960 avec l'utilisation croissante des ordinateurs pour ajuster les données expérimentales. Le nom de "recherche directe" a été introduit en 1961 par Hooke et Jeeves[33].

La première méthode de recherche directe basée sur le simplex a été proposé par Spendley, Hext et Himsworth en 1962 [34]. Elle utilise seulement deux types de transformations pour former un nouveau simplex dans chaque étape:

- la réflexion loin du pire sommet (celui avec la plus grande valeur de la fonction), ou
- retrait vers le meilleur sommet (celui avec la plus petite valeur de fonction).

Dans ces transformations, les angles entre les bords dans chaque simplex restent constants tout au long des itérations, de sorte que le simplex de travail peut changer de taille, mais pas de forme.

En 1965, Nelder et Mead modifie la méthode originale de Spendley et al. en incluant deux transformations-de dilatation et de contraction supplémentaires, qui permettent au simplex de travail pour changer non seulement sa taille, mais aussi sa forme.

La méthode simplex de Nelder-Mead a gagné en popularité très rapidement. A ce moment, en raison de ses exigences en matière de simplicité et de faible stockage, il est parfaitement adapté pour une utilisation sur des mini-ordinateurs, en particulier dans les laboratoires. Dans les années 1970, la méthode est devenue un membre au niveau de plusieurs bibliothèques de logiciels majeurs.

Sa popularité a grandi encore plus dans les années 1980, quand il est apparu avec l'appellation "algorithme Amoeba" dans le manuel largement utilisé « recettes numériques » [35], et en logiciel Matlab, où il est maintenant appelé " fminsearch"[36].

Malgré son âge et de récents progrès dans les méthodes de recherche directe, la méthode Nelder-Mead reste encore parmi les méthodes les plus populaires de recherche directe dans la pratique.

4.3 Principe de base

L'algorithme Nelder-Mead est conçu pour résoudre le problème classique d'optimisation sans contrainte de minimisation d'une fonction non linéaire donnée $f: \mathbb{R}^n \to \mathbb{R}$. Il :

- Utilise seulement les valeurs de fonction de certains points dans $\mathbb{R}^n$, et
- ne cherche pas à former un gradient approximatif à aucun de ces points.

Par conséquent, il appartient à la classe générale des méthodes de recherche directe [37].

La méthode Nelder-Mead est basée sur le simplex. Un simplex S dans $\mathbb{R}$ est défini comme un polytope convexe de $n+1$ sommets $x_0, x_1, \ldots\ldots, x_n \in \mathbb{R}^n$. Par exemple, dans un simplex $\mathbb{R}^2$ est un triangle, et en un simplex $\mathbb{R}^3$ est un tétraèdre.

L'algorithme de Nelder-Mead simplex a une interprétation géométrique naturelle vive. Un simplex est un polytope géométrique qui comporte $n+1$ sommets $x_0, x_1, \ldots\ldots, x_n \in \mathbb{R}^n$, par exemple, un segment de droite dans l'espace de dimension 1, un triangle (illustré dans la figure 4.1) dans un plan, un tétraèdre (illustré dans la figure 4.2) dans un espace à 3 dimensions et ainsi de suite. Dans la plupart des cas, la dimension de l'espace correspond au nombre de paramètres indépendants qui doivent être optimisées afin de minimiser la valeur d'une fonction: (prend un vecteur de n paramètres), où n est la dimension de l'espace [32].

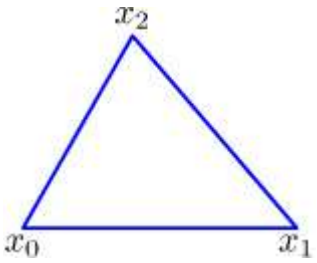

Figure 4.1: triangle

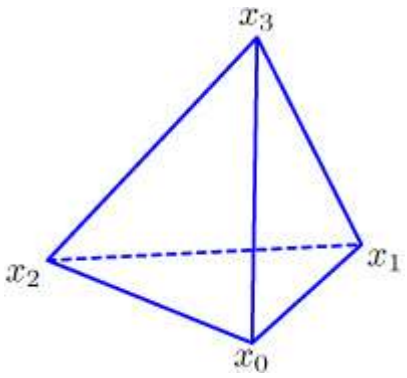

Figure 4.2: tétraèdre

Une méthode de recherche directe fondée simplex commence avec un ensemble de $n+1$ sommets $x_0, x_1, \ldots\ldots, x_n \in \mathbb{R}^n$ considérés comme les sommets d'un simplex de travail S, et l'ensemble correspondant des valeurs de la fonction aux sommets $f_j := f(x_j)$, pour $j = 0, \ldots\ldots, n$, le simplex initial de travail S doit être non dégénéré, c.-à-d. $x_0, \ldots\ldots, x_n$ ne doivent pas se trouver dans le même hyperplan. La méthode qui effectue une séquence de transformations du travail simplex S, visant à diminuer les valeurs de la fonction à ses sommets. A chaque étape, la transformation est déterminée par le calcul d'un ou plusieurs points de mesure, ainsi que leurs valeurs de fonction, et par comparaison de ces valeurs de fonction avec ceux au niveau des sommets. Ce processus est terminé lorsque le travail simplex S devient suffisamment faible dans un certain sens, ou lorsque les valeurs de la fonction f_j sont assez proches dans un certain sens (à condition que f soit continue).

L'algorithme Nelder-Mead exige habituellement seulement un ou deux évaluations de la fonction à chaque étape, tandis que de nombreuses autres méthodes de recherche directe utilisent n ou même plus d'évaluations de fonction.

4.4 L'algorithme

Même si la méthode est assez simple, elles est mise en œuvre de nombreuses façons différentes. En dehors de quelques détails mineurs de calcul de l'algorithme de base, la principale différence entre les différents modes de réalisation réside dans la construction du simplex initial, et à la sélection des tests de convergence ou de terminaison servant à mettre fin au processus d'itération. L'algorithme général est comme suit :

- construire la simplex initial S.
- Répétez les étapes suivantes jusqu'à ce que le test d'arrêt est satisfait:
    - Calculer les informations de test d'arrêt;

- si le test d'arrêt est pas satisfait, transformer le simplex de travail.

- Retour au meilleur sommet du simplex courant S et la valeur de la fonction associée.

4.4.1 Le simplex initial

Le simplex initial S est généralement construit en générant $n+1$ sommets $x_0, x_1, \dots\dots, x_n$ autour un point d'entrée $x_{in} \in \mathbb{R}^n$. Dans la pratique, le choix le plus fréquent est $x_0 = x_{in}$ pour permettre le redémarrage appropriée de l'algorithme. Les n sommets restants sont ensuite générés pour obtenir l'une des deux formes standard de S:

- S est à l'angle droit en x_0, basée sur des axes de coordonnées, ou

$$x_j := x_0 + h_j e_j, \qquad j = 1, \dots\dots, n,$$

où h est une taille de pas dans la direction du vecteur unité e_j dans $\mathbb{R}^n$.

- S est un simplexe régulier, où toutes les arêtes ont la même longueur spécifiée.

4.4.2 Algorithme de transformation simplex

Chaque itération de la méthode de Nelder-Mead est constituée des trois étapes suivantes :

Classement: Déterminer les indices h, s, l du pire, le deuxième pire et le meilleur sommet, respectivement, dans le simplex de travail courant $Sf_h = max_j f_j, \quad f_s = max_{j \neq h} f_j, \quad f_l = min_{j \neq h} f_j$.

Dans certaines implémentations, les sommets de S sont classés selon les valeurs de la fonction f, pour satisfaire $f_0 \leq f_1 \leq \cdots \dots \leq f_{n-1} \leq f_n$. Ensuite $l = 0, s = n-1, h = n$.

Barycentre : Calculer le centre de gravité c de tous les points sauf x_h, le meilleur côté est le seul en face du pire sommet x_h,$c := \frac{1}{n} \sum_{j \neq h} x_j$

Transformation : Calculer le nouveau simplex de travail. Tout d'abord, essayer de remplacer seulement le pire sommet x_h avec un meilleur point en utilisant la réflexion, expansion ou la contraction par rapport au meilleur côté. Tous les points de mesure sont situés sur la ligne définie par x_h et$c$, et au plus deux d'entre eux sont calculés dans une itération. Si cela réussit, le point accepté devient le nouveau sommet de simplex de travail. Si cela échoue, rétracter la simplex vers le meilleur sommet x_l. Dans ce cas, n nouveaux sommets seront calculés.

les transformations Simplexe dans la méthode de Nelder-Mead sont commandées par quatre paramètres [38] :

α pour la réflexion, β pour la contraction , γ pour l'expansion et δ pour le rétraction. Ils doivent satisfaire les contraintes suivantes $\alpha > 0, 0 < \beta < 1, \gamma > 1, \gamma > \alpha, 0 < \delta < 1$.

Les valeurs standards, utilisés dans la plupart des implémentations, sont $\alpha = 1,\ \beta = \frac{1}{2}2, \gamma = 2, \delta = \frac{1}{2}$.

L'algorithme suivant décrit les transformations de simplex de travail à l'étape 3 , et les effets de diverses transformations sont présentés dans les figues correspondantes. Le nouveau simplex de travail est affiché en rouge.

Réflexion : Calculer le point de réflexion $x_r := c + \alpha(c - x_h)$ et $f_r := f(x_r)$.Si $f_l \leq f_r < f_s$, accepter x_r et terminer l'itération(illustré dans La figure 4.3).

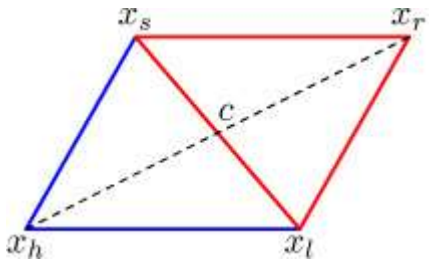

Figure 4.3 : Réflexion

Expansion : si $f_r < f_l$ calculer le point d'expansion $x_e := c + \gamma(x_r - c)$ et $f_e := f(x_e)$. Si $f_e < f_r$Accepter x_e et terminer l'itération, sinon ($f_e \geq f_r$) accepter x_r et terminer l'itération.

Cette approche de "minimisation gourmande" comprend le meilleur des deux points x_r et x_e dans le nouveau simplex, et le simplex est élargi que si $f_e < f_r < f_l$. Il est utilisé dans la plupart des mises en œuvre et en théorie [38].

Le Nelder-Mead original utilise « l'expansion gourmande », où il est accepté si $f_e < f_l$ et $f_r < f_l$, quel que soit la relation entre f_r et f_e, il peut arriver que $f_r < f_e$, afin que x_r serait un meilleur nouveau point de x_e, et x_e est toujours accepté pour le nouveau simplex. Le simplex de travail est maintenue aussi grand que possible, afin d'éviter une terminaison prématurée d'itérations (illustré dans la figure 4.4).

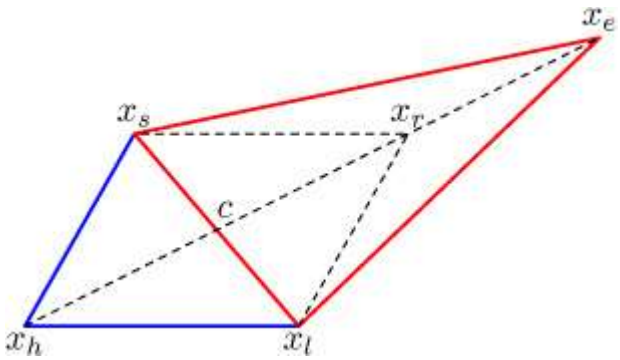

Figure 4.4 : Expansion

Contraction : si $f_r \geq f_s$ **,** calculer le point de contraction x_c en utilisant le meilleur des deux point x_h et x_r.

A l'extérieur : si $f_s \leq f_r < f_h$, calculer $x_c := c + \beta(x_r - c)$ et $f_c := f(x_c)$. Si $f_c \leq f_r$, accepter x_c et terminer l'itération (illustré dans la figure 4.5).

Sinon effectuer une transformation de rétraction.

A l'intérieur : si$f_r \geq f_h$, calculer $x_c := c + \beta(x_h - c)$ et $f_c := f(x_c)$.Si $f_c < f_h$, accepter x_c et arrêter l'itération (illustré dans la figure 4.6).

Sinon effectuer une transformation à la rétraction.

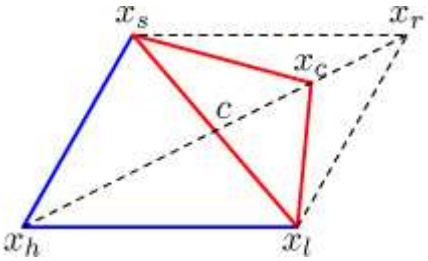

Figure 4.5 : Contraction à l'extérieur

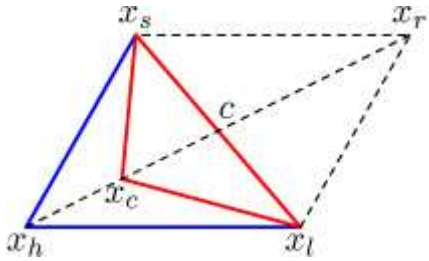

Figure 4.6 : Contraction à l'intérieur

Rétraction : calculer n nouveaux sommets $x_j := x_l + \delta(x_j - x_l)$ et $f_j := f(x_j)$, pour $j = 0, \ldots\ldots, n$ avec $j \neq l$.

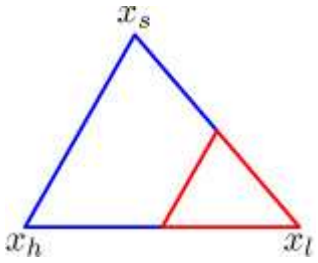

Figure 4.7 : Rétraction

La transformation de rétraction a été introduite pour empêcher l'algorithme d'échouer dans le cas où la contraction augmente la fonction $f$ce qui n'arrive quand on est assez proche d'un point minimum non singulier(illustré dans la figure 4.7).

4.4.3 Les tests d'arrêt

Une mise en œuvre pratique de la méthode de Nelder-Mead doit inclure un test qui assure l'arrêt dans un laps de temps fini. Le test d'arrêt est souvent composé de trois parties différentes :

- *term_x* ,
- *term_f*
- et *fail* .

term_x : est la convergence de domaine ou un test d'arrêt devient vrai lorsque la simplex S est suffisamment faible dans un certain sens (certains ou tous les sommets x_j sont assez proche).

term_f : le test de convergence valeur de fonction. Il devient vrai quand (toutes ou une partie) des valeurs de la fonctionf_j sont assez proches dans un certain sens.

Fail : est le test de non-convergence. Il devient vrai si le nombre d'itérations ou évaluations de la fonction dépasse une certaine valeur maximale prescrite permis.

L'algorithme se termine dès qu'au moins l'un de ces tests devient vrai.

Si l'algorithme est censé travailler pour les fonctions f discontinues, alors il doit avoir une certaine forme d'un test *term_x*. Ce test est également utile pour des fonctions continues, quand un point de minimisation raisonnablement précis est nécessaire, en plus de la valeur de la fonction minime. Dans de tels cas, un test *term_f* est seulement une garantie pour les fonctions "plates".

La détermination du simplex initiale est importante, comme un premier simplex trop petit peut conduire à une recherche locale, augmentant le risque d'échec. Pour cette raison le simplex initial doit être construit avec soin en tenant en compte la nature de problème.

4.5 Mise en œuvre efficace

Après tant d'années d'expérience numérique avec la méthode Nelder-Mead, il y a des preuves accablantes que les transformations de rétraction presque ne pourront jamais se produire dans la pratique. Par conséquent, une itération non rétractable typique de l'algorithme est extrêmement rapide, car il calcule seulement un ou deux points de test et les valeurs de fonction associés. En outre, dans ce cas, le nouveau simplex de travail contient un seul nouveau sommet-le point accepté qui remplace le pire sommet dans l'ancien simplex. Par conséquent, les deux premières étapes dans l'itération suivante peuvent être effectuées de manière plus efficace que dans la mise en œuvre évidente :

- l'ordre des sommets peut être mis à jour en temps linéaire (au plus n comparaisons) par une étape de droite tri par insertion, et
- le nouveau centre de gravité peut également être calculée par la mise à jour du précédent en $O\ (n)$ opérations, avec presque sans stockage supplémentaire.

Une analyse d'une seule itération Nelder-Mead a révélé un goulot d'étranglement de calcul potentiel dans le test de convergence de domaine. Il devient un grave

problème si l'évaluation de chaque fonction est assez rapide par rapport à l'ensemble itération. Pour contourner ce problème, Sasa et Singer (2004) ont proposé un test simple et efficace convergence de domaine basé sur le suivi du "volume linéarisé" relative du simplex de travail. Ce document montre aussi combien peut être acquise par la mise en œuvre efficace des différentes étapes de l'algorithme Nelder-Mead [39].

4.6 La convergence

Une analyse rigoureuse de la méthode Nelder-Mead semble être un problème mathématique très difficile. Résultats de convergence connus pour les méthodes de recherche directe [40], en terme simplex, comptent sur l'une ou l'autre des propriétés suivantes:

a) Les angles entre les bords adjacents des simplex de travail sont uniformément bornées loin de 0 et π à travers les itérations, à savoir le simplex reste uniformément non dégénéré.
b) une certaine forme de condition de descente "suffisante" pour les valeurs de la fonction aux sommets est nécessaire à chaque itération.

En général, la méthode de Nelder-Mead originale ne satisfait pas l'une de ces propriétés. De par sa conception, la forme du simplex de travail peut dégénérer presque pendant qu'il "s'adapte au paysage local", et la méthode utilise seulement simple diminution des valeurs de la fonction aux sommets pour transformer le simplex. Par conséquent, on connait très peu sur les propriétés de convergence de la méthode avec des résultats principalement négatifs.

4.7 Les avantages et les inconvénients

Dans de nombreux problèmes pratiques, comme l'estimation des paramètres et le contrôle des processus, les valeurs de la fonction sont incertaines ou soumises au bruit. Par conséquent, une solution de haute précision n'est pas nécessaire, et il peut être impossible à calculer. Tout ce qui est souhaité est une amélioration de la valeur de la fonction, plutôt que de l'optimisation complète.

La méthode Nelder-Mead donne fréquemment des améliorations significatives dans les premières itérations et produit rapidement des résultats tout à fait satisfaisants. En outre, la méthode nécessite généralement et seulement un ou deux évaluations de la fonction par itération, sauf dans les transformations de rétraction, qui sont extrêmement rares dans la pratique. Ceci est très important dans les applications où

chaque évaluation de la fonction est très coûteuse ou prend beaucoup de temps. Pour ces problèmes, la méthode est souvent plus rapide que les autres méthodes, en particulier ceux qui ont besoin d'au moins de n évaluations de la fonction par itération.

- Dans de nombreux tests numériques, la méthode Nelder-Mead réussit à obtenir une bonne réduction de la valeur de la fonction en utilisant un nombre relativement faible d'évaluations de fonction.

En plus d'être simple à comprendre et à utiliser, c'est la principale raison de sa popularité dans la pratique.

D'autre part, le manque de théorie de la convergence se reflète souvent dans la pratique comme une rupture numérique de l'algorithme, même pour des fonctions dérivables et bien élevés.

- La méthode peut prendre un énorme nombre d'itérations avec une amélioration négligeable de la valeur de la fonction, en dépit d'être loin à un minimum.

Cela se traduit généralement la résiliation prématurée d'itérations. Une approche heuristique pour traiter de tels cas est de redémarrer l'algorithme à plusieurs reprises, avec raisonnablement petit nombre d'itérations permises par chaque série.

4.8 La méthode Nelder-Mead avec le recuit simulé

Lorsque la fonction possède de nombreux minima locaux, il arrive fréquemment de converger vers l'un d'eux et de manquer la solution. Dans tel cas, il possible d'introduire dans la méthode un couplage avec le mécanisme empirique du recuit simulé : à chaque itération, les valeurs effectives de la fonction aux divers sommets sont perturbé par un bruit de fond « thermique » aléatoire dont l'importance décroit au fur et à mesure que l'algorithme progresse [34].

4.9 Conclusion

L'algorithme de simplex heuristique Nelder-Mead pour l'optimisation sans contrainte a été largement utilisé pour résoudre les problèmes d'estimation des paramètres et d'autres problèmes depuis 50 ans. Malgré son âge, il est encore la méthode préférée pour de nombreux praticiens dans plusieurs citons les statistiques, l'engineering, les sciences physiques et médicales, car il est facile à implémenter et

très facile à utiliser. Il appartient à une classe de méthodes qui ne nécessitent pas de dérivés et qui sont souvent prétendu être robuste pour des problèmes avec des discontinuités où les valeurs de la fonction sont bruitées.

Le Simplex algorithme Nelder-Mead a connu une grande popularité. De toutes les méthodes de recherche directe, l'algorithme simplex de Nelder-Mead est le plus trouvé dans les logiciels numériques.

En plus il est facile à coupler avec les autres algorithmes heuristiques comme le recuit pour améliorer les solutions trouvées en cas de plusieurs minimas locaux, cependant le choix des simplex initial est très important pour assurer la convergence de la méthode.

Le chapitre 5 est consacré à la conception et l’implémentation de notre approche.

Chapitre 5 : Implémentation et expérimentation

Résumé

L'approche proposée consiste principalement à construire un modèle de prédiction de la ROP à l'aide des forêts aléatoires, puis utiliser Nelder-Mead Simplex pour trouver les paramètres opérationnels optimisés.

Nous avons utilisé l'outil « learning machine » open source WEKA développé par l'université Waikato [41], Nouvelle-Zélande, les API's de cet outil sont facile à intégrer dans les application java.

Les données de terrain obtenues à partir des puits on shore verticaux forés dans le champs de Hassi Terfa au Sud de l'Algérie en 2012 ont été utilisés dans cette étude.

Le modèle de prédiction est basé sur les paramètres suivants : Le poids sur l'outil, La vitesse de rotation, débit entrant de la boue, la densité de la boue, la pression dans les tiges , le moment de torsion et la résistance à la compression uni-axiale.

La prédiction du ROP qui est une valeur continue, est définie comme problème régression. Trois algorithmes efficaces dans ce genre de problèmes ont été comparé en utilisant l'outil WEKA : Les RNA, les SVM pour régression et forêts aléatoires. Les résultats de comparaison ont montré l'avantage de l'algorithme des forêts aléatoires.

Après le choix de l'algorithme des forêts aléatoires pour modéliser notre prédicteur nous avons optimisé les paramètres de notre prédicteur via un méta-classifieur de l'environnement WEKA.

Nous avons ensuite utilisé l'algorithme Nelder-Mead simplex pour optimiser la vitesse de progression (ROP) au moyen de la sélection du meilleurs poids sur l'outil (WOB) et vitesse de rotation (RPM).

5.1 Introduction

Ce chapitre présente l'implémentation et l'expérimentation de notre approche de prédiction et optimisation du ROP.

L'optimisation efficace de ROP est un élément crucial de la réussite processus de forage pétrolier. Avec l'approfondissement d'exploration et le développement de pétrole et gaz, le succès de forage de puits est devenu de plus en plus difficile. Dans de nombreux cas, des réservoirs de pétrole et de gaz se trouvent profondément sous terre. Il est fréquent que les profondeurs varient de 5.000 à 10.000 mètres sous terre. En raison des propriétés de formation complexes, le temps non productif(NPT) prend une proportion très élevée dans le temps total de forage [40]. Tous ces facteurs conduisent finalement à l'allongement des cycles de forage, et un faible ROP global, qui ralentit sérieusement le progrès de l'exploration et du développement.

En raison de la complexité de pénétration et l'hétérogénéité de la formation, l'approche traditionnelle comme équations mathématiques de ROP et analyse de régression sont confinés par leurs limitations dans la prédiction des paramètres forage, il est donc nécessaire de trouver une méthode de prédiction de ROP commode et relativement précise. Au cours des dernières années, les applications des méthodes d'intelligence artificielle dans l'ingénierie de l'essence ont évolué progressivement.

Dans cette optique plusieurs expériences réalisées dans ce travail sont décrite dans le chapitre 1. Ces expériences ont montré l'avantage des techniques de l'intelligence artificielle pour la prédiction de ROP. Notre approche est basé sur l'algorithme des forêts Aléatoires « Random Forest » et l'algorithme Nelder-Mead simplex.

L'approche consiste principalement à construire un modèle de prédiction de la ROP à l'aide des forêts aléatoires, puis utiliser Nelder-Mead Simplex pour trouver les paramètres opérationnels optimisés. Il est montré dans les résultats que l'approche proposée est en mesure d'obtenir de meilleurs paramètres opérationnels qui optimisent efficacement ROP.

5.2 Les outils utilisés

WEKA (Waikato Environment for Knowledge Analysis)

WEKA est une boite à outils open source d'apprentissage automatique. Ecrite en java, développée à l'université Waikato [41], Nouvelle-Zélande. Il est désormais

utilisé dans beaucoup de domaines différents, en particulier l'éducation et la recherche. Les principaux points forts de WEKA sont :

- librement disponible (en particulier gratuitement) sous la licence publique générale GNU,
- très portable car entièrement implémenté en Java et donc fonctionne sur quasiment toutes les plateformes modernes, et en particulier sur quasiment tous les systèmes d'exploitation actuels,
- contient une collection complète de préprocesseurs de données et de techniques de modélisation, et
- facile à utiliser par un novice en raison de l'interface graphique qu'il contient et facile aussi à intégrer dans les applications JAVA.

Dans notre approche nous avons utilisé l'outil logiciel open source WEKA pour la comparaison des algorithmes de prédiction afin de bien justifier notre choix de l'algorithme des forêts aléatoires, pour l'implémentation le langage JAVA a été choisi avec l'environnement NETBEANS 8.

5.3 Description des données

Les données de terrain obtenues à partir des puits on shore verticaux forés dans le champs de Hassi Terfa au sud de l'Algérie en 2012 ont été utilisés dans cette étude . L'emplacement et le nom du champ sont confidentiels. Donc, ils ne peuvent pas être mentionnés directement. En outre, la formation de la lithologie disponible pour ce domaine est présentée dans le tableau 5.1:

Tableau 5.1 : Description des formations géologiques

Age	**Formation**	**Lithologie**	**La profondeur (m)**
MIO-PLIOCENE		Sand – Sand – Clay - Limestone	9
EOCENE		Dolomitic Limestone – Sand - Clay	191
SENONIAN CARBONATE		Shale - Anhydrite - Calcareous Shale – Dolomite	303
SENONIAN		Anhydrite – Shale –	404

ANHYDRITIQUE		Dolomite - Calcareous Dolomite – Clay - Anhydrite	
SENONIAN SALIFERE		Salt- Clay	604
TURONIAN		Limestone - Clay	746
CENOMANIAN		Anhydrite – Shale - Limestone	863
ALBIAN		Silty Shale – Dolomite – Sand	1025
APTIAN		Dolomitic Limestone - Silty Shale	1372
BARREMIAN		Silty Shale – Dolomite – Siltstone - Sand	1398
NEOCOMIAN		Shale - Anhydrite	1673
MALM		Dolomite – Shale - Anhydrite	1888
DOGGER ARGILEUX		Anhydrite – Silty Clay	2115
DOGGER LAGUNAIRE		Anhydrite - Clay	2344
LIAS LD1		Anhydrite - Shale	2436
LIAS LS1		Salt - Anhydrite - Shale	2477
LIAS LD2		Dolomite – Anhydrite – Shale	2601
LIAS LS2		Salt	2660
Horizon B		Dolomitic Limestone	2719
TRIAS S1+S2		Anhydrite - Salt	2754
TRIAS S3		Salt	2967
TRIAS ARGILEUX (G10)		Shale	3195

ERUPTIFS TRIASIQUE		Shale	3281
ORDOVICIEN	GRES DE OUARGLA		3291
	QUARTZITES EL HAMRA	Sandstone	3298
	Gres d'El Atchane	Sandstone –Silty Shale -Siltstone	3455
	Argile d'El Gassi	Silty Shale – Siltstone	3474

Deux types d'outils (PDC et de diamants imprégnés) ont été utilisés à travers les opérations de forage dans ce champ. En raison de la similitude de la lithologie et de la conception des puits.

Nous avons choisi la phase 6 pouces pour notre application sur trois puits de champ Hassi-Terfa.

Notre modèle de prédiction de la ROP est basé sur les paramètres suivants :

1) WOB : Le poids sur l'outil.
2) RPM : La vitesse de rotation.
3) Flow In : débit entrant de la boue.
4) MWI : la densité de la boue.
5) SPP : la pression dans les tiges.
6) Torque : le moment de torsion.
7) UCS : la résistance à la compression uni-axiale :

Tous les paramètres sont de type surface (mud logging), sauf UCS qui est un paramètre de fond qui peut se calculer par les tests de laboratoire ou à partir des logs sonic (gamma ray).

5.4 Notre démarche

5.4.1 **Schéma global**

Pour implémenter notre approche nous allons passer par les étapes suivantes (illustré dans la figure 5.1)

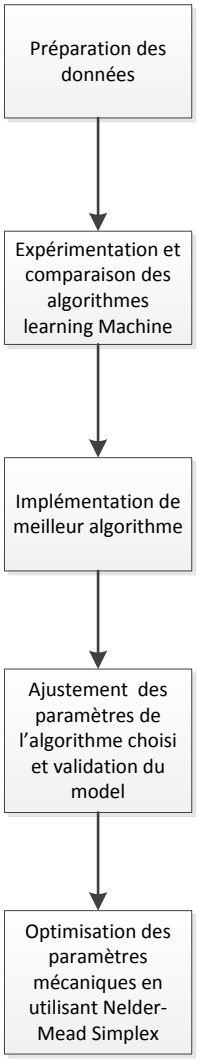

Figure 5.1 : Schéma général de l'approche.

5.4.2 Préparation des données

Trois bases de données ont été préparées pour tester et comparer les algorithmes de learning machine pour régression. Ces bases de données sont scindées en deux parties (66% pour apprentissage et 34% pour test) avec les paramètres cités dans la description des données.

5.4.3 Expérimentation et comparaison des algorithmes

La prédiction de la ROP qui est une valeur réelle (continue) est considéré comme un problème de régression, l'outil WEKA est doté de plusieurs algorithmes pour prendre en charge de ce type de problème, parmi eux nous avons choisi les plus performants d'après les résultats des expérimentations dans l'environnement WEKA et les travaux rencontrés dans la littérature. Citons les RNA, les SVM et les forets aléatoires qui feront l'objet de notre expérimentation et comparaison.

- **Les réseaux de neurones artificiels**

Un réseau de neurones artificiels est un assemblage d'objets informatiques dont l'organisation et le fonctionnement sont inspirés de ceux des neurones biologiques.

Les réseaux de neurones sont utilisés en analyse décisionnelle dans sa partie datamining pour la prédiction, la classification et l'analyse des données, il existe plusieurs architectures pour les réseaux de neurones, L'architecture choisie est de type MLP (Multi Layer Perceptron), l'apprentissage du classifier MLP implémenté sous WEKA est optimisé en minimisant l'erreur quadratique plus une pénalité quadratique avec la méthode de BFGS (Broyden-Fletcher-Goldfarb-Shanno : est une méthode permettant de résoudre un problème d'optimisation non linéaire sans contraintes)[43].

En ce qui concerne la couche cachée le nombre choisi par défaut est 4(illustré dans la Figure 5.2).

Légende

WOB : le poids sur l'outil en Tonnes

RPM : La vitesse de rotation en Tour/Minute

Flow In : le débit la boue en Litre/Minute

MWI : la densité de la boue en kilogramme/Litre

SPP : la pression dans les tiges en PSI (pound-force per square inch).

Torque : le moment de torsion en Pieds livre-force.

UCS : la compression à la résistance uni axiale en méga pascal.

ROP : la vitesse de progression mètre/heure

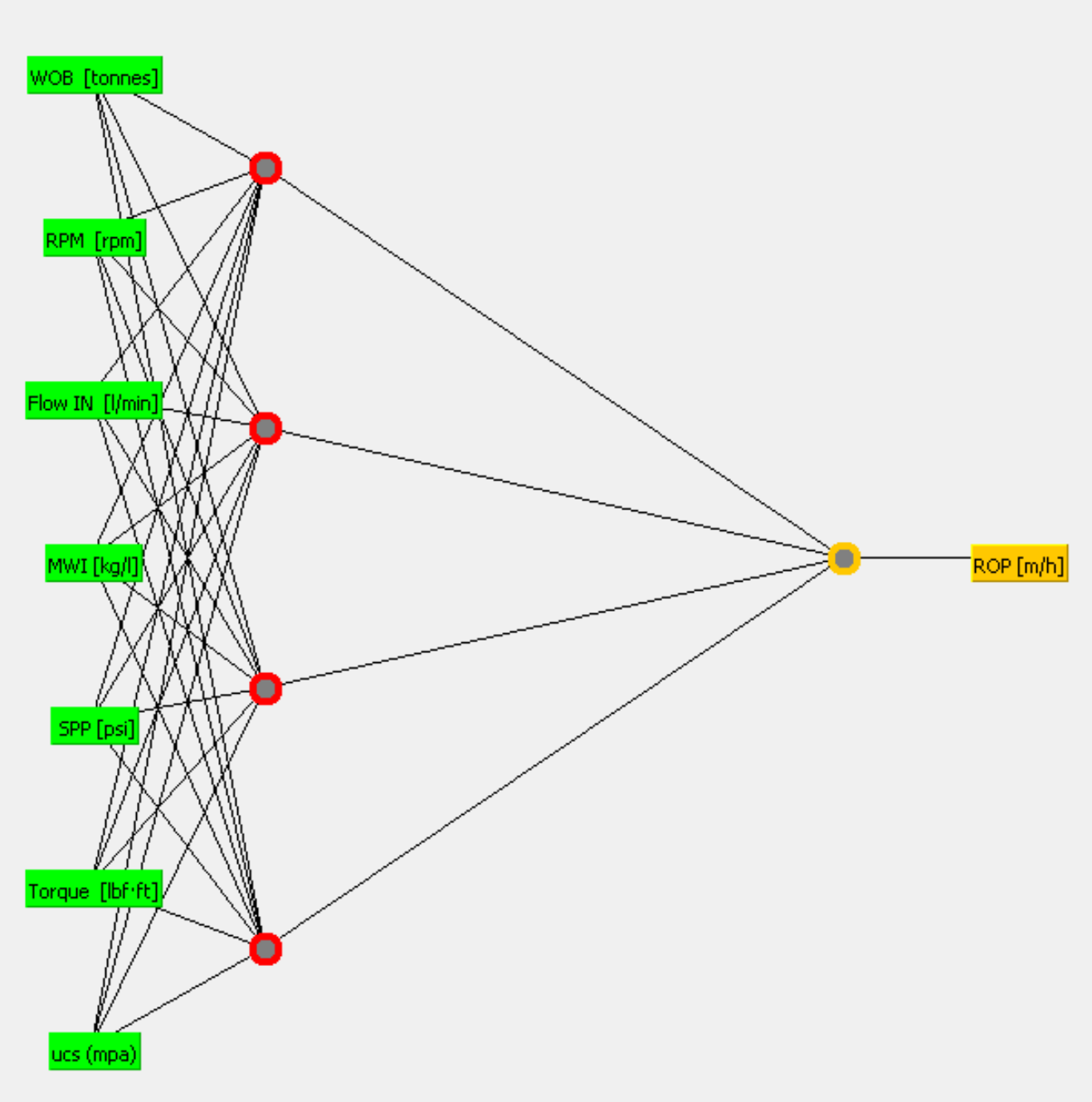

Figure 5.2 : Architecture de RN utilisé pour la prédiction du ROP

- **Les machines à vecteurs de support (SVM)**

Les machines à vecteurs de support ou séparateurs à vaste marge (en anglais Support Vector Machine, SVM) sont un ensemble de techniques d'apprentissage supervisé destinées à résoudre des problèmes de classification et de régression. Les SVM sont une généralisation des classifieurs linéaires.

Les SVM ont été développés dans les années 1990 à partir des considérations théoriques de Vladimir Vapnik sur le développement d'une théorie statistique de l'apprentissage : la théorie de Vapnik-Chervonenkis. Les SVM ont rapidement été

adoptés pour leur capacité à travailler avec des données de grandes dimensions, le faible nombre d'hyper paramètres, leurs garanties théoriques, et leurs bons résultats en pratique.

Les SVM ont été appliqués à de très nombreux domaines (bio-informatique, recherche d'information, vision par ordinateur, finance…). Selon les données, la performance des machines à vecteurs de support est de même ordre, ou même supérieure, à celle d'un réseau de neurones ou d'un modèle de mélanges gaussiens [44].

Le classifier WEKA utilisé s'appelle SMOreg (sequential mininmal optimization for regression) crée par Alex J.Smola and Bernhard Scholkopf qui implémente les support vecteur machines pour régression dont l'apprentissage est effectué en utilisant les noyaux polynomiaux ou RBF. Ce dernier est plus performant que les autres classifieurs des SVM comme SVR suite aux expérimentations effectuées dans l'outil WEKA.

- **Les forêts aléatoires**

Les forêts d'arbres décisionnels (ou forêts aléatoires de l'anglais « Random decision forest ») ont été formellement proposées en 2001 par Leo Breiman et Adèle Cutler. Elles font partie des techniques d'apprentissage automatique. Cet algorithme combine les concepts de sous-espaces aléatoires et de « bagging ». L'algorithme des forêts d'arbres décisionnels effectue un apprentissage sur de multiples arbres de décision entraînés sur des sous-ensembles de données légèrement différents (voir le chapitre 3).

Les paramètres les plus importants : nombre d'arbres par défaut est 100, nombre de paramètres choisi aléatoirement par défaut est $\frac{p}{3}$ pour régression (p : nombre total de paramètres).

Le critère de comparaison est le taux de corrélation, il existe deux mode d'évaluation :

1) **Division en ensemble d'apprentissage et test**

Ce mode consiste à diviser la base de données en deux parties, la première (66%) pour l'apprentissage et le reste pour le test. Le modèle est bâti sur l'échantillon d'apprentissage et validé sur l'échantillon de test. L'erreur est estimée en calculant un test, une mesure ou un score de performance du modèle sur l'échantillon de test, par exemple l'erreur quadratique moyenne [45].

2) Validation croisée (cross-validation).

La validation croisée(« cross-validation ») est une méthode d'estimation de fiabilité d'un modèle fondé sur une technique d'échantillonnage, on divise l'échantillon original en k échantillons, puis on sélectionne un des k échantillons comme ensemble de validation et les (k-1) autres échantillons constitueront l'ensemble d'apprentissage. On calcule comme dans la première méthode l'erreur quadratique moyenne. Puis on répète l'opération en sélectionnant un autre échantillon de validation parmi les (k-1) échantillons qui n'ont pas encore été utilisés pour la validation du modèle. L'opération se répète ainsi k fois pour qu'en fin de compte chaque sous-échantillon ait été utilisé exactement une fois comme ensemble de validation. La moyenne des k erreurs quadratiques moyennes est enfin calculée pour estimer l'erreur de prédiction [46].

3) L'expérimentation

L'expérimentation s'est déroulée avec les paramètres suivants :

- 10 itérations de contrôle
- 66% ensemble d'apprentissage et 34% ensemble de test
- 10 échantillons pour la validation croisée
- Le critère de comparaison choisi est le taux de corrélation.

Après l'expérimentation avec le premier mode d'évaluation (train/test) nous avons obtenu les résultats présentés dans le tableau 5.2

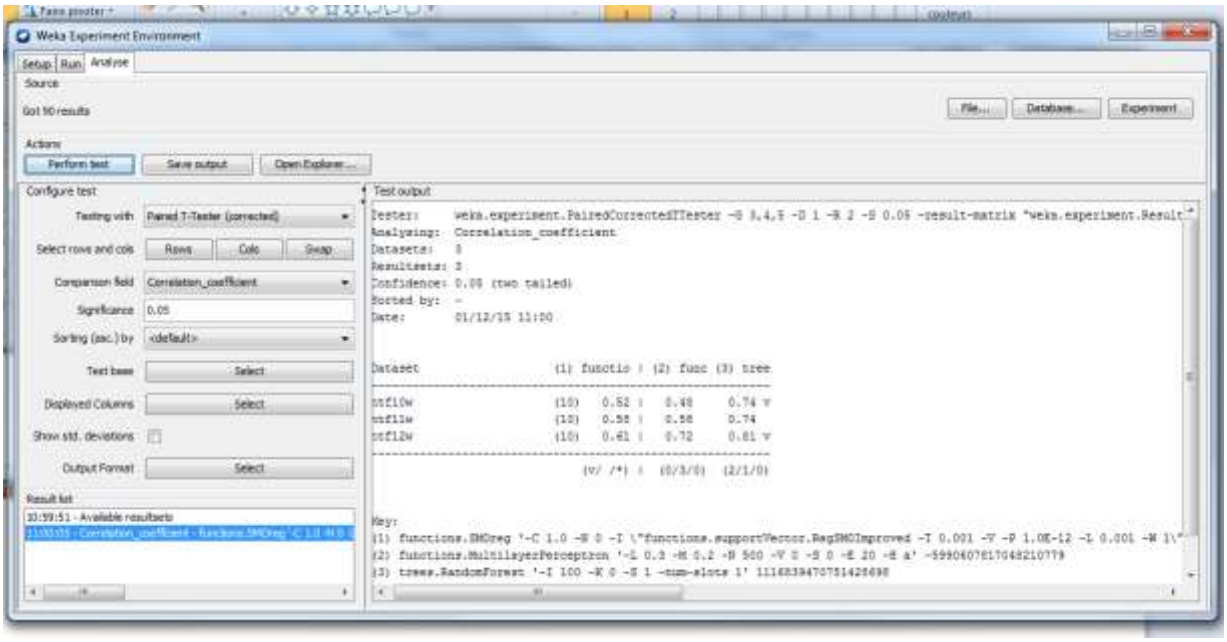

Figure 5.3 : l'expérimentation et l'évaluation en mode (train\test) sous WEKA.

Tableau 5.2 : Résultats de comparaison entre les algorithmes de régression en mode d'évaluation (train/test) source « Figure 5.3 »

	RN **MLP**	Les SVM **SMOReg**	Les forêts aléatoires **Random forest**
Base de données 1	0.48	0.52	**0.74**
Base de données 2	0.58	0.58	**0.74**
Base de données 3	0.72	0.61	**0.81**

Avec la validation croisée (10 échantillons) nous avons obtenu les résultats présentés dans le tableau 5.3

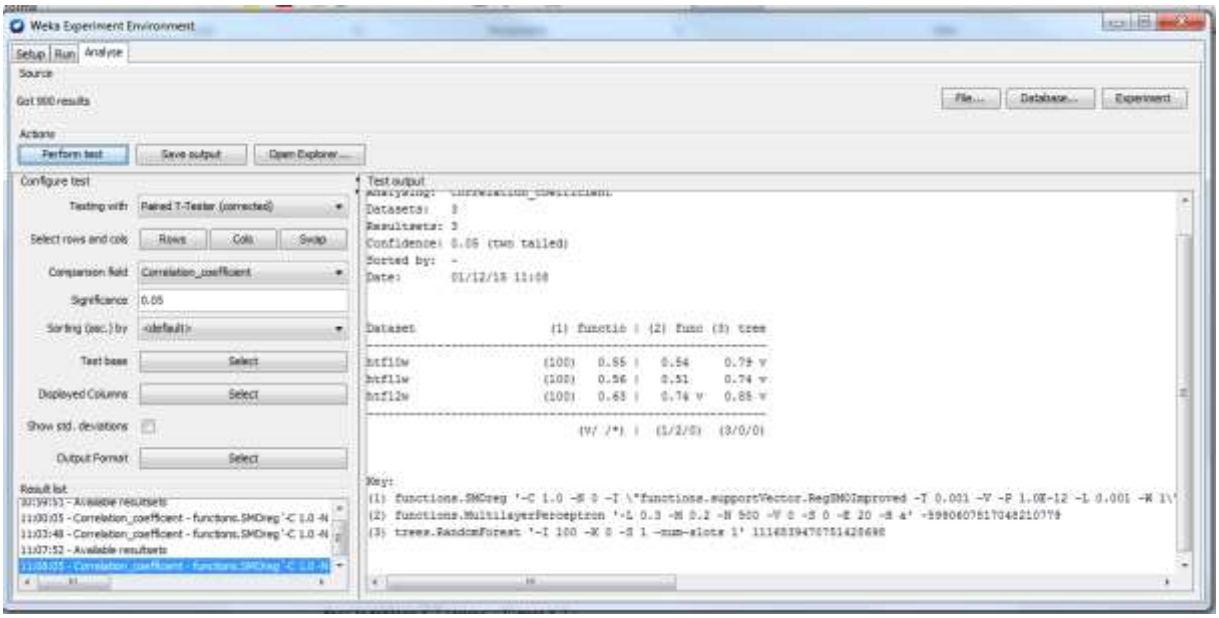

Figure 5.4 : : l'expérimentation et l'évaluation en mode de validation croisée sous WEKA.

Tableau 5.3 : Résultats de comparaison entre les algorithmes de régression en mode de validation croisée source figure 5.4

	RN **MLP**	Les SVM **SMOReg**	Les forêts aléatoires **Random forest**
Base de données 1	0.54	0.55	**0.79**
Base de données 2	0.51	0.56	**0.74**
Base de données 3	0.74	0.63	**0.85**

Les résultats des tableaux (5.2 et 5.3) montrent que L'algorithme des forêts aléatoires a le plus grand taux de corrélation. Alors il est le meilleur choix pour prendre en charge notre problème de prédiction.

5.4.4 Réglage des paramètres de l'algorithme des forets aléatoires

La configuration des paramètres pourraient avoir un grand impact sur la précision de la prédiction du modèle formé. La configuration optimale des paramètres est différente souvent pour les différents ensembles de données. Par conséquent, ils doivent être réglés pour chaque ensemble de données. Depuis le processus de formation ne définit pas les paramètres, il doit y avoir un processus réglage des paramètres (illustré dans la figure 5.5).

Les modèles d'apprentissage de la machine sont paramétrés de sorte que leur comportement peut être ajusté pour un problème donné. Les modèles peuvent avoir de nombreux paramètres et de trouver la meilleure combinaison de paramètres peuvent être traités comme un problème de recherche [48].

Le réglage de l'algorithme est une étape finale dans le processus de l'apprentissage appliquée de la machine avant la présentation des résultats.

La recherche des paramètres optimaux pour un classificateur peut être un processus assez fastidieux, WEKA propose quelques façons d'automatiser ce processus un peu. Les méta classificateurs suivants permettent d'optimiser certains paramètres de classificateur de base[47]:

- **CVParameterSelection :** Ce méta classifier peut optimiser sur un nombre arbitraire de paramètres en utilisant la validation croisée.
- **GridSearch:** est une approche de réglage des paramètres qui seront méthodiquement construire et évaluer un modèle pour chaque combinaison de paramètres de l'algorithme spécifiés dans une grille.
- **MultiSearch :** est similaire à GridSearch, plus générale et plus simple en même temps. Elle permet l'optimisation d'un nombre quelconque de paramètres, et pas seulement deux.

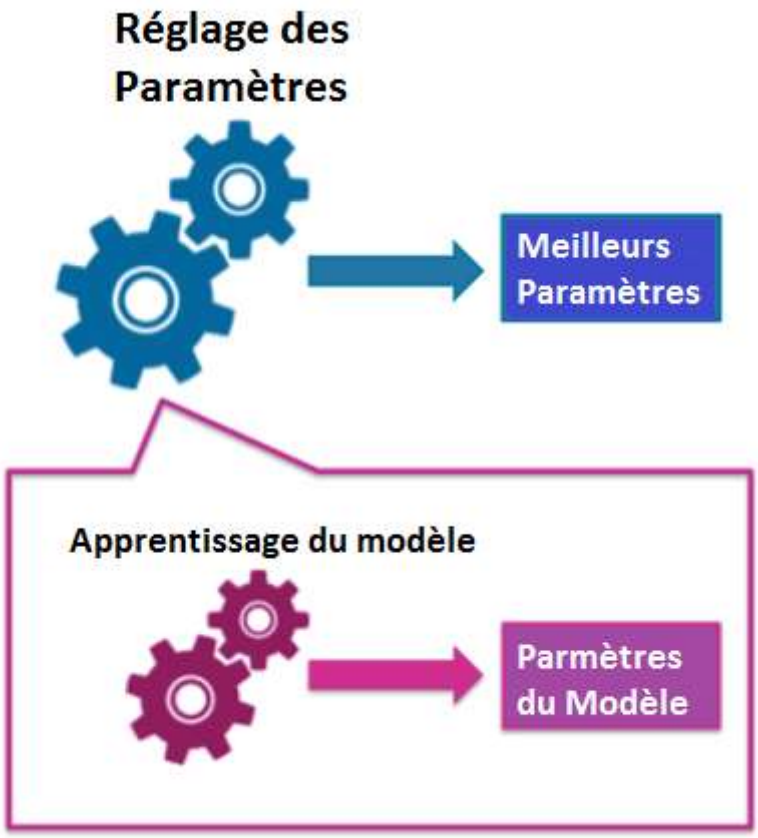

Figure 5.5: Optimisation des paramètres d'une machine d'apprentissage [48]

Nous avons utilisé le méta classifier **CVParameterSelection** pour optimiser le paramètre de nombre d'arbre ***'ntree '*** dans notre algorithme Random Forest, la base de données 3 est utilisée pour évaluation, après l'exécution du méta-classifier nous avons obtenu un coefficient de 0.87 avec le nombre d'arbres égal à 70.

5.4.5 Résultats de prédiction du ROP

Nous prenons la base de donnée 3 (puits 3 phase 6 pouces) dans nos données collectée. Cette zone est un réservoir ORDOVICIEN, notre prédicteur crée avec l'algorithme des forêt aléatoires est capable de prédire la ROP avec une très bonne précision (coefficient de corrélation) de 0,87.

La base de données est divisée en deux parties (66% pour apprentissage) et le reste pour le test.

Les résultats de prédiction sont illustrés dans la figure 5.6

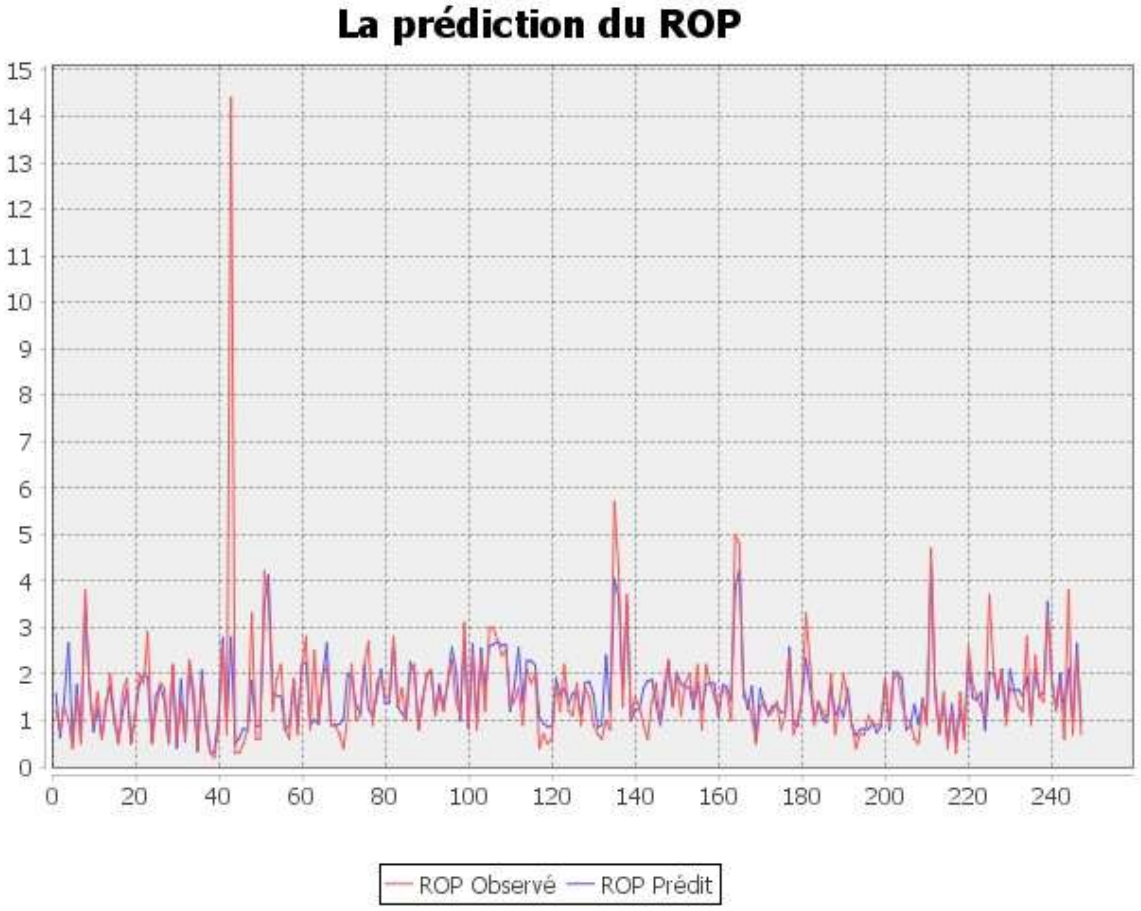

Figure 5.6 : Comparaison entre ROP observé et prédit

Après avoir validée et testé notre prédicteur nous passons à l'optimisation des paramètres mécaniques (WOB, RPM).

5.4.6 Optimisation du ROP

Après l'obtention d'un modèle de prévision de ROP satisfaisant, la prochaine tâche est d'essayer de maximiser la valeur de sortie (ROP) du modèle. Il y a un petit nombre de paramètres qui peuvent être ajustés. Selon la pratique de forage, nous avons choisi deux principaux paramètres de fonctionnement: le poids sur l'outil(WOB) et la vitesse de rotation(RPM). Les deux paramètres sont autorisés et facilement pour régler dans la pratique, et ils sont généralement réglés manuellement dans de nombreux processus de forage de monde réel. L'ajustement manuel se repose principalement sur des expériences humaines. Notre modèle de prédiction élaboré est une fonction boite noire, non linéaire et non dérivable, pour optimiser ce modèle la méthode heuristique Nelder-Mead Simplex (voir chapitre 4) est le meilleur choix à cause de sa simplicité et rapidité.

5.4.7 Implémentation de Nelder-Mead Simplex pour optimisation du ROP

Cette méthode est conçu pour résoudre les problèmes de minimisation, une adaptation est nécessaire aux problèmes de maximisation.

Nous avons deux paramètres à optimiser WOB et RPM, alors le simplex aura deux dimension (2-D), le simplex peut être considéré comme un polygone de $n+1$ sommets, dans notre cas $n=2$, le simplex est un triangle.

Considérons trois points $[u, v, w]$ dans un plan $p1-p2$, ces trois point sont reliés entre eux par un triangle et la fonction objectif est évaluée aux trois points : $f(u), f(v)\ et\ f(w)$.Nous allons suivre les étapes suivantes pour améliorer de manières itératives les sommets de triangle de manière à maximiser $f(p)$.

Prise en compte des bornes dans l'algorithme Nelder Mead Simplex

Les paramètres mécaniques de forage pétrolier (WOB, RPM) sont bornés. Pour leur appliquer la méthode de Nelder-Mead il faut traiter les points qui sortent des bornes. Pour régler ce problème nous devons faire une projection sur les bornes. Pour $i=1,..,n,$ le projeté x^p de x sur les bornes est défini par :

Si $\left(x_i < x_i^{min}\right)$ alors $x_i^p = x_i^{min}$

Si $(x_i > x_i^{max})$ alors $x_i^p = x_i^{max}$

La projection peut intervenir après les étapes de réflexion ou d'expansion.

Les étapes de l'algorithme Nelder-Mead

1. **Trier** les sommets de telle sorte que $f(u) > f(v) > f(w)$. le point u est le meilleur point, le point v est le deuxième pire et le point w est le pire point.
2. **Réfléchir** le pire point w, à travers le barycentre des points restant $(u\ et\ v)$ pour obtenir le point de réflexion r, et évaluer $f(r)$. Projeter r si il est hors les bornes.

   Si $f(u) > f(r) > f(v)$ remplacer le pire point w par le point réfléchi r , et aller à l'étape 5.
3. Si $f(r) > f(u)$, **étendre** le point de réflexion r, vers le point e

   Projeter e si il est hors les bornes.

   a) Si $f(e) > f(r)$, remplacer le pire sommet w par le point d'extension e, et aller à l'étape 5.

   b) Sinon remplacer le pire sommet w par le point de réflexion r, et aller à l'étape 5.

4. Si les tests 2 et 3 ne sont pas satisfaits, alors il est certain que le point de réflexion r est pire que le deuxième pire point$v$, $(f(v) > f(r))$, et une valeur supérieure de f, alors on essaye de **contracter** le pire point au point de contraction c, il **existe** deux points de contraction intérieur et extérieur notés par $c_i et\ c_o$.
    a) Si $max[f(c_i), f(c_o)] > f(v)$, alors remplacer w par le meilleur des deux point de contraction $c_i et\ c_o$ et aller à l'étape 5.
    b) Sinon rétracter le simplex vers le meilleur point u , et aller l'étape 5.
5. Tester la convergence
    a) Si $2 \max \left|\frac{[u,v]-[v,w]}{[u,v]+[v,w]}\right| < \epsilon_p$

    Et $max \left|\frac{f(u)-[f(v),f(w)]}{f(u)+10^{-9}}\right| < \epsilon_f$

    Alors
    - la différence de paramètres entre les sommets adjacents est inférieure à ϵ_p fois la moyenne de paramètres de sommets adjacents et,
    - la fonction objectif f, à tous les sommets est dans ϵ_f fois la meilleure valeur de la fonction objectif.

      Ainsi, les itérations ont convergé, et l'algorithme est terminé.

    b) Sinon, si le nombre d'évaluations de la fonction a dépassé une limite spécifiée alors l'algorithme est terminé.
    c) Sinon, retournez à l'étape 1 pour la prochaine itération.

La figure 5.7 illustre l'organigramme de notre algorithme proposé.

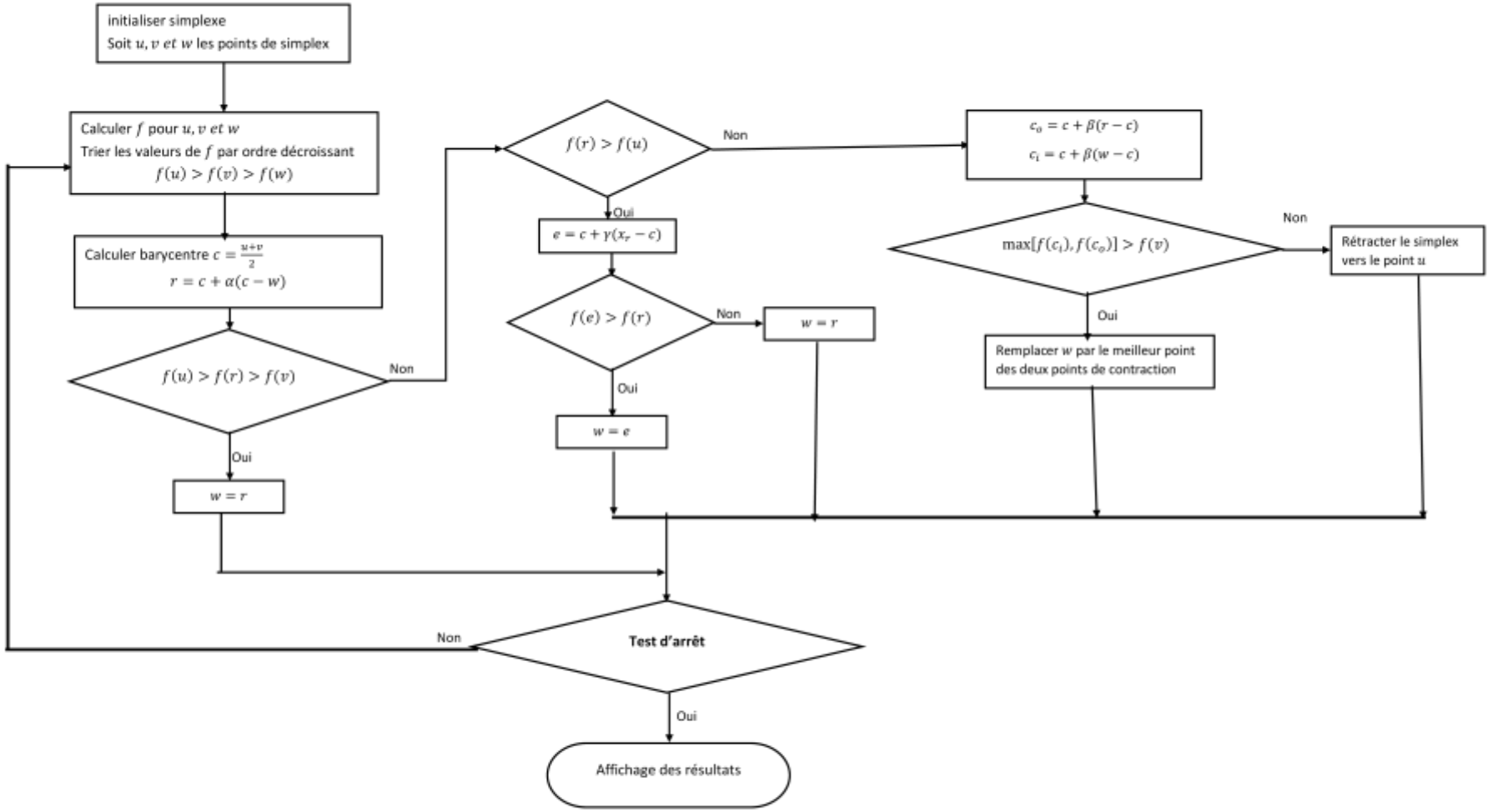

Figure 5.7 : Organigramme de la méthode Nelder-Mead pour maximisation d'un problème 2D

Liste des variables utilisée

u, v, w: points de simplex

c : barycentre

r: point de réflexion

e: point d'expansion

c_i : point de contraction intérieur

c_o : point de contraction extérieur

α : coefficient de réflexion

β: coefficient de contraction

γ : coefficient d'expansion

Nous avons implémenté Nelder-Mead simplex en langage java sous l'environnement de développement intégré Netbeans 8 pour optimiser ROP.

5.5 Discussion

Comme la figure 5.8 montre que les paramètres optimisés mènent directement à un meilleure ROP de la sortie originale. La séquence des paramètres d'exploitation sont en mesure d'être sauvé pour les futures applications telles que la simulation ou de contrôle de forage. Le résultat de l'expérience a montré que Nelder-Mead simplex est un moyen efficace pour trouver les meilleurs paramètres d'entrée dans le problème d'optimisation ROP. Le résultat est passionnant, et est en mesure d'éviter les inconvénients de réglage manuel.

Il est intéressant, même si notre modèle de prédiction n'est toujours pas précis à 100%, comme le montre la figure 5.6, l'approche proposée peut encore obtenir un meilleur ROP que le processus réel dans la plupart du temps.

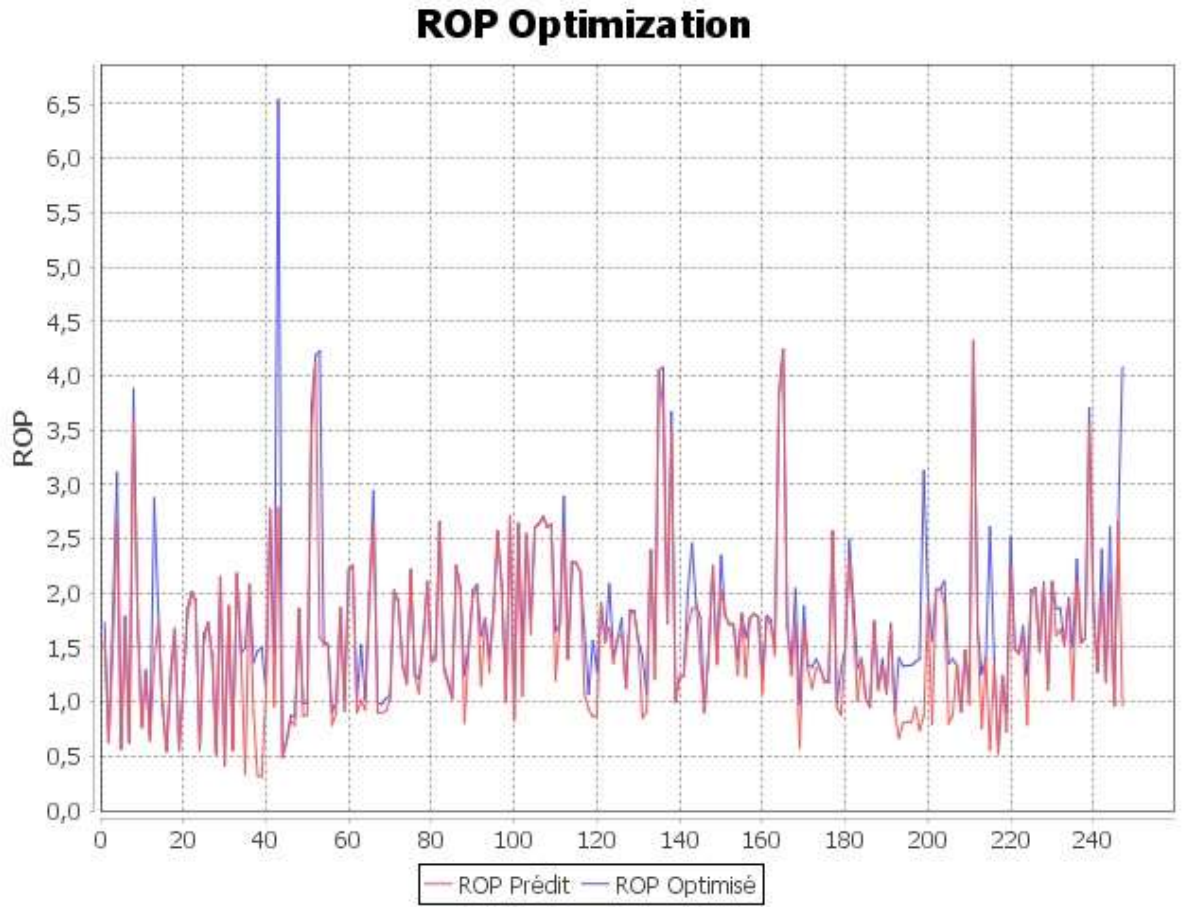

Figure 5.8 : Comparaison entre ROP prédit et optimisé

5.6 L'application développée

Nous présentons ici des captures d'écran de l'application développée.

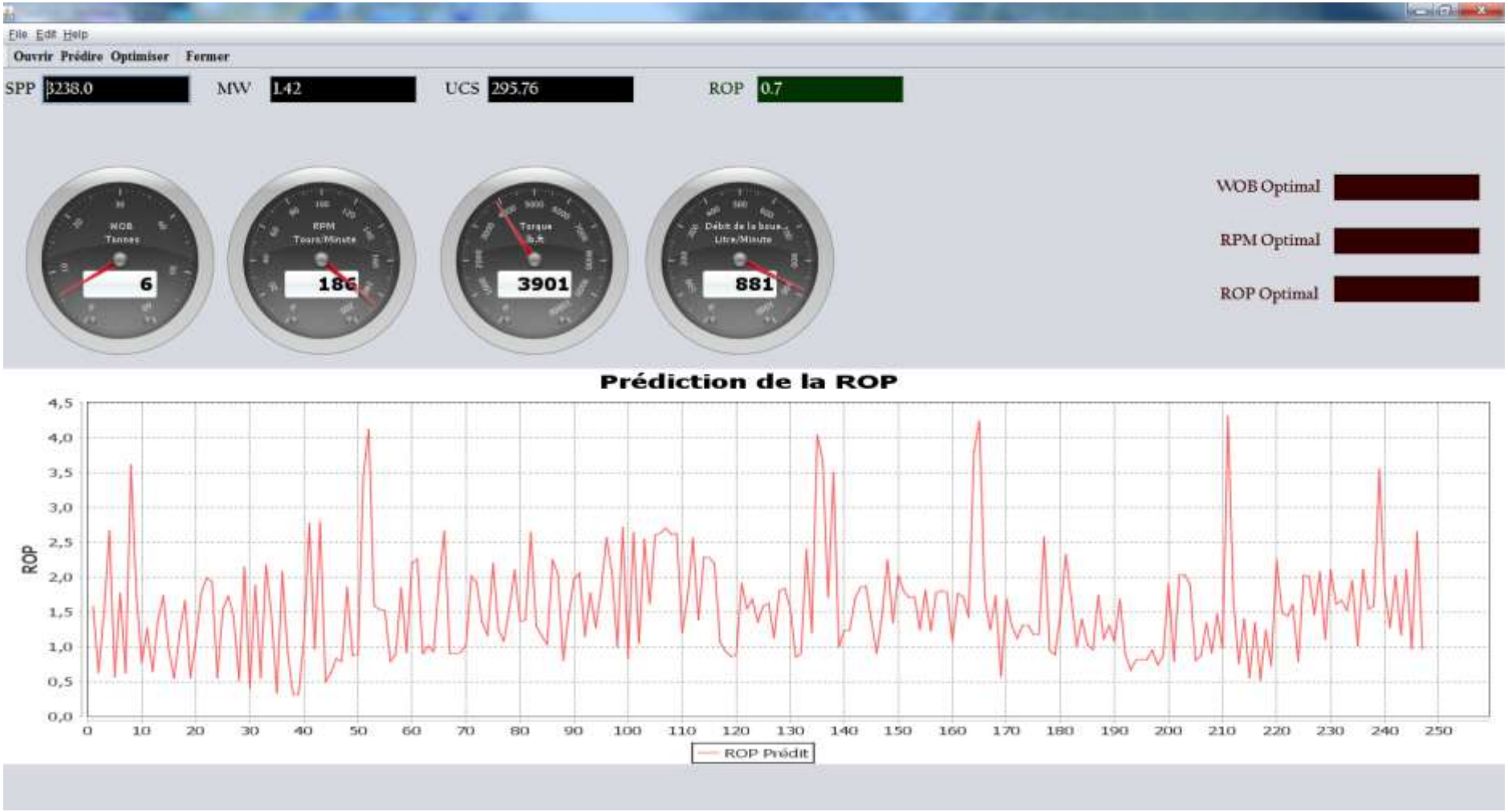

Figure 5.9 : Capture d'écran 1 de l'application SIADDRILL

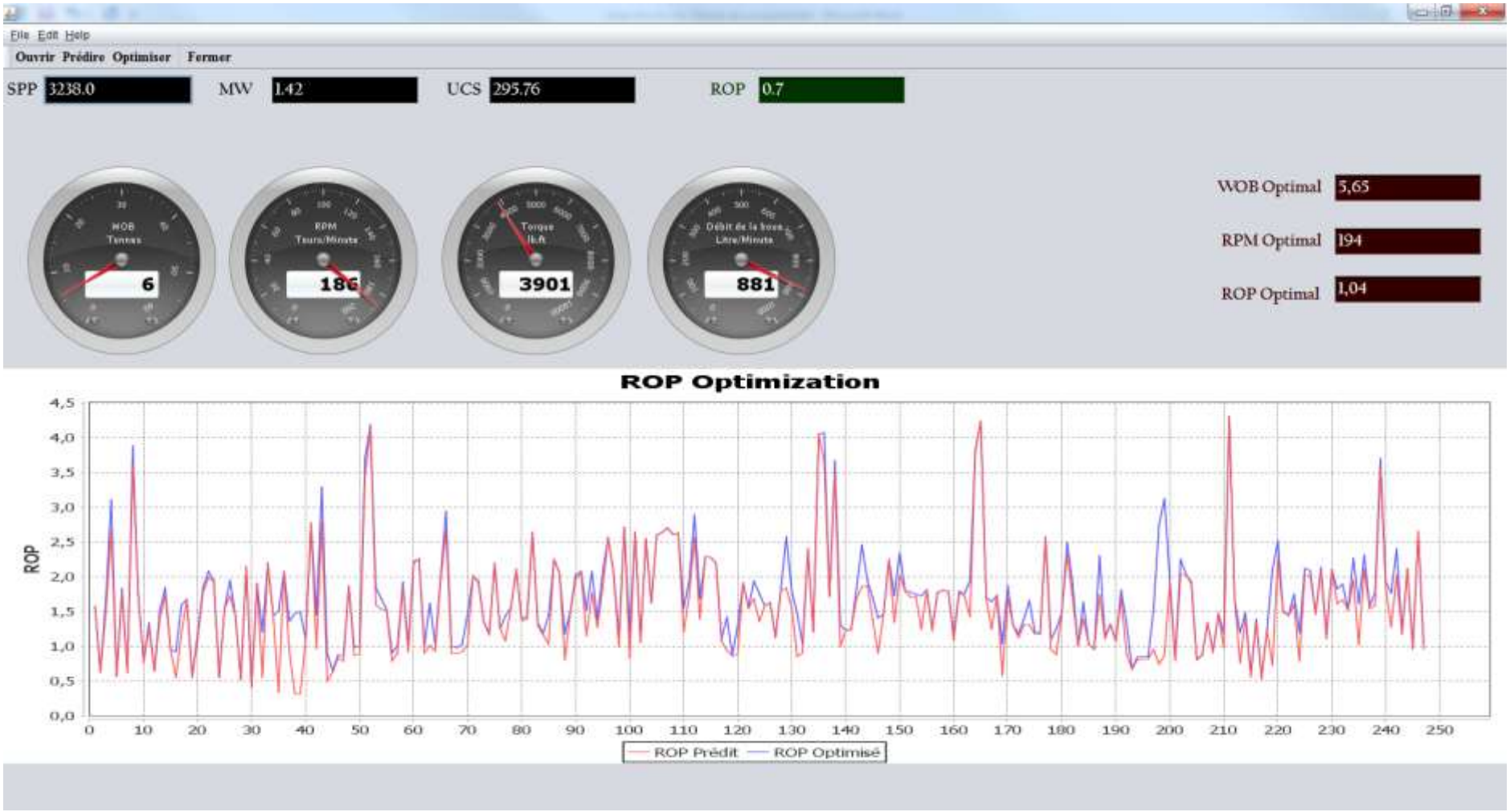

Figure 5.10 : Capture d'écran 2 de l'application SIADDRILL

5.7 Conclusion

Dans ce travail de recherche, nous avons présenté une approche d'optimisation de ROP qui permet d'utiliser pleinement la capacité d'apprentissage automatique des forêts aléatoires et la capacité de la solution de recherche rapide de simplex heuristique Nelder-Mead. L'approche obtient également les meilleurs paramètres de fonctionnement grâce à l'algorithme Nelder-Mead simplex, les résultats de l'étude de cas montrent que notre approche est fiable et stable. Le résultat d'optimisation sont également passionnante, l'approche s'est avérée être en mesure d'augmenter le ROP efficacement. Le travail dans ce document indique que nous avons proposé une méthode simple et efficace pour optimiser la ROP pour l'industrie pétrolière moderne. La mise en pratique de notre application peut réduire en moyenne le temps des opération de forage de 20%. Elle peut servir comme système d'aide à la décision pour les opérateur et les ingénieurs forage.

Conclusion générale

Une méthodologie de prédiction et d'optimisation de la vitesse de forage ROP a été développée, démontrée et appliquée afin atteindre des paramètres de forage contrôlables optimales. La première tâche de cette étude est la prédiction de la vitesse de progression de forage (ROP). Les données utilisées dans le cadre de cette étude appartiennent à des puits forés en Sud de l'Algérie au champ de Hassi Terfa. La vitesse de progression de forage(ROP) est prédite à l'aide l'algorithme des forêts aléatoires pour régression qui est plus performant que les autres algorithmes de régression à savoir les RNA et les SVM d'après les comparaisons que nous avons effectuées dans l'outil Learning-machine WEKA présentées sur les tableaux 5.2 et 5.3. Les résultats ont indiqué que la vitesse de forage peut être prédite avec une haute précision, le taux de corrélation est 0.87 après le choix des meilleurs paramètres de notre algorithme des forêts aléatoires.

L'algorithme de simplex heuristique Nelder-Mead a été utilisé pour optimiser les paramètres mécaniques WOB et RPM afin d'avoir un meilleur avancement. Cet algorithme se caractérise par la vitesse et la simplicité de mise en œuvre. Les résultats obtenus ont montré son efficacité.

Au vu des autres approches, la nôtre se distingue par :

- Le classifieur utilisé « Forêts aléatoires » est de type ensembliste qui combine plusieurs classifieur apprennent sur des sous-ensembles de données d'apprentissage et agrège ces derniers pour produire un classifieur très performant.
- La comparaison entre les classifieurs les plus répondus pour notre problème est effectué à l'aide de l'expérimenter de l'outil WEKA qui répète l'expérience n fois (n=10 dans notre cas) avec partitionnement aléatoire de la base de données en deux modes d'évaluation « train\test » et «validation croisée ».
- L'optimisation avec l'algorithme Nelder-Mead simplex est appliquée directement sur le modèle de prédiction où la fonction objectif « fitness » est une boite noire, au lieu créer une fonction linéaire à la sortie du modèle [40].

Notre application peut être utilisée comme aide à décision pour élaborer les programmes de forage, elle permet de choisir paramètres mécaniques (WOB et RPM) pour avoir une meilleure vitesse de forage (ROP).

Elle peut être utilisée en temps réel à condition avec la technologie LWD qui signifie la diagraphie pendant le forage pour capter tous les paramètres nécessaires à notre prédicteur.

L'utilisation de notre application peut réduire le temps de forage en moyenne de 20%.

Elle peut servir comme un système d'aide à la décision pour optimiser les opérations de forage en temps réel .

Perspectives

Notre modèle de prédiction et optimisation de vitesse de forage a donné des résultats encourageants. Il peut être testé sur le terrain afin de l'améliorer et rentabiliser.

Les méthodes utilisées dans ce travail peuvent être étendues. à :

- utiliser les forêts aléatoires en ligne qui travaillent avec le principe d'apprentissage incrémental (en ligne).
- prendre en charge des données bruitées en utilisant des algorithmes de filtrage de données.
- utiliser Nelder-Mead simplex ou l'optimisation bayésienne pour l'optimisation des paramètres de notre algorithme des forets aléatoires.
- Hybrider des algorithmes génétiques avec Nelder-Mead simplex pour améliorer l'optimisation du ROP.
- prendre en charge la partie usure de l'outil en utilisant d'autres paramètres de formations tel que la porosité et la forabilité.
- développer un modèle de prédiction du paramètre mécanique comme UCS (la compression à la résistance uni axiale) qui est un peu difficile à calculer ce qui permet d'améliorer notre modèle de prédiction et optimisation.
- développer un système à base d'agents qui prend en charge tous les aspects de forage pour servir un système d'aide à la décision plus complet.

Références

[1] F. Akgun, Drilling rate at the technical limit, Int. J. Pet. Sci. Tech., vol.1, no.1, pp.99-118, 2007.

[2] J. Ricardo, P. Mendes and T. C. Fonseca, Applying a genetic neuro-model reference adaptive controller in drilling optimization, World Oil Mag., vol.288, no.10, pp.29-36, 2007.

[3] T. Bourgoyne and F. S. Young, A multiple regression approach to optimal drilling and abnormal pressure detection, Soc. Pet. Eng. J., vol.14, no.4, pp.371-384, 1974.

[4] A. Bahari and A. Baradaran Seyed, Trust-region approach to find costant of Bourgoyne and Young penetration rate rate model in Khangiran Iranian gas field, Proc. of SPE Latin American and Caribbean Petroleum Engineering Conf., Buenos Aires, Argentina, pp.5-18, 2007.

[5] T. F. Coleman and Y. Li, On the convergence of reflective Newton methods for large-scale nonlinear minimization subject to bounds, Mathematical Programming J., vol.67, no.2, pp.189-224, 1994.

[6] T. F. Coleman and Y. Li, Interior, Trust region approach for nonlinear minimization subject to bounds, Siam J. Optim., vol.6, no.2, pp.418-445, 1996.

[7] H. Moradi, M. H. Bahari, M. B. Naghibi-S and A. Bahari, Drilling rate prediction using an innovative soft computing approach, Sci. Res. Assays, vol 5, 2010.

[8] M.H. Bahari, A. Bahari and H. Moradi, Intellegent drilling rate predictor, Int J of Innovative computing, information and control, vol 7, num 4,pp 1511-1519, 2011.

[9] A. F. Al-Rashidi, Designing Neural Networks for the Prediction of the Drilling Parameters for Kuwait Oil and Gas Fields, Mémoire de master en ingénierie de gaz et pétrole, univérsité de Viginia ouest, Etats Unies 1999.

[10] M. Monazami, A. Hashemi and M. Shahbazian, Drilling rate of penetration prediction using artificial neural network : A case study of one of Iranian Southern Oil Fields, Electronic scientific journal “Oil and Gas Business”, 2012, № 6.

[11] Carlos M. C. Jacintoa, Paulo J. Freitas Filho,b, Sílvia M. Nassarb,Mauro Roisenbergb, Diego G. Rodriguesb, Mariana D. C. Lima, Optimization Models and Prediction of Drilling Rate (ROP) for the Brazilian Pre-Salt Layer, Chemical Engineering transaction , Vol. 33, 2013.

[12] J Ning, F Honghai, Z Yinghu, L Tianyu, A New Model of ROP Prediction for Drilling

Engineering with Data Mining Technology, Advances in information Sciences and Service Sciences(AISS), Vol 5, 2013.

[13] Altmis, U., "Estimation of Drilling Parameters Using Neural Networks." Mémoire de Master ingénierie de pétrole et gaz, West Virginia University, Morgantown, WV, 1996.

[14] Wojtanowicz, A.K., and Kuru, E.: "Minimum-Cost Well Drilling Strategy Using Dynamic Programming", Joumal of Energy Resources Technology, Transactions of the ASME, December 1993.

[15] B. Esamael, Arghab Arnaout, Rudolf K. Fruhwirth et Gerhard Thonhauser, Automated System for Drilling Operations Classification Using Statistical Features, IEEE,2011.

[16] R. Arabjamaloeia et S. Shadizadehb, Modeling and Optimizing Rate of Penetration Using Intelligent Systems in an Iranian Southern Oil Field (Ahwaz Oil Field), Petroleum Science and Technology, Volume 29, Issue 16, 2011.

[17] R. Arabjamaloeia et B. K Dehkordib, Investigation of the Most Efficient Approach of the Prediction of the Rate of Penetration, Energy Sources, Part A: Recovery, Utilization, and Environmental Effects, Volume 34, Issue 7, 2012.

[18] R. Arabjamaloeia, S. Edalatkhahb & E. Jamshidib, A New Approach to Well Trajectory Optimization Based on Rate of Penetration and Wellbore Stability, Petroleum Science and Technology, Volume 29, Issue 6, 2011.

[19] S. Edalatkhaha, R. Rasoula & A. Hashemia, Bit Selection Optimization Using Artificial Intelligence Systems, Petroleum Science and Technology, Volume 28, Issue 18, 2010.

[20] J.P Nguyen, Techniques d'exploitation pétrolière le forage, Editions Téchnip, 1993.

[21] A. Farage, Commande non-linéaire dans les systèmes de forage pétrolier : contribution à la suppression du phénomène de « Stick-Slip », thèse de doctorat, Université de paris XI Orsay, 2006.

[22] Amadou-Abdoulaye BA, contribution à la surveillance d'un processus de forage pétrolier, thèse de doctorat « automatique et traitement de signal », ParisTech –institut des sciences et technologies-, 2010.

[23] R. E. Osgouei, Rate of penetration estimation model for directional and horizontal wells, Thesis Master of sciences in petroleum and natural gaz engineering, Middle east technical university, 2007.

[24] H. Horra, Approche adaptative d'optimisation des paramètres mécaniques de forage,

Mémoire de magister en exploitation des gisements pétroliers, Université M'hamed Bougara- Boumerdes, 2010.

[25] K Amar and A Ibrahim, Rate of penetration prediction and optimization using advances in Artificial Neural Networks, A comparative study, International Joint Conference on Computational Intelligence IJCCI, 2012.

[26] R. Genuer, Forêts aléatoires : aspect théoriques, sélection de variables et applications, Thèse de Doctorat Mathématiques, Université de Paris-Sud XI, 2010.

[27] A. Hammyani et S. Allioua , Amélioration des forêts aléatoires : Application au diagnostic médical, Mémoire de Master II en informatique, Université de Abou Bakr Belkaid Tlemcen, 2013.

[28] S Bernard, Forêts aléatoires : De l'analyse des mécanismes de fonctionnement à la construction dynamique, Thèse de doctorat en informatique, Université de Rouen, 2009.

[29] T.Hastie, R. Tibshirani and J. Friedman, The Elements of Statistical Learning : Data Mining, Inference and Prediction- Second Edition, Springer, 2008.

[30] Y. Brostaux , Etude du classement par forêts aléatoires d'échantillons perturbés à forte structure d'interaction, Thèse de Doctorat agronomique et ingénierie biologique, Université de Gembloux , 2005.

[31] H. Chouaib, Sélection de Caractéristiques : Méthodes et Applications, Thèse de doctorat en informatique, Université Paris Descartes,2011.

[32] J.A Nelder and R. Mead, "A simplex method for function minimization", Comput. J., 7, pp. 308–313, 1965.

[33] Hooke, R. and Jeeves, "Direct Search Solution of Numerical and Statistical Problems", J. ACM 8, pp. 212–229, 1961.

[34] Spendley, W., Hext, G.R., and Himsworth, "Sequential Application of Simplex Designs in Optimisation and Evolutionary Operation", Technometrics, Vol. 4, pp. 441–461, 1962.

[35] Press, W.H., Teukolsky, S.A., Vetterling, W.T., and Flannery, Numerical Recipes in Fortran : The Art of Scientific Computing, (second ed.), Cambridge University Press, Cambridge, New York, 1992.

[36] The MathWorks, "Matlab 2015, Function Reference", Natick, Massachusetts, 2015.

[37] M.H Wright,"Direct Search Methods: Once Scorned, Now Respectable", in Numerical Analysis 1995, Proceedings of the 1995 Dundee Biennial Conference in Numerical

Analysis, D.F. Griffiths and G.A. Watson (Eds.), Addison Wesley Longman, Harlow, UK, pp. 191–208, M.H, 1996.

[38] J.C Lagarias., J.A Reeds., M.H Wright., and P.E Wright,"Convergence Properties of the Nelder-Mead Simplex Method in Low Dimensions", Siam J. Optim. 9, pp. 112–147,1998.

[39] S Singer and S Singer, "Efficient Implementation of the Nelder-Mead Search Algorithm", Appl. Numer. Anal. Comput. Math. 1, No. 3, pp. 524–534, 2004.

[40] Audet, C. and Dennis, "Analysis of Generalized Pattern Searches", SIAM J. Optim. 13, pp. 889–903, , 2003.

[40] J Duan, J Zhao, L Xiao, C Yang, and C Li, "A ROP Optimization Approach Based on Improved BP Neural Network PSO", pp. 11–18, Springer International Publishing Switzerland 2015.

[41] https://fr.wikipedia.org/wiki/Weka_(apprentissage_automatique), date consultation 22/11/2015.

[42] Final well report Hassi terfa,17/01/2012.

[43] http://weka.sourceforge.net/doc.packages/multiLayerPerceptrons/weka/classifiers/functions/MLPClassifier.html, date de Consultation 22/11/2015.

[44] http://weka.sourceforge.net/doc.dev/weka/classifiers/functions/SMOreg.html, date de consultation 22/11/2015.

[45] http://machinelearningmastery.com/design-and-run-your-first-experiment-in-weka/, date de consultation 25/11/2015.

[46] https://en.wikipedia.org/wiki/Cross-validation_(statistics), date de consultation 25/11/2015.

[47] https://weka.wikispaces.com/Optimizing+parameters, date de consultation 26/11/2015.

[48] http://blog.dato.com/how-to-evaluate-machine-learning-models-part-4-hyperparameter-tuning, date de consultation 26/11/2015.

Production scientifique

1. Publications dans des revues internationales

- Nafissa Rezki, Okba Kazar, Leila Hayet Mouss, Laid Kahloul, Djamil Rezki , A novel approach for multivariate process monitoring using several intelligences, International Journal of Industrial and Systems Engineering, ISSN 1748-5045, 2015,(Article sous press).
- Nafissa Rezki, Okba Kazar, Leila Hayet Mouss, Laid Kahloul, Djamil Rezki (in press) .On the use of multi-agent systems for the monitoring of industrial systems, Journal : Journal of Industrial Engineering International, ISSN : 2251-712X, 2015, Vol.11, Springer Berlin Heidelberg, (Article sous press).
- Aitouche Samia, Mouss Mohamed Djamel, Mouss Kinza Nadia, Kaanit Abdelghafour, Boutarfa Youcef and Rezki Djamil, A hybrid method to develop a knowledge management system, eKNOW 2014 : The Sixth International Conference on Information, Process, and Knowledge Management, Copyright (c) IARIA, 2014. ISBN: 978-1-61208-329-2.

2. Publications dans des conférences internationales

- Djamil REZKI, Leila Hayet MOUSS, and Nafissa REZKI . An automatic control of cement production quality. La troisième conférence internationale sur l'ingénierie industrielle et Productique ICIEM'2014 `a l'Université de Batna Algérie, le 11 – 13 mai 2014.
- Nafissa REZKI, Okba KAZAR, Leila Hayet MOUSS, Khadija ABID, and Djamil REZKI. Joint quality and maintenance in manufacturing system. La troisième conférence internationale sur l'ingénierie industrielle et Productique ICIEM'2014 `a l'Université de Batna Algérie, le 11 - 13 mai 2014.
- Aitouche Samia, Mouss Mohammed Djamel, Kanit Abdelghafour, Boutarfa Youcef et Rezki djamil. SKACICM : Méthode de développement de management des connaissances. La troisième conférence internationale sur l'ingénierie industrielle et Productique ICIEM'2014 `a l'Université de Batna Algérie, le 11 - 13 mai 2014.
- Aitouche Samia, Mouss Mohammed Djamel, Kanit Abdelghafour, Boutarfa Youcef et Rezki Djamil. Proposition d'un système de e-management des connaissances. La troisième conférence internationale sur l'ingénierie industrielle et Productique ICIEM'2014 `a l'Université de Batna Algérie, le 11 - 13 mai 2014.