PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA
Ministry of Higher Education and Scientific Research

Serial N°: ……. / 2024

Kasdi Merbah Ouargla University

Faculty of Hydrocarbons, Renewable Energies and Sciences of Earth and the Universe

Production Department

Dissertation
To obtain the Master's degree
Option: Professional Production

Submitted by:

Salah Eddine KAIFAS

-THEME-

Reservoir Oil Rate Forecasting Using Machine Learning Regression Algorithms

Defended on: 06/04/2024

Jury:

President:	**GHALI Ahmed**	**MAA**	Univ. Ouargla
Supervisor:	**BAZZINE Zineb**	**MCB**	Univ. Ouargla
Examiner:	**ALI ZERROUKI Ahmed**	**Prof**	Univ. Ouargla

Academic Year: 2023/2024

Dedication

To my dearest family and friends,

This dissertation is dedicated to you.

To my parents, Dad Abdelkader and Mom Rahma, your unwavering love and support have been the foundation upon which I built this dissertation. Your constant encouragement and belief in me have fueled my determination throughout this journey.

To my family, your love and understanding have provided a constant source of strength. Thank you for always being there for me, celebrating my successes and offering support during challenging times.

To my friends, your friendship has been a source of joy and inspiration. Your willingness to listen and offer encouragement has been invaluable.

This work is a testament to the collective support of all of you. Thank you for being a part of my life.

- ***Salah Eddine***

Acknowledgements

This dissertation would not have been possible without the invaluable support and guidance of several individuals. I would like to express my sincere gratitude to:

Mentor Zineb Bazzine: Your unwavering support and encouragement throughout this journey have been a constant source of motivation. I am grateful for your patience, feedback, and dedication to my success.

Engineers Brahim, Abdelbaset, and Adel: Thank you for your technical expertise and willingness to share your knowledge throughout this project. Your insights and guidance were instrumental in shaping my understanding of the field and overcoming technical challenges.

I am truly fortunate to have had such a supportive network during this endeavor.

Abstract

Accurate prediction of oil rates is critical for optimizing production and reservoir management. This dissertation investigates the effectiveness of machine learning (ML) regression algorithms techniques as an alternative to traditional numerical simulation methods in oil rate prediction. The work focuses on pre-processing and cleaning well data, including outlier removal, handling missing values, and label encoding for well names. Feature selection utilizes correlation analysis to identify relevant features impacting oil rates, such as Gas-Oil Ratio (GOR), water cut, choke size, and wellhead pressure. Five regression algorithms – Linear Regression, Random Forest Regressor, Gradient Boosting Regressor, Support Vector Regressor (SVR), and XGBoost – are evaluated for their performance in predicting oil rate by using different Python libraries. XGBoost emerges as the best-performing model with the lowest Mean Squared Error (MSE). Random Forest Regressor and Gradient Boosting Regressor also demonstrate promising results. Linear Regression and SVR exhibit significantly higher MSE, indicating lower accuracy. The dissertation showcases the effectiveness of XGBoost in predicting oil rates with high accuracy (over 90% for sample wells). This approach offers a valuable alternative to traditional numerical simulation methods, potentially leading to improved production forecasting and reservoir management strategies.

Keywords: Artificial intelligence, Machine learning, Oil rate prediction, Regression algorithm, Python, XGBoost algorithm.

ملخص

يعد التنبأ الدقيق لمعدل تدفق النفط أمر بالغ الأهمية في تحسين الانتاج و إدارة المكمن البترولي, تهدف هذه الدراسة الى التطرق لمختلف خوارزميات التراجع لتعلم الآلة و مدى فعاليتها مقارنة بالطرق التقليدية للمحاكاة الرقمية لمعدل تدفق البترول, تركز الدراسة على معالجة بيانات الآبار، بما في ذلك إزالة البيانات الغير مألوفة ومعالجة القيم الغير مكتملة و الترميز بالارقام لأسماء الآبار. كذلك تم دراسة مقدار إرتباط مختلف الميزات مع بعضها البعض لتحديد الأكثر تأثيرا على معدل تدفق النفط، مثل نسبة الغاز إلى النفط (GOR), نسبة الماء في المائع ,نصف قطر الصمام الخانق و ضغط رأس البئر. تم دراسة أداء خمسة خوارزميات تراجع - النموذج الخطي، الغابة العشوائية، النزول الإشتقاقي المدعم ,المتجهات الداعمة (SVR) ، و نموذج XGBoost - في التنبؤ بمعدل تدفق النفط بإستعمال مختلف مكتبات لغة البرمجة Python. يتبين أن XGBoost أفضل خوارزمية بأقل خطأ تربيعي متوسط(MSE) . كما تظهر خوارزميات الغابة العشوائية و النزول الإشتقاقي المدعم نتائج جيدة. تظهر خوارزمية التراجع الخطي و ,المتجهات الداعمة خطأ تربيعي متوسط أعلى بكثير، مما يشير إلى دقة أقل. يعرض البحث مدى فعالية خوارزميات تعلم الآلة و خاصة XGBoost في التنبؤ بمعدل تدفق النفط بدقة عالية (أكثر من 90٪ لآبار العينة). تقدم هذه النتائج

حلا فعالا مقارنة بأساليب التقليدية للمحاكاة الرقمية للمكامن البترولية، مما يؤدي إلى تحسين تنبؤات الإنتاج واستراتيجيات إدارة مكامن البترول.

الكلمات المفتاحية: الذكاء الصناعي، تعلم الآلة, التنبؤ بمعدل تدفق النفط, خوارزميات التراجع, Python, XGBoost.

Résumé

La prédiction avec précision du débit de pétrole est essentielle pour l'optimisation de la production et la gestion des réservoirs. Cette memoir étudie l'efficacité des techniques d'apprentissage automatique (ML) comme alternative aux méthodes de simulation numérique traditionnelles dans la prédiction du débit de pétrole. Les travaux se concentrent sur le prétraitement et le nettoyage des données de puits, y compris la suppression des valeurs aberrantes, la gestion des valeurs manquantes et l'encodage par étiquette des noms de puits. La sélection des caractéristiques utilise l'analyse de corrélation pour identifier les facteurs pertinents affectant le débit de pétrole, tels que le rapport gaz-pétrole (GOR), le pourcentage d'eau, le diamètre de la Duse et la pression de tête de puits. Cinq algorithmes de régression - régression linéaire, régression Random Forest, régression par gradient boosting, régression à vecteurs de support (SVR) et XGBoost - sont évalués pour leurs performances en matière de prédiction du débit de pétrole. XGBoost s'impose comme le modèle le plus performant avec la plus faible erreur quadratique moyenne (MSE). La régression Random Forest et la régression par gradient boosting montrent également des résultats prometteurs. La régression linéaire et le SVR présentent une erreur quadratique moyenne (MSE) significativement plus élevée, indiquant une précision moindre. La recherchef met en évidence l'efficacité de XGBoost dans la prédiction du débit de pétrole avec une grande précision (plus de 90 % pour des puits échantillons). Cette approche offre une alternative précieuse aux méthodes de simulation numérique traditionnelles, ce qui pourrait conduire à une amélioration des prévisions de production et des stratégies de gestion des réservoirs.

Mots-clés : Intelligence artificielle, Apprentissage automatique, prédiction du débit de pétrole, Algorithme de régression, Python, XGBoost algorithme.

Table of contents

Chapter I: Basics of Oil Reservoir Simulation and Optimization

Chapter II: Data-Driven Technologies: Machine learning

Chapter III: Practical Part

List of Figures

List of Tables

List of Abbreviations

HMD	Hassi Messaoud
SN. REPAL	National Petroleum Research and Exploitation Company
C.F.P.A	Compagnie Française des Petroles Algeria
MBE	Material Balance Equation
OWC	Oil Water Contact
OGC	Oil Gas Contact
GOR	Gas Oil Ratio
WC	Water Cut
IMPES	Implicit Pressure, Explicit Saturation
MER	Maximum efficient rate
HP	High pressure
HT	High Temperature
ML	Machine Learning
ANNs	Artificial Neural Networks
AI	Artificial Intelligence
NaN	Not a Number
MSE	Mean Squared Error
XAI	Explainable Artificial Intelligence

Nomenclatures

N	Initial reservoir oil	STB
Boi	Initial oil formation volume factor	bbl/STB
Np	Cumulative produced oil	STB
Bo	Oil formation volume factor	bbl/STB
G	Initial reservoir gas	SCF
Bgi	Initial gas formation volume factor	bbl/SCF
Gf	Amount of free gas in the reservoir	SCF
Rsoi	Initial solution gas-oil ratio	SCF/STB
Rp	Cumulative produced gas-oil ratio	SCF/STB
Rso	Solution gas-oil ratio	SCF/STB
Bg	Gas formation volume factor	bbl/SCF
W	Initial reservoir water	bbl
Wp	Cumulative produced water	STB
Bw	Water formation volume factor	bbl/STB
We	Water influx into reservoir	bbl
C	Total isothermal compressibility	psi–1
Δp	Change in average reservoir pressure	psia
Swi	Initial water saturation	

Vf	Initial pore volume	bbl
cf	Formation isothermal compressibility	psi–1
I	Injectivity	bbl/day/psi
iw	Injection rate	bbl/day
ΔP	Difference between injection pressure and producing well bottom hole flowing pressure	psi
ibase	Initial (base) water injection rate	bbl/day
Δpbase	Initial (base) pressure difference between injector and producer	psi
ibase	Base (initial) water injection rate	bbl/day
h	Net thickness	ft
k	Absolute permeability	md
Δpbase	Base (initial) pressure difference	psi
d	Distance between injector and producer	ft
rw	Wellbore radius	ft
kro	Oil relative permeability as evaluated at Swi	
θ	Contact angle of liquid-solid interface	degrees
M	Mobility	
xj	Input nodes	
wij	Weights from the input layer	

W	Weight matrix
uj	Biases

General Introduction

General Introduction

Decision-making in the oil and gas industry relies heavily on accurate models of reservoirs. These models are crucial for estimating reserves, predicting future production, and optimizing it. However, building reliable reservoir models is a complex task due to the uniqueness and complexity of each reservoir.

Traditionally, reservoir modeling has involved a step-by-step process of constructing a geological model and then simulating fluid flow through numerical methods. While this approach has yielded successes, it also has limitations. Complexities like incorporating vast amounts of data and performing extensive simulations can lead to models with large computational footprints. These computationally expensive models can become impractical for tasks like sensitivity analysis and optimization, hindering effective reservoir management.

The limitations of traditional modeling methods necessitate a new approach. This dissertation explores how machine learning can be applied to bypass traditional modeling steps and forecast production outcomes, potentially leading to a more efficient and data-driven approach in the field.

To achieve the overall goal of this dissertation, the research was divided into three distinct chapters:

Chapter 1 dives into traditional reservoir modeling techniques. While these methods have proven successful, they also have limitations, particularly the computational issue associated with large reservoirs.

Chapter 2 examine machine learning and its various algorithms principles, and its potential for production prediction. The chapter emphasizes the ability of machine learning to handle vast datasets and find patterns.

Chapter 3 translates theory into practice by applying machine learning techniques to the Hassi Messaoud reservoir data. This section starts by introducing Hassi Messaoud field and then delves into the details of the data preparation steps, the selection of suitable algorithms, and the training and evaluation of the prediction models. By analyzing the models' performance in predicting production outcomes.

Chapter I: Basics of Oil Reservoir Simulation and Optimization

I.1 Introduction

Within the oil and gas industry, reservoir simulation has become the established methodology for addressing reservoir engineering challenges. This technique utilizes a combination of fundamental physical principles, mathematical formulations specific to reservoir behavior, and advanced computational programming to construct digital models of subsurface hydrocarbon reservoirs. These models possess the capability to predict the performance of the reservoir under a variety of production scenarios. This predictive ability empowers engineers to make informed decisions regarding production strategies aimed at optimizing hydrocarbon recovery.(Abou-Kassem et al., 2006)

I.2 Basics of Reservoir Simulation

A reliable reservoir model is built on two core ideas:

1. The Integration of all available data, including detailed well data.
2. History matching to the past performance before predictions about future behavior can be made.

The model should incorporate all available reservoir information, with a particular focus on detailed well data. This ensures a complete picture of the subsurface and its characteristics. Furthermore, the model's ability to predict future behavior hinges on its accuracy in replicating past reservoir performance, which is crucial for validating the model's predictive capabilities. (Shahab D. Mohaghegh, 2017)

Although simulation in the petroleum industry is not new, the new aspects are that more detailed reservoir features, and thus more accurate simulations, have become practical because of the capability afforded by the computers now available.

I.3 Reservoir Models

There are three types of models:

1. Material balance models
2. Numerical models
3. Petrophysical models

I.3.1 Material Balance (MBE)

Fluid production from a hydrocarbon reservoir doesn't leave a void space behind. As pressure drops during this process, the remaining fluids and/or rock expand, or nearby water fills the void. The volume of produced oil helps engineers determine the extent of this expansion or encroachment.

Material balance is a method that can be used to account for the movement of reservoir fluids within the reservoir or to the surface where they are produced. The material balance accounts for the fluid produced from the reservoir through expansion of existing fluid, expansion of the rock, or the migration of water into the reservoir. The material balance equation includes factors that compare the various compressibility of fluids, consider the gas saturated in the liquid phase, and include the water that may enter into the hydrocarbon reservoir from a connected aquifer.(Terry et al., 2015)

Figure I-1 shows a typical oil reservoir with three distinct zones: an oil zone in the center, an aquifer below, and a gas cap above. During oil production, the pressure within the reservoir will decline. This pressure decline has two key consequences:

- Water influx: Water from the underlying aquifer will migrate into the oil zone, displacing the oil and altering the original oil-water contact (OWC) to a new position.
- Gas cap expansion: As pressure decreases, gas previously dissolved in the oil will come out of solution and migrate upwards, expanding the gas cap and pushing the original oil-gas contact (OGC) to a new location.

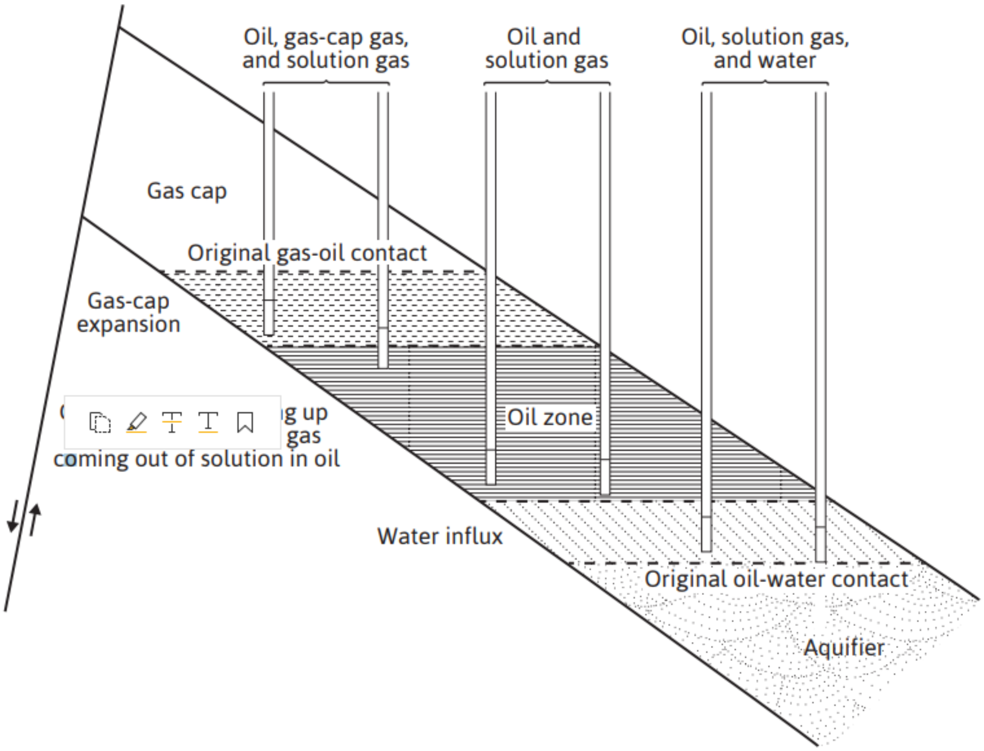

Oil, gas-cap gas, and solution gas
Oil and solution gas
Oil, solution gas, and water
Gas cap
Original gas-oil contact
Gas-cap expansion
g up
gas
coming out of solution in oil
Oil zone
Water influx
Original oil-water contact
Aquifier

Figure I-1 : Cross section of a combination drive reservoir (Terry et al., 2015).

The concept the MBE was presented by Schilthuis in 1936 and is simply based on the principle of the volumetric balance. It states that the cumulative withdrawal of reservoir fluids is equal to the combined effects of fluid expansion, pore volume compaction, and water influx. In its simplest form, the equation can be written on a volumetric basis as:(Ahmed & McKinney, 2011)

Initial volume = volume remaining + volume removed

Since oil, gas, and water are present in petroleum reservoirs, the MBE can be expressed for the total fluids or for any one of the fluids present. There are Three different forms of the which are:

- Generalized MBE.
- MBE as an equation of a straight line.
- Tracy’s form of the MBE.

For simplicity purpose, we will discuss the general volumetric material balance equation only which is presented by: (Terry et al., 2015)

Oil expansion + Gas expansion + Formation and water expansion + Water influx = Oil and gas production + Water production

Which can be written mathematically by the equation I-1):

$$N(B_t - B_{ti}) + \frac{NmB_{ti}}{B_{gi}}\left(B_g - B_{gi}\right) + (1+m)NB_{ti}\left[\frac{c_w S_{wi} + c_f}{1 - S_{wi}}\right]\Delta\bar{p} + W_e = N_p\left[B_t + \left(R_p - R_{soi}\right)B_g\right] + B_w W_p \qquad \text{I-1}$$

Where:

N Initial reservoir oil, STB

B $_{oi}$ Initial oil formation volume factor, bbl/STB

N $_p$ Cumulative produced oil, STB

B $_o$ Oil formation volume factor, bbl/STB

G Initial reservoir gas, SCF

B $_{gi}$ Initial gas formation volume factor, bbl/SCF

Gf Amount of free gas in the reservoir, SCF

R $_{soi}$ Initial solution gas-oil ratio, SCF/STB

R $_p$ Cumulative produced gas-oil ratio, SCF/STB

R $_{so}$ Solution gas-oil ratio, SCF/STB

B $_g$ Gas formation volume factor, bbl/SCF

W Initial reservoir water, bbl

W $_p$ Cumulative produced water, STB

B $_w$ Water formation volume factor, bbl/STB

W $_e$ Water influx into reservoir, bbl

c Total isothermal compressibility, psi–1

Δ*p* Change in average reservoir pressure, psia

S $_{wi}$ Initial water saturation

V $_f$ Initial pore volume, bbl

c_f Formation isothermal compressibility, psi–1

These material balances are mainly used for preliminary rough calculations before going on to more advanced models, particularly when a new reservoir is discovered. They are run on microcomputers.(Cossé, 1993)

I.3.2 Numerical Modeling

Early reservoir simulation models, like the "Black Oil" models developed decades ago, were designed for conventional oil production and relied on fundamental equations like mass conservation, Darcy's Law, and thermodynamics and doesn’t account temperature variations.

While efficient for pressure calculations, the IMPES (implicit pressure, explicit saturation) method used in these models introduced numerical dispersion issues in saturation calculations. Sometimes addressed by modifying permeability curves (pseudo permeabilities), these early programs had few lines of code laid the groundwork for the field. However, with increasing computing power, reservoir simulation has evolved beyond Black Oil models, embracing more complex approaches and advanced programming techniques.

Reservoir simulations use a variable number of blocks (from dozens to thousands) and time steps (ranging from fractions of a day to months) depending on the complexity of the reservoir. A complex reservoir, with varying geological features and potentially three fluid phases (gas, oil, water), requires a more intricate model compared to a simpler one.

The key is to find a balance between the model's detail and the computational cost. Running a complex model with many blocks and small-time steps provides more accurate results, but takes longer, depending on the specific scenario, engineers can choose from simpler models (e.g., two fluids, one-dimensional flow) or more complex ones (e.g., three fluids, three-dimensional flow) as shown in Figure I-2

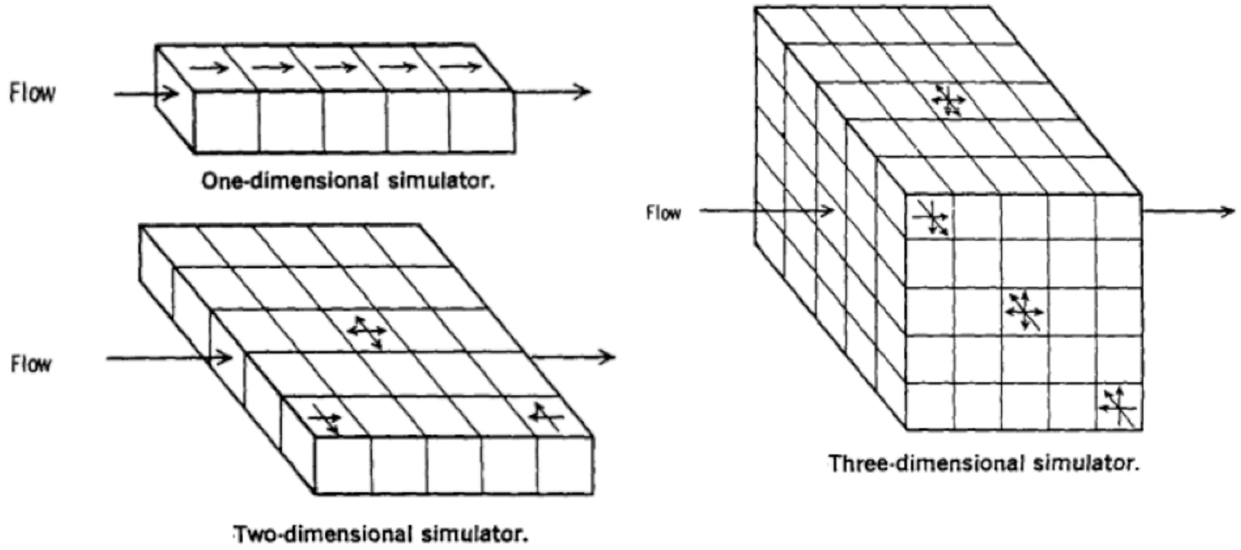

Flow
One-dimensional simulator.
Flow
Two-dimensional simulator.
Flow
Three-dimensional simulator.

Figure I-2 : Different types of simulators dimensions (Odeh, 1969)

Reservoir simulation models not only predict future production (forecasts) but also help engineers understand the reservoir itself. Through history matching, engineers can adjust unknown parameters to match the model's predictions with past production data. This dual functionality allows them to not only forecast production but also conduct specialized studies (like analyzing fluid segregation or coning) to gain a deeper understanding of the reservoir's behavior. Specialized modeling

Other more specialized models have been developed to deal with specific cases. These include the following:

I.3.3 Compositional and miscible models

Compositional models take a more detailed approach, representing oil and gas with multiple components each. This allows for a more accurate representation of how the fluids change in the reservoir, particularly for volatile oils, condensate gases, and situations involving gas injection. The miscible model offers a better representation of the flows of injected fluids that are miscible with the fluid in place, see Figure I-3 .

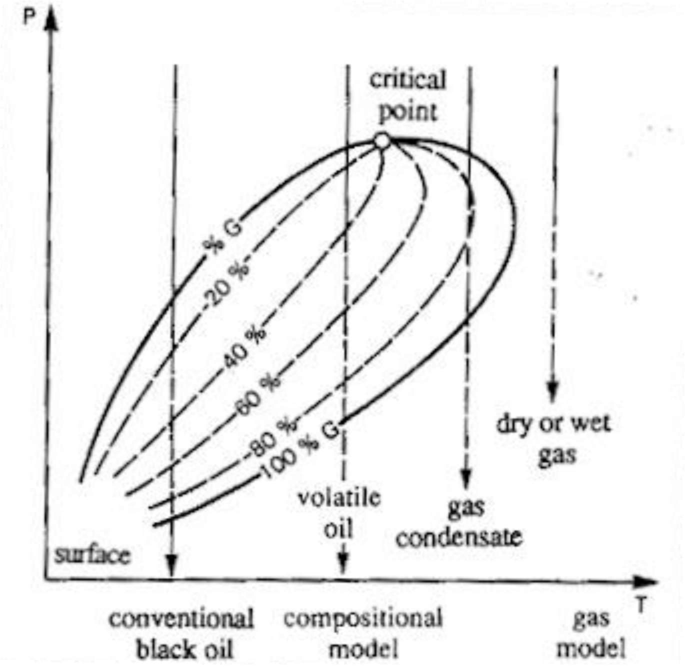


Figure I-3 : Compositional model (Cossé, 1993).

I.3.3.1 Chemical Models

These allow the simulation of the injection of polymers and surfactants in Particular.

I.3.3.2 Thermal Models

Steam injection and in situ combustion models exist, added to the equations already discussed is a temperature equation. The vaporization of water and (possibly) of oil must also be accounted for.

I.3.3.3 Fractured Models

The representation of the medium is more complex. One approach uses a dual porosity system, where separate "blocks" represent the pressure and porosity within the fractures and the rock matrix. Equations account for fluid flow between these regions based on pressure differences. While this is the most challenging reservoir type to model, such dual porosity models are actively used alongside ongoing research to improve their accuracy in capturing the complex physics of flow in fractured rocks.(Cossé, 1993)

Modeling fractured reservoirs is a complex task due to two main challenges. First, the inherent complexity of the fracture network (Figure I-4) makes it difficult to accurately represent its characteristics in a model. Perhaps a more significant challenge lies in our incomplete understanding of the driving mechanisms that govern fluid flow between the rock matrix and the fractures. These limitations necessitate ongoing research to improve our ability to capture the intricate flow dynamics within fractured formations.

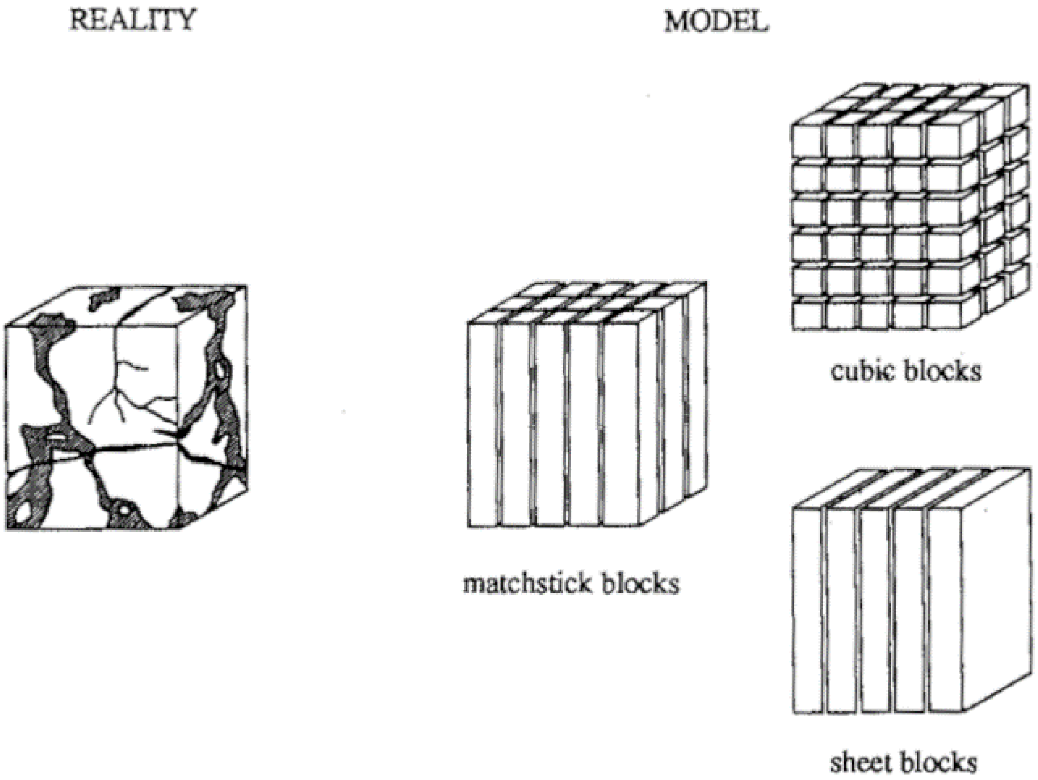


Figure I-4 : Modeling of fractured environment (Cossé, 1993).

I.3.3.4 Petrophysical models

While numerical reservoir simulation models are powerful tools, they are not the only way to investigate reservoir behavior. Petrophysical models offer a complementary approach. These are physical scale models built in the laboratory using materials that mimic the rock properties of the actual reservoir. They are often saturated with fluids to represent oil, water, and gas.

Petrophysical models are particularly useful for studying specific flow mechanisms in detail, such as the distortion of fluid fronts (where the boundary between different fluids becomes uneven) or fingering (where one fluid preferentially channels through the rock, bypassing others). These models provide valuable insights that can be difficult to capture with purely numerical simulations. Historically, electrical models were also used for reservoir studies. These models employed resistors and capacitors to represent the flow properties of the reservoir rock and fluids. While not as widely used today as numerical models, electrical models were valuable tools for certain specific problems encountered in the past. (Terry et al., 2015)

By combining numerical simulations with physical and electrical analogs, reservoir engineers gain a more comprehensive understanding of fluid flow behavior within the reservoir, leading to better informed decision-making for oil and gas production.

I.4 Production enhancement strategies through choke size and water injection

I.4.1 The Selection of the Optimum Choke Management Strategy

When wells are drilled and completed the size of the production tubing remains constant. Consequently, the rate at which hydrocarbons are produced requires control at the surface. This control is achieved through a surface choke, a valve specifically designed to regulate flow rate and pressure of produced fluids at the wellhead.

While chokes serve various purposes downhole and at the surface, their primary function in production optimization revolves around managing flow characteristics.

There are different types of chokes, Fixed choke (positive choke), Needle and seat choke, Plug and cage choke (or adjustable choke), however the most commonly used in the oil field are the adjustable and fixed choke.

1- **Positive chokes (**Figure I-5**):** a removable flow bean with a circular orifice of fixed dimensions. The dimension is usually specified in diametric increments of 1/64 inch.

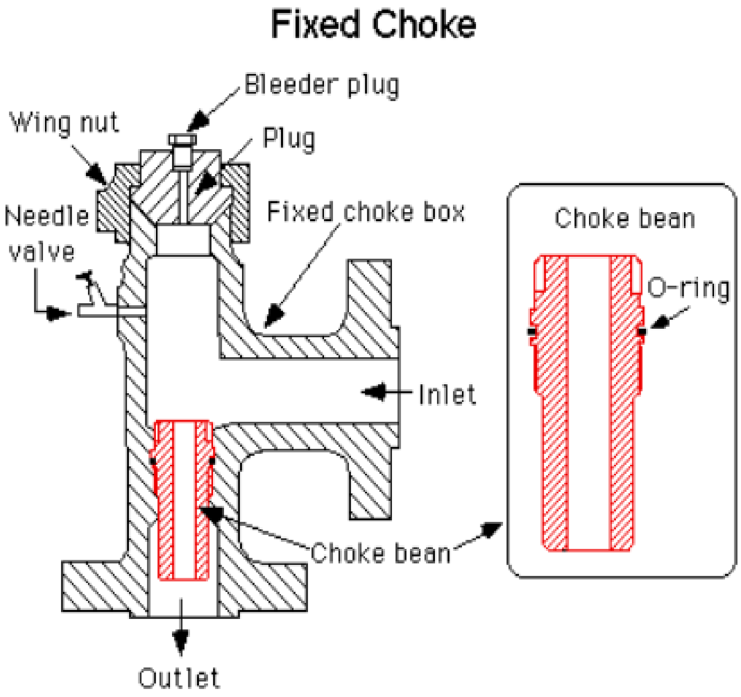


Figure I-5 A Fixed Choke (Scott, 2008).

2- **Adjustable choke (**Figure I-6**)**: it allows the size of the orifices that fluids flow through to be changed has an externally controlled variable orifice and a visible area indication mechanism. This mechanism called the barrel or stem is calibrated in terms of diameters of equivalent circular orifices in increments of 1/64 inch. The adjustable choke.

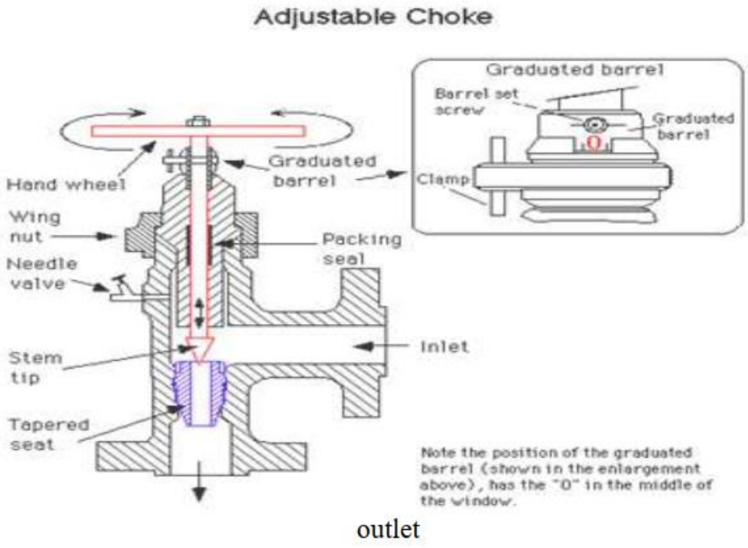


Figure I-6 Adjustable choke (Scott, 2008).

Field experience has established the existence of a "maximum efficient rate" (MER) for each well. This MER represents the optimal production rate that balances hydrocarbon recovery with wellbore and reservoir sustainability. Exceeding this rate can lead to adverse consequences, including sand production. Sand influx, as noted by (Scott, 2008) ,occurs due to the pressure differential between the producing formation and the wellbore. Excessive sand production can damage both downhole and surface equipment, significantly impacting well productivity.

(Arnold & Stewart, 1998)emphasizes the importance of controlling flow rate through the wellhead, particularly for naturally flowing wells (without artificial lift). Precise choke sizing is crucial to achieve optimal production while minimizing formation damage. Uncontrolled flow rates can also induce gas and/or water coning, further hindering hydrocarbon recovery. Additionally, excessive pressure associated with high flow rates can lead to tubing or casing collapse.

The surface choke serves as a vital tool for mitigating these production risks. By enabling controlled flow through careful choke size selection and management, operators can achieve a sustainable and optimized production strategy, maximizing hydrocarbon recovery while ensuring wellbore integrity and reservoir health.

Choke size management methods vary from purely empirical to analytical models and sophisticated numerical schemes. However, since the parameters associated with numerical modeling are not routinely measured, major operating companies typically deploy analytical models that are based on either shear or tensile failure criteria. Such analytical models typically capture a single failure mechanism and assume that formation or completion failure is concomitant with the onset of sand production. (Hans et al., 2002)showed that analytical models generally provide a high level of conservatism in predicting the maximum allowable drawdown, especially in HP/HT wells. Additionally, massive reservoir depletion and/or water-breakthrough limit the applicability and reliability of analytical methods (Nouri et al., 2004).

I.4.2 Water injection strategy

Water injection stands as the most widely employed secondary oil recovery technique. It involves strategically injecting water into an oil reservoir to displace and propel trapped oil towards production wells. This displacement occurs at the microscopic level within the reservoir's pore spaces.

The effectiveness of waterflooding, as water injection is often called, is heavily influenced by factors like oil viscosity and the interplay between rock and fluid properties. To estimate the ultimate recovery factor (the percentage of original oil in place that can be ultimately recovered), engineers quantify water-oil flow measurements. These measurements consider the effects of geological formations, mineral composition, gravitational forces, and well spacing within the reservoir.(Ahmed & McKinney, 2011)

I.4.2.1 Factors affecting water injection

A. Viscosity

In most oil reservoirs, water viscosity at operating temperatures is either significantly lower than or equal to that of the oil. This translates to a water-oil viscosity ratio typically less than 1. This ratio plays a crucial role in the efficiency of water-oil displacement, with a lower ratio indicating a more favorable displacement process. Fluid mobility in a porous medium is another key factor. As per the definition, a low viscosity (≤ 1 cP) signifies a highly mobile fluid, assuming relative permeability isn’t extremely low. Conversely, a low-API crude oil (≤ 20°API) indicates a highly viscous oil with low mobility, except at elevated temperatures where viscosity can decrease. (Abou-Kassem et al., 2006)

B. Reservoir Rock

To further understand the influence of reservoir rock (lithology) on incremental oil recovery (the additional oil recovered due to water injection), it's essential to examine fluid-rock interactions. These interactions play a significant role in wettability alteration, residual oil saturation to water injection, and relative oil permeability when water saturation is high.

C. Wettability

Wettability refers to the tendency of an immiscible liquid (like oil or water) to spread and adhere to a solid surface in the presence of another immiscible liquid (Ahmed, 2018). This propensity is quantified by measuring the contact angle (θ) at the liquid-solid interface. A contact angle of 0° indicates complete wettability by the measured liquid, while 180° signifies complete non-wettability Click or tap here to enter text..

I.4.2.2 Water injection problem

Many oilfields, experience a high water cut stage during their later stages of development. This stage presents two key challenges: a rapid increase in water cut and a steep decline in oil production rate.

Injection rate is a key economic variable that must be considered when evaluating a waterflooding project. The waterflood project's life and, consequently, the economic benefits will be directly affected by the rate at which fluid can be injected and produced. Estimating the injection rate is also important for the proper sizing of injection equipment and pumps. (Ahmed, 2010)Although injectivity can be best determined from small-scale pilot floods, empirical methods for estimating water injectivity for regular pattern floods have been proposed by Muskat (1948) and Deppe (1961).

based on the following assumptions:

- Steady-state conditions
- No initial gas saturation
- Mobility ratio of unity

Water injectivity is defined as the ratio of the water injection to the pressure difference between the injector and producer in the equation I-2):

$$I = \frac{i_w}{\Delta P} \quad \text{I-2}$$

Where:

- **I** injectivity, bbl/day/psi
- $\mathbf{i_w}$ injection rate, bbl/day
- **ΔP** difference between injection pressure and producing well bottom hole flowing pressure.

When the injection fluid has the same mobility as the reservoir oil (mobility ratio M = 1), the initial injectivity at the start of the flood is referred to as Ibase in the equation I-3:

$$I_{base} = \frac{i_{base}}{\Delta P_{base}} \qquad \text{I-3}$$

Where:

- $\mathbf{i_{base}}$ = initial (base) water injection rate, bbl/day
- $\mathbf{\Delta P_{base}}$ = initial (base) pressure difference between injector and producer

For a five-spot pattern that is completely filled with oil, i.e., $S_{gi} = 0$, Muskat (1948) proposed the following injectivity equation I-4:

$$I_{base} = \frac{0.003541hkk_{ro}\Delta P_{base}}{\mu_o \ln\frac{d}{r_w} - 0.619} \qquad \text{I-4}$$

Therefore:

$$\frac{i}{\Delta P_{base}} = \frac{0.003541hkk_{ro}}{\mu_0 \ln\frac{d}{r_w} - 0.619} \qquad \text{I-5}$$

Where:

- $\mathbf{i_{base}}$ = base (initial) water injection rate, bbl/day
- **h** = net thickness, ft
- **k** = absolute permeability, md
- $\mathbf{k_{ro}}$ = oil relative permeability as evaluated at Swi
- $\mathbf{\Delta P_{base}}$ = base (initial) pressure difference, psi

- **d** = distance between injector and producer, ft
- **r_w** = wellbore radius, ft

Several studies have been conducted to determine the fluid injectivity at mobility ratios other than unity. All of the studies concluded the following:

- At favorable mobility ratios, i.e., $M < 1$, the fluid injectivity declines as the areal sweep efficiency increases.
- At unfavorable mobility ratios, i.e., $M > 1$, the fluid injectivity increases with increasing areal sweep efficiency.

I.5 Conclusion

Reservoir simulation plays a crucial role in the oil and gas industry by combining physical principles, mathematical models, and computational techniques to create detailed digital representations of subsurface hydrocarbon reservoirs. These simulations are essential for predicting reservoir performance under various production scenarios and optimizing hydrocarbon recovery.

Building a reliable reservoir model involves integrating comprehensive data sets, particularly detailed well data, and matching the model to historical reservoir performance. This process ensures the accuracy of the models and their ability to make reliable future predictions, which is vital for effective reservoir management.

Chapter II: Data-Driven Technologies: Machine learning

II.1 Introduction

Reservoir engineering traditionally relies on physical principles and theoretical models to predict fluid flow behavior within subsurface reservoirs. While these established methodologies provide a strong foundation, data-driven technologies offer synergistic capabilities. These techniques prioritize real-world data (facts), alongside theoretical understanding, to create a more comprehensive picture of the reservoir.(Shahab D. Mohaghegh, 2017)

In the context of reservoir modeling, field measurements serve as the data foundation, representing real observations of the reservoir and fluid flow characteristics. However, it's crucial to acknowledge that these measurements may contain inherent noise. Fortunately, data-driven methodologies incorporate techniques to account for such noise, allowing robust utilization of the gathered information.

II.2 Data mining

Data mining can be defined as the exploration and analysis of large quantities of data in order to discover meaningful patterns and rules(Berry & Linoff, 2008). Data Mining is used to discover knowledge.

Oil and gas companies used to analyze information based on educated guesses. For instance, an engineer might check well depth against pressure, assuming deeper wells meant more pressure. Data mining is different. It's like a powerful searchlight that digs into all the information collected during oil exploration and production, without needing any initial ideas. It can automatically find the most important things that affect a specific outcome. This might uncover surprising connections, like seismic data helping drill wells faster, or sensor readings predicting equipment breakdowns before they happen. By finding these hidden clues, data mining helps oil and gas companies make smarter decisions based on real information.(Moitra et al., 2021)

II.2.1 Data Minning Process

1. Data Preparation: The first step involves collecting and cleaning the data. This ensures the information is accurate, consistent, and ready for analysis.

2. Data Exploration: Data mining algorithms then delve into the prepared data, identifying interesting trends, outliers, and potential relationships between different data points.

3. Pattern Recognition: Sophisticated algorithms analyze the data to uncover hidden patterns and relationships that might not be obvious through traditional methods.

4. Model Building: Based on the discovered patterns, data mining can build models to predict future outcomes or identify areas for improvement.

5. Evaluation and Deployment: The effectiveness of the models is evaluated, and if successful, they are deployed to guide decision-making within the oil and gas company.

II.3 Artificial intelligence

Artificial intelligence is a suite of innovative analytical methodologies striving to mimic life. These AI techniques demonstrate the capacity to learn and adapt to novel circumstances (Zurada et al., 1994). Key technologies falling under the umbrella of artificial intelligence include artificial neural networks, evolutionary programming, and fuzzy logic, all of which exhibit various reasoning attributes like generalization, discovery, association, and abstraction (Eberhart et al., 1996).

Over the past decade, artificial intelligence has evolved into a refined arsenal of analytical tools enabling the resolution of previously daunting or insoluble problems. Presently, these AI tools are effectively employed, often integrated with traditional statistical analysis methods, to construct intricate systems capable of tackling complex challenges.

The widespread utilization of these tools' spans across diverse domains, permeating into commercial applications. Artificial intelligence finds application in various sectors, including medical diagnosis, credit card fraud detection, bank loan approval, smart home devices, public transportation systems, automotive technologies like automatic transmissions and driverless cars, financial management, robot navigation, among others. In industries such as oil and gas, artificial intelligence aids in addressing issues pertaining to pressure transient analysis, well log interpretation, reservoir characterization, and the identification of suitable well candidates for stimulation, among myriad other tasks.(Shahab D. Mohaghegh, 2017)

II.4 Regression in Machine learning

Regression is a statistical method used to examine the relationship between a dependent variable (target) and one or more independent variables. The goal is to identify the optimal function that describes this relationship. This model can then be used for making predictions or drawing inferences.

II.4.1 Regression algorithms

II.4.1.1 Linear regression

The linear regression algorithm demonstrates a linear relationship between a dependent variable (y) and one or more independent variables (x), which is why it is termed linear regression. This algorithm identifies how the value of the dependent variable changes in response to changes in the independent variable (Montgomery et al., 2012).

The linear regression model yields a sloped straight line that represents this relationship between the variables as shown in Figure II-1:

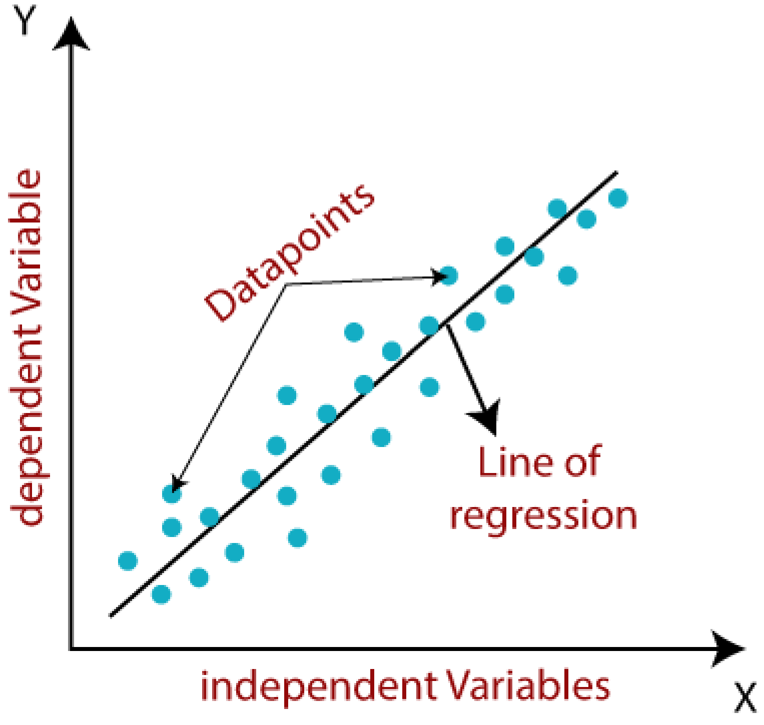


Figure II-1 Linear regression concept (Montgomery et al., 2012)

Mathematically, linear regression can be represented as:

$$y = a_0 + a_1 x \ + \ \varepsilon \qquad \text{II-1}$$

Where:

Y: Dependent Variable (Target Variable)

X: Independent Variable (predictor Variable)

a0: intercept of the line (Gives an additional degree of freedom)

a1: Linear regression coefficient (scale factor to each input value).

ε : random error

II.4.1.1.1 Determining the Optimal Fit Line

In linear regression, the primary goal is to find the best fit line, which minimizes the error between predicted values and actual values. The best fit line is characterized by having the least error.

Different values for the weights or coefficients of the line (a0, a1) result in different regression lines. To determine the best values for a0 and a1, we use a cost function.

The cost function estimates the values of the coefficients that produce the best fit line. It optimizes the regression coefficients or weights by measuring the performance of the linear regression model. The cost function helps determine the accuracy of the mapping function, which maps input variables to the output variable, also known as the hypothesis function.(Montgomery et al., 2012)

For linear regression, we use the Mean Squared Error (MSE) cost function, which is the average of the squared errors between the predicted values and actual values. It is calculated as shown in equation II-2:

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N} \left(Y_i - (a_1 x_i + a_0)\right)^2 \qquad \text{II-2}$$

Where:

N: Total number of observations

Yi: Actual value

($a_1x_i+a_0$): Predicted value.

A linear regression model can be trained using the optimization algorithm gradient descent by iteratively modifying the model's parameters to reduce the mean squared error (MSE) of the model on a training dataset. To update a1 and a2 values in order to reduce the Cost function (minimizing RMSE value) and achieve the best-fit line the model uses Gradient Descent. The idea is to start with random a1 and a2 values and then iteratively update the values, reaching minimum cost.

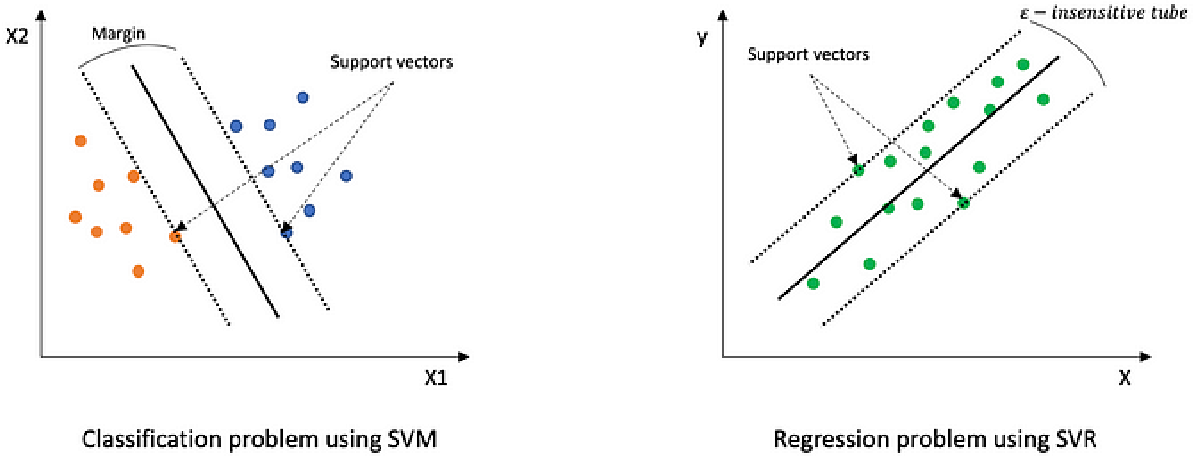
A gradient is the derivative that defines the effects on outputs of the function with a little bit of variation in inputs.(Olive, 2017)

II.4.1.2 Support Vector Regression (SVR)

Support Vector Regression (SVR) is a type of Support Vector Machine (SVM) algorithm commonly used for regression analysis. SVMs are powerful supervised learning algorithms mainly used for classification tasks.(Awad & Khanna, 2015)

SVMs aim to find the optimal hyperplane that best separates two classes in the input data. A hyperplane is a flat subspace of dimension p-1 in a p- dimensional space, where p is the number of input features. The optimal hyperplane is the one that maximizes the margin, which is the distance between the hyperplane and the nearest data points from each class, known as support vectors.

SVR, similar to SVM, employs the concept of a hyperplane and margin, but their definitions differ. In SVR, the margin is defined as the error tolerance of the model, also called the ε-insensitive tube. This tube allows some deviation of the data points from the hyperplane without being counted as errors. The hyperplane in SVR is the best fit possible for the data points that fall within the ε-insensitive tube. The differences between SVM and SVR are illustrated in Figure II-2.



Classification problem using SVM

Regression problem using SVR

Figure II-2 : The difference of SVM and SVR (Awad & Khanna, 2015)

II.4.1.3 Random forest

Random Forest Regressor is an ensemble learning method used for regression tasks, combining the predictions of multiple decision trees to enhance accuracy and reduce overfitting. Each tree in the forest is trained on a different subset of the data, created through bootstrapping (sampling with replacement), and at each split, a random subset of features is considered. This randomness helps in creating diverse trees, leading to a more robust model that can handle high-dimensional data and provide insights into feature importance as shown in Figure II-3.(Sullivan, 2018)

While Random Forest Regressor effectively reduces overfitting and can handle large datasets with numerous features, it is computationally intensive and less interpretable compared to a single decision tree. Key hyperparameters include the number of trees, which balances performance with computational cost, and the maximum depth of each tree (max depth), which can prevent overfitting by limiting tree complexity.

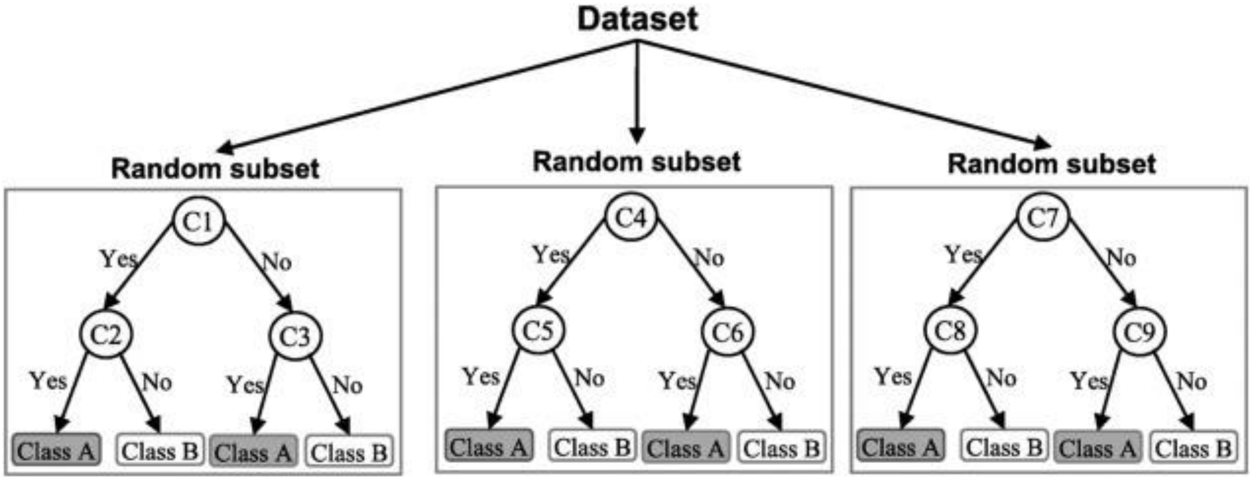


Figure II-3 : An illustration of a random forest (Sullivan, 2018)

II.4.1.4 Gradient boosting

Gradient boosting is defined along the lines of the building of a weighted sum in relation to weak learners which uses a gradient descent method to clarify the designated problem. This is achieved through gradient boosting at each individual repetition, where a weak learner is assigned with extra weight in the weighted sum and adapted to the opposite gradient, with the on-going fitting error and in relation to the on-going ensemble. The main advantage of Gradient boosting is its very fast estimation, which in turn makes it implementable in real-time applications.(Hastie et al., 2013)

Gradient Boosting Regressor works by iteratively improving predictions. It starts with a simple decision tree to predict the target variable (e.g., oil rate). Then, it analyzes the errors (differences between actual values and predictions) and builds another decision tree specifically focused on correcting those errors. The predictions from this new tree are added to the original ones, creating a more accurate ensemble. This process repeats, with each subsequent decision tree tackling the remaining errors from the previous ensemble. Finally, the final Gradient boost model is a weighted combination of predictions from all the individual trees as shown in Figure II-4, resulting in a more robust and accurate predictor.

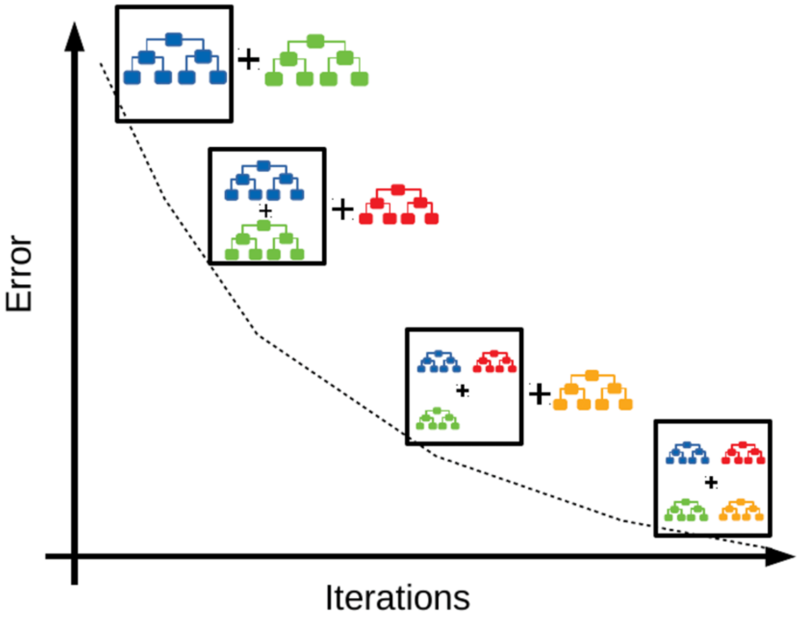


Figure II-4 : Gradient boosting concept (Hastie et al., 2013)

II.4.1.5 Extreme Gradient Boosting

Extreme Gradient Boosting (XGBoost) is an advanced implementation of gradient boosting designed for speed and performance. It builds an ensemble of decision trees sequentially, where each tree corrects the errors of its predecessor by focusing on the residuals. XGBoost introduces regularization (L1 and L2) to prevent overfitting, supports parallel processing to expedite training, and uses a sophisticated tree pruning algorithm to optimize the tree structure. These features make XGBoost particularly effective for large datasets and complex tasks, often leading to superior predictive accuracy compared to other boosting algorithms.(Géron, 2019)

One of XGBoost's key strengths is its flexibility. It allows the use of various loss functions, can handle missing data gracefully, and supports custom evaluation metrics. This versatility, combined with its efficiency and scalability, has made XGBoost a popular choice in data science competitions and real-world applications, such as finance, healthcare, and marketing analytics. However, the model's complexity and numerous hyperparameters require careful tuning to achieve optimal performance, which can be computationally intensive and time-consuming.

II.5 Conclusion

We conclude from this chapter that data-driven technologies can enhance traditional reservoir engineering by incorporating real-world data. In addition, we have seen how data mining can discover patterns in large datasets, and how regression algorithms can uncover relationships between variables and making continuous predictions., offering even more powerful tools for data analysis and decision-making.

Chapter III: Practical Part

III.1 Introduction

In the practical part of this dissertation, a structured workflow was developed to predict oil rates using machine learning techniques (Figure III-1). The process starts with collecting field data from Hassi-Messaoud field, and then a rigorous data preparation follows, involving ensuring a robust dataset. Feature selection is performed by examining correlation matrices to identify the most relevant variables. The model training phase involves selecting suitable machine learning algorithms, splitting the data into training, validation, and test sets. The model's accuracy is evaluated then, and if results are unsatisfactory, the workflow iterates back to data preparation for further refinement. This approach ensures reliable and actionable predictions for oil production decision-making.

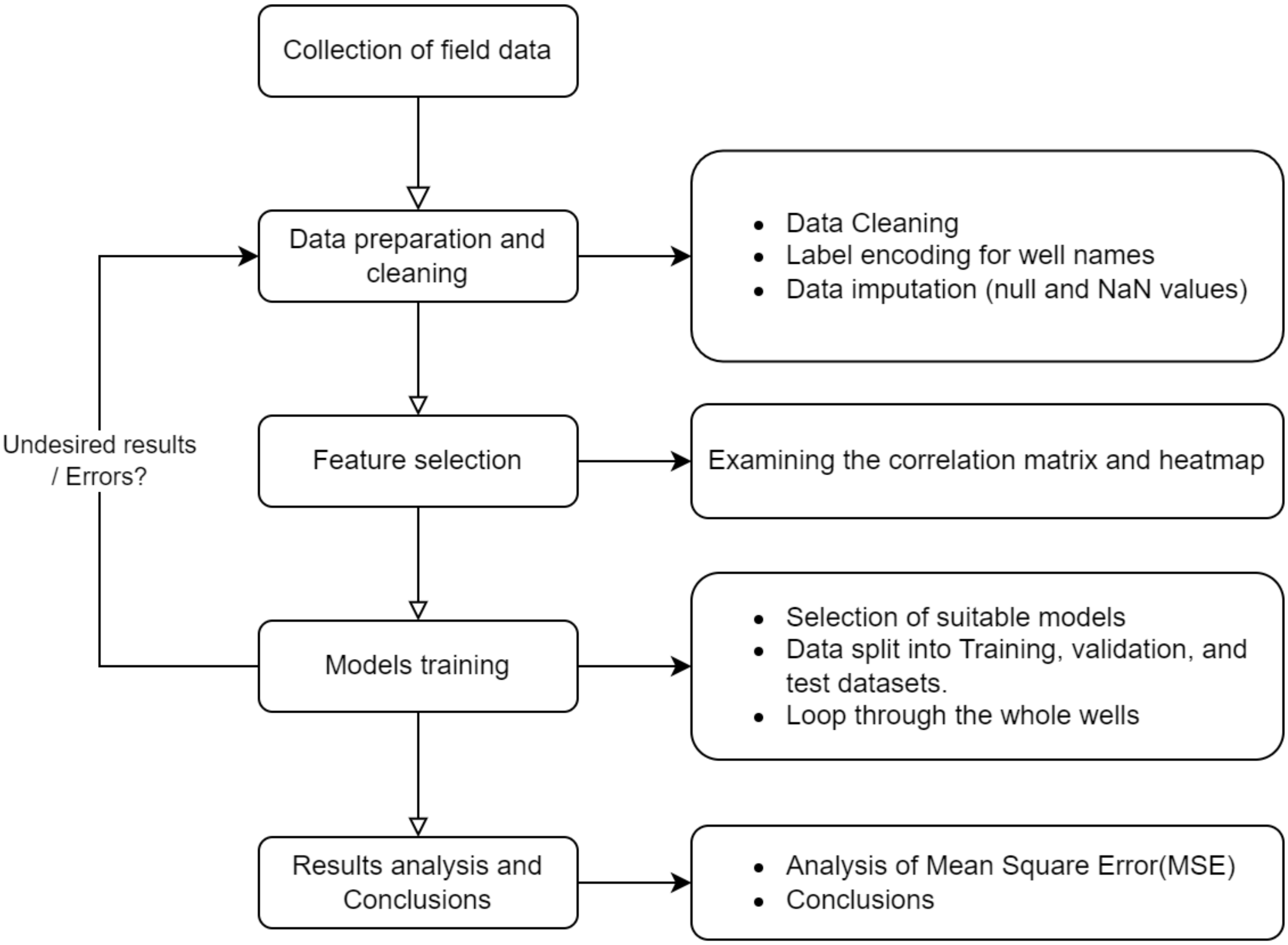

Collection of field data
Data preparation and cleaning
• Data Cleaning
• Label encoding for well names
• Data imputation (null and NaN values)
Undesired results / Errors?
Feature selection
Examining the correlation matrix and heatmap
Models training
• Selection of suitable models
• Data split into Training, validation, and test datasets.
• Loop through the whole wells
Results analysis and Conclusions
• Analysis of Mean Square Error(MSE)
• Conclusions

Figure III-1 : Workflow for Predicting Oil Rates Using Machine Learning

III.2 Presentation of Hassi-Messaoud field

The Hassi-Messaoud field represents one of the most complex fields in the world. During geological history, this field has undergone, on the one hand, an intense tectonic evolution characterized by distinctive compressive phases. On the other hand, by the diagenetic transformation in the reservoir, during its burial over geological time, until the deposit took the current shape or configuration. These events can sometimes improve the petrophysical parameters (natural hydraulic fracturing, dissolution, etc.) as well as reduce them (reduction of porosity, cementation due to solution pressure phenomena, creation of matrices of small grains, etc…).

III.2.1 Geographic location

The Hassi Messaoud field is located 850 km southeast of Algiers (650 km as the crow flies) and 350 km from the Tunisian border. The dimensions of the field reach 2500 km2 with an oil-impregnated surface of approximately 1600 km2(KECHAR, 2020). Its location in Lambert South Algeria coordinates is as follows:

- 790,000 to 840,000 East,
- 110,000 to 150,000 North.

In geographic coordinates:

- To the north by latitude 32° 15′,
- To the west by longitude 5° 40′,
- In the South by latitude 31° 30′,
- To the East by longitude 6° 35′.

III.2.2 Geological setting

The Hassi Messaoud field occupies the central part of the Triassic province, east of the Oued Mya depression in district IV which, by its surface area and its reserves, is the largest oil deposit in Algeria which covers an area of almost 2500 km^2 (KECHAR, 2020). It is limited:

- In the North-West by the Ouargla deposits (Gellala, Ben Kahla and Haoud Berkaoui),

- To the South-West by the deposits of El Gassi, Zotti and El Agreb,
- To the South-East by the deposits; Rhourde El Baguel and Mesdar.

Geologically, it is limited:

- To the West by the Oued M'ya,
- To the South by the Amguid El Biod mole,
- To the North by the Djammâa-Touggourt structure,
- To the east by the Dahar shoals, Rhourde El Baguel and the Ghadames.

Figure III-2 shows the geological situation of the Hassi Messaoud field.

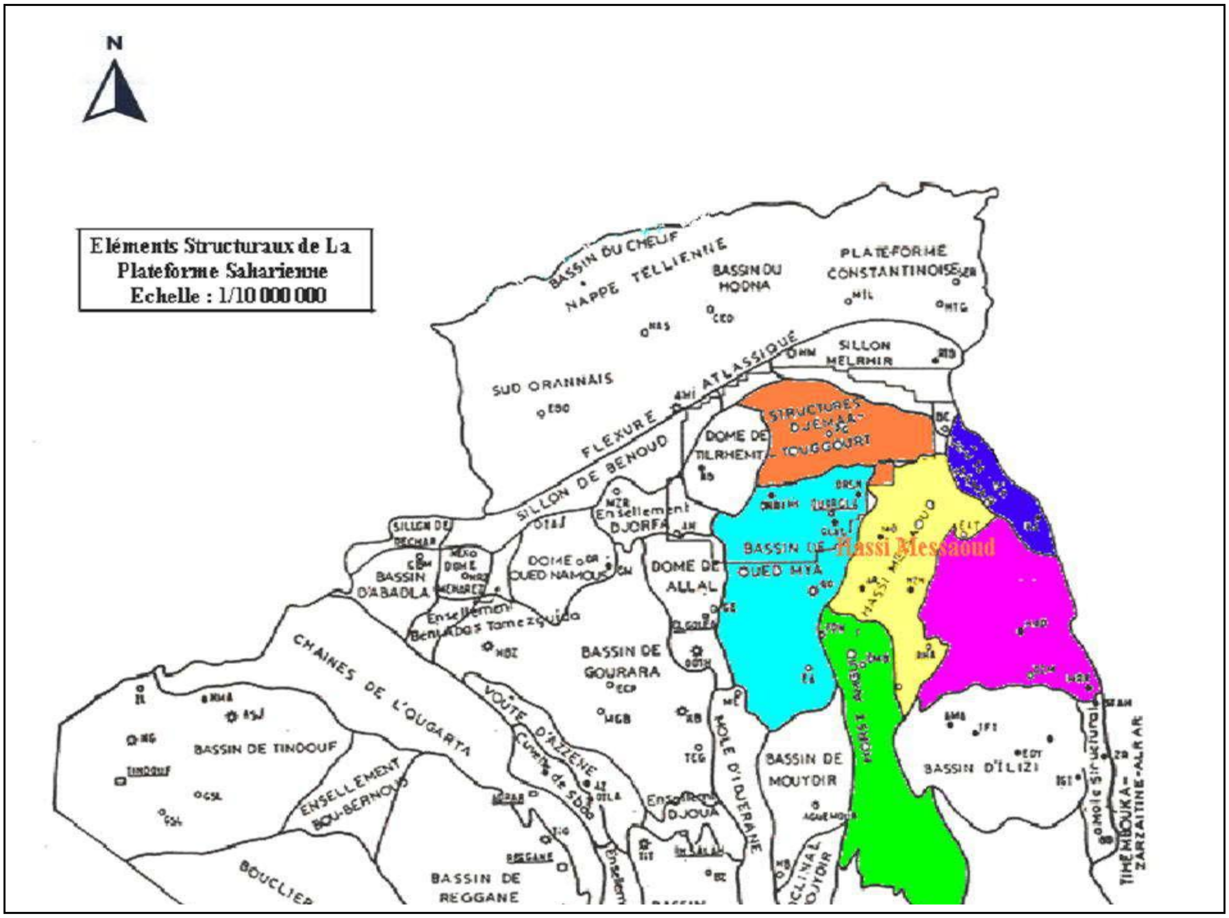


Figure III-2 : Geological situation of the Hassi Messaoud field (KECHAR, 2020).

The Hassi Messaoud structure develops into a vast sub-circular anticline 45 km in diameter, direction: North – East / SOUTH-West. It is partially cracked and the cracks are due to plate tectonic movements which caused the structure to become anticlinal. The reservoirs have undergone natural hydraulic fracturing(TRABELSI, 2019).

Accidents affecting the reservoir are of two types:

- The faults in the submeridian direction and as well as the other faults, perpendicular in the northwest/southeast direction, highlight the tectonic character of the region,
- Breaks without releases which have a great effect on reservoir fracturing

From the reservoir characteristic point of view, the Hassi Messaoud deposit is defined in a perfect trilogy:

- Heterogeneous on the vertical and horizontal plane,
- Discontinuous from the point of view of fluid flow,
- Anisotropic: by the presence of silt and the existence of a matrix of small grains.

III.2.3 Well areas and numbering

The Hassi Messaoud field remains traditionally divided into Hassi Messaoud North and Hassi Messaoud South, Currently, the field is subdivided into 25 production zones. These zones are relatively independent, corresponding to a set of wells which communicate with each other lithologically and behave in the same way from a pressure point of view.(TRABELSI, 2019)

The Hassi Messaoud field is divided from east to west into two distinct parts: The field South and the North field, each has its own numbering.

III.2.3.1 North Field

It is a geographical numbering supplemented by a chronological numbering,

example: Omo38, Onm14, Ompz12 Where:

- **O:** Capital letter, Ouargla permit.
- **m:** area of the oil zone: 1600 km2.
- **o:** Tiny, surface area of the oil zone of 100 km2

- 3: x, and 8: y.

III.2.3.2 South Field

The numbering of the zones is chronological. Ex : MD1, MD2, MD3,…MDZ509*, MDZ557* (Figure III-3).

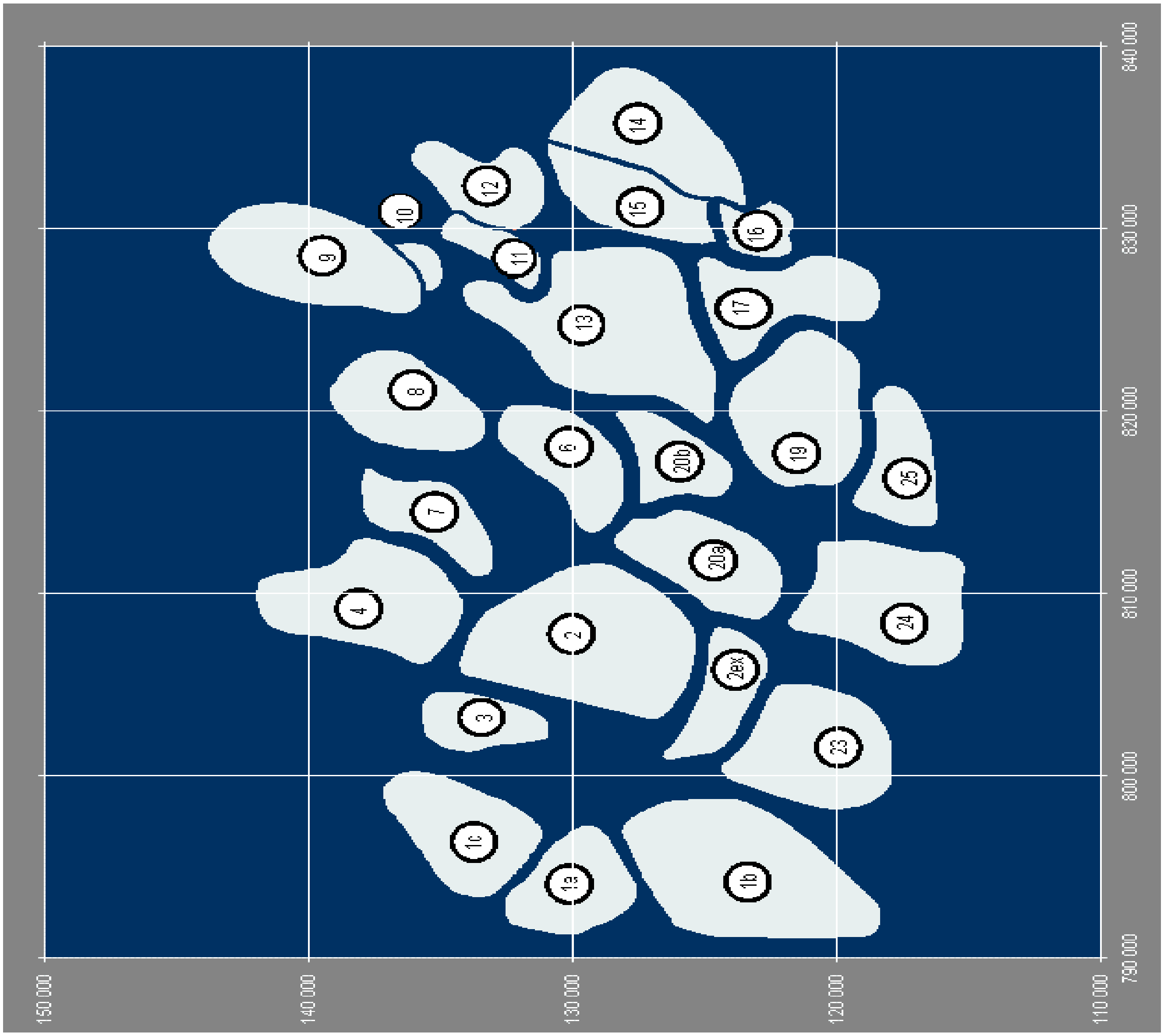


Figure III-3 : Hassi Messaoud reservoir zones numbering (KECHAR, 2020)

III.2.4 Field history

The Hassi Messaoud deposit was discovered on January 16, 1956 by the first MD1 drilling, installed following a seismic refraction survey not far from the Hassi Messaoud camel well.

On June 15 of that same year, this drilling discovered oil in the Cambrian sandstones at a depth of 3,338 meters.

In May 1957, 7 km north-northwest of MD1, the OM1 borehole, drilled by the C.F.P.A, confirmed the existence of a very significant quantity of oil in the Cambrian sandstones.(KECHAR, 2020)

The deposit was therefore covered by two distinct concessions:

- In the North the C.F.P.A,
- To the south the SN. REPAL.

The boundary cuts the field in the East - West direction into two approximately equal parts.

III.2.5 Reservoir Description

The Hassi Messaoud field is part of the eastern province of the Saharan platform. This province contains the main hydrocarbon accumulations of the Sahara, the reservoirs are mainly made up of different sandstone levels from the Cambro-Ordovician and Triassic. Its depth varies between 3100 and 3380 m. Its thickness is up to 200 m. The oil is light with an API rating of 45.4. Its initial pressure evaluated in the well is 482 kg/cm2 for a bubble point between 140 kg/cm2 and 200 kg/cm2.(TRABELSI, 2019)

At Hassi Messaoud the hydrocarbons are found in the Cambro-Ordovician which is subdivided from top to bottom into:

- Sandstone from Hassi Messaoud,
- EL-GASSI sandstone (lower part of the clay-sandstone of Oued Maya). Due to the Hercynian unconformity, a large part of it has been eroded and it is the salt Triassic which constitutes the cover of the reservoir.

The black Silurian clays, 40 km northwest of Hassi Messaoud, rich in Kerogen (organic matter), are supposed to be the source rock.

III.3 Application of ML regression algorithms on Hassi-Messaoud field data

III.3.1 Data preparation and cleaning

This pre-processing part is essential in machine learning projects that transform raw data into a well-structured format suitable for ML algorithms. In this part, we will be uncovering its patterns, and eliminating inconsistencies and errors in data to use it for the next ml models.

By importing the data as an Excel file, these data are available for the different wells are represented in Table III-1.

Table III-1 : Features description

Feature	**Description**	**Unit**
UWI	Well name	/
Jaugeage.Date	Well test date	dd/mm/yy
DEBIT_HUILE	Oil flow rate	m3/h
DEBIT_GAS	Gas flow rate	sm3/h
DEBIT_EAU	Water flow rate	m3/h
DIAM_DUSE	Choke size	mm
GOR	Gaz oil ratio	/
WC	Water cut	/
TEMP_PROD	Time of production	Hour
AVG_WHP_P	Average Wellhead Pressure	bar
WELL_TYPE	Type of Well	Object, unitless

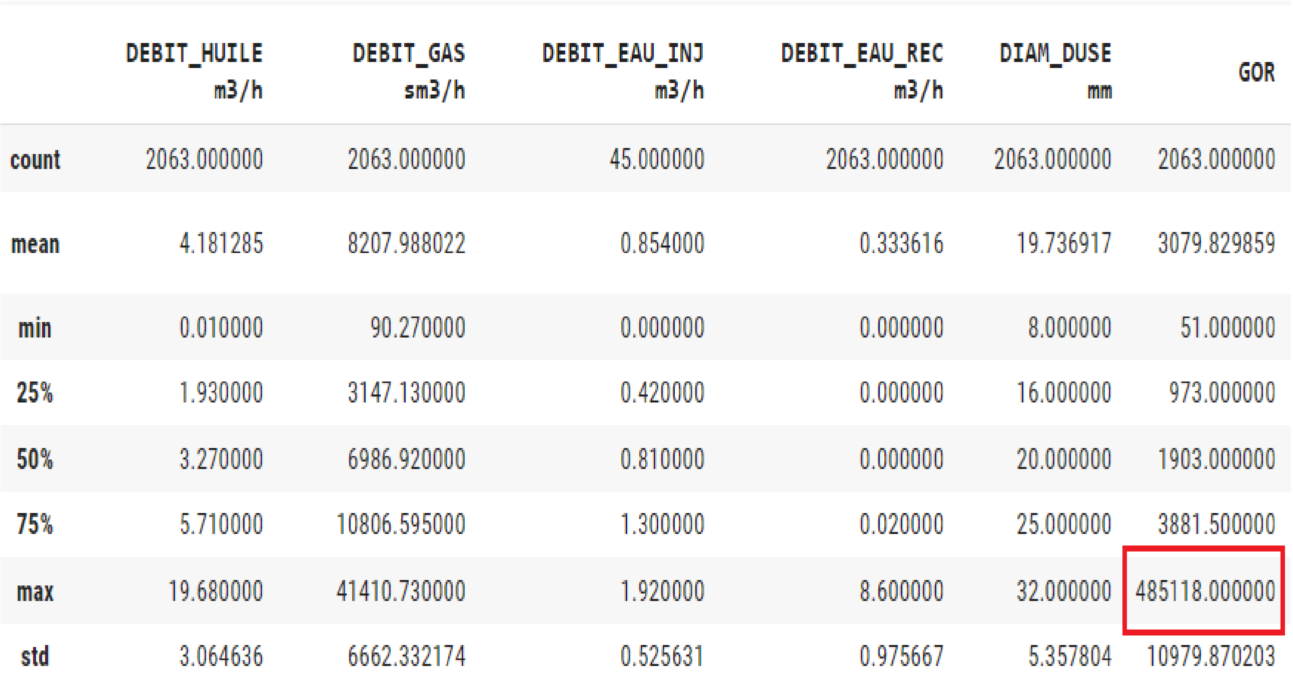
III.3.2 Data cleaning

Since captures, and humans can make mistakes in inputting data, we have to remove unusual values from our datasets, to get an idea about our data and to be more familiarized with it, we have used **describe()** function in pandas library, which is represented in Table III-2.

we can see that for the maximum value of GOR in our data is 485118 which seems to be wrong value, in order to fix that we have limited the range of GOR from 51 to 25000 which is the usual GOR value, and also since more than 75% of rows has a GOR around 4000.

In addition, for other features, data distribution seems to be normal however, there are some null and NaN values which should be removed.

Table III-2 Features statistics



	DEBIT_HUILE	DEBIT_GAS	DEBIT_EAU_INJ	DEBIT_EAU_REC	DIAM_DUSE	GOR
	m3/h	sm3/h	m3/h	m3/h	mm	
count	2063.000000	2063.000000	45.000000	2063.000000	2063.000000	2063.000000
mean	4.181285	8207.988022	0.854000	0.333616	19.736917	3079.829859
min	0.010000	90.270000	0.000000	0.000000	8.000000	51.000000
25%	1.930000	3147.130000	0.420000	0.000000	16.000000	973.000000
50%	3.270000	6986.920000	0.810000	0.000000	20.000000	1903.000000
75%	5.710000	10806.595000	1.300000	0.020000	25.000000	3881.500000
max	19.680000	41410.730000	1.920000	8.600000	32.000000	485118.000000
std	3.064636	6662.332174	0.525631	0.975667	5.357804	10979.870203

III.3.3 Label encoding for well names

Since our data is combined and separated by well name "UWI" feature, One-hot encoding can be beneficial for that feature in our data for several reasons. It allows machine learning algorithms to effectively handle categorical features, enables the model to capture potential interactions between different well names, and facilitates the assessment of individual well names' importance in influencing the target variable.

In this step we have used **LabelEncoder** library in python to facilitate the work, and finally we have separated each dataset according to a digit instead of a “well name”, hence the first well is encoded with the number 1, and the last one with the number 53 (Figure III-4)

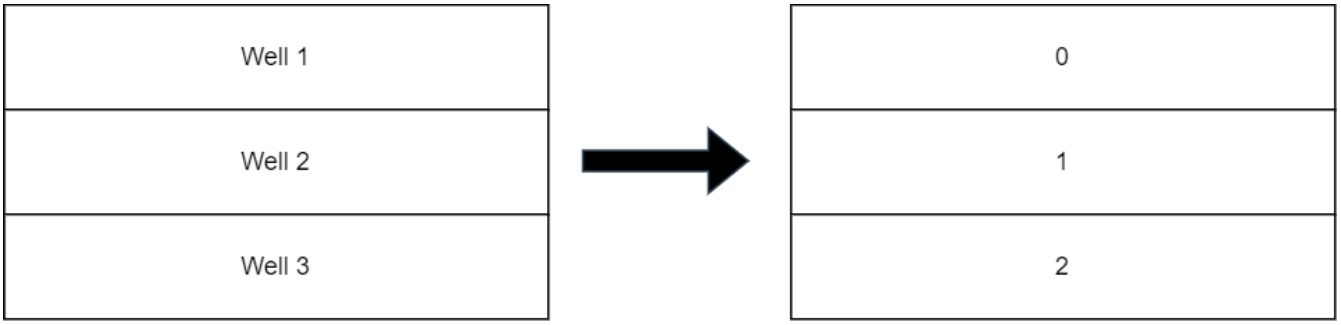


Figure III-4 : Label encoding for well names

III.3.4 Determining null and NaN values and data imputation of missing values

Most of datasets contains null and NaN (Not A Number) values which can occurs due to the data capturing and transmission issues, incorrectly imputing missing values can lead to major discrepancies between the imputed data and the real data. This, in turn, can render the entire analysis and resulting model useless.

To get an idea about these values, we used **isna()** function in panda and **missingno** python libraries to get an idea and the number of those values, Figure III-5 shows that some features has huge missing values, which necessitates a removal.

In addition Figure III-6 shows that the **production time** feature has more than 3000 missing value, however for the other features the missing values varies from 0 to 63 for GOR, WC Pressure and oil production, hence we dropped the production time column, and used the **KNNImputer** approaches to impute missing rows, since that the latter features are correlated to other features (pressure, water cut etc...).

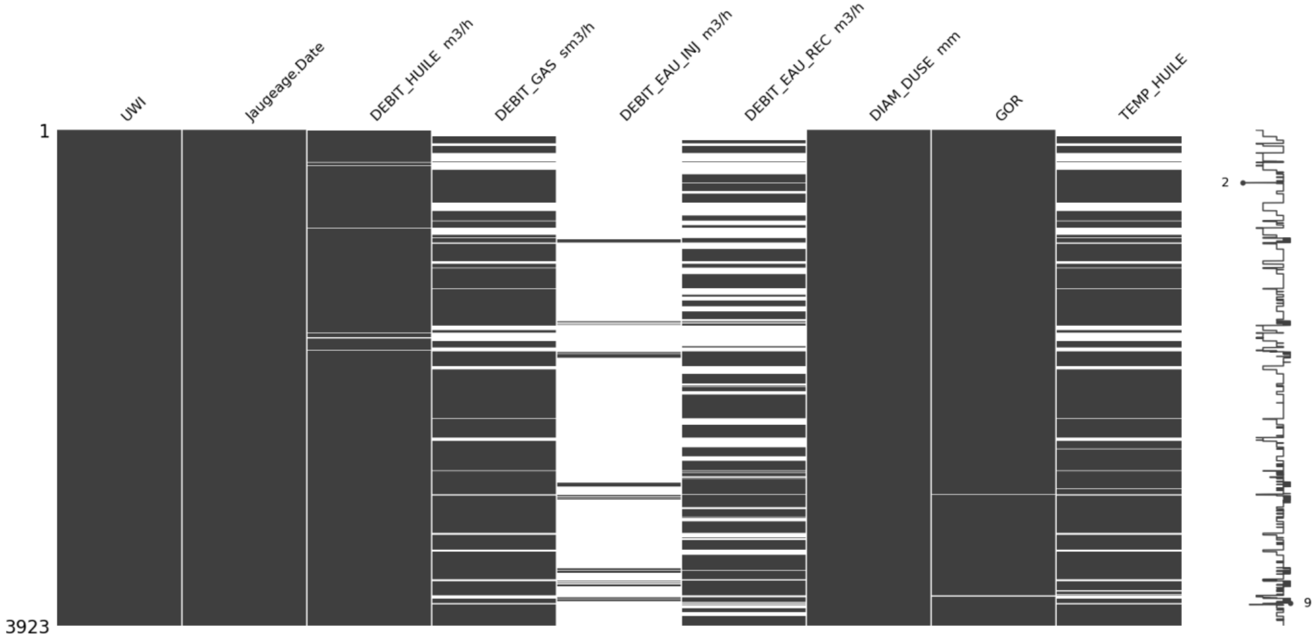
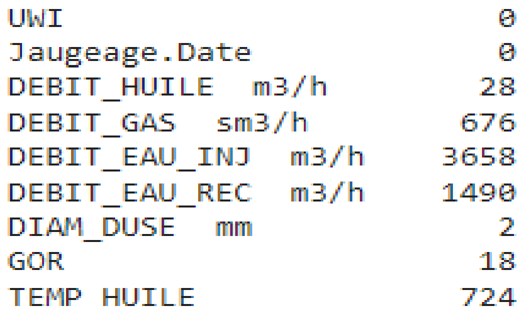


Figure III-5 : Missing values from dataset

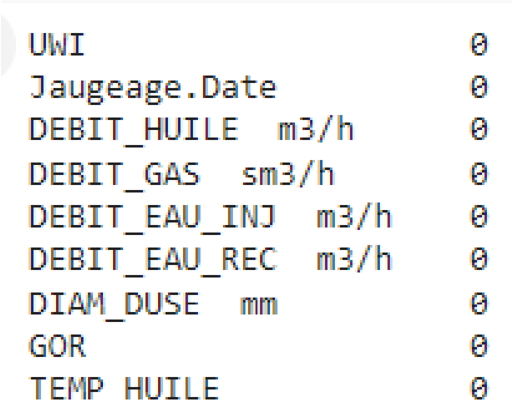


```
UWI                      0
Jaugeage.Date            0
DEBIT_HUILE  m3/h       28
DEBIT_GAS  sm3/h       676
DEBIT_EAU_INJ  m3/h   3658
DEBIT_EAU_REC  m3/h   1490
DIAM_DUSE  mm            2
GOR                     18
TEMP HUILE             724
```

Figure III-6 : NAN data values sum

KNN imputation addresses missing data in variables by using information from similar data points. This technique assumes that observations with close characteristics (nearest neighbors) likely share similar values for missing features. By identifying these nearest neighbors, KNN imputation estimates the missing values based on the values of their close counterparts, for that purpose we have used fancy impute library in python to achieve that.

After treating the data with KNN imputation, Figure III-7 show that the whole dataset is fully filled and doesn't have NAN values.



```
UWI                    0
Jaugeage.Date          0
DEBIT_HUILE  m3/h      0
DEBIT_GAS  sm3/h       0
DEBIT_EAU_INJ  m3/h    0
DEBIT_EAU_REC  m3/h    0
DIAM_DUSE  mm          0
GOR                    0
TEMP HUILE             0
```

Figure III-7 : New NAN data values sum

Furthermore, in order to deal with time column which is in nanoseconds unit, we need to convert the datetime64[ns] to useable format, we use pandas **to_datetime** method, and we specify a format which needs to match exactly the data provided in the Dates column:

- %Y Full year
- %m Month
- %d Day of the month
- %I Hour (12hr clock)
- %p AM or PM
- %M Minute
- %S Second

III.3.5 Feature Selection

Analyzing the correlation matrix and its corresponding heatmap reveals valuable insights into the relationships between variables in our data. Strong positive correlations (values close to 1) indicate variables that tend to move together, while strong negative correlations (values close to -1) suggest variables that move in opposite directions. Values close to 0 signify a weak or nonexistent linear relationship. This visual tool allows us to identify patterns, dependencies, and potential relationships (high correlation between variables).

After dropping the columns that theoretically has no strong relationship with oil rate, the Figure III-8 shows that the most parameters affecting oil rate are:

- GOR

- Water cut

- Choke size
- Well head pressure

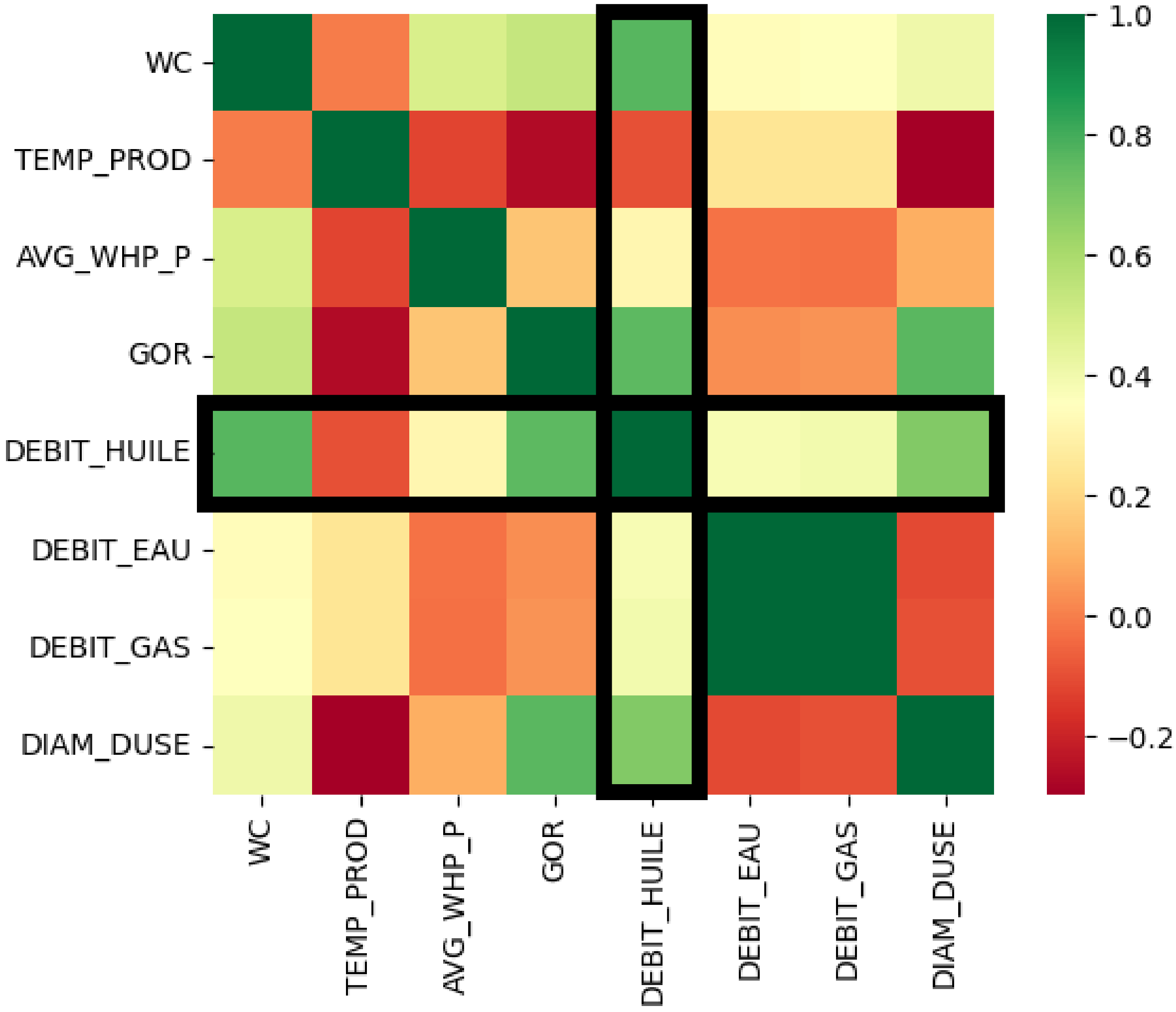


Figure III-8 Features heatmap

III.3.6 Model training

III.3.6.1 Selection of test algorithms

In order to select the most suitable model for our training model, we had to write a code that evaluates the performance of several regression algorithms on our data and identifies the one with the lowest mean squared error (MSE) and the closest to zero, indicating the best fit for our target variable prediction, which is the flowrate of oil.

The regression algorithms that are used are:

- LinearRegression
- RandomForestRegressor
- GradientBoostingRegressor
- SVR (Support Vector Regressor)
- xgboost.XGBRegressor

III.3.6.2 Data splitting

In addition, to ensures the model is evaluated on unseen data, we have to divide the data into training and testing sets using a technique like "train_test_split" from scikit-learn, we used the recommended data split percentage with is:

- 60% for training data.
- 20% for validation data.
- 20% for testing data.

where the testing data is used for studying how much is our model is accurate when trained on training data.

III.3.6.3 Model training and results

Finally, in order to calculate the MSE for every well, we have to write a loop for each well encoded name (from 1 to 53) and then, we calculate the mean of MSE, which is desired to be as close as possible to 0.

The result of training which is shown in Table III-3, indicates that best model for our study is XGBoost, since it has the lowest MSE, however, Random Forest Regressor and Gradient Boosting shows a good and similar MSE and can be used too, however, Linear Regression and SVR have bigger MSE in comparison to other models, which will result less accuracy model.

Table III-3 : MSE comparison of different models

Model	MSE
XGBoost	0.218
Random Forest Regressor	0.225
Gradient Boosting	0.316
Linear Regression	7.514
SVR	8.511

To get an idea of the predicted oil flowrate, we have chosen random wells (Well number 9, 26 and 38) and plotted the actual, and the model values, for the test data.

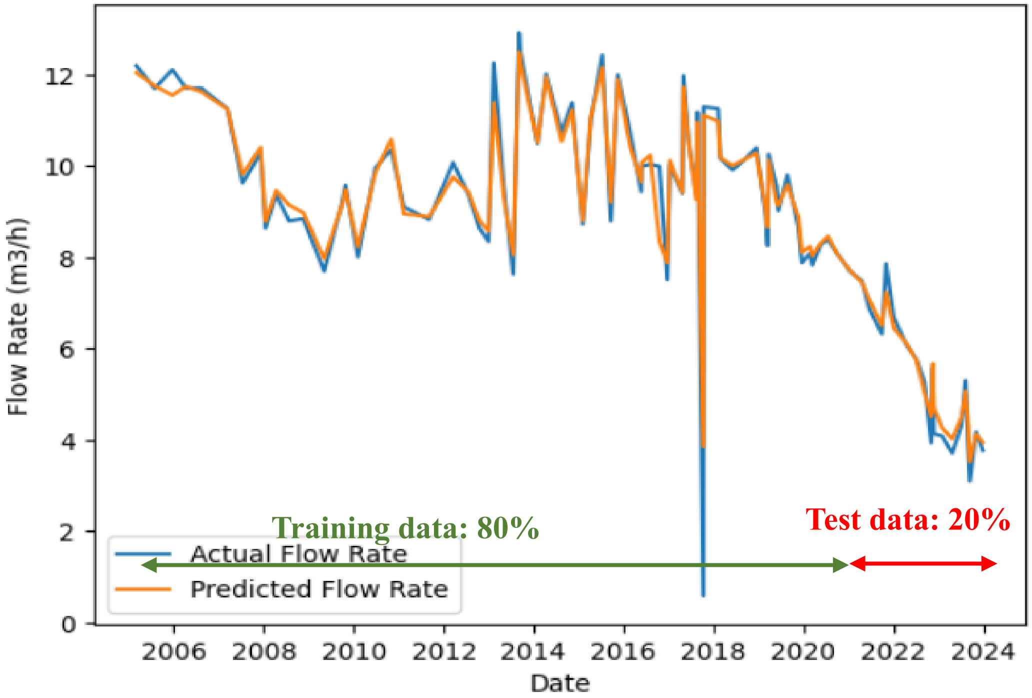


Figure III-9 Actual vs. Predicted Flow Rate for Well num 9

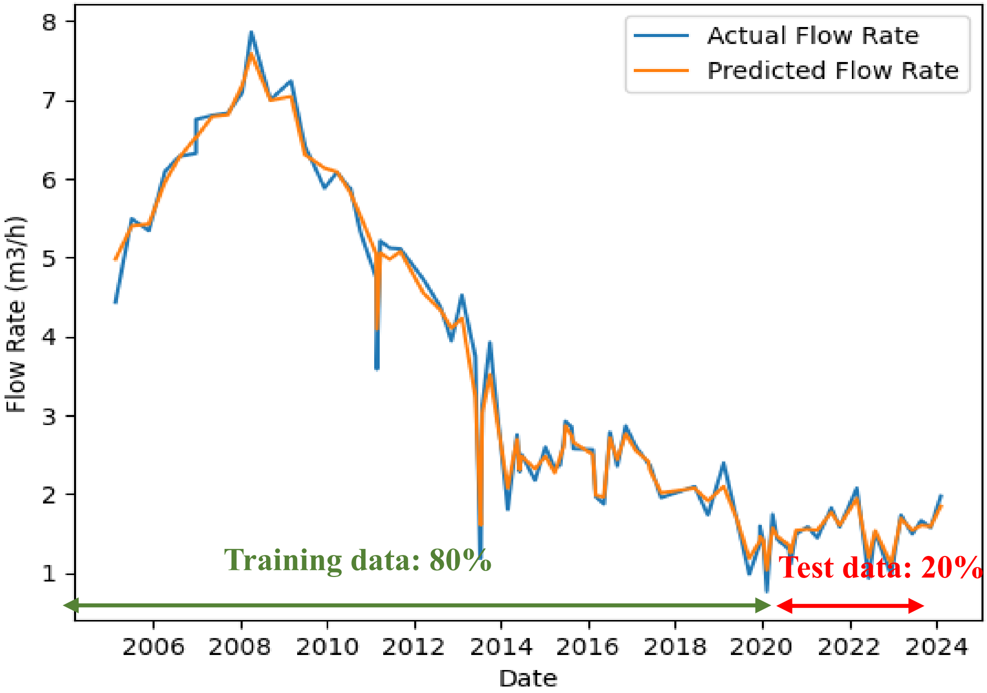


Figure III-10 : Actual vs. Predicted Flow Rate for Well num 26

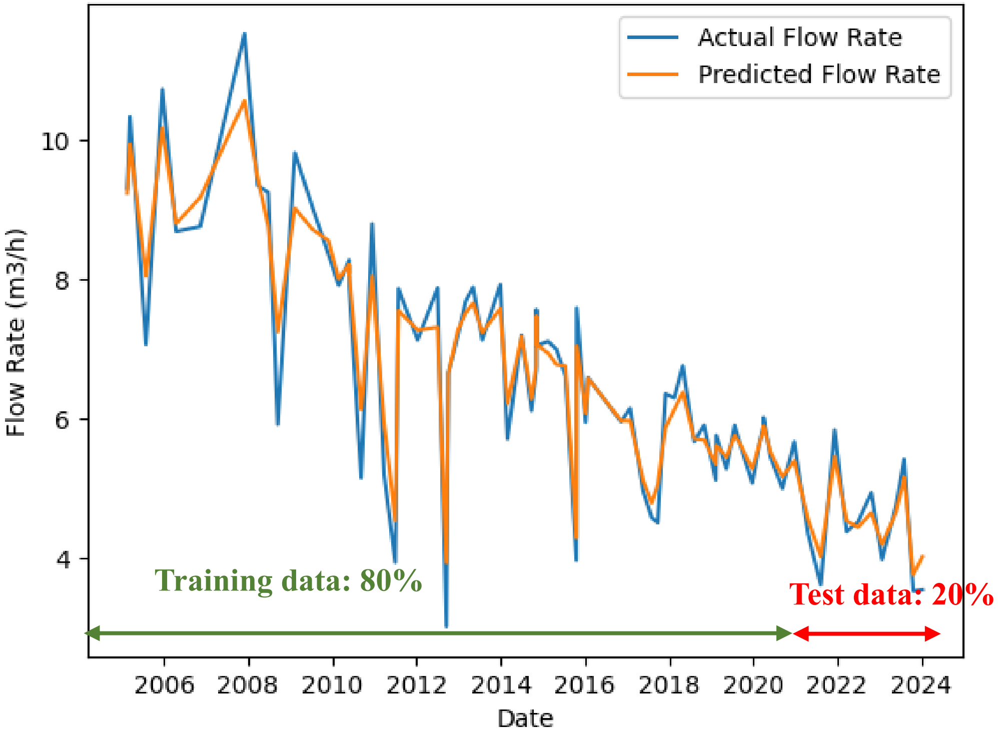


Figure III-11 : Actual vs. Predicted Flow Rate for Well num 38

III.4 Results Comments:

Previous Figures shows that the model prediction is close to the actual data in average (more than 90%), which is a good sign of the accuracy of the model, however, it shows some fluctuations at some points, especially in well number 9, And in some cases, the predicted model curve does not reach the peak, even with the training data, that’s due to the complexity of non-linear equations, therefore, there's potential in the future to refine the model to better capture peak values and reduce fluctuations.

III.5 Conclusion

After the application of different machine learning algorithms, and evaluating its cost, requirements and time, a comparison has been done with traditional numerical simulation is shown in Table III-4 :

Table III-4 : Comparison of ML modeling methods with traditional reservoir simulation

Feature	**Traditional Reservoir Modeling (Values & Explanation)**	**Data-Driven (Machine Learning) Reservoir Modeling**
Model Development Time	• **3 - 12 Months** (Lengthy due to data acquisition, geological modeling, reservoir simulation, and history matching) for data preparation phase • **1-4 Days** for computational simulation operation. (Requires Powerful Workstations for processing large datasets and running simulations)	• **1 – 4 weeks** (Faster due to automated data processing and model training) • **1 minute – 1 hour** for model real time prediction call (Runs on Standard Computers).
Average Cost	**$250,000 - $2 Million+** (High due to personnel costs, software licenses, computational resources, and data acquisition)	**$300 - $200,000** (Lower due to less data requirements, open-source software options, and reduced computational needs)

Data Requirements	**Extensive geological data** (core samples, well logs, seismic surveys), fluid data (PVT analysis), historical production data	**Primarily historical production data and well logs** (May incorporate additional data if available)
Data Source Examples	Geological surveys, wellbore logs, PVT lab reports, SCADA systems, well completion reports	Production data: SCADA systems, well completion reports, well logs (porosity, permeability data)
Calibration	**Manual, Time-Consuming Iteration** (Adjusting numerous model parameters, rerunning simulations, and comparing results with production data)	**Automated Through Machine Learning Algorithms**
Strengths	**Highly accurate** for complex reservoirs and strong models. Provides a deeper understanding of reservoir physics.	**Faster, cheaper, and less data-intensive.** Automated calibration process. Can be used for real-time optimization.
Weaknesses	**Time-consuming and expensive**. Requires significant expertise. Less flexible for rapidly changing reservoir conditions.	**Less interpretable results**. May not be as accurate for complex reservoirs or require significant data pre-processing. Limited to predicting trends based on historical data.

General Conclusion and Recommendations

General conclusion

This research has explored the potential of machine learning for predicting oil production rates. The research investigated various machine learning algorithms, their effectiveness in capturing the complex dynamics of oil production, and their advantages over traditional forecasting methods.

The key findings of this research are:

- Machine learning models can achieve accurate predictions of oil production rates, and compared to traditional reservoir simulation, machine learning models are generally less expensive to develop and deploy. This advantage makes them a more scalable solution, especially for complex reservoirs or for a large number of wells.

- Specific algorithms, such as XGBoost and Random Forest Regressor, demonstrate a strong ability to learn from historical data and identify non-linear relationships between influencing factors and production rates.

- The integration of machine learning with domain knowledge from reservoir engineering can further enhance the accuracy and interpretability of the predictions.

These findings suggest that machine learning holds significant promise for the oil and gas industry. By leveraging this technology, companies can gain valuable insights into future production, optimize resource allocation, and make informed decisions regarding well management and development strategies.

Recommendations

- Increase model complexity by using a more complex model architecture that can better capture the non-linearities and sharp changes in the data.

- Using noise reduction techniques by applying techniques like filtering or smoothing to the training data to reduce the impact of noise.

- Incorporating image analysis techniques using convolutional neural networks (CNNs) to analyze downhole wellbore images and seismic data. This could allow for the extraction of valuable features related to reservoir properties and fluid flow patterns.

- Exploring the integration of additional data sources, such as PVT data and Well tests, to improve model accuracy.

- Developing hybrid models that combine machine learning with physics-based reservoir simulation techniques.

- Develop and integrate Explainable AI (XAI) methods to gain deeper understanding of the factors influencing model predictions. This will increase trust in the models and allow for targeted data collection or model refinement efforts.

References

Abou-Kassem, J. H. (Jamal H., Farouq Ali, S. M. (Syed M., & Islam, R. (2006). *Petroleum reservoir simulation : a basic approach*. Gulf Pub. Co.

Ahmed, T. (2010). *Reservoir Engineering Handbook*. Elsevier Science.

Ahmed, T., & McKinney, P. (2011). *Advanced Reservoir Engineering*. Elsevier Science.

Arnold, K., & Stewart, M. (1998). *Surface Production Operations, Volume 1:: Design of Oil-Handling Systems and Facilities*. Elsevier Science.

Awad, M., & Khanna, R. (2015). *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers*. Apress.

Berry, M. J. A., & Linoff, G. S. (2008). *MASTERING DATA MINING: THE ART AND SCIENCE OF CUSTOMER RELATIONSHIP MANAGEMENT*. Wiley India Pvt. Limited.

Cossé, R. (1993). *Basics of Reservoir Engi...* Editions OPHRYS.

Eberhart, R. C., Simpson, P. K., Dobbins, R. C., & Dobbins, R. W. (1996). *Computational Intelligence PC Tools*. AP Professional.

Géron, A. (2019). *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, Incorporated.

Hans, V., Yuing, X., & Ian, P. (2002). Assessment Of Several Sand Prediction Models With Particular Reference To HPHT Wells. *Proceedings of the SPE/ISRM Rock Mechanics in Petroleum Engineering Conference*. https://doi.org/10.2118/78235-MS

Hastie, T., Tibshirani, R., & Friedman, J. (2013). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer New York.

KECHAR. (2020). *ETUDE DES CHITINOZOAIRES DE L'ORDOVICIEN. EXEMPLE DU PUITS DE BORDJ NILI-2 (NL-2) DANS LE BASSIN D' OUED MYA*.

Moitra, A. K., Bhattacharya, J., Kayal, J. R., Mukerji, B., & Das, A. K. (2021). *Innovative Exploration Methods for Minerals, Oil, Gas, and Groundwater for Sustainable Development*. Elsevier Science.

Montgomery, D. C., Peck, E. A., & Vining, G. G. (2012). *Introduction to Linear Regression Analysis*. Wiley.

Nouri, A., Vaziri, H., Belhaj, H., & Islam, R. (2004). Sand Production Prediction: A New Set of Criteria for Modeling Based on Large-Scale Transient Experiments and Numerical Investigation. *SPE Journal*, *11*. https://doi.org/10.2118/90273-MS

Odeh, A. S. (1969). Reservoir Simulation ...What Is It. *Journal of Petroleum Technology*, *21*(11), 1383–1388. https://doi.org/10.2118/2790-PA

Olive, D. J. (2017). *Linear Regression*. Springer International Publishing.

Scott, J. (2008). *Method and System for Automated Choke Control on a Hydrocarbon Producing Well*.

Shahab D. Mohaghegh. (2017). *Data-Driven Reservoir Modeling*. Society of Petroleum Engineers.

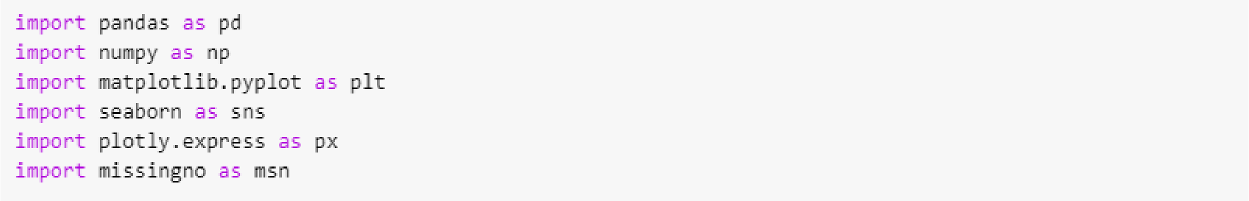
Sullivan, W. (2018). *Decision Tree and Random Forest: Machine Learning and Algorithms: The Future Is Here!* CreateSpace Independent Publishing Platform.

Terry, R. E., Rogers, J. B., & Craft, B. C. (2015). *Applied Petroleum Reservoir Engineering*. Prentice Hall

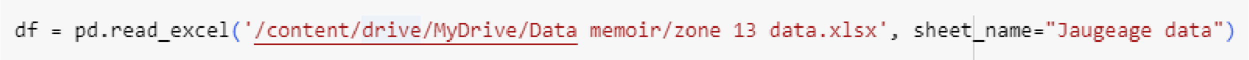
TRABELSI, K. G. (2019). *Caractérisation Pétro-physique d'un Réservoir cambroordovicien de la zone 13 du champ HMD Par l'utilisation de Diagraphies et des Mesures sur Carottes*.

Zurada, J. M., Marks, R. J., & Robinson, C. J. (1994). *Computational Intelligence: Imitating Life*. IEEE.

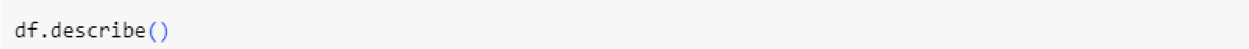
Appendices



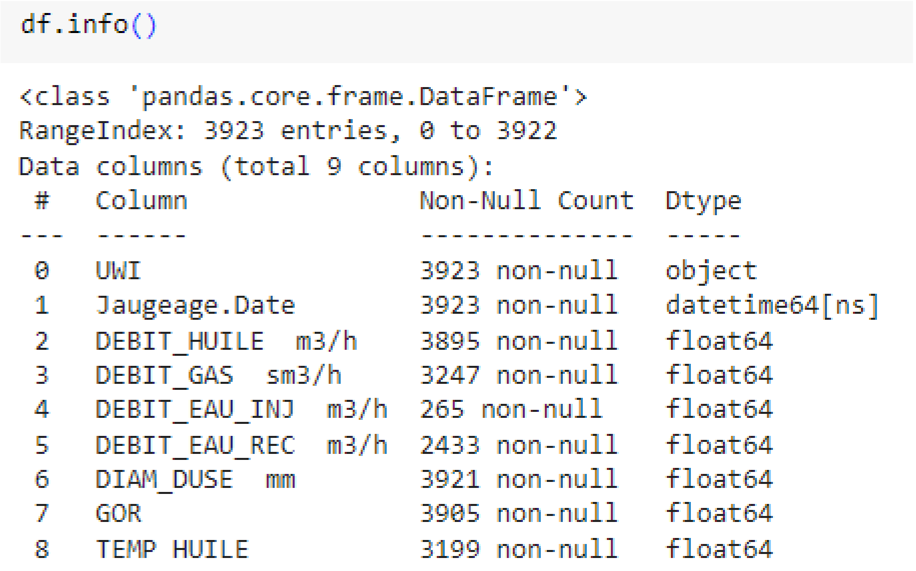
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import missingno as msn
```



```
df = pd.read_excel('/content/drive/MyDrive/Data memoir/zone 13 data.xlsx', sheet_name="Jaugeage data")
```

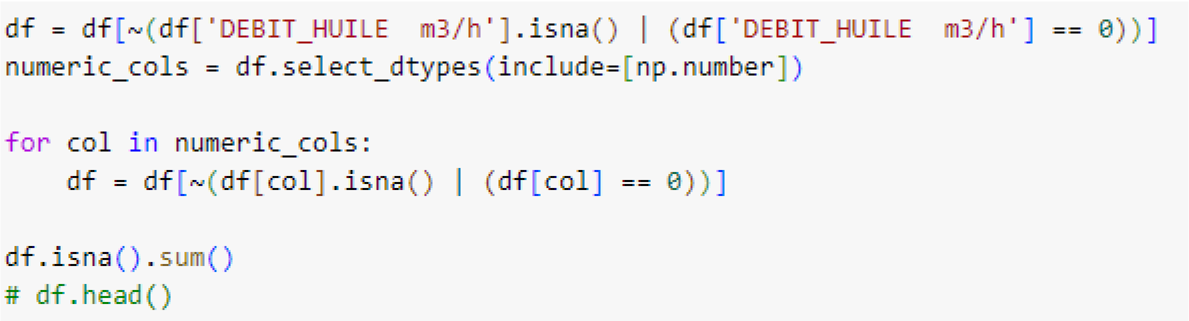


```
df.describe()
```

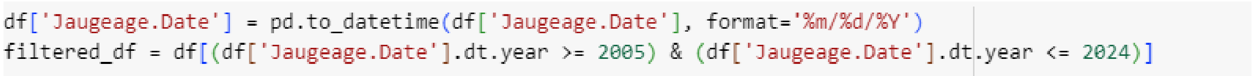


```
df.info()
```

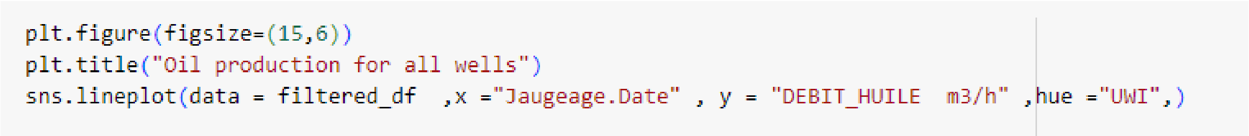
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3923 entries, 0 to 3922
Data columns (total 9 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   UWI                   3923 non-null   object
 1   Jaugeage.Date         3923 non-null   datetime64[ns]
 2   DEBIT_HUILE  m3/h     3895 non-null   float64
 3   DEBIT_GAS  sm3/h      3247 non-null   float64
 4   DEBIT_EAU_INJ  m3/h   265 non-null    float64
 5   DEBIT_EAU_REC  m3/h   2433 non-null   float64
 6   DIAM_DUSE  mm         3921 non-null   float64
 7   GOR                   3905 non-null   float64
 8   TEMP HUILE            3199 non-null   float64
```



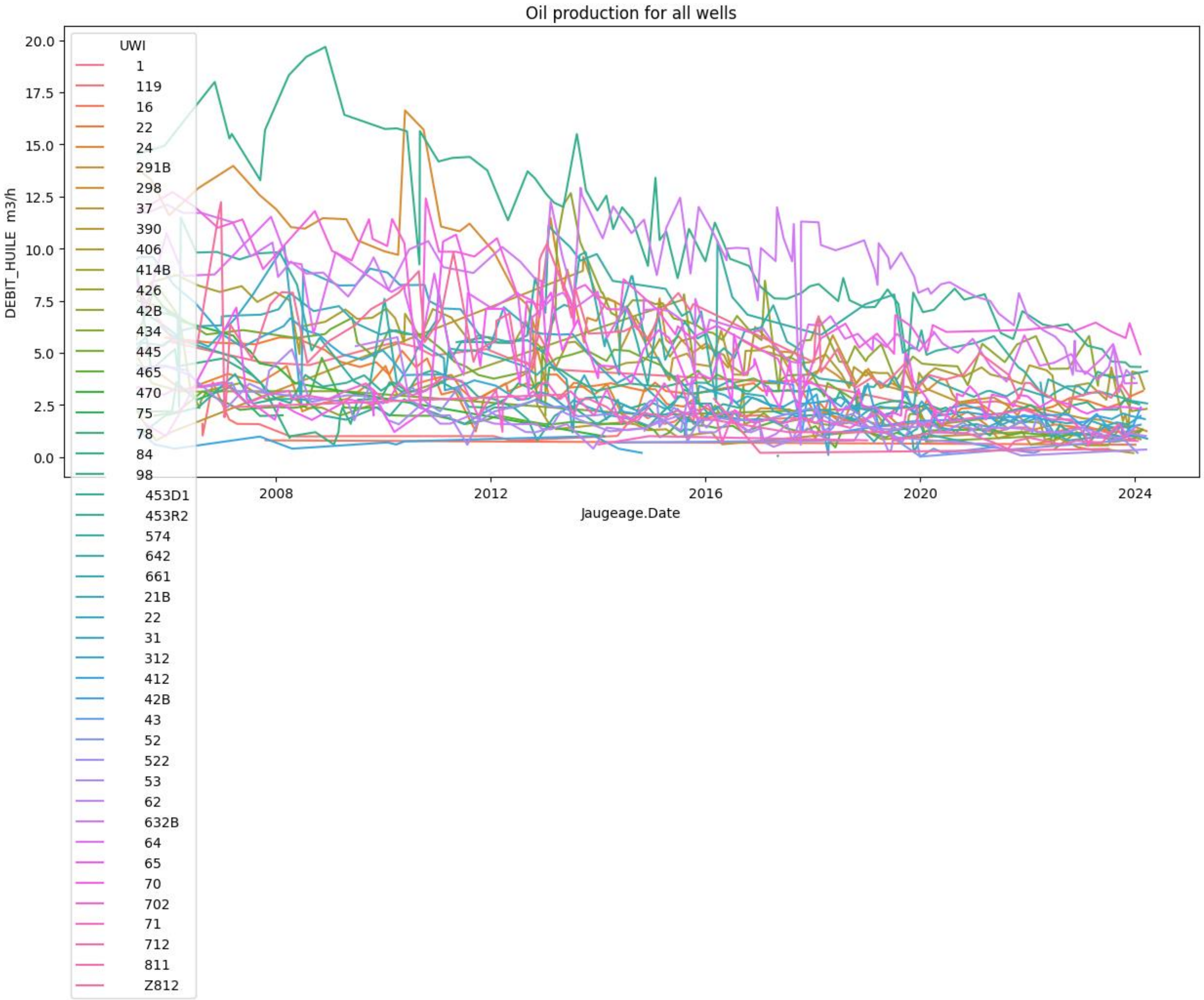
```
df = df[~(df['DEBIT_HUILE  m3/h'].isna() | (df['DEBIT_HUILE  m3/h'] == 0))]
numeric_cols = df.select_dtypes(include=[np.number])

for col in numeric_cols:
    df = df[~(df[col].isna() | (df[col] == 0))]

df.isna().sum()
# df.head()
```

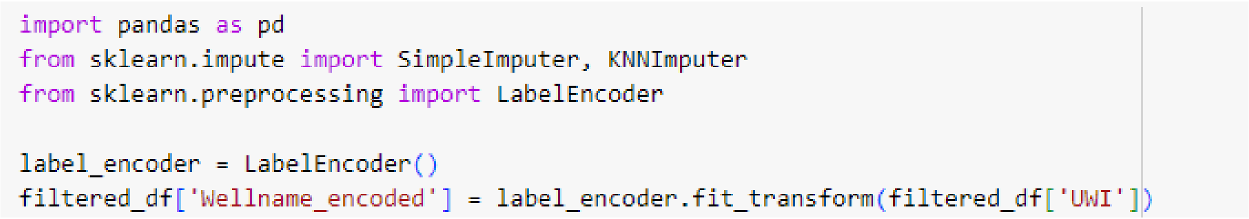


```
df['Jaugeage.Date'] = pd.to_datetime(df['Jaugeage.Date'], format='%m/%d/%Y')
filtered_df = df[(df['Jaugeage.Date'].dt.year >= 2005) & (df['Jaugeage.Date'].dt.year <= 2024)]
```

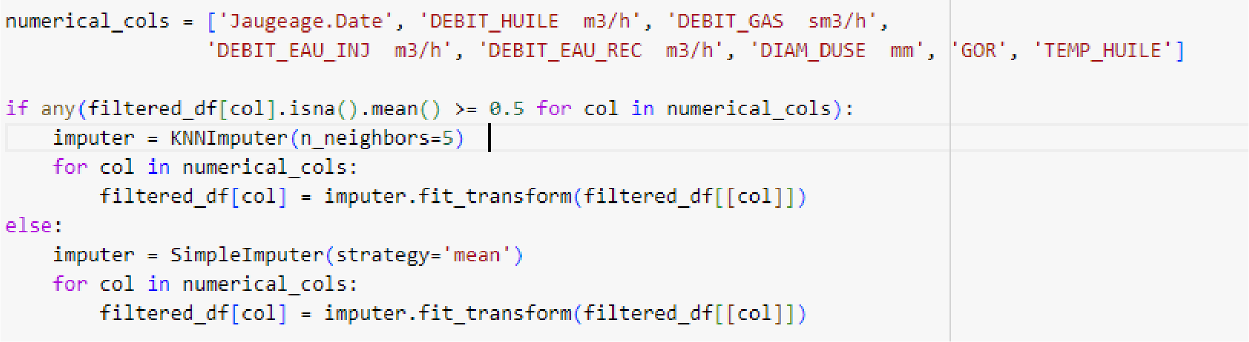


```
plt.figure(figsize=(15,6))
plt.title("Oil production for all wells")
sns.lineplot(data = filtered_df  ,x ="Jaugeage.Date" , y = "DEBIT_HUILE  m3/h" ,hue ="UWI",)
```

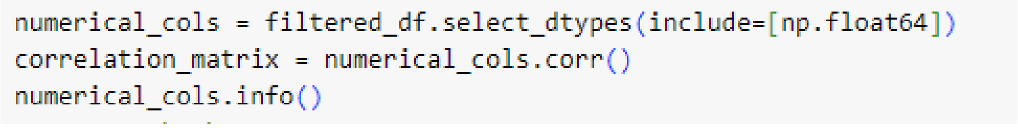




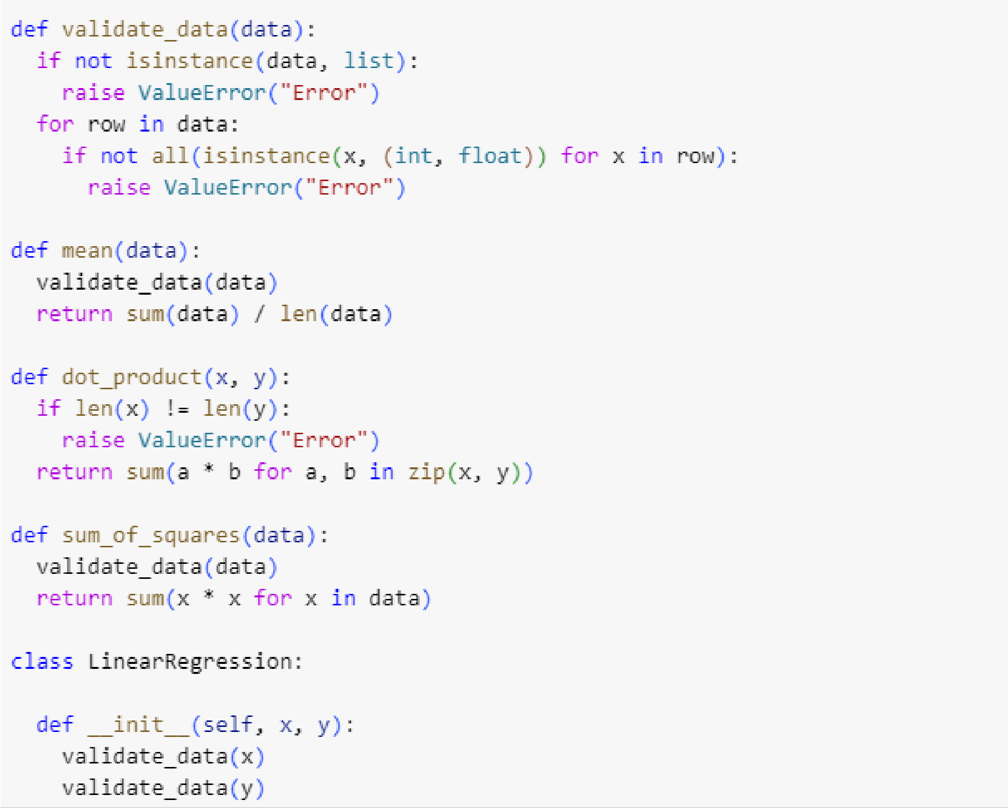
```
import pandas as pd
from sklearn.impute import SimpleImputer, KNNImputer
from sklearn.preprocessing import LabelEncoder

label_encoder = LabelEncoder()
filtered_df['Wellname_encoded'] = label_encoder.fit_transform(filtered_df['UWI'])
```



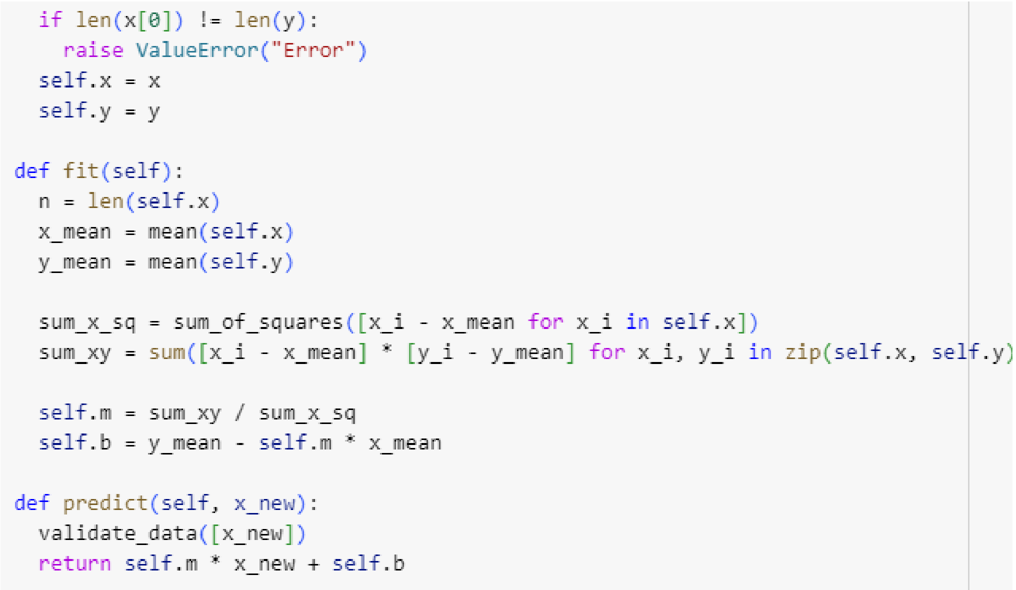
```
numerical_cols = ['Jaugeage.Date', 'DEBIT_HUILE  m3/h', 'DEBIT_GAS  sm3/h',
                  'DEBIT_EAU_INJ  m3/h', 'DEBIT_EAU_REC  m3/h', 'DIAM_DUSE  mm', 'GOR', 'TEMP_HUILE']

if any(filtered_df[col].isna().mean() >= 0.5 for col in numerical_cols):
    imputer = KNNImputer(n_neighbors=5)
    for col in numerical_cols:
        filtered_df[col] = imputer.fit_transform(filtered_df[[col]])
else:
    imputer = SimpleImputer(strategy='mean')
    for col in numerical_cols:
        filtered_df[col] = imputer.fit_transform(filtered_df[[col]])
```



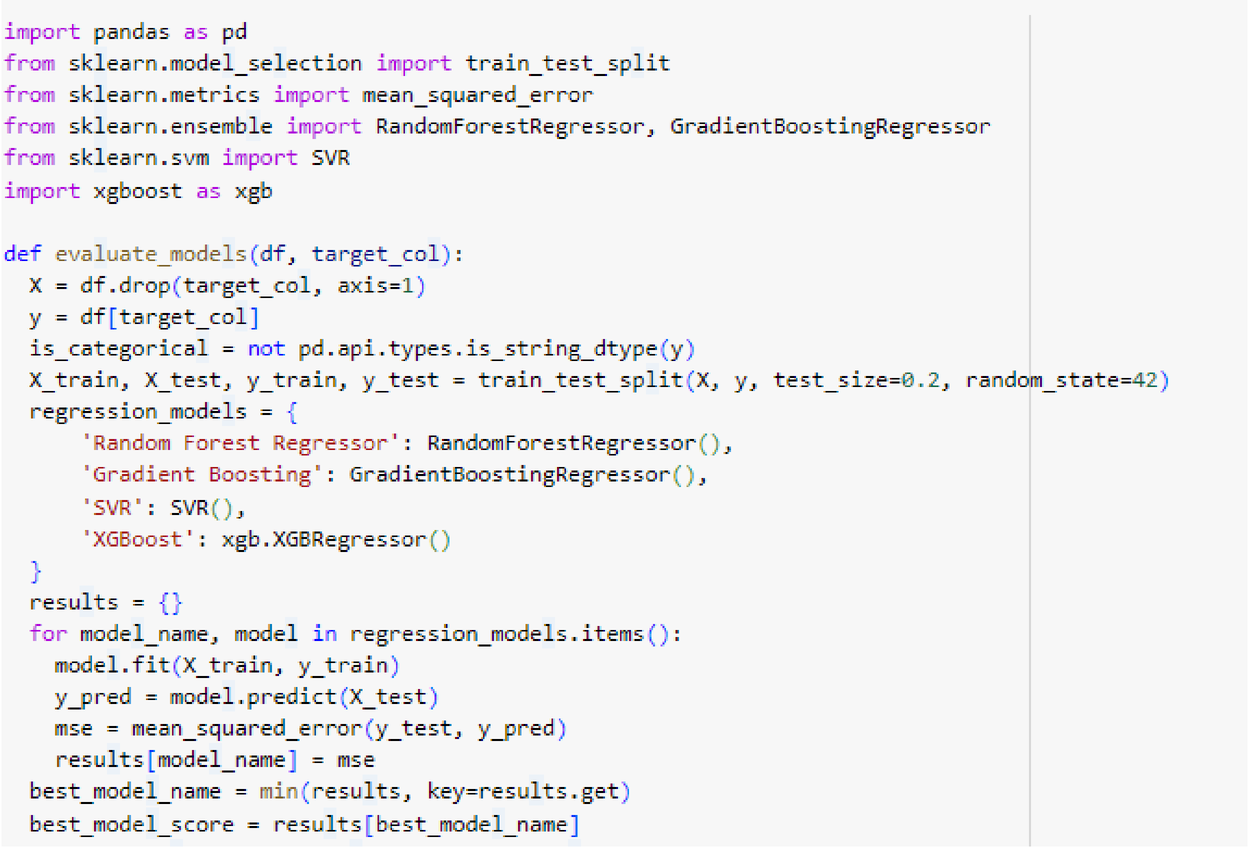
```
numerical_cols = filtered_df.select_dtypes(include=[np.float64])
correlation_matrix = numerical_cols.corr()
numerical_cols.info()
```



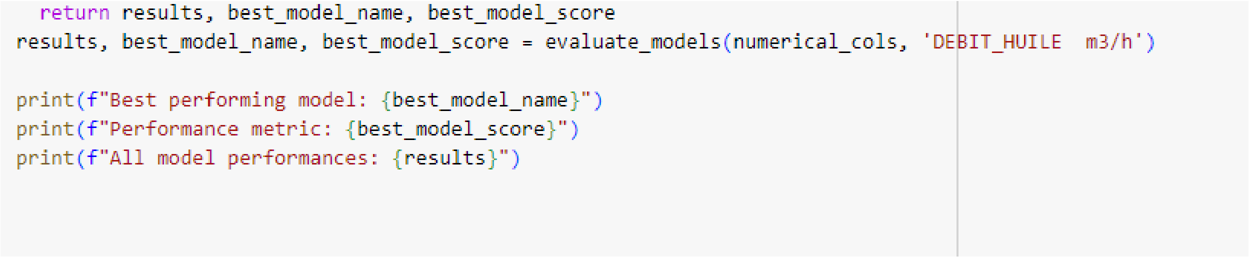
```
def validate_data(data):
  if not isinstance(data, list):
    raise ValueError("Error")
  for row in data:
    if not all(isinstance(x, (int, float)) for x in row):
      raise ValueError("Error")

def mean(data):
  validate_data(data)
  return sum(data) / len(data)

def dot_product(x, y):
  if len(x) != len(y):
    raise ValueError("Error")
  return sum(a * b for a, b in zip(x, y))

def sum_of_squares(data):
  validate_data(data)
  return sum(x * x for x in data)

class LinearRegression:

  def __init__(self, x, y):
    validate_data(x)
    validate_data(y)
```



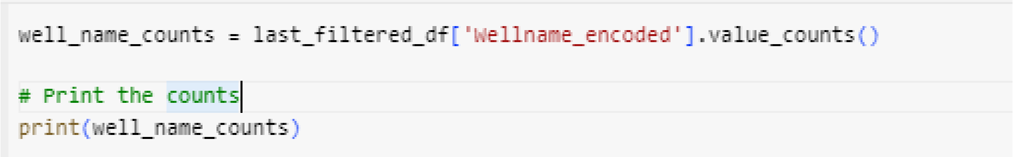
```
  if len(x[0]) != len(y):
    raise ValueError("Error")
  self.x = x
  self.y = y

def fit(self):
  n = len(self.x)
  x_mean = mean(self.x)
  y_mean = mean(self.y)

  sum_x_sq = sum_of_squares([x_i - x_mean for x_i in self.x])
  sum_xy = sum([x_i - x_mean] * [y_i - y_mean] for x_i, y_i in zip(self.x, self.y)

  self.m = sum_xy / sum_x_sq
  self.b = y_mean - self.m * x_mean

def predict(self, x_new):
  validate_data([x_new])
  return self.m * x_new + self.b
```



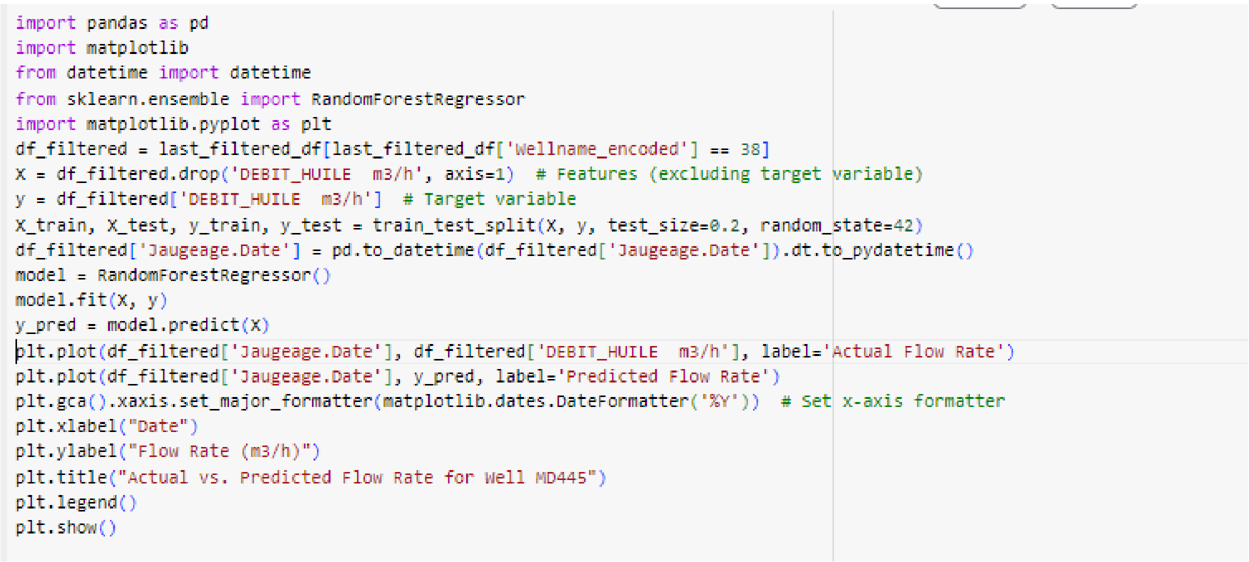
```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
from sklearn.ensemble import RandomForestRegressor, GradientBoostingRegressor
from sklearn.svm import SVR
import xgboost as xgb

def evaluate_models(df, target_col):
  X = df.drop(target_col, axis=1)
  y = df[target_col]
  is_categorical = not pd.api.types.is_string_dtype(y)
  X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
  regression_models = {
      'Random Forest Regressor': RandomForestRegressor(),
      'Gradient Boosting': GradientBoostingRegressor(),
      'SVR': SVR(),
      'XGBoost': xgb.XGBRegressor()
  }
  results = {}
  for model_name, model in regression_models.items():
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    mse = mean_squared_error(y_test, y_pred)
    results[model_name] = mse
  best_model_name = min(results, key=results.get)
  best_model_score = results[best_model_name]
```



```
  return results, best_model_name, best_model_score
results, best_model_name, best_model_score = evaluate_models(numerical_cols, 'DEBIT_HUILE  m3/h')

print(f"Best performing model: {best_model_name}")
print(f"Performance metric: {best_model_score}")
print(f"All model performances: {results}")
```



```
well_name_counts = last_filtered_df['Wellname_encoded'].value_counts()

# Print the counts
print(well_name_counts)
```



```
import pandas as pd
import matplotlib
from datetime import datetime
from sklearn.ensemble import RandomForestRegressor
import matplotlib.pyplot as plt
df_filtered = last_filtered_df[last_filtered_df['Wellname_encoded'] == 38]
X = df_filtered.drop('DEBIT_HUILE  m3/h', axis=1)  # Features (excluding target variable)
y = df_filtered['DEBIT_HUILE  m3/h']  # Target variable
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
df_filtered['Jaugeage.Date'] = pd.to_datetime(df_filtered['Jaugeage.Date']).dt.to_pydatetime()
model = RandomForestRegressor()
model.fit(X, y)
y_pred = model.predict(X)
plt.plot(df_filtered['Jaugeage.Date'], df_filtered['DEBIT_HUILE  m3/h'], label='Actual Flow Rate')
plt.plot(df_filtered['Jaugeage.Date'], y_pred, label='Predicted Flow Rate')
plt.gca().xaxis.set_major_formatter(matplotlib.dates.DateFormatter('%Y'))  # Set x-axis formatter
plt.xlabel("Date")
plt.ylabel("Flow Rate (m3/h)")
plt.title("Actual vs. Predicted Flow Rate for Well MD445")
plt.legend()
plt.show()
```