RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
Ministère de L'Enseignement Supérieur et de la Recherche Scientifique
ECOLE NATIONALE POLYTECHNIQUE
Département de Génie Minier

End-of-study project dissertation for obtaining the State Engineer's degree in Mining Engineering

Theme

Predictive and Comparative Study of Petrophysical Parameters Based on AI

Realized by:
IMADALOU Karine Anaïs
MIMOUNI Aya Fella

Under the supervision of:
Mr. Larouci CHANANE (MAA)
Mr. Aziz KHELALEF (Geophysicist)

Publicly presented and defended on June 21st 2025, in front of the jury composed of :

President	M.	Sami Yahyaoui	Professor	at ENP
Examiner	M.	Arezki Akkal	Professor	at ENP
Supervisor	M.	Larouci Chanane	Senior Lecturer	at ENP
Co-Supervisor	M.	Aziz Khelalef	Senior Petroleum Geophysic Engineer	at SONATRACH

ENP 2025

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
Ministère de L'Enseignement Supérieur et de la Recherche Scientifique
ECOLE NATIONALE POLYTECHNIQUE
Département de Génie Minier

End-of-study project dissertation for obtaining the State Engineer's degree in Mining Engineering

Theme

Predictive and Comparative Study of Petrophysical Parameters Based on AI

Realized by:
IMADALOU Karine Anaïs
MIMOUNI Aya Fella

Under the supervision of:
Mr. Larouci CHANANE (MAA)
Mr. Aziz KHELALEF (Geophysicist)

Publicly presented and defended on June 21st 2025, in front of the jury composed of :

President	M.	Sami Yahyaoui	Professor	at ENP
Examiner	M.	Arezki Akkal	Professor	at ENP
Supervisor	M.	Larouci Chanane	Senior Lecturer	at ENP
Co-Supervisor	M.	Aziz Khelalef	Senior Petroleum Geophysic Engineer	at SONATRACH

ENP 2025

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
Ministère de L'Enseignement Supérieur et de la Recherche Scientifique
ECOLE NATIONALE POLYTECHNIQUE
Département de Génie Minier

Mémoire de Projet de Fin d'Études en vue de l'obtention du diplôme d'Ingénieur d'État en Génie Minier

Thème

Étude Prédictive et Comparative des Paramètres Pétrophysiques Basée Sur l'Intelligence Artificielle

Réalisé par :
IMADALOU Karine Anaïs
MIMOUNI Aya Fella

Sous la direction de :
M. CHANANE Larouci (MAA)
M. KHELALEF Aziz (SR Géophysicien)

Présenté et soutenu publiquement, le 21/06/2025, devant le jury composé de :

Président	M.	Sami Yahyaoui	Professeur	à l'ENP
Examinateur	M.	Arezki Akkal	Professeur	à l'ENP
Encadrant	M.	Laarouci Chanane	MAA	à l'ENP
Co-Encadrant	M.	Aziz Khelalef	Ingénieur SR en Géophysique Pétrolière	à SONATRACH

ENP 2025

"Nothing of me is original.
I am the combined effort of everyone I have ever known."

— Chuck Palahniuk

To everyone who shaped me,
I am a mosaic of everyone I have encountered, of everyone I shared laughs with, of everyone who taught me even only a single word. I dedicate my work, to my family for being my support, for all this time, and never doubting me, even when I was barely confident myself, my mum, for being simply the strongest woman I know, my dad, for always making sure to provide with the best conditions for my success, to my two brothers, thanks to whom, I am trained to work under chaos, and be unphased, of course, in addition to their unconditional love. To my cousins, the sisters I need in my life. To every teacher I have ever had, who put effort into my growth. To my bestie, the instant match, you made these years incredible, and I strongly believe that the best is yet to come, thank you for always being around, on brightest and darkest times –but mostly brightest, as you make every day a good day– To all my friends, especially the two I met in middle school, for their untemporal love, and the one that saved my preparatory years, with lots of math and tea. I would also like to thank, me, for never giving up, for always pushing myself to the best, and for reaching it sometimes.
Lastly, to the unnamed chapter that quietly changed the entire story. Thank you for everything, you *remain always one unique figure.*
I am carrying a piece of each one of you, no matter where I am, nor where you are.
I am happy to be in the same sliver of cosmic time as you — out of billions of years and infinite possibilities, yet here we are.

Karine Anaïs IMADALOU

First and foremost,
I express my deepest gratitude and dedicate this work to my beloved parents: to my mother, a model of strength and intelligence, and to my father, whose quiet sacrifices made every step of my academic journey possible.

To my younger sister, thank you for always being there and for somehow finding a way to make me laugh, even during the most stressful moments; to my little brother, thanks for distracting me from the crushing weight of academic stress... mostly by being louder than my thoughts.

To my dear friends Ikram and Meriem, from high school until now, thank you for growing with me through this journey.

To my friends Raouf, Sonia, Ilhem, Insaf, and Soraya, thank you for the wonderful moments and your friendship along the way.

I would also like to extend my sincere thanks to all the teachers and professors who have guided me, challenged me, and played a vital role in shaping the person I've become.

To my friend the Engineer Raouf Boudia, thank you for your help and support; to the Engineer Idriss Merah, thank you for patiently introducing me to LaTeX and saving my computer from a paperweight fate!

To my best friend and peer, Karine, for being a constant pillar of strength throughout these years. Your determination and ambition have inspired me beyond words, and I'm grateful to have walked this path alongside you.

And finally, to the one who stayed up, showed up, and never gave up!
Thank you all.

Aya Fella MIMOUNI

Acknoledgement

This thesis marks the conclusion of a significant chapter in our academic journey, and it is with immense pride and gratitude that we reflect on the path that brought us here, and to everyone who contributed directly or indirectly to this project.

First and foremost, we are sincerely grateful to our academic supervisor, Mr. CHANANE Laarouci, from the National Polytechnic School, for his trust, his exemplary rigour, and his invaluable guidance throughout this project. His mentorship, insight, and availability have been key pillars in the development of our work. In addition to our head of department, M. YAHYAOUI Sami, for creating a suitable environment for thriving.

We would like to extend our most sincere thanks to the president of the jury, Mr. YAHYAOUI Sami, Professor at the National Polytechnic School and Head of the Mining Engineering Department, as well as to the examiner, Mr. AKKAL Arezki, Professor at the National Polytechnic School. We are grateful to them for accepting to be part of the review committee and for dedicating their time to a thorough evaluation of our work. Their constructive feedback, expertise, and teaching over the past three years have greatly contributed to our education and development.

Also, we would like to express our deep gratitude to all the professors of the Mining Engineering Department. Their dedication, pedagogical excellence, and inspiring professional paths have greatly shaped our perspective and ambitions. Their impact will remain with us throughout our careers.

Finally, we would like to express our deep gratitude to SONATRACH for welcoming us during the entire duration of our internship. We are particularly indebted to Mr. KHELAFEL Aziz, our industrial supervisor, whose infectious enthusiasm, patience, and clarity transformed every learning moment into a memorable experience. We also thank the entire team on site for their kindness and generous support.

Contents

List of Figures

List of Tables

Listings

List of Acronyms

1D One-Dimensional

2D Two-Dimensional

AI Artificial Intelligence

ATP Adenosine Triphosphate

BHA Bottom Hole Assembly

CNN Convolutional Neural Network

CT Computed Tomography

DL Deep Learning

DTP Compressional Sonic Transit Time

EDA Exploratory Data Analysis

EOR Enhanced Oil Recovery

FC Fully Connected

GBDT Gradient Boosting Decision Tree

GEN AI Generative Artificial Intelligence

GMM Gaussian Mixture Model

GR Gamma Ray

HI Hydrogen Index

IQR Interquartile Range

KNN K-Nearest Neighbors

LR Logistic Regression

LWD Logging While Drilling

MAE Mean Absolute Error

MDP Markov Decision Process

ML Machine Learning

MLP Multilayer Perceptron

MSE Mean Squared Error

MW Mud Weight

MWD Measurement While Drilling

NLP Natural Language Processing

NPHI Neutron Porosity

PCA Principal Component Analysis

PFE Photoelectric Effect

PHIE Effective Porosity

POTA Potassium Concentration

REL Rectified Linear Unit

ResNets Residual Networks

RF Random Forest

RHOB Bulk Density

RL Reinforcement Learning

ROP Rate of Penetration

SCAL Special Core Analysis

SEM Scanning Electron Microscope

SVM Support Vector Machine

SWAP Shale Water Absorption Parameter

TAGI Truly Autonomous Gradient Inference

TAGS Tagged Analysis of Geological Samples

THOR Thorium Concentration

TOC Total Organic Carbon

URAN Uranium Concentration

VCL Volume of Clay

VSH Volume of Shale

XRD X-ray Diffraction

ملخص

تهدف هذه الدراسة إلى استكشاف استخدام التعلم الآلي كأداة قوية من أدوات الذكاء الاصطناعي لتطوير خوارزمية مناسبة لتقدير والتنبؤ بثلاثة معايير بتروفيزيائية أساسية، وهي حجم الطين (V_{CL})، والمسامية الفعالة $(PHIE)$، وتشبع الماء (S_W)، وذلك انطلاقاً من بيانات السجلات الخام لعدة آبار إنتاجية في حوض بركين. تكمن الإشكالية الرئيسية في التنبؤ الدقيق بتشبع الماء. تمت مقارنة عدة نماذج، من بينها $XGBoost$، MLP وCNN. وقد أظهرت النتائج المتحصل عليها، خاصة باستخدام نموذج CNN، فعالية ممتازة لتقنيات التعلم الآلي، حيث تم تحقيق معامل تحديد إجمالي قدره $R^2 = 0{,}81$ لتشبع الماء، وهو المعيار الأصعب من حيث التقدير.

الكلمات المفتاحية: التعلم الآلي، الذكاء الاصطناعي، التنبؤ، حجم الطين، المسامية الفعالة، تشبع الماء، السجلات، الخزانات، حوض بركين.

Résumé

L'objectif de cette étude est d'explorer l'utilisation de l'apprentissage automatique comme un outil puissant d'intelligence artificielle pour développer un algorithme permettant d'estimer et de prédire trois paramètres pétrophysiques essentiels : le volume d'argile (V_{CL}), la porosité effective $(PHIE)$ et la saturation en eau (S_W), à partir de données diagraphiques brutes issues de plusieurs puits de production du bassin de Berkine. La principale problématique réside dans la prédiction précise de la saturation en eau. Plusieurs modèles ont été comparés, notamment **XGBoost**, **MLP** et **CNN**. Les résultats obtenus, en particulier avec **CNN**, démontrent une efficacité remarquable des techniques d'apprentissage automatique, avec un coefficient de détermination global de $R^2 = 0{,}81$ pour la saturation en eau, paramètre le plus difficile à estimer.

Mots-clés : apprentissage automatique, intelligence artificielle, prédiction, volume d'argile, porosité effective, saturation en eau, diagraphies, réservoirs, bassin de Berkine.

Abstract

This study aims to explore the use of machine learning as a powerful artificial intelligence tool to develop an algorithm capable of estimating and predicting three essential petrophysical parameters: clay volume (V_{CL}), effective porosity $(PHIE)$, and water saturation (S_W), based on raw log data from several production wells in the Berkine Basin. The main challenge lies in the accurate prediction of water saturation. Several models were compared, including **XGBoost**, **MLP**, and **CNN**. The results obtained, especially with the **CNN** model, demonstrate the high efficiency of machine learning techniques, achieving a global determination coefficient of $R^2 = 0.81$ for water saturation, which is the most complex parameter to predict.

Keywords: machine learning, artificial intelligence, prediction, clay volume, effective porosity, water saturation, logs, reservoirs, Berkine Basin.

General Introduction

Petrophysical analysis plays a pivotal role in the exploration of hydrocarbons, by giving key insights into the rock properties, the fluid characteristics and behavior and reservoir performance. With the rise of new technologies applied to the field of Earth sciences, the integration of artificial intelligence (AI) into petrophysical analysis represents a promising opportunity to improve the interpretation and predictive modeling of well log data. AI-assisted petrophysical analysis has the high potential to refine reservoir evaluation, reduce uncertainties, and optimize decision-making processes in the oil and gas industry.

Despite significant innovations in logging tools and data acquisition methods, the interpretation of petrophysical parameters remains a complex task for the petroleum professionals. Historically, the oil and gas industry has always relied on conventional well log analysis techniques, which involve analysing logs to detect deviations from baseline trends. These deviations often indicate variations in either lithology, fluid saturation, porosity, or borehole conditions. The objective behind such analyses is to identify depth intervals of interest that require further investigation, for their hydrocarbon potential. However, these traditional approaches meet a number of limitations that can compromise the accuracy and efficiency of the reservoir characterization process.

The difficulties faced during the training of the models stem from multiple factors, such as the inherent heterogeneity of the subsurface, with layered formations, fractures, and uneven fluid distributions, the subjective nature of visual interpretation and the variation in data quality. Moreover, well log data can be compromised by physical factors such as the limitations in the tool resolution, the environmental noise referring to the signal disturbances due to borehole conditions (e.g., mud invasion, borehole rugosity, temperature and pressure variations) or tool-related effects, and the borehole irregularities, all of which leads to inaccurate interpretation of the graphs. These traditional approaches, dependent on empirical correlations and subjective expertise, are prone to human error and bias. This highlights the urgent demand for streamlined, automated alternatives to enhance petrophysical analysis. AI and machine learning (ML) techniques present an interesting pathway to address these challenges by allowing faster and more accurate log interpretation. Algorithms capable of pattern recognition, anomaly detection, and predictive modeling can help in identifying important reservoir parameters while minimizing human error.Furthermore, they enable the combination of multidisciplinary data—such as core samples, seismic attributes, and production history—into AI-driven workflows to

enhance the responses, thus, the reliability of reservoir assessments, to improve hydrocarbon exploration, and ultimately, maximize profitability by economizing time, energy and resources. It is safe to say that the future, undoubtedly, lies in associating domain expertise with AI power.

This thesis is structured in two main parts: a theoretical component and a practical one. The theoretical part introduces the fundamental concepts of petrophysics and artificial intelligence, providing the necessary background to understand both the problem treated and the proposed approach. The practical part presents the methodology for the selected models being, *Linear Regression, MLP, XGBoos*t and *CNN*, covering from exploratory data analysis (EDA), the modeling pipeline, the obtained results, to a pilot log generated from the best performing model CNN.

Part One
Theoretical Framework

Chapter 1

Fundamentals of Petrophysical Parameters

Objective: *establish the theoretical and practical foundations of petrophysics with a view to setting up an intelligent platform for applications of the Artificial Intelligence tool.*

1.1 Definition and Scope

Petrophysics, from the Greek petra "*rock*" and physis "*nature*" [1] is a sub-discipline of the geosciences, also known as the main branch of petroleum geology and geophysics, which is concerned with the study of the physical and chemical properties of rocks and their interactions with fluids, such as water, hydrocarbons, and gases, in subsurface geological formations. Integrating physics, chemistry, geology and engineering principles allows us to quantitatively analyse the petrophysical characteristics of reservoir rocks, providing key information about the interconnected network of pore spaces and the distribution and circulation of fluids in these spaces across this related network [2].

Although the term "*petrophysics*" may have been used informally in earlier industrial circles. Its first well-documented appearance in a published article is attributed to Gustavus Archie's 1950 book [3], "*Introduction to Petrophysics of Reservoir Rocks*". However, Archie's empirical models (e.g. the Archie equation) clearly reinforced the scientific rigour of this study field [4].

The adoption of scientific terms accelerated in the mid-20th century, driven by the need to quantify the petrophysical properties of reservoirs. Gustavus Archie played a central role in this effort. In this seminal work, he introduced a revolutionary perspective to oil and gas exploration, highlighting the importance of studying a series of physical properties: porosity, permeability, capillary pressure, clay volume, water and hydrocarbon saturation in relation to electrical resistivity, fluid properties, natural radioactivity potential and their interrelationships in reservoir rocks, and considering them together when interpreting their interactions in order to detect and evaluate more effectively the presence of zones of interest corresponding to bearing layers containing hydrocarbons in economically exploitable grade [3].

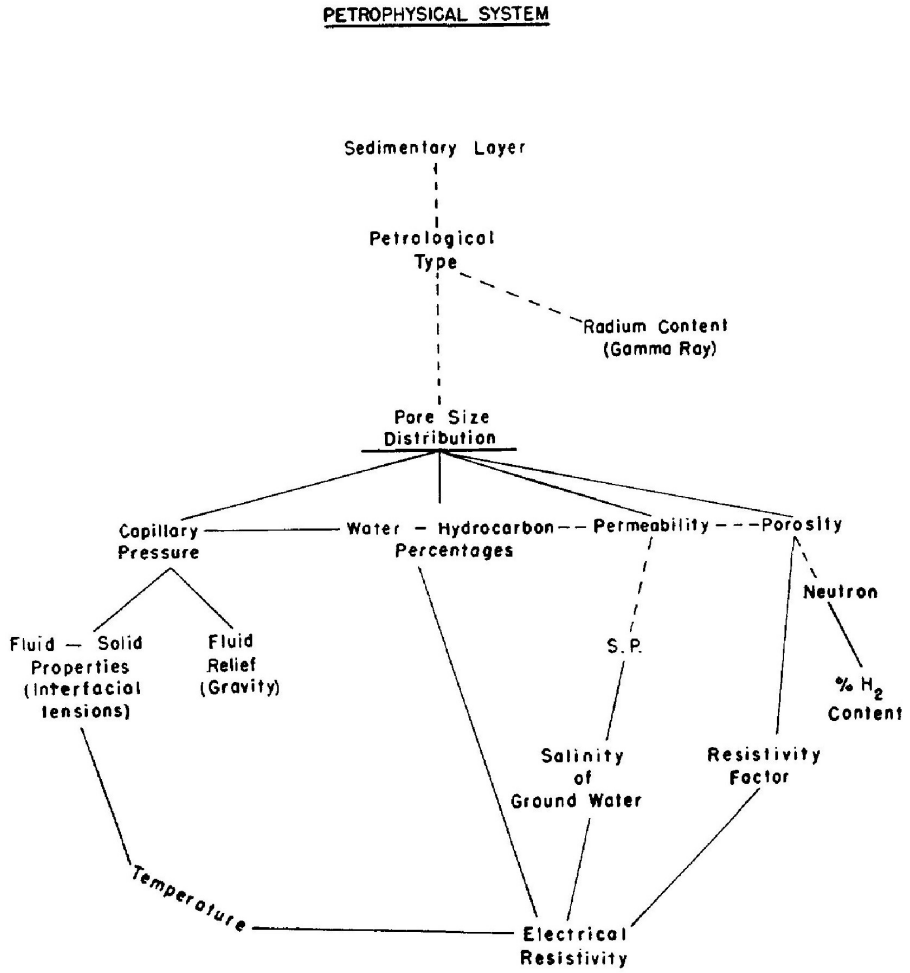


Figure 1.1: Scheme of petrophysical system [3]

In the "*petrophysical system diagram*", he schematized the relationships between the main physical properties of the reservoir rocks and the factors that influence them. His framework has become the basis for the modern assessment of reservoir petrophysics.

The core objectives of petrophysics include the following:

- **Rock Property Analysis:** Measurement of porosity, permeability, and saturation.
- **Reservoir Characterization:** Assessing the storage capacity, fluid flow, and economic potential of either hydrocarbon or geothermal reservoirs.
- **Fluid Behaviour Prediction:** Modeling fluid dynamics through rock formations under different conditions of pressure and temperature.

Petrophysical interpretation plays a crucial role in most subsurface work, as it implies the integration of the pore system, fluid and flow properties, and their interrelationships, in order to identify and assess geological formations. This analysis is fundamental to exploration; enabling hydrocarbon reservoirs, aquifers, and seals to be characterized as follows[5]:

- **Characterizing Reservoir Rocks:** Petrophysics determines the essential properties required to assess reservoir quality and potential.

- **Evaluating Fluid Distribution:** Petrophysics is used to predict the production rates and identify zones of interest, by analysing the fluid distribution within the pore system of rocks.
- **Integrating Data for Reservoir Models:** Building accurate subsurface models based on the integration of different relevant data sources. This integration helps to make enlightened decisions about drilling operations, production, and reservoir management in general.
- **Supporting Reservoir Management:** Applied petrophysics provides information on in situ bed limits, net pay and fluid contacts, which is needed for economic decision-making and efficient resource management.
- **Reducing Uncertainty and Risk:** Through quantitative analysis, petrophysics reduced overall uncertainty in reservoir evaluation, mitigating risks and improving the efficiency of the upstream industry[6].
- **Enhancing reservoir surveillance:** Operational petrophysics focuses on real-time data acquisition, analysis, and interpretation of downhole parameters, to support immediate decision-making processes.

This explains the wide-range applications of petrophysics, it is indispensable during every step in the oil and gas industry. Its main industrial applications include:

- **Reservoir characterization and evaluation:** The quality and potential of reservoirs are assessed by determining their keys properties, logging technologies and core analysis are used to identify production zones. Enhanced petrophysical evaluation through machine learning and well logging data in an Iranian oil field [7].
- **Well Logging and Formation Evaluation:** Well logs (e.g. resistivity, neutrons, density, and gamma rays) and borehole images provide in situ measurements that enable real-time decisions to be made during drilling and completion operations, facies analysis and thin layer identification[8].
- **Production Optimization and Reservoir Management:** Monitoring fluid movement[9], evaluating enhanced oil recovery (EOR) techniques (e.g., water flooding, gas injection)[10], and managing reservoir performance over time. Theoretical advancements in operational petrophysics for enhanced reservoir surveillance
- **Broader Industrial Applications:** Beyond the oil and gas industry, petrophysics is being applied in the energy transition sector as in geothermal energy[11] (characterizing heat reservoirs), water resource management (aquifer assessment), mining (orebody characterization) and carbon capture and storage[12] (CO2 sequestration site assessment) by using practically the same processes as for hydrocarbon exploration.

1.2 Key Petrophysical Parameters

The fundamental petrophysical parameters required to characterize a rock are mainly its porosity, permeability, and fluid saturation. These measurements can be obtained from petrophysical loggings, core analysis, or pressure measurements [13, p. 57].

1.2.1 Porosity (ϕ)

Porosity (Phi or ϕ) is defined as the ratio of the pore volume within a rock to its bulk volume. It is strongly affected by the uniformity of grain shape and size, sorting, and the degree of consolidation. Porosity is dimensionless and usually expressed as a percentage:

$$\phi = \frac{V_p}{V_b}$$

where V_p is the pore volume and V_b is the bulk volume.

1.2.1.1 Primary vs. Secondary Porosity

In the context of sedimentary geology, primary porosity refers to the original pores spaces formed from the lithification process, as a result of mineral precipitation or partial dissolution during sediment deposition of the sand beds in the early stages of rock formation. However, this type of porosity does not necessarily have a significant impact on the rock's permeability, i.e, its ability to let the fluid flow.

On the other hand, if the voids occurred after the rock formation through geological processes like diagenesis, catagenesis, geodynamic stresses or dissolution, it is referred to as *secondary* or porosity. This type of voids is more important in carbonate reservoir rocks. Nevertheless, both can generally be found in the same rock matrix [14, 2, p. 123].

1.2.1.2 Total vs. Effective Porosity

Porosity can also be classified as either total porosity (ϕ_T) or effective porosity (ϕ_E), depending on the connection between pores. In fact, during sedimentation, some of the spaces initially formed become isolated due to geological processes such as compaction, while others remain interconnected. The total (absolute) porosity is the ratio of the total space contained in a bulk volume to the total volume, whereas effective porosity refers to spaces that are interconnected and have the capacity to conduct fluids. The equation then becomes the ratio of the effective pores or connected pores to the bulk volume.

- **Total porosity** (ϕ_T) includes all pore spaces, whether connected or isolated.
- **Effective porosity** (ϕ_E) includes only the interconnected pores contributing to fluid flow.

1.2.2 Permeability (k)

Permeability is another parameter, related to porosity. It describes the ability of the rock to allow fluid movement through its interconnected pores, related to its effective porosity. It depends on grain size, shape, sorting, consolidation cementation and clay content. The type of clay or cementing material between the grains determines permeability, especially when wet, as some are known to expand under the effect of water (such as smectites), forming an impermeable barrier, leading to a significant reduction of permeability [15].

1.2.2.1 Darcy's Law and Modifications

The permeability is measured in *Darcies*, referring to the French engineer Henry Darcy (1803–1858), who formulated a simple empirical equation describing the fluid flow through porous media, as a function of flow rate and differential pressure.

$$Q = -kA\frac{dH}{dL}$$

Where:

- Q: Volumetric flow rate (m^3/s),
- k: Permeability (m/s),
- A: Cross-sectional area (m^2),
- $\frac{dH}{dL}$: Hydraulic gradient (change in head per unit length).

The negative sign indicates flow occurs from high to low hydraulic head.

One *Darcy* corresponds to a flow of 1 cm^3/s of 1 cP (centiPoise) fluid through 1 cm^2 of cross-section under a pressure gradient of 1 atm/cm.

The law assumes laminar flow (*Reynolds number* < 1), a homogeneous isotropic medium, incompressible Newtonian fluid, and no phase changes [16].

Alternative Form in Petroleum Engineering Darcy's equation has been used as a starting point for modifications and correction, to include several parameters such as velocity, with a view to generalize the application of the formula to reservoir rocks [17].

One of the suggested forms for petroleum measurement is :

$$\nu = -\frac{k}{\mu}\frac{dP}{dx}$$

Where:

- ν : Darcy velocity or apparent fluid velocity (m/s)
- k : Permeability of the medium (m^2)

- μ : Dynamic viscosity of the fluid (Pa·s)
- $\frac{dP}{dx}$: Pressure gradient (Pa/m)
- x : Distance in the direction of flow (cm) (always positive).

Absolute vs. Effective Permeability

- **Absolute permeability** refers to the permeability of a rock when its pores are 100% saturated with a single fluid.
- **Effective permeability** (k_o, k_w, k_g) (effective permeability of oil, gas and water respectively) refers to the permeability of a specific fluid in the presence of others [13, 14, p. 83, 129].

Relative Permeability Relative permeability is defined as the ratio of effective permeability at a given saturation to absolute permeability, expressed as a percentage or fraction:

$$k_{\text{rel}} = \frac{k_{\text{eff}}}{k}$$

1.2.3 Fluid Saturation (S_w, S_o, S_g)

Fluid saturation assessment aims to quantify the fraction of pore space occupied by a fluid phase in the reservoir rock, which is essential for assessing hydrocarbon potential. S_w, S_o and S_g are respectively the standard notation for water, oil and gas saturation.

1.2.3.1 Archie's equation

Archie conducted a number of experiments using clean, clay-free sandstone samples saturated with a brine of resistivity noted R_w [18]. In his seminal work, he showed the inverse relationship between the resistivity of the brine saturating the rock and the resistivity of a clean formation [14], leading to the establishment of a quantitative relationship between porosity (ϕ), rock resistivity (R_o), and hydrocarbon saturation of reservoir rocks. He then suggested an empirical approach that estimates the water saturation of clean sands as follows:

$$S_w = \left(\frac{aR_w}{R_t \phi^m} \right)^{1/n}$$

Where:

- R_w: Formation water resistivity,
- R_t: True formation resistivity,
- ϕ: Porosity,
- a, m, n: Empirical constants, to be determined from core analysis [19].

Limitations of Archie's Equation Archie revolutionised reservoir characterization, however, still carried some limitations:

- **Empirical Basis :** Archie carried out his experimental work on clean, consolidated sandstone reservoirs with high humidity and clay-free. His parameters (a, m, n) are empirical and formation-specific, and must therefore be determined for each reservoir rock.
- **Non-Archie Pore Geometries :**The equation assumes intergranular pore spaces; it is not valid for rocks with complex pore geometries.
- **Presence of Conductive Minerals :** Rocks containing conductive minerals violate *Archie*'s assumptions, as they modify the resistivity values, leading to an underestimation of water saturation.
- **Fresh Formation Waters :** Archie's equation is less accurate when the formation water is very fresh (low salinity). The lack of electrolytes increases its resistivity, which becomes similar to that of hydrocarbons.
- **Heterogeneous and Shaly Formations :** In shale sands or heterogeneous reservoirs, Archie's equation requires modifications or alternative models to adapt to this type of complex lithology.

1.2.3.2 Wettability

Wettability is a term used to describe the tendency of a solid to be wetted by one fluid rather than another, by spreading or adhering to its surface. It is determined by the balance between the adhesive forces (attraction between the fluid and the surface) and the cohesive forces (attraction within the fluid itself).

In a water–brine–oil–rock system, the water tends to cover the rock surface and fill the smallest pores, while the oil occupies the largest pore spaces.

If a rock is water-wet and initially saturated with oil, it will imbibe water when exposed to it, displacing the non-wetting fluid—in this case, oil—from the small pores.

In contrast, if the rock is oil-wet, it will imbibe oil even when it is saturated with water, pushing water out of the rock.

Wettability can vary depending on how the brine interacts with the rock surface.

- **Water-wet** $\rightarrow$ absorbs water
- **Oil-wet** $\rightarrow$ absorbs oil
- **Mixed-wet** $\rightarrow$ no strong preference (about 50/50)

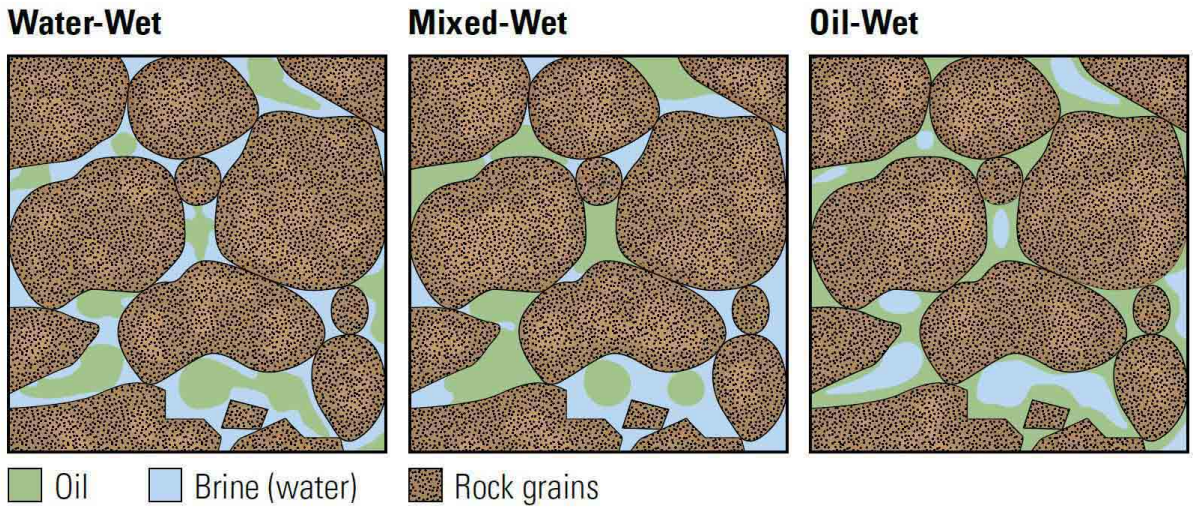


Figure 1.2: In water-wet pores, oil stays in the center; in oil-wet pores, it coats the surfaces. In mixed-wet cases, oil occupies some surfaces, but stays centered in water-wet pores. All three cases may show similar fluid saturations. [20]

Wettability is an important parameter in petroleum engineering, as it strongly affects oil recovery. It influences oil production rates, the water/oil ratio after water breakthrough, the effectiveness of enhanced oil recovery methods, and the amount of oil remaining in the reservoir at the time of abandonment.

For instance, a *water-wet system* allows **greater** primary oil recovery, whereas as the system becomes *more oil-wet*, oil recovery becomes more **challenging** at any given amount of water injected into the rock [14].

The role of reservoir engineers is therefore to plan Enhanced Oil Recovery (EOR) processes to modify the wettability to water-wet in order to capture the oil coating the solid surface. Alternatively, in some cases, making the formation more oil-wet may lead to better ultimate oil extraction, depending on the reservoir characteristics [20].

1.2.4 Clay and Shale Volumes

The volume of clay is an important petrophysical parameter, denoted as V_{cl}. It is a valuable measure considering the fact that the clay content modifies the rock's physical properties, such as *the effective porosity*, *permeability*, and *electrical conductivity*.

Physically, it describes the proportion of clay present in a rock matrix, quantified as a fraction or percentage of the rock volume occupied by this mineral.

The term *shale volume* (V_{sh}) is often misused to refer to the volume of clay, although it is a closely related concept, the distinction is that V_{sh} refers to the volume of the shale-filled matrix, containing up to 70% of clay, plus a minor proportion of hydrogen-free silt-sized particles with quartz (SiO2) and some lithic minerals such as feldspars and plagioclases, forming the common shale minerals (e.g., kaolinite, illite, smectite) [2].

Clay minerals have a strong affinity for water, which can lead to an overestimation of water saturation (S_w). Their tendency to swell in the presence of water also reduces porosity and permeability. In addition, clay affects some well log responses, particularly

those sensitive to the presence of radioactive elements and hydrogen—namely, gamma-ray and neutron logs—due to the natural radiation emitted by shale formations from *thorium* and *potassium* associated with clay minerals, in addition to *uranium* which is often fixed by phosphatic or organic matter [21].

V_{cl} values can typically range from close to 0% in clean (clay-free) sandstones or carbonates to close to 100% in shale or clay-rich formations.

1.2.4.1 V_{cl} Estimation Methods

The estimation of the clay volume (V_{cl}) is based on the results of various well logs related to radioactivity and hydrogen occurrence.

- **Gamma ray method:** The Gamma ray logging measures the natural radioactivity of the formations, due to the presence of radioactive elements in clay-rich rocks. The V_{sh} (volume of shale) is estimated by normalizing the gamma ray readings between clean sand and shale, using the following formula :

$$V_{\text{sh}} = \frac{GR_{\text{log}} - GR_{\text{clean}}}{GR_{\text{shale}} - GR_{\text{clean}}}$$

- **Density-neutron log separation:** The difference between neutron porosity and density readings may indicate the presence of clay minerals containing formation-water which appears as additional hydrogen on the neutron logs, resulting in apparent increases in porosity, while the density log remains less affected; in fact, this difference indicates the presence of clay.

- **Nuclear Magnetic Resonance (NMR):** This is used to distinguish between free fluids and formation-water bound to clay, which gives an indication of the amount of clay present in the formation.

1.2.5 Other Parameters

In addition to the petrophysical parameters listed above, other measurements provide complementary information on reservoir characteristics, contributing to fluid identification, lithology estimation, and mechanical property assessment [22]. These include resistivity, bulk density, and acoustic velocity.

Resistivity Resistivity, expressed in ohm-m, describes the extent to which the formation is able to resist the flow of electric current. Sedimentary formations filled with saline water contain electrolytes capable of conducting an electric current and therefore have low resistivity values, due to the ions present in the brine, whereas zones filled with hydrocarbons show a higher resistivity. This behaviour is dictated by Archie's law, which links resistivity to porosity and water saturation in a clean formation [18]. Nevertheless, factors such as clay content and formation water considerably affect resistivity readings and must be taken into account during interpretation [23].

Bulk Density Bulk density, also known as apparent density, is the measure of the rock mass per unit volume, including the inherent interstitial spaces. It is generally measured from logs and is essential for identifying lithology and estimating porosity. When the density values of the rock matrix and fluid are known, bulk density can be used to calculate porosity using the following equation:

$$\phi = \frac{\rho_{ma} - \rho_b}{\rho_{ma} - \rho_f}$$

where:

- ϕ: Porosity (fraction)
- ρ_{ma}: Matrix density (g/cm^3)
- ρ_b: Bulk density (g/cm^3)
- ρ_f: Fluid density (g/cm^3)

[24]

Acoustic Velocity Acoustic velocity, or sonic velocity, is a measure of the propagation speed of sound waves through a medium. In the formation, it is influenced by the lithology, pores, and fluid content of the rock. The velocity is obtained by converting the travel time of compressional waves (Δt) acquired by sonic tools, which is then correlated with porosity using models such as Wyllie's time-average equation [25]:

$$\frac{1}{\Delta t} = \frac{\phi}{\Delta t_f} + \frac{1 - \phi}{\Delta t_{ma}}$$

where:

- Δt: Measured travel time (µs/ft)
- Δt_f: Travel time in fluid (µs/ft)
- Δt_{ma}: Travel time in matrix (µs/ft)

Chapter 2

Acquisition and Pre-processing of Petrophysical Data

Objective: *Introduce the conventional methods used in petrophysical data acquisition, highlight their practical limitations and identify common data issues.*

2.1 Traditional Measurement Methods

The data used in petrophysics originates from various sources and is collected at different stages of the well life cycle, either during the drilling, the completion or production; however, the tools may differ. The primary source of this data is: *well logging*, *core analysis* and *well testing*. These methods provide either direct characterization, as it is for the case of lab tests, or indirect, like log results.

2.1.1 Well Logging

Well logging tools are the specialized downhole hardware run in the wellbore, to make logs; they are equipped with sensors for different measurements. The type, diameter and length of these instruments depend on the stage of the well; they are divided into two main categories: *drilling logging tools* and *wireline logging tools*.

2.1.1.1 Drilling Logging

During the drilling process, engineers operate without a direct visibility into wellbore conditions, to address this challenge, Measurements-while-drilling (MWD) logging tools, or also known as logging while drilling (LWD) tools, are used to provide the crucial insight required to complete the task safely, while gathering initial information about the geological formation. The data obtained can be either physical, such as cuttings, operational, such as rate of penetration (ROP) or mud weight (MW), or formation evaluation data, from (LWD) sensors measurements, including gamma ray, density, neutron porosity and resistivity data, displayed as tracks on mud-logs or as digital outputs [2].

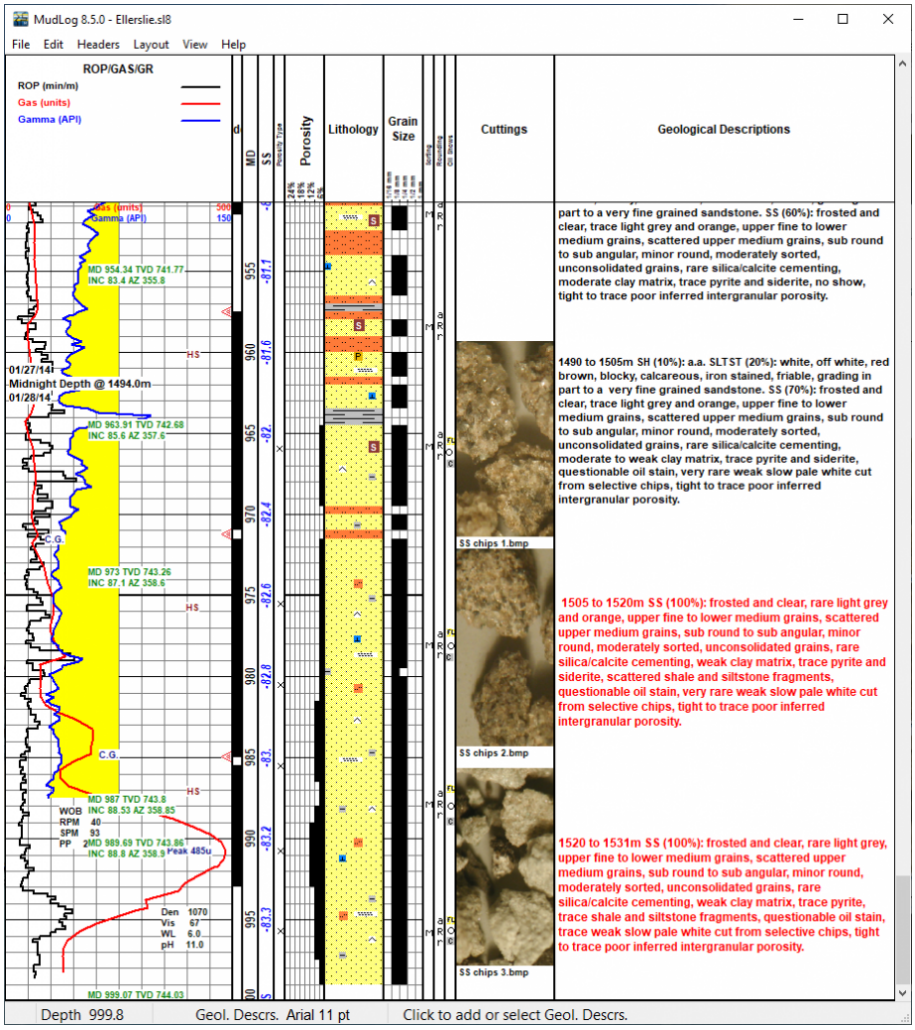


Figure 2.1: Excerpt from a basic mud log [26].

Logging While Drilling (LWD) measurements are acquired near the drill bit as part of the bottom hole assembly (BHA) using self-contained tools. The data points are recorded downwards (continuously while drilling) and referenced against time-while-drilling and then processed to be converted to depth-base measurements. The distance between the position of the tools and the bit may limit the use and effectiveness; hence, it is crucial to plan the type and order of the tool string assembly. Then, the data can be stored in the tool's memory during drilling LWD to be retrieved later to the surface.

Although unreliable and expensive in some cases, this type of technology has the advantage of measuring properties of a formation before drilling fluids invade the wellbore. In addition, these data are sufficient in the case of exploratory wells, as they form the primary evaluation of the downhole in terms of lithology, presence of hydrocarbon and even water saturation. The physical samples, of cuttings and gas, are also a strong indicator of the stratigraphy and the source rock.

2.1.1.2 Wireline Logging

After drilling operations are completed, the wireline logging is performed in order to collect continuous measurements of wellbore properties. This type of logging involves lowering

tools equipped with sensors (sondes) into the wellbore using an armoured electrical cable that serves two main functions: *mechanical support of the tool's weight* and *ensuring real-time transmission* of power and data between the surface and the downhole tools, via various telemetry systems. During the operations, surface equipment is required, as the tool is lowered and raised using a motorized winch system mounted on the wireline unit near the rig floor, manipulated by operators to maintain proper speed in order to avoid tool sticking and high-quality acquisition. The wireline cable is counter-helically armoured, to prevent twisting and ensure resistance to high tensions and hostile conditions.

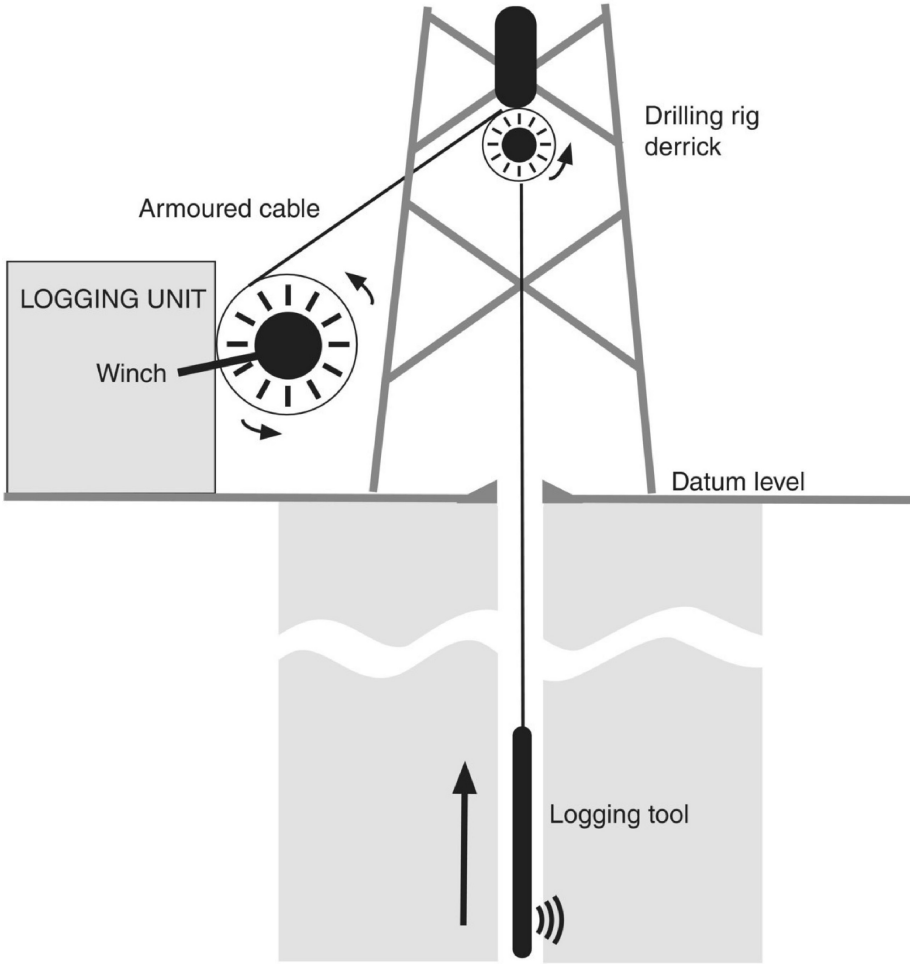


Figure 2.2: Schematic of a standard Wireline operation set-up [2].

Regarding shape, wireline tools are typically cylindrical, with a diameter ranging from 1.5 to 5 inches (3.8 to 12.7 cm), making them smaller and more manageable than the tools used for LWD. These tools are capable of measuring a wide range of parameters related to the formation and the borehole including, but not limited to electrical resistivity, to differentiate between hydrocarbon or water bearing zones ; acoustic velocity, that infers porosity and mechanical properties of the rock ; radioactive response, to estimate the volume of clay ; dimensional measurements, to modelise the well geometries (diameter, depth...), in addition to formation pressure and temperature, using specialized gauges and samplings.

The following tables summarize the most used Wireline Tools, their applications and relevant units.

Table 2.1: Common Wireline Logging Tools and Their Applications [2]

Tool / Log	**Physical Measurement**	**General Use**	**Applications**
GR (Gamma Ray)	Natural radioactivity	Lithology identification	Shale volume estimation, well correlation
DTP (Sonic Log)	Acoustic velocity	Porosity evaluation	Matrix porosity, lithology discrimination
AT (Array Tools)	Attenuation of electromagnetic waves	Resistivity proxy	Water saturation estimation via Archie's law
RHOB (Density)	Bulk density	Porosity evaluation	Total and matrix porosity, lithology discrimination
NPHI (Neutron)	Hydrogen index	Porosity evaluation	Total porosity, fluid type indicator
URAN (Uranium)	Uranium concentration	Lithology	Identification of radioactive anomalies
THOR (Thorium)	Thorium concentration	Lithology	Differentiation between clastic and carbonate facies
POTA (Potassium)	Potassium concentration	Lithology	Shale typing and sediment provenance analysis

Table 2.2: Common Log Tracks, Units, and Display Scales [2]

Log (Track)	**Measurement (Units)**	**Left**	**Right**
GR (1)	API units	0	150
SP (1)	Millivolts (mV)	–10	+10
CAL (1)	Inches (in)	6	16
BIT_SIZE (BS) (1)	Inches (in)	6	16
RES (2)	Resistivity – log scale (Ω·m)	0.2	200
SONIC (3)	Slowness (μs/ft)	140	40
DENS (2)	Bulk density (g/cm^3)	1.95	2.95
NEUT (2)	Limestone porosity units (p.u.)	0.45	–0.15
PEF (2)	Barns/electron (B/e)	0	10

Principles of density, neutron and resistivity logs

1. **Density log**

   The tool diameter typically ranges from 1.5 inches to 5 inches (3.8 cm to 12.7 cm), depending on the borehole size.

The log density measures the bulk density, which is both the density of the rock and the fluids contained in the pore spaces, symbolized by the letter ρ **(rho)** commonly denotes density. The typical scale for density ranges between 1.95 and 2.95 g/cm^3. In order to determine the porosity from the density tool, it is crucial to determine first, the density of the matrix and all the fluids it potentially contains.

The tool is mounted on a skid and a caliper arm to maximize the contact of the emitter side with the borehole wall. It uses a radioactive source (such as Cesium-137 or Cobalt-60), or a modern accelerator to emit gamma rays. The emitted rays interact with the electrons in the formation by Compton scattering, each collision causes an energy loss of the gamma particle, and the tool's two detectors, placed about 50cm from the source, measure the radiation of the returned scattered particles.

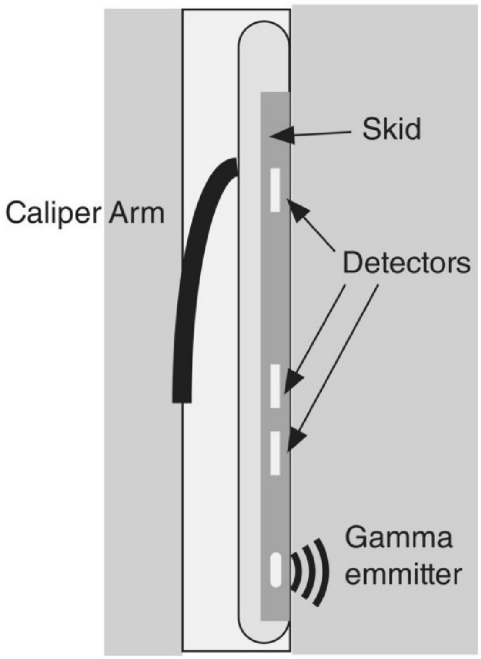


Figure 2.3: Schematic of a density tool [2].

The returned rays are divided into two categories, high-energy rays, related to electron density, used to calculate the rock bulk density, or low-energy rays, governed by the photoelectric effect (PFE), used to identify rock and fluid types.

2. **Neutron logging**

   Neutron logging measures the **hydrogen index (HI)**, which reflects the amount of hydrogen atoms in the rock, in order to estimate the formation porosity. The main sources of hydrogen in the subsurface are water and hydrocarbons; hence, the tool essentially detects fluid-filled porosity.

   The tool contains a neutron source (typically americium-beryllium) to emit fast neutrons, which collide with atoms in the rock. When they hit hydrogen atoms, their energy is quickly lowered due to their similar mass. After slowing down, the neutrons are absorbed by the formation and emit gamma rays, which are detected by the

tool. The more hydrogen, the more gamma rays, the higher porosity readings, gaseous zones contain less hydrogen than water and oil, which means a lower porosity reading, known as the gas effect. When neutron porosity appears lower than density porosity, it creates a "*crossover*" on the log, indicating a potential gas-bearing zone.

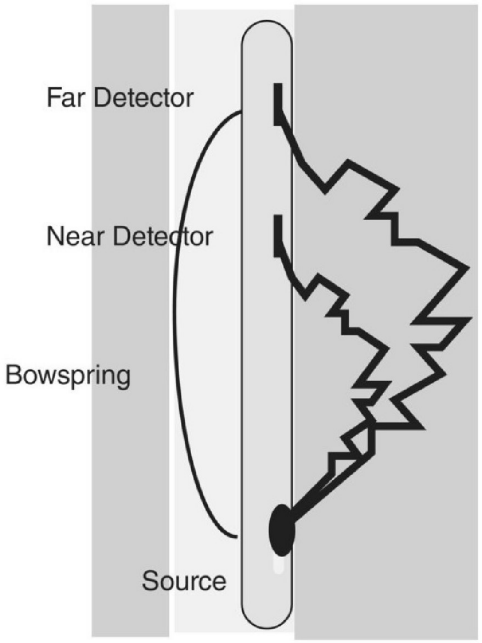


Figure 2.4: Schematic of a neutron tool [2].

The log is calibrated, based on a standard limestone response, being near zero, and displaying scales usually ranging from 0.45 to – 0.15 in limestone porosity units. However, it's important to correct, according to lithology for:

- Shales, which trap water in clay and overestimate porosity.
- Hydrocarbons and salt, which affect the HI and require corrections.

The **Hydrogen Index (HI)** is defined as:

$$HI = \frac{\text{Hydrogen atoms per unit volume of rock}}{\text{Hydrogen atoms per unit volume of pure water}}$$

This index serves as an indicator of porosity, but is not a direct measurement, as it also depends on lithology and fluid type, in addition to tool-specific corrections.

Neutron-density cross plot equations The density-neutron logging is a combined log that simultaneously records neutron and density porosity, providing valuable cross-validation for porosity estimation and lithology identification [27]. In some areas, porosities recorded on the logs differ for three reasons:

- Incorrect matrix density assumption in computing porosity logs.

- The gas effect.
- Shale/clay presence, containing bound water, which elevates the porosity readings.

3. **Array Induction Resistivity Log**

   Resistivity tools work by inducing an electrical current into the formation and measuring the resulting voltage response, which is affected by the salinity and volume of the formation water. As highly saline water exhibits better conductivity resulting in lower resistivity values, hydrocarbon-rich zones show higher resistivity values. The resulting logs are displayed on a logarithmic scale ranging from $0.2\ \Omega \cdot$ m to $2000\ \Omega \cdot$ m. Induction tools generate a magnetic field to induce eddy currents in the formation and measure conductivity. During drilling, mud filtrate invades the permeable formations, pushing formation water away from the borehole and forming a flushed zone. This creates three zones of interest:

   - The rinsed (invaded) zone near the borehole.
   - The transition zone (annular).
   - The uninvaded zone (true formation).

   Each zone has different fluid saturations and electrical properties. Resistivity readings taken at different depths of investigation allow interpretation of these zones.

2.1.2 Core Analysis

Core drilling is the process of extracting cylindrical samples of the formation, either during drilling or later, respectively, conventional coring or cable and sidewalls coring Petrophysics. Conventional coring involves using a special core bit with the drill string, the main advantage of this method is the recovery of large diameters samples, varying from 3 to 5 inches in diameter and 30 to 90 feet long, but it requires the entire drill string to be removed in order to retrieve the core. However, for the wireline coring, the sample is carried to the surface using a drill pipe (a downhole tool used for fishing and recovery operations according to *SLB Glossary*). The cores obtained are relatively small, from 1 to 2 inches in diameter and 10 to 20 feet in length.

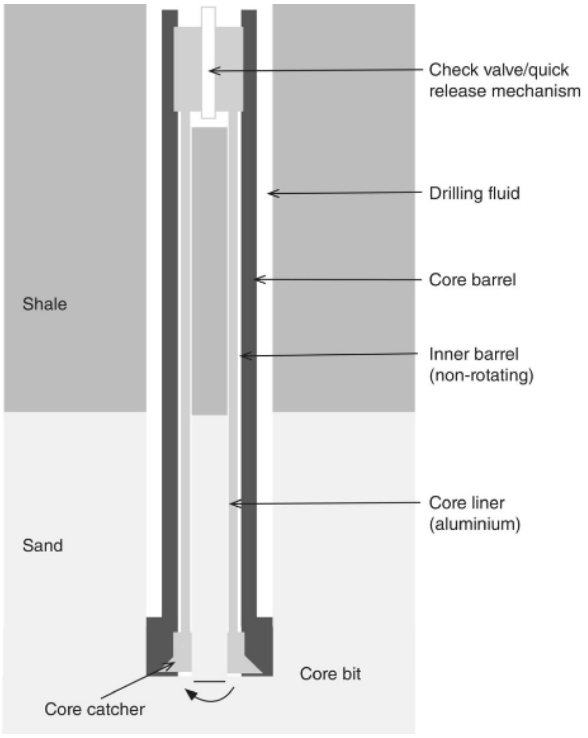


Figure 2.5: Schematic diagram of a coring assembly and barrel prior to retrieval [2].

Furthermore, lateral coring is operated to obtain core samples from a particular zone already drilled, especially in soft rocks.

Figure 2.6: Core Samples [28].

These cores are essential for understanding the depositional environment and acquiring direct measurements of reservoir properties such as porosity and permeability.

Laboratory measurements (porosity–permeability)

Upon delivery to the laboratory, cores are arranged to collect the information required for identification and archiving, including description and measurements, surface features such as fractures are noted prior to sampling and gamma-ray scanning. A typical core analysis implies systematic samplings at regular intervals (e.g., every 25 cm).

Thin sections (slices of rock) prepared in a laboratory from the cores are used for petrographic analysis (mineralogy, texture, diagenesis), often complemented by scanning electron microscope (SEM) imaging for pore size distribution.

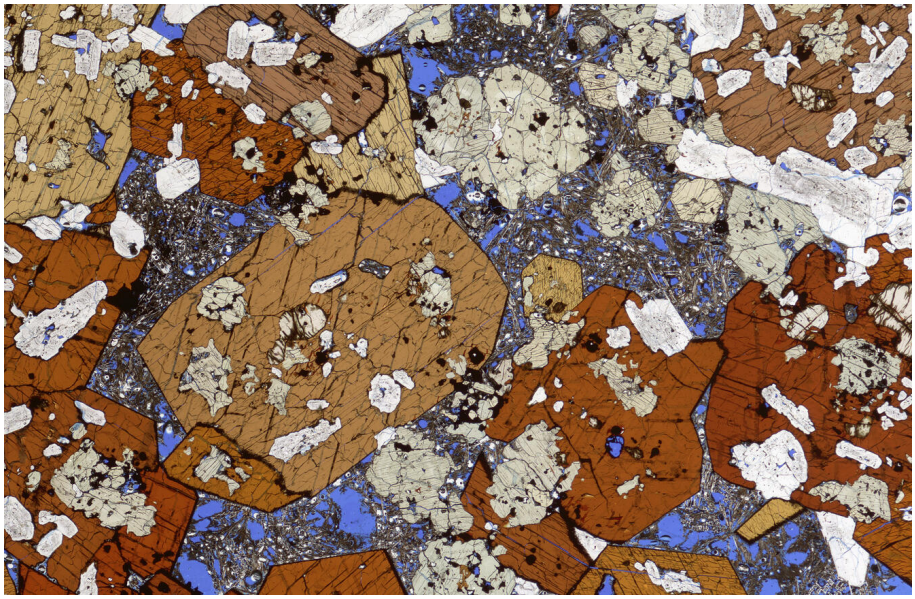


Figure 2.7: Photomicrograph of rock thin section: Gabbroic inclusion [29].

Representative core samples from different facies are subjected to electrical experiments to determine Archie's parameters (a, m, n), allowing better estimates of water saturation based on resistivity logs and empirical relationships. Additionally, core samples are crucial for the dynamic evaluation of reservoir properties, including wettability, capillary pressure and relative permeability. Despite the valuable information they provide, cores lose some of their representativeness once extracted. Pressure and temperature changes alter fluid distribution, and physical manipulations can disturb the rock fabric, leading to changes in porosity and permeability. Nonetheless, core data remains indispensable for calibrating wireline logs during log integration and for building accurate geological and petrophysical models.

Special Tests

Specific core Analysis (SCAL) samples, targeting specific facies, are selected jointly by geologists and petrophysicists. These whole-cores samples are described, photographed, and preserved before being cut and stored. Routine sampling is performed using a water-cooled diamond drill to extract 2.5 to 3.8 cm plugs perpendicular to the bedding. These are labelled and cleaned in solvent chambers to remove hydrocarbons and water. Drying then takes place in humidity-controlled ovens; however, care must be taken, as processes such as drying can affect fragile clay structures (e.g., illite), altering permeability without significantly impacting porosity. SCAL experiments fall into two main categories: (1) *electrical measurements for saturation models* and (2) *dynamic flow measurements for reservoir simulation*. Due to their complexity and cost, SCAL experiments are limited to a few carefully selected samples deemed homogeneous and representative, often verified by computed tomography (CT) scanning. Without such high-quality core data, petrophysical models would carry much greater uncertainty and are less reliable for decision-making.

2.1.3 Well Test Data

Historically, well testing was performed to obtain rudimentary production-related data, such as fluid type, deliverability, reservoir pressure, and permeability. Nowadays, well testing has become a vital tool for comprehensive reservoir evaluation, as a direct result of the rapid development of high-precision gauges and analysis software programs that enable better understanding and improved data quality.

With access to accurate pressure measurements, it is now possible to detect large-scale reservoir heterogeneities, such as faults or facies boundaries. Nevertheless, interpretations remain very different and must be corroborated by additional seismic or geological data to ensure reliability.

Well tests are particularly valuable for estimating permeability. Unlike core or log data that provide localized or averaged values, the flow rate measured during a well test provides a dynamic, in situ representation of the reservoir behaviour. The resulting permeability–thickness values can be compared across wells to identify the most productive intervals and prioritize reservoir development.

2.2 Pre-processing Techniques in Traditional Petrophysical Workflows

Before using petrophysical data for quantitative analysis or predictive modeling, it has to go through a series of pre-processing steps to improve its quality, consistency, and reliability. These steps are mandatory when dealing with large volumes of well log data, which may be affected by noise, missing values, or incompatible measurement scales.

The most commonly applied pre-processing techniques in petrophysical analysis include:

1. **Outlier Detection and Removal:** Logs may contain anomalous readings due to tool malfunctions, borehole conditions, or sudden lithological changes. Statistical methods such as the Interquartile Range (IQR), Z-score thresholding, or visual inspection (e.g., box plots or cross-plots) are used to identify and remove these outliers.

2. **Handling Missing Data:** Missing data is a frequent problem, particularly for older wells or in cost-constrained environments. Traditional imputation techniques include filling gaps using linear interpolation, nearest-neighbor values, or constant substitution. In more sophisticated workflows, geostatistical methods or empirical correlations may be used to estimate missing values.

3. **Depth Matching and Log Alignment:** When combining data from different logging runs or tools, slight mismatches in depth registration can occur. Depth

shifting or resampling is applied to align measurements from multiple sources to a unified depth scale.

4. **Environmental and Borehole Corrections:** Raw log data may be affected by borehole diameter, mud properties, or tool standoff. Traditional workflows involve applying environmental corrections using correction charts or proprietary software to obtain true formation responses.

5. **Data Filtering and Smoothing:** To reduce high-frequency noise in the logs, filtering techniques such as moving average, median filters, or Savitzky–Golay filters are often applied. These help preserve geological trends while removing noise from the data.

2.3 Limitations of Conventional Approaches

The conventional approaches for predicting petrophysical parameters such as porosity, Clay Volume, and water saturation face several significant limitations that affect the efficiency; accuracy and scalability in reservoir characterization.[30]
Some of These restrictions are listed below:

1. **Time-consuming and Expensive:** Traditional methods like core analysis and well logging are both time-consuming and expensive, making them impractical for comprehensive or real-time reservoir evaluation.

2. **Limited Data and Coverage:** Direct measurements typically come from a few wells or limited depth intervals. As a result, they may not adequately represent the spatial variability and heterogeneity of the entire reservoir.

3. **Complexity in Heterogeneous Reservoirs:** Empirical correlations and conventional models often fail to capture the nonlinear, non-uniform distribution of petrophysical properties in complex, layered, or heterogeneous reservoirs. This leads to reduced accuracy in parameter estimation.

4. **Sensitivity to Reservoir Conditions:** Parameters such as water saturation are highly sensitive to reservoir-specific factors including mineral composition, cementation, and fluid salinity, complicating accurate estimation using traditional resistivity-based methods like Archie's equation, may not perform well under varying geological conditions.

5. **Lack of Real-Time Prediction:** Conventional methods do not provide real-time predictions, which limits timely decision-making during drilling and production.

6. **Requirement of Expertise and Specialized Equipment:** These methods depend heavily on the availability of domain experts and specialized tools, adding to operational complexity and cost.

7. **Nonlinear and Complicated Relationships:** The relationship between well log attributes and petrophysical parameters is often complex and nonlinear, making conventional empirical or correlation-based approaches less effective.[30]

These challenges have driven the industry toward data-driven approaches such as machine learning (ML) and artificial intelligence (AI) techniques, which are capable of handling nonlinearities, provide real-time predictions, and improve accuracy and efficiency in petrophysical parameter predictions.

2.4 Data-Related Challenges in Predicting Petrophysical Parameters

One of the main obstacles in predicting petrophysical parameters lies in the quality, completeness, and consistency of the available data.

Several key issues must be addressed:

1. **Data Quality and Measurement Errors:** Logging data can be affected by borehole conditions, tool calibration differences, and environmental factors, introducing noise and inconsistencies [31]. These errors complicate the interpretation and reduce the reliability of conventional prediction methods.

2. **Noise in the Data:** Well log measurements are often affected by various types of noise resulting from mechanical, electrical, or environmental disturbances. This noise can obscure meaningful patterns and reduce the accuracy of predictive models.

3. **Missing or Incomplete Data:** It is common to encounter missing logs or incomplete records, especially in older wells or cost-constrained drilling operations. This limits the volume and quality of usable data for training machine learning models.

4. **Variable Resolutions and Scales:** Data may originate from multiple sources with differing sampling rates and resolutions (e.g., high-resolution logging tools vs. low-resolution seismic data). This heterogeneity complicates data integration and may introduce bias into the modelling process.[31]

Chapter 3

Bibliographic summary of artificial intelligence

Objective: *Provide a concise and comprehensive overview of the fundamental concepts, historical evolution, and key techniques of artificial intelligence (**AI**).*

3.1 Introduction

Artificial intelligence has grown in popularity throughout various industries, it has been a transformative technology by revolutionizing numerous scientific and industrial fields due to its ability to process vast amount of data, recognize patterns and make prediction has led to its widespread adoption in many sectors such as engineering, health care and finance. In petroleum engineering, **AI** plays an important role in analysing petrophysical parameters, enhancing decision-making process and also optimizing reservoir management. **AI** has been studied for decades and is still one of the most elusive branches of Computer Science. This is mainly because of how large the subject is. This chapter aims to provide a global understanding of **AI** by exploring its basics; concepts and applications. Furthermore, it will enlighten the relationship between **AI**; *Machine learning* (ML), *Deep Learning (DL)* and *Generative **AI*** (gen. AI) explaining their significance and methods in data-driven, decision-making and prediction across various industries.[32]

3.2 Artificial intelligence

AI is known the use of a machine or computer intelligence rather than human or animal intelligne.it's a branch of computer science that studies the simulation of human intelligence processes such as learning, problem-solving and self-correction by computers [33].**AI** is a technology that allow computers and machines to reproduce human comprehension, learning, problem-solving, decision-making and autonomy [34]. The term Artificial intelligence was first coined in 1956 by *John McCarthy* during the first academic conference on the subject that he held[32]. Still, the journey to figuring out whether computers can truly think began much earlier. In The groundbreaking 1945 essay *"As We May Think"*,

Vannevar Bush envisioned a system that amplifies human knowledge and understanding.Five years later, *Alan Turing* created the notion of machines simulating human intelligence and performing tasks requiring human [32].

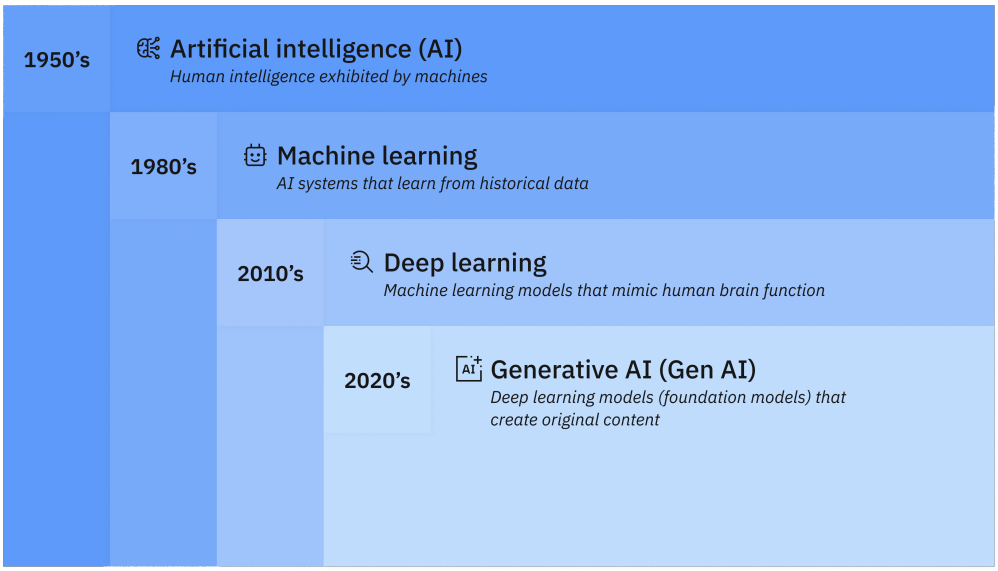


Figure 3.1: Evolution of Artificial Intelligence: From AI to Generative AI[35].

ibm$_a i_t opic$

3.3 Maching Learning

Underneath **AI**, there is Machine Learning (ML) that is defined as the collection of various algorithms used to teach computers to find patterns in data for future estimation and forecasting or as a quality check for performance optimization, it involves creating models to make decision and predictions [34]. It encloses a wide range of techniques that provides the ability to machines to learn from and make inferences based on date without being explicitly programmed for specific tasks, and it can be categorized into three main types, which are Supervised Learning, Unsupervised Learning and Reinforcement Learning [33].

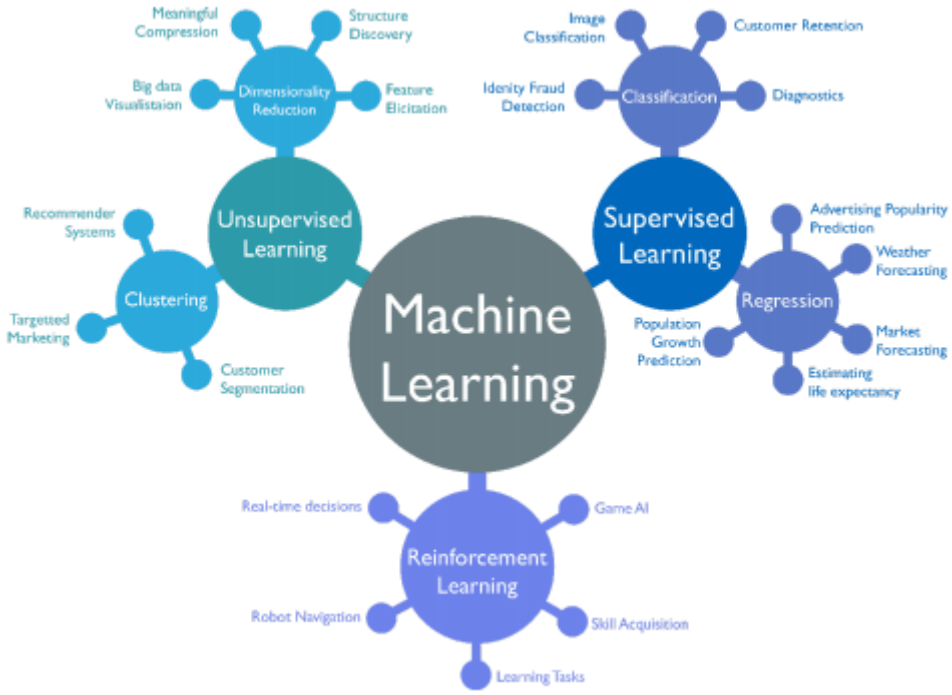


Figure 3.2: Machine Learning Techniques[36].

3.3.1 Supervised Learning

Supervised learning is a method that trains the model, using labelled data, where input features are paired with corresponding output labels [37]. The objective is to establish a mapping between inputs and outputs, enabling the mode to recognize accurate predictions on unseen data. This approach solves various problems at scale and is used to develop highly accurate predictive models [38].

Supervised Learning Common Models

1. **Classification:** this technique classes data by recognizing specific entities in the dataset and determining how those should be labelled.Its objective is to predict the category to which a given input belongs.[38]

   Common classification algorithms include:

   - **K-Nearest Neighbour (KNN):** This algorithm assumes that similar data points are located near to one another when represented mathematically; it classifies the data points according to their similarity and proximity. It is simplicity make it useful for image recognition and recommendation systems.But, as the data size increases; processing time will also increase, making it less efficient [38].
   - **Random Forest:** this supervised **ML** approach can be used in both classification and regression models. It consists of numerous uncorrelated decision trees,

forming a forest. By merging their predictions, the algorithm will reduce divergence and higher accuracy [38].

- **Support Vector Machine (SVM):** this operation distinguishes different classes by setting a decision boundary, known as a *hyperplane*. The purpose of this algorithm is to determine the hyperplane that expands the combination between the data point groups, confirming an optimal separation [38].

2. **Regression:** It is a model that is employed to understand the relationship between variables. In regression problems, the output is a continuous value and models attempt to predict the target variable [37].

   There are three common types of regression algorithms:

   - **Linear regression:** It's a predictive modelling method that estimates a dependent variable based on independent variables. It established a linear relationship between these variables by estimating the coefficients of a linear equation that fits the data. The goal of this technique is to reduce the difference between predicted and actual outputs [39].
     The simplest form for a linear regression model consists of a linear combination of the input variables and takes the form :

     $$y(\mathbf{x}, \mathbf{w}) = w_0 + \sum_{j=1}^{m} w_j x_j \quad \text{where } \mathbf{x} = [x_1, x_2, \ldots, x_m]$$

     The important property of the linear regression model is that it is a linear function of the regression coefficients $w_1, w_2, \ldots, w_m$ [40].
     This type of regression model remains linear even if one of the regressors is a non-linear function of the other regressor or of the all data predictors.
   - **Logistic regression:** This approach is considered as a statistical model that evaluates the probability of an event occurring based on a given set of independent variables. Commonly referred to as the logic model, it is used for classification and forecasting analytics. Considering that the output represent a probability, the dependent variable ranges between 0 and 1 [41].
   - **Polynomial regression:** Like other regression models, this one establishes the relationship between variables but, does so using polynomial functions of varying degrees. It captures non-linear patterns by incorporating exponential terms of the independent variables [37].

3.3.2 Unsupervised Learning

Unsupervised learning builds upon complex inputs without any labels, its purpose is to deduce the underlying structure or patterns of a system from observed data. It demands using machine learning algorithms to cluster and analyse unlabelled data sets

into significant groupings called subsets and clusters.These algorithms reveal intrinsic patterns or data structures without any human intervention based on similarities. It's ability to detect resemblances and distinctions in data makes it ideal for exploratory data analysis, cross-selling strategies, customer segmentation and image recognition [42].

Unsupervised Learning common modals

1. **Clustering:** This technique classifies and organizes objects, data points or observations into groups or "clusters" based on patterns. Dissimilar to supervised learning, clustering does not rely on labelled data, instead it identifies built-in structures within the dataset this method is commonly used in exploratory data analysis to reveal the hidden patterns and to understand underlying trends, patterns, and outliers. It upgrades the ability to recognize natural groupings, facilitating hypothesis generation and deeper insight. Furthermore, it is an essential in dimensionality reduction where large datasets are segmented into small meaningful subsets. In this case, clustering can be a step in preprocessing. There are numerous clustering algorithms as there are multiple ways to define clusters, the approach of the algorithm depends on different factors like the size of the input data, its dimensionality, the rigidity of the categories and the number present clusters [42].

   These algorithms can be divided into four types:

   - **K-means Clustering:** this algorithm is one of the most widely used centroid based clustering techniques in data segmentation, pattern recognition, and exploratory data analysis across various domains due to its simplicity and efficiency. This optimization process involves separating a dataset into **'K'** clusters, where each data point is assigned to a cluster with the nearest centroid. K-means performs effectively when clusters are approximately equal in size, and there are no significant variations in density across the data [43].
   - **Hierarchical Clustering:** Also known as connectivity-based clustering, groups data points based on their distance and connectivity across all dimensions. The idea behind this approach is that objects that are closer to each other are more related than those farther apart. This method can be performed using agglomerative strategy (where each point starts as a single cluster, then clusters are iteratively merged based on their similarity) or divisive strategy (here all data points are in one single large cluster, that is then recursively split into smaller clusters). Unlike K-means clustering, this process does not need a specification regarding the numbers of clusters [43].
   - **Distribution-based clustering:** Sometimes called probabilistic clustering, groups data points based on their probability distribution rather than Metrics like Euclidean distance [42]. It considers a process generating normal distributions

across different dimensions. Instead of relying on measuring distances, it identifies statistical distributions that best represent the data across each dimension. One of the most used probabilistic clustering techniques is the Gaussian Mixture Model (GMM) that aims to determine the Gaussian probability distribution to which a data point belongs and to operate under the assumption that these parameters are unknown [43].

- **Density-based clustering:** This category detects high density areas within a dataset while differentiating low density regions. Density-based clustering can discover clusters of any shape, size, or density in a dataset. It also distinguishes between data points which are part of a cluster and those that have to be labelled as noise. This algorithm classifies as outliers the points that are isolated in low-density areas and clusters the densely packed points. It performs well for analysing complex datasets [42].

2. **Dimensionality Reduction Models:** It's a technique employed when a dataset contains a high number of features or dimensions, it aims to reduce the number of the data inputs to a more manageable size while keeping the essential structure and integrity of the dataset as much as possible. This process is mostly used in the data processing stage to reduce computational costs and improve model efficiency [43].

There are a few different dimensionality reduction methods:

- **Principal Component Analysis:** This method is designed to eliminate redundancies and compress datasets. It applies a linear transformation to the data that generates a new representation defined by a set of components. The first component captures the maximum variance in the dataset, on the other hand, the second one maximizes the variance as well but remains completely uncorrelated with the first one, ensuring that it is orthogonal to it. This operation continues iteratively, where each new principal component is orthogonal to the previous ones and captures the next highest variance, allowing the PCA to retrain relevant information [42].

- **Singular value decomposition:** It is another technique that divides a matrix **A** to three lower-rank matrices$\boldsymbol{A = USV^T}$, where **U** and **V** are orthogonal matrices, and **S** is a diagonal Matrix that contains the singular values. Like PCA, it is usually used to reduce noise and compress data [42].

- **Autoencoders:** It utilises neural networks to compress, then reconstruct a new representation of the original data's input. This approach consists of two stages: encoding where the input in compressed into a hidden layer and decoding where the date is reconstructed. The hidden layer acts as a bottleneck, ensuring that the essential features are kept [42].

3.3.3 Reinforcement Learning

Reinforcement Learning (RL) is a branch of machine learning that focuses on sequential decision-making by autonomous agents, systems capable of acting independently in a response to their environment, needless to any guidance or direct instruction by a human user. The process of this learning method involves an interaction between the agent, the environment, and a defined goal. This relationship is commonly modelled using a *Markov Decision Process* (MDP), which furnishes a mathematical framework for decision-making in unknown environments [44].

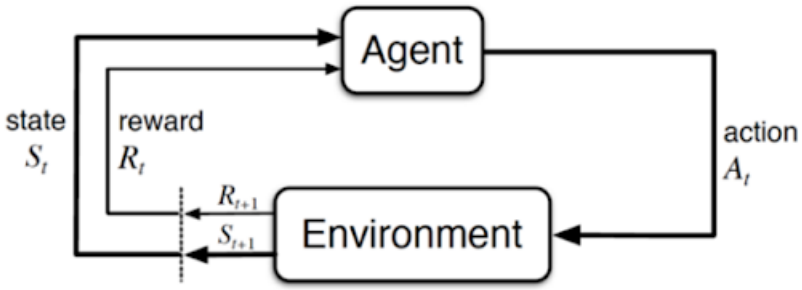


Figure 3.3: graphical representation of the MDP model [45].

At each step, the environment presents the agent with its current state. The agent then determines an action to take based on that state. If the action obtains a reward signal from the surrounding environment, the agent is encouraged to repeat that action in similar future situations. Over time, the agent learns from rewards and punishments to take actions within the environment that meet a specified goal.

3.4 Deep Learning

Deep leaning is the latest achievement of machine learning, it achieves its learning based on deep neural networks, which are a multilayered structure designed to simulate the complex decision-making power of the human brain with its own complex computational and recognition abilities. It consists of multiple layers of interconnected nodes, each building on the previous layer, which take more features or details from the previous layer in order to optimize and fine-tune the prediction or categorization. Data travels through the network through a process called forward propagation, in which each layer processes the data it received in a way that refines the prediction or classification of the model [46].

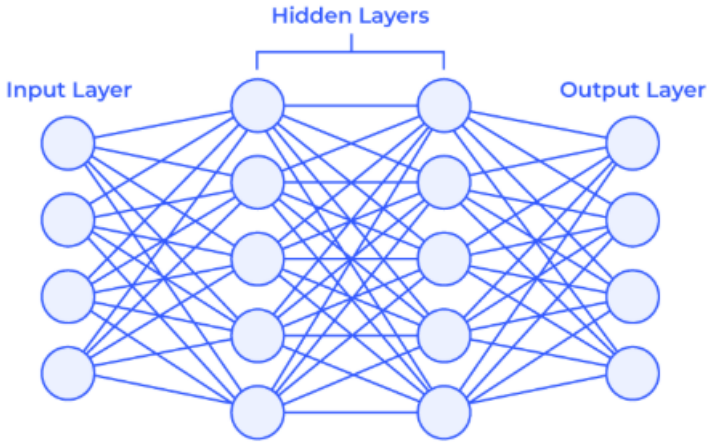


Figure 3.4: Neural-Networks-Architecture[47].

An additional operation called backpropagation uses algorithms including gradient descent to determine errors in predictions, then modify the weights and biases in the function by working backward through the layers to train the system to learn from its errors, and thus get better at predicting and improving its accuracy over time.
Deep learning is widely used today in many artificial intelligence applications, particularly in computer vision and natural language processing (NLP).[46]

Deep Learning Common Models

1. **Multi-Layer Perceptron (MLP):**
   The word *"perceptron"* originates from a simple neural model designed for binary classification, which maps input features to a single output decision. It is termed *"multilayer Perceptron"* because it contains an input layer, one or more hidden layers and an output layer, in this architecture, neurons in each layer maintain complete connections with the neurons in the subsequent layer, enabling the network to perform advanced non-linear data transformations of the input data [48]. This approach shows strong performance in regression and classification functions because it accurately detects complex hidden connections among data variables [49].
   A common MLP is composed of key components that work together to process information and make predictions, whose role is set:

   - **Input Layer:** Each neuron or node in this layer corresponds to an input feature. For instance, if you have three input features the input layer will have three neurons.
   - **Hidden Layer:** MLP can have any number of hidden layers with each layer containing any number of nodes. These layers process the information received from the input layer.
   - **Output Layer:** The output layer generates the final prediction or result. If there are multiple outputs, the output layer will have a corresponding number of neurons [40].

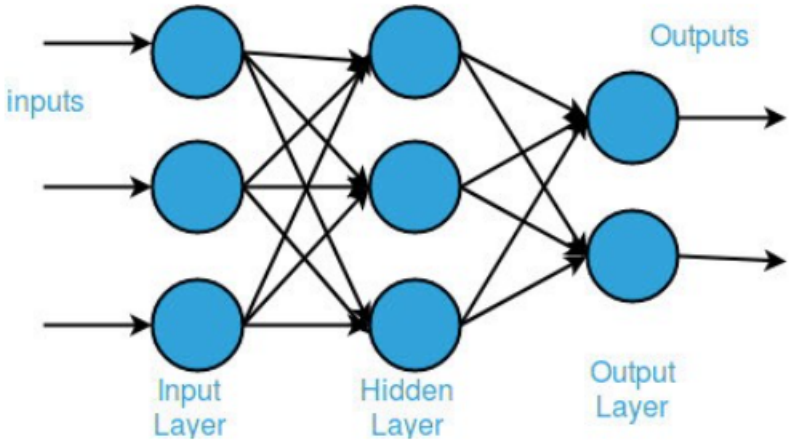


Figure 3.5: Multi-Layer Perceptron Structure[50].

2. **Convolutional Neural Networks:** CNNs are a subset of deep learning models, specifically tailored for dealing with data that has 'grid' like topology and images. They are famous for their great success in image classification, object detection, and pattern recognition. CNNs are especially suitable due to their capability of automatically learning and extracting high-level features from raw data, and therefore are especially powerful for structural and spatial data [51].

A common CNN network is composed of key components whose role is set:

- **Convolutional Layer:** It's considered as the core building block of a CNN, where the majority of computation occurs. It requires a few components, which are input data, a filter and a feature map. These filters detect unique patterns, such as edges or curves, which are systematically moved over all the spatial expanse of the input while preserving spatial information.

- **Pooling Layer:** These layers intend to redact the spatial dimensions of the feature maps, reducing computational load and helping with spatial invariance. This operation retains details to reduce computational burden and support the prevention of overfitting.

- **Activation Function:** Both convolutional and pooling layers are improved with non-linearity using activation functions. This process leaves only positive input values identifiable by their lack of alteration, enabling the model to explore complex relations and patterns within the data set.

- **Weights:** Both convolutional and pooling layers are improved with non-linearity using activation functions. This process leaves only positive input values identifiable by their lack of alteration, enabling the model to explore complex relations and patterns within the data set.

- **Fully Connected Layer (FC):** Positioned at the end of the network, these layers accumulate spatial feature information, produced by earlier convolutional and pooling layers, towards the classification or regression result. They are mainly used to

interpret the high-level features extracted by the previous layers as a single-dimensional array, allowing for a complex analysis and reasonable conclusions [52].

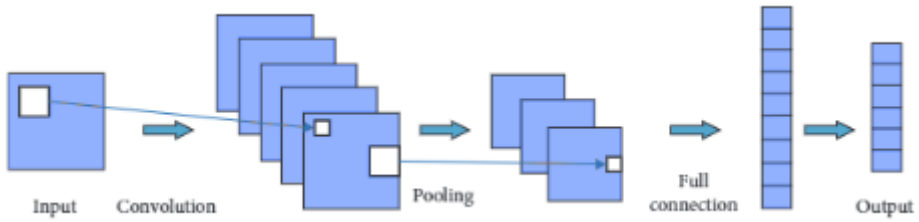


Figure 3.6: Convolutional Neural Network Structure[53].

Most popular deep CNNs

- **1D Convolutional Neural Network:** This type of CNN is specifically designed for processing one-dimensional sequential data, like time series, text or any data where the structure varies along a single axis. It extends the traditional CNN's ability to recognize patterns in images to handle sequences of data. The core component is the 1D convolutional layer, which applies filters that slide across the sequence to capture local patterns and features, enabling the network to automatically extract meaningful patterns and dependencies within the data [53].

- **2D Convolutional Neural Network:** A 2D Convolutional Neural Network is an architecture primarily designed to analyse two-dimensional data, such as images. Processes data by applying convolutional filters that slide across the image in both horizontal (x) and vertical (y) directions, capturing spatial features such as edges, textures, and patterns. Each filter generates a 2D feature map that retains spatial information about where specific features appear [53].

3.5 Algorithms and Applications of AI in Petrophysics

1. **Prediction of Petrophysical Parameters**
   AI developments have improved many industries, with the petroleum sector being one of the main ones, making it easier and better to study underground rocks. In particular, the prediction of petrophysical parameters such as porosity and permeability has greatly benefited from these developments.
   The following section presents recent studies and examples that illustrate the application of deep learning techniques for the prediction of petrophysical parameter:

   - **Direct Mineral Content Prediction from Drill Core Images via Transfer Learning:** Boiger et al. (2024) employed convolutional neural networks (CNNs) to analyse drill core images, achieving 96.7% accuracy in classifying formation types. Additionally, a CNN model was trained to evaluate mineral content,

demonstrating performance comparable to laboratory X-ray diffraction (XRD) measurements.[54]

- **Neural Machine Translation of Seismic Waves for Petrophysical Inversion:** Teixeira et al. (2024) introduced a deterministic petrophysical inversion technique based on a language model that decodes seismic wave velocity measurements to infer soil petrophysical and mechanical parameters as textual descriptions. This approach delivered comprehensive geological insights 2,000 times faster than conventional methods [55].
- **Machine Learning-Based Prediction of Well Logs Guided by Rock Physics:** A 2025 study utilized four machine learning algorithms —Random Forests (RF), Gradient Boosting Decision Trees (GBDT), Multilayer Perceptrons (MLP), and Linear Regression (LR)— to predict porosity and clay volume fraction from well logs. The predictions were guided by rock physics principles, and SHapley Additive exPlanations (SHAP) analysis uncovered consistent patterns across the algorithms [56].

2. **Facies and Lithology Classification:**

Reservoir characterization depends greatly on properly identifying facies and lithology, which guides decisions in both exploration and production. Before, facies and lithology were recognized by looking at well logs, core samples and seismic data, which often took time and could be affected by the interpreter's views. Current advancements in deep learning have provided tools that are able to automate tagging more accurately. When using large data and complex neural network settings, deep learning models are able to notice microscopic changes and patterns in geological information that are not easy for people to analyse nor to detect.

This section highlights recent deep learning advances in facies and lithology classification.

- **Lithofacies Prediction from Well Log Data Using Deep Learning:** A 2024 study utilized Convolutional Neural Networks (CNNs) and Residual Neural Networks (ResNets) to classify lithofacies such as coal, sandstone, and limestone from well log data. The models achieved up to 88% accuracy in predicting various lithologies, demonstrating the effectiveness of deep learning in automating lithology classification [57].
- **Automated Lithology Classification of Drill Core Images Using CNN:** In 2024, researchers developed a lightweight CNN model for classifying lithologies such as carbonate, sandstone, and shale from drill core images. The model achieved an accuracy of 96.9% with only 69,600 parameters, showcasing its efficiency and potential for real-time applications [58].

Conclusion of the Theoretical Framework

The theoretical framework presented in this first part has laid the essential foundations for understanding the petrophysical parameters at the heart of this study: clay volume (V_{CL}), effective porosity ($PHIE$), and water saturation (S_W). We explored the physical principles behind well logging, the interpretation of log data, and the relevance of each parameter to reservoir characterization. Additionally, we reviewed the fundamentals of machine learning, focusing on its growing importance in solving complex prediction problems in geosciences.

This theoretical grounding now allows us to move forward with confidence into the practical part, where we apply these concepts using real data and advanced machine learning models to predict petrophysical properties in the Berkine Basin.

Part Two
Practical Study

Chapter 4

General Characteristics of the Berkine Basin

Objective: *Provide an overview of the geological and petrophysical framework of the Berkine Basin, with a particular focus on its petroleum systems, source rocks, reservoir characteristics, and hydrocarbon generation history.*

4.1 Introduction

The Berkine Basin is a major intracratonic sedimentary basin, representing the most subsided part of the syneclise of the oriental province in the eastern part of Algerian Saharan Platform. The thickness of the sedimentary terrains reaches about 7500 m deep, resting directly on a crystalline basement, testifying to a considerable geological subsidence. The basin covers an area of about 350,000 km^2, spread over three North African countries: 50,000 km^2 in eastern Algeria, 200,000 km^2 in western Libya and about 100,000 km^2 in southern Tunisia.

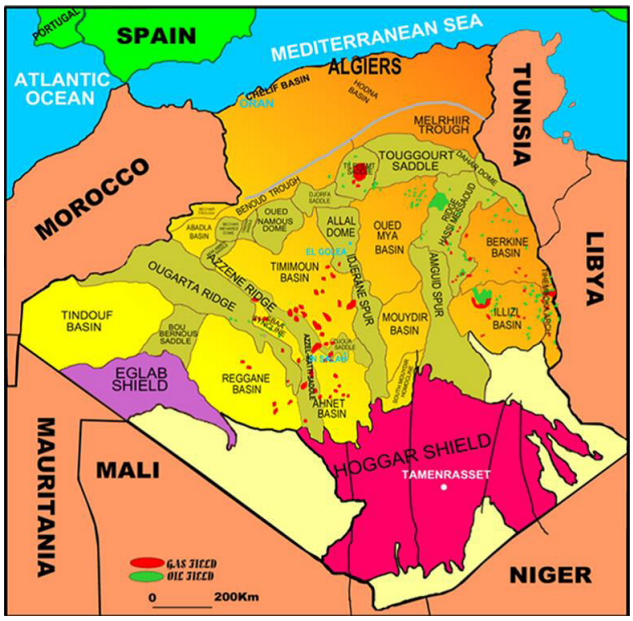


Figure 4.1: Map of the Sedimentary Basins of the Saharan Platform[59].

Geologically, it is located in the oriental province of the eastern side of the Algerian Saharan Platform. It is divided into three distinct geological complexes based on tectono-structural and facies relationships:

- he central Berkine sub-basin, constituting an extension of the eastern sector of the larger Ghadames Basin in Libya;
- The Triassic Hassi-Messaoud Ridge and AMGUID-Spur, located to the west of Berkine basin;
- The Dahar Dome, located to the north, characterized by high Hercynian structural relief.

Tectonic evolution reflects a complex interaction between marine and continental processes. During the Namurian, at the basin scale, the region experienced a significant marine regression, which was followed during the Westphalian by a new marine transgression, originating from the northeast, extending over large areas of the eastern Sahara. During the final phase of the Westphalian, a new phase of marine regression led to the formation of sedimentary environments, mainly lagoonal and continental.

4.2 Geographical Setting and Boundaries

The Berkine Basin is geographically bounded between latitudes 29°30'N and 33°40'N and extends eastward from longitude 5°55'E to the Tunisian border. It is bordered to the east by the Algerian–Tunisian and Algerian–Libyan frontiers. The basin covers a total area of approximately 102 395 km^2, divided into 28 exploration blocks.

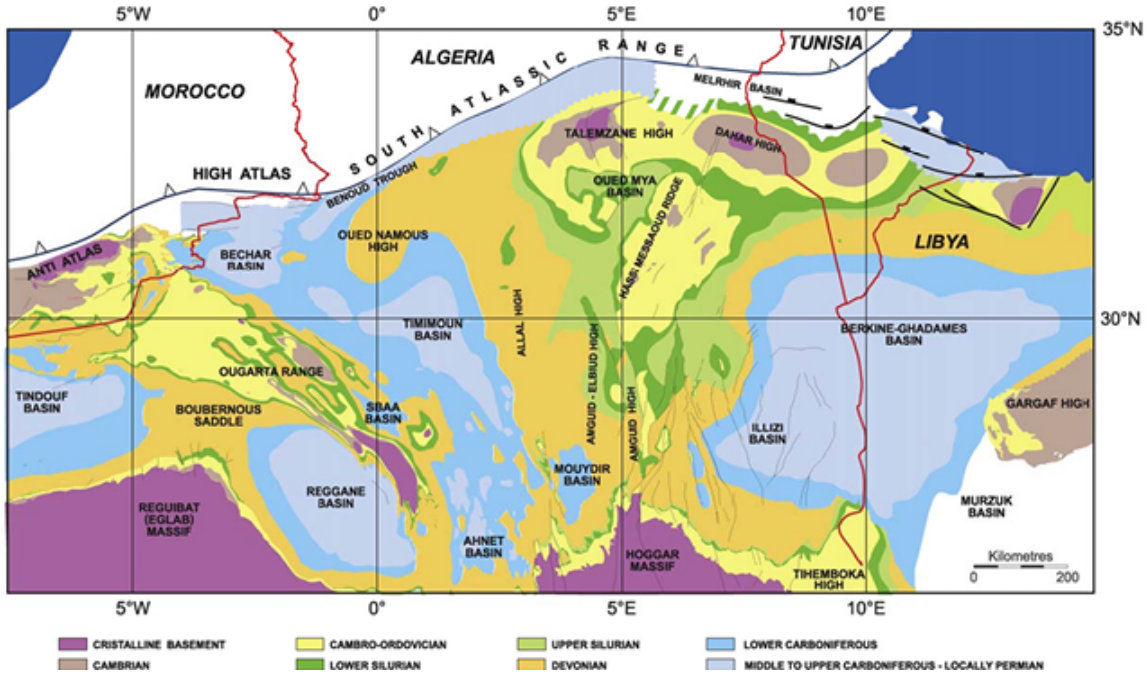


Figure 4.2: Geographical Location of the Berkine Basin [60]

4.3 Geological context

4.3.1 Lithostratigraphy

The stratigraphic conditions of the region have posed several challenges, mainly due to lateral facies variations and the scarcity of macrofauna. These factors have led to the establishment of regional lithostratigraphic nomenclatures. However, the lack of reliable chronological markers has often complicated the correlation between sedimentary series.
The Berkine Basin has preserved a sedimentary fill exceeding 6,000 meters thick at its center, ranging from the Paleozoic to the Quaternary. This entire sequence rests on a Precambrian granitic basement.
The central part of the basin has been relatively unaffected by Hercynian erosion, allowing the preservation of the upper Carboniferous series. In contrast, towards the structural highs and basin margins, the Paleozoic sequences have been progressively eroded by Hercynian events. The basin margins are characterized by the development of Silurian-Devonian units underlying the Mesozoic cover.

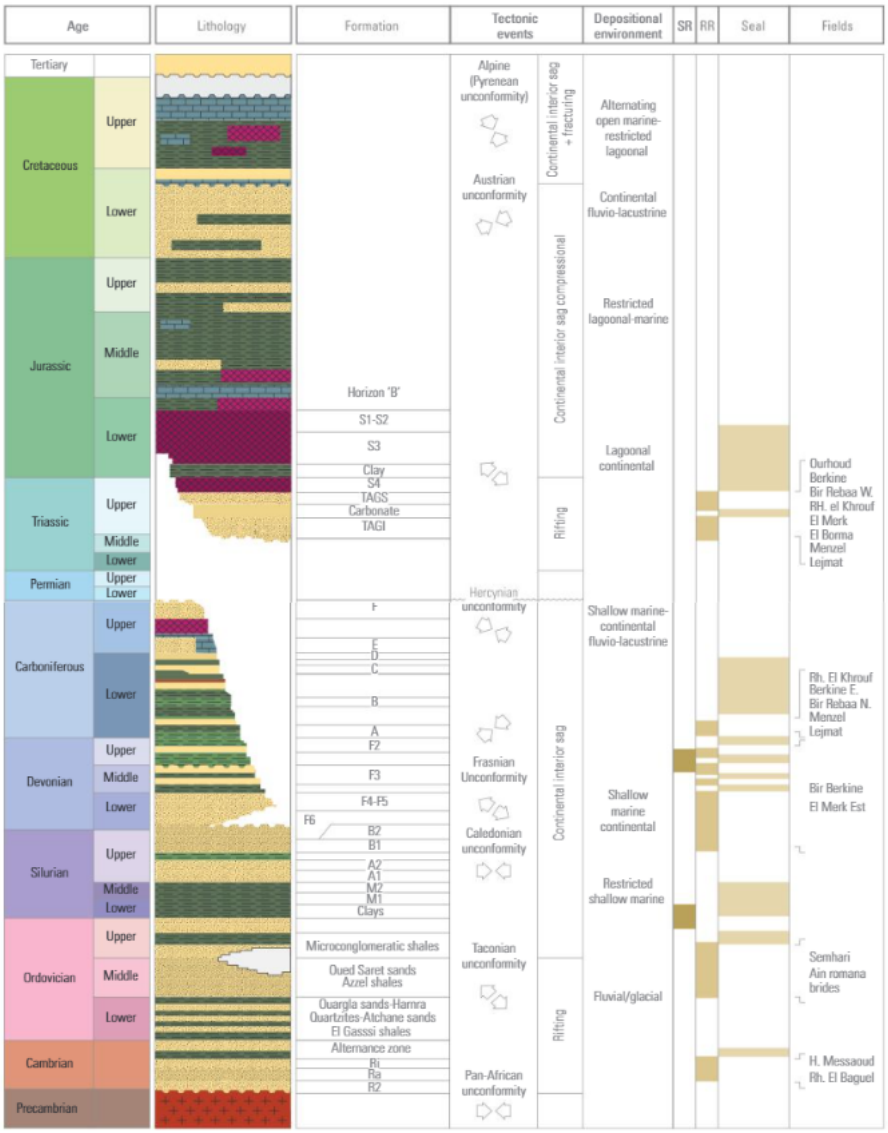


Figure 4.3: Stratigraphic column of Berkin Bassin[61]

The Paleozoic

The Paleozoic succession is subdivided into three main stages:

- **Cambrian:**
  This interval corresponds to three main reservoir units: R1 (Ri, Ra), R2, and R3, generally composed of quartzitic sandstones. These Cambrian sandstones are known for their oil productivity and have an average thickness of around 300 meters.

- **Ordovician:**
  The principal reservoirs are composed of quartzites (Hamra), Ouargla sandstones, El Atchane sandstones, El Gassi clays, Azzel clays, Ramade sandstones, Oued Saret sandstones, and micro-conglomeratic clays. These reservoirs have an average thickness of about 250 meters, and their productivity is primarily associated with natural fracturing.

- **Silurian:**
  Unconformably overlying the Paleozoic, it is represented by clay-sandstone and lagoonal deposits (salt and anhydrite) and is subdivided into three units:

  - Argillaceous Silurian: Comprising mainly gray to light gray clays, sometimes dark brown, silty with fine siltystone interbeds. Towards the base, the clays become dark and highly organic-rich, forming the main source rock of the basin.
  - Argillaceous-Sandstone Silurian: A mixed lithological unit including dark gray to greenish-gray clays, silts, white fine- to very fine-grained quartzitic sandstones, compact in some layers, with occasional siltystone interbeds.

The Mesozoic

- **Jurassic:**
  Composed of lagoonal marine sediments, it begins with a characteristic dolomitic level known as Horizon B, which is widespread and easily recognizable.

- **Cretaceous:**
  This system is widely developed across the Saharan platform and consists of alternating sandstones, clays, dolomites, and limestones, along with some interbeds of anhydrite, gypsum, and salt. Toward the top, the formation becomes predominantly carbonate in nature.

The Cenozoic

The Cenozoic rests unconformably on the Mesozoic and is mainly composed of fine- to coarse-grained sandstones interbedded with sandy clays.

4.3.2 The Petroleum Systems of the Berkine Basin

The Berkine Basin is well known for its petroleum potential. Previous geological and geophysical studies in this region have primarily focused on the Lower Argillaceous-Sandstone Triassic (TAGI), the Carboniferous, the Devonian, and the Upper Silurian formations. The primary petroleum systems on the Saharan platform are predominantly hosted in Palaeozoic and Triassic formations, which collectively account for approximately 43% of the known oil reserves and 84% of the gas reserves. Nearly all of these hydrocarbon accumulations are concentrated in the eastern part of the Sahara platform.
Within the Berkine Basin, the estimated in-place resources include roughly 1,609 million cubic meters of oil, 72 million cubic meters of condensate, and 765 billion cubic meters of gas. These hydrocarbons are primarily sourced from the basin's most prolific source rocks, notably the Silurian black shales rich in Graptolites and the Middle to Upper Devonian formations.

Source Rocks

1. **Silurian Shales (Oued Imerhou Formation)**
   The Silurian source rocks, dating from the Llandoverian-Wenlock to Gothlandian stages, comprise black, pyritic marine shales characterized by high organic content derived from marine phytoplankton and zooplankton. These formations exhibit thicknesses ranging from 200 to 300 meters across the basin, thinning toward the northwest and northern margins.
   The Silurian interval can be subdivided into three distinct organic-rich layers:

   - A basal layer, 10 to 25 meters thick, is the richest in organic matter and recognized as the principal hydrocarbon-generating horizon.
   - An intermediate layer with variable thickness (0 to 275 meters) containing moderate organic matter content.
   - An upper layer with minimal hydrocarbon potential.

2. **Upper Devonian (Frasnian-Famennian)**
   These source rocks, predominantly preserved in the southeastern portion of the basin, were deposited during a marine transgression and are divided into two zones:

   - The lower Frasnian zone, which develop higher total organic carbon (TOC) contents and thicknesses between 50 and 200 meters.
   - The upper Famennian zone, characterized by low radioactivity and a thickness close to 50 meters.

Hydrocarbon Generation

The hydrocarbon generation history of the basin is linked to two major tectonic events: the Paleozoic and Mesozoic stages.

- During the Paleozoic, Silurian and Devonian source rocks matured sufficiently only in the basin's southeast, where burial depths allowed entry into the oil generation window.
- The Hercynian orogeny caused uplift and erosion, reducing burial depths and temporarily halting or slowing organic maturation in certain areas.
- Subsequent Early Mesozoic subsidence and increased geothermal gradients reinitiated maturation and hydrocarbon generation.
- In the Tertiary, subsidence ceased while thermal flow increased, promoting further maturation and cracking of oil to gas in the deepest basin sections.

Overall, Silurian source rocks are currently in the dry gas window across most of the basin but remain within the oil window in the northern and some southeastern areas.

Reservoirs Rock

1. **TAGS (Upper Triassic clayey-Sandstone)**
   The TAGS reservoir is located in the southeastern Triassic depression, in the southwestern part of the basin. It consists of fluvial and deltaic channel sequences, primarily composed of medium to coarse-grained sandstones, indicating proximity to sediment sources. The unit thins out southeastward near the Maouar High and disappears to the west against the Ramade Fault and the El Biod High. Northward, it transitions into more argillaceous and eventually evaporitic facies (equivalent to the S4 unit). The average thickness ranges from 100 to 150 meters.
   From a petroleum standpoint, TAGS represents one of the main reservoirs in the southeastern Triassic depression. Notable hydrocarbon discoveries, including oil and gas condensate, have been made in fields such as Nezla, Hassi Touareg, and Hassi Chergui. The unit is effectively sealed by a thick Triassic evaporite sequence.

2. **TAC (Carbonate Triassic) - TAGin (Intermediate Triassic)**
   This unit formed during the Triassic rifting phase and is characterized by argillaceous and commonly dolomitic facies, along with interbedded sandstones. It is well-developed within the southeastern Triassic depression. Its thickness varies depending on syn-rift fault activity.
   Although its reservoir potential is generally low, hydrocarbons have been encountered in areas such as Rhourde En Nouss and Hassi Chergui. In the Berkine Basin, localized sandy intervals have yielded oil, especially in the SFSW, SF, and BRSE formations.

3. **TAGI (Lower Clayey-Sandstone Triassic)**
   The TAGI formation represents the basal Mesozoic sequence and consists of predominantly fluvial (occasionally aeolian) deposits, distributed widely across the basin. Its thickness varies from 65 to 80 meters, largely controlled by faulting and regional paleotopography.
   This reservoir displays stacked sandy units, each approximately ten meters thick, separated by clayey intervals. Sediment input likely originated from the southwest, with paleo-flow directed toward the northeast. TAGI is considered a key target for exploration due to its excellent petrophysical properties, with porosity ranging between 7% and 26% and average permeability values between 27 and 35 millidarcies (md).

4. **Carboniferous**
   The Carboniferous reservoirs, dated from the Strunian to the Viséan, consist of sandstone intervals interbedded with argillaceous sequences deposited in shallow marine settings. The basal Carboniferous has limited areal extent, primarily occurring in the central and western edges of the Berkine Basin.
   The coarse-grained, proximal nature of the sandstones reflects the influence of ancient paleohighs such as the D'Amguid-Messaoud and D'Ahar massifs, which acted as major sediment sources. These reservoirs have average thicknesses ranging from 20 to 50 meters and exhibit very good petrophysical properties.

5. **Lower Devonian**
   Lower Devonian reservoirs include two main sequences: the Gedinnian, characterized by thick, post-Caledonian fluvial sandstones averaging 200 meters in thickness, and the Siegenian, a transgressive sequence with coastal and deltaic sandy deposits. Their distribution was influenced by major uplifts like the Amguid-Messaoud High and detrital input from the southeast.
   These reservoirs display good petrophysical characteristics and have proven productive for light oil and gas.

6. **Ordovician**
   Ordovician reservoirs, with average thicknesses around 250 meters, occur primarily in the southeastern Triassic depression and thin out northeastward toward the Touggourt-Semhari region. Hydrocarbon production from these quartzitic units, such as in the Hamra and Rhourde Nouss fields (gas and oil) and Nezla (oil), is largely associated with natural fracturing.

7. **Cambrian**
   Cambrian reservoirs are represented by four units: Ri, Ra, R2, and R3. The best reservoir quality is found in the quartzitic sandstones of the Ri and Ra units. These sandstones have yielded oil in several fields, including Rhourde El Baguel,

Ain Romana, and Damrane. To date, Cambro-Ordovician reservoirs have mainly been identified along the northern and western flanks of the Berkine Basin.

Seal Rocks

1. **Seals of the Cambrian Reservoirs**
   The seal is provided by clay formations, and lateral closure is ensured by vertical displacement along faults associated with the regional structural trend.

2. **Seals of the Ordovician Reservoirs**
   Typical clays from the Ordovician and Silurian periods can act as seals for the Ordovician sandstones, providing containment for accumulations, except in areas where erosion has occurred due to tectonic activity, especially in the northern region.

3. **Seals of the Devonian Reservoirs**
   Clays within the Carboniferous and Devonian formations serve as seals for the Devonian reservoirs; specifically, those from the late Devonian.
   The Ordovician and Cambrian reservoirs are sealed by clay formations.

4.3.3 Selecting the Berkine Region

The Berkine region was selected for this study due to its high hydrocarbon potential and the availability of a rich and diverse dataset. As one of the most productive basins in Algeria, Berkine offers extensive well logging data across various formations, which is essential for building robust machine learning models. The abundance and quality of the data in this region contribute to effective model training and also allow for reliable evaluation on the test wells, ensuring that the models can generalize well within the same geological context.

Chapter 5

Exploratory Data Analysis (EDA)

Objective: *Perform an Exploratory Data Analysis (EDA) to investigate the structure and distribution of the petrophysical dataset, with a particular focus on feature relationships, missing values, outliers, and data consistency.*

5.1 Introduction To Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) is a fundamental step in any data-driven projects. It represents a set of tools and techniques that allows to better understand the structure, distribution, and quality of the data to discover patterns, spot anomalies, test a hypothesis, or check assumptions [62]. It is a key step before modeling, helping to identify outliers, skewed distributions, variable correlations, and potential data quality issues such as missing or non-physical values.

In the context of this study, EDA was conducted on petrophysical well log data in order to assess data completeness, investigate feature distributions, identify anomalies, and explore the relationships between predictor variables and the target variables, Volume of Clay (V_{cl}, Effective Porosity ($PHIE$) and the water saturation (S_w). The analysis was conducted on the full dataset prior to the train-test split, and subsequent preprocessing steps, such as imputation and feature scaling, were applied using parameters derived from the training set only, to prevent data leakage.

The data analysis and cleaning processes were conducted using Python. Several libraries were used to perform data manipulation, handle missing values, visualize distributions, and compute descriptive statistics. The listing below presents the main packages required for this stage.

Listing 5.1: Main imports for EDA and data cleaning

```
# For loading and manipulating tabular data
import pandas as pd
# For numerical computations and array handling
import numpy as np
# For static plotting
import matplotlib.pyplot as plt
```

```
# For statistical data visualization
import seaborn as sns
# For statistical transformations
from scipy.stats import skew, kurtosis, boxcox
```

5.2 Data Description

5.2.1 Data Sources

The dataset used in this study consists of well log measurements acquired during the evaluation phase of hydrocarbon reservoir development In the Berkine region.
These measurements were collected pretty thoroughly down borehole using various wireline logging tools that provide super high resolution data continuously from different wells from the same field.

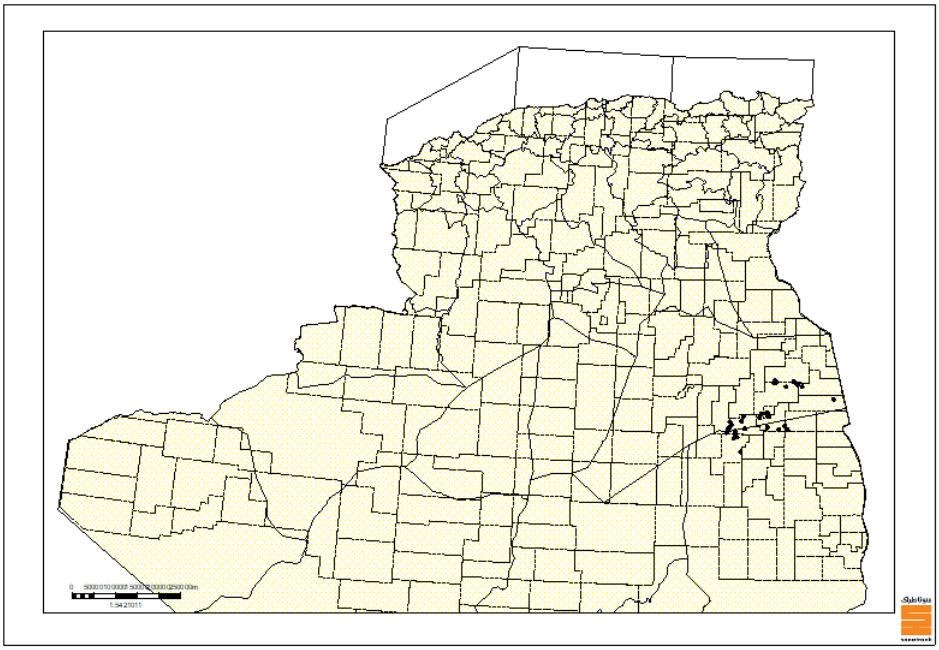


Figure 5.1: Geographical Location of the Berkine Basin[63].

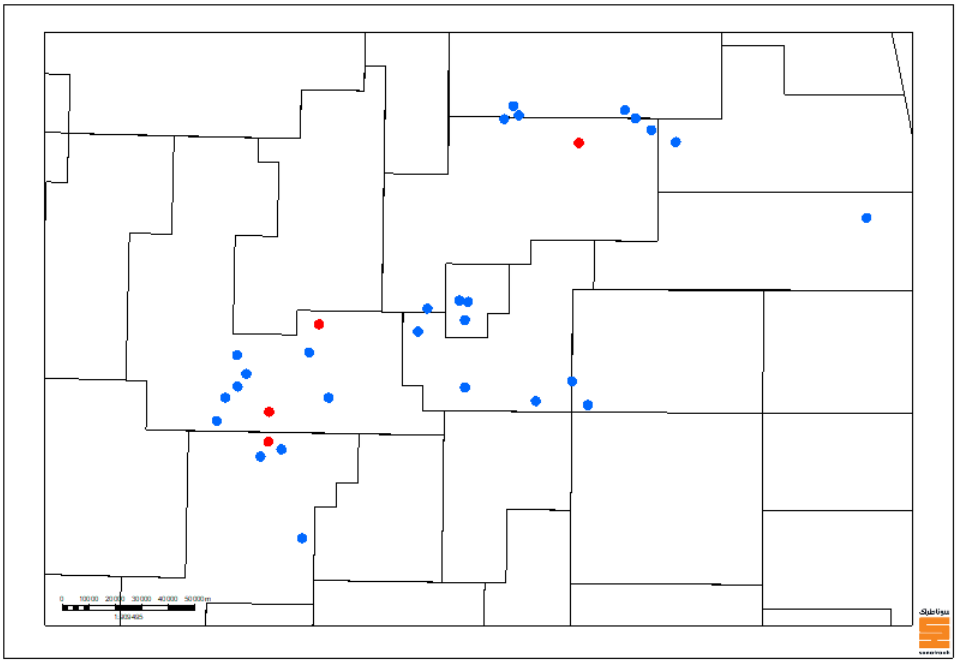


Figure 5.2: Wells distribution. [63].

Where the blue points represent the wells used in the training, and the red points represent the wells used in the testing

5.2.2 Structure of the Dataset

The combined dataset comprises a total of 91,167 data points of 19 wells in total. Each row represents a depth-specific log measurement, and the dataset includes 16 columns collected from well logging tools during the evaluation and exploration phases of subsurface formations. These features represent a variety of geophysical measurements commonly used in petrophysical analysis.

The dataset includes both input features and target variables, each group of features corresponds to a specific category of well logging tools, such as gamma-ray, sonic, resistivity, density, neutron, and spectral gamma-ray logs.
The following table presents the full list of variables grouped by their logging category, along with a brief description of each:

5.2.3 Data types

This step aims to check for the data types, as in some cases, the numerical data is stored as a string, hence, requires a conversion from string to integer or float — depending on the known type —, to be able to display plots of the data via graphs. In this case, the data extraction wasn't affected, and our data types match the nature.

Category	Features
General	Well, DEPTH
Gamma-Ray and Sonic	GR (Gamma Ray), DTP (Sonic Travel Time)
Resistivity Logs	AT10, AT20, AT30, AT60, AT90
Density & Neutron Logs	NPHI (Neutron Porosity), RHOB (Bulk Density)
Spectral Gamma Ray	URAN (Uranium), THOR (Thorium), POTA (Potassium)
Target Variables	VCL (Clay Volume), PHIE (Effective Porosity), SW (Water Saturation)

Table 5.1: Categories and corresponding features used in the dataset

Table 5.2: Types of the data in the dataset.

Column	Data Type
Well	object
DEPTH	float64
GR	float64
DTP	float64
AT10	float64
AT20	float64
AT30	float64
AT60	float64
AT90	float64
NPHI	float64
RHOB	float64
URAN	float64
THOR	float64
POTA	float64
VCL	float64
PHIE	float64
SW	float64

5.2.4 Number of Data Points Per Well

The identification of the data distribution is essential during exploratory data analysis (EDA), through a computation of data points recorded for each well, then visualisation of the results using a bar chart. This following plot 5.3 highlights a slight imbalance in the dataset, where some wells contribute with more data than others. Such disparities might impact the model performance. This analysis informs decisions related to sampling strategies and data weighting.

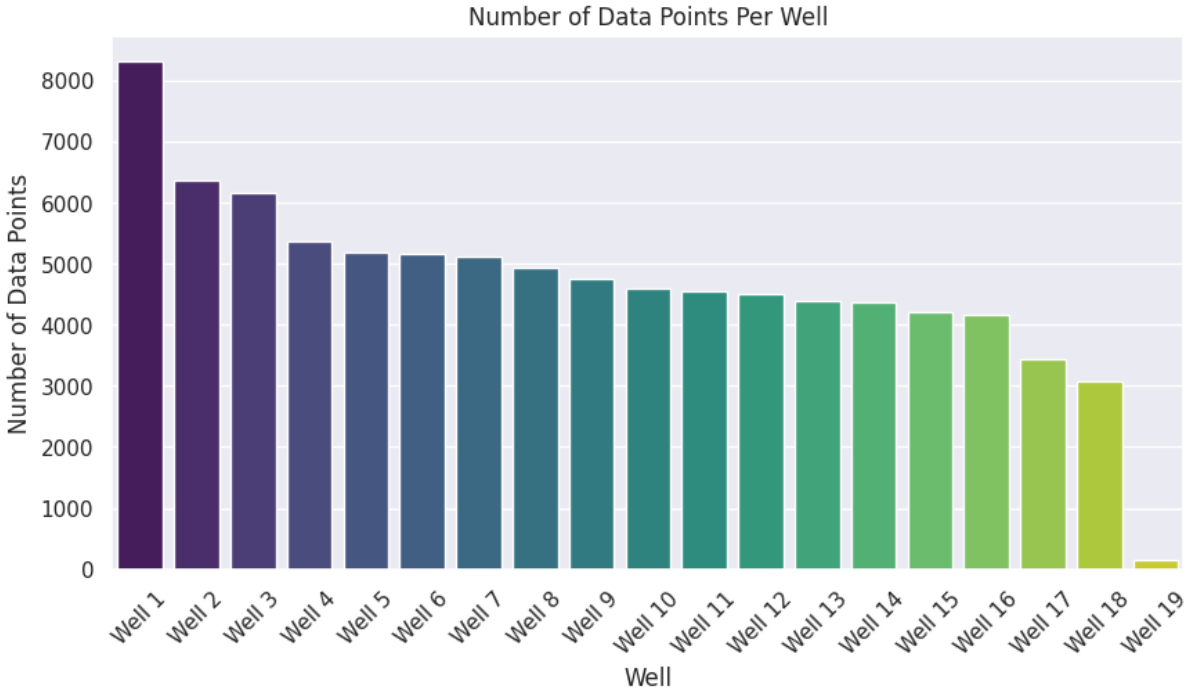


Figure 5.3: Data Point Distribution Per Well

5.2.5 Histogram of Data Points

A histogram was plotted 5.4 to explore the distribution of values for the variables. The resulting plot reveals the range, central tendency, and frequency of observed values, enabling a quick look about potential skewness, outliers, or abnormal concentrations. Most values appear clustered around a central range, suggesting a relatively consistent distribution, while tails or sparse regions may indicate anomalous readings or natural geological variability. This visual inspection is crucial for understanding the data spread and guiding the selection of the next appropriate preprocessing steps, such as normalization, transformation, or outlier handling, prior to model training.

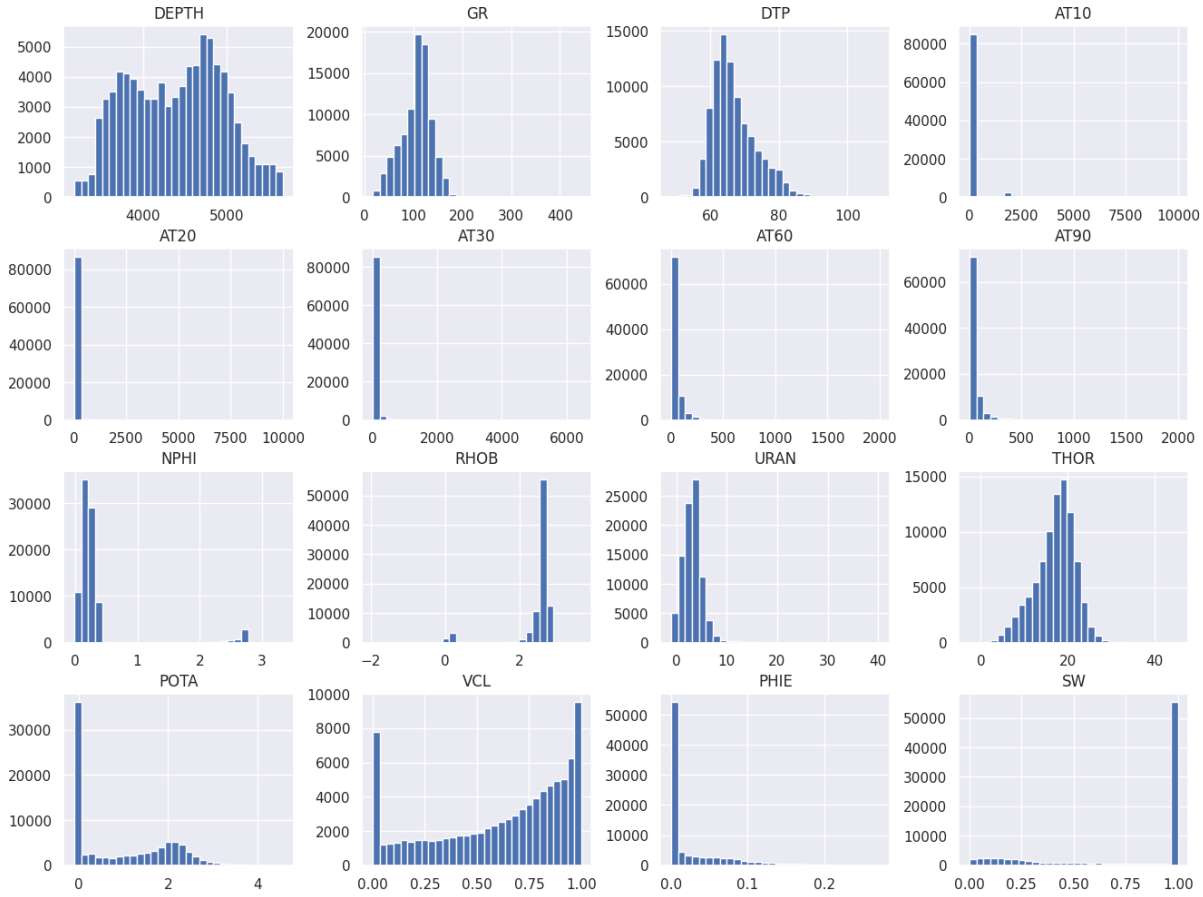


Figure 5.4: Histogram of Raw Data

5.2.6 Heatmap

A heatmap was generated to visualize the pairwise correlation coefficients between numerical features. This graphical representation helps identify linear relationships between variables, each cell in the heatmap represents the correlation coefficient (typically Pearson's r) between two variables, ranging from -1 to 1. A value close to 1 (displayed in darker warm colors) indicates a strong positive linear relationship—meaning the two features tend to increase together. A value near -1 (cooler tones) suggests an inverse relationship—when one increases, the other decreases. Values close to 0 (neutral colors) imply no linear correlation. The heatmap reveals clusters of highly correlated features, which may suggest redundancy or multicollinearity. Understanding these relationships is essential for feature selection, engineering, and model interpretation. For instance, strong correlations between porosity-related logs or radioactive indicators such as GR and V_{cl} confirm geological coherence in the dataset. Conversely, weak or near-zero correlations suggest that the variables are likely to contribute independently to the model's predictive power.

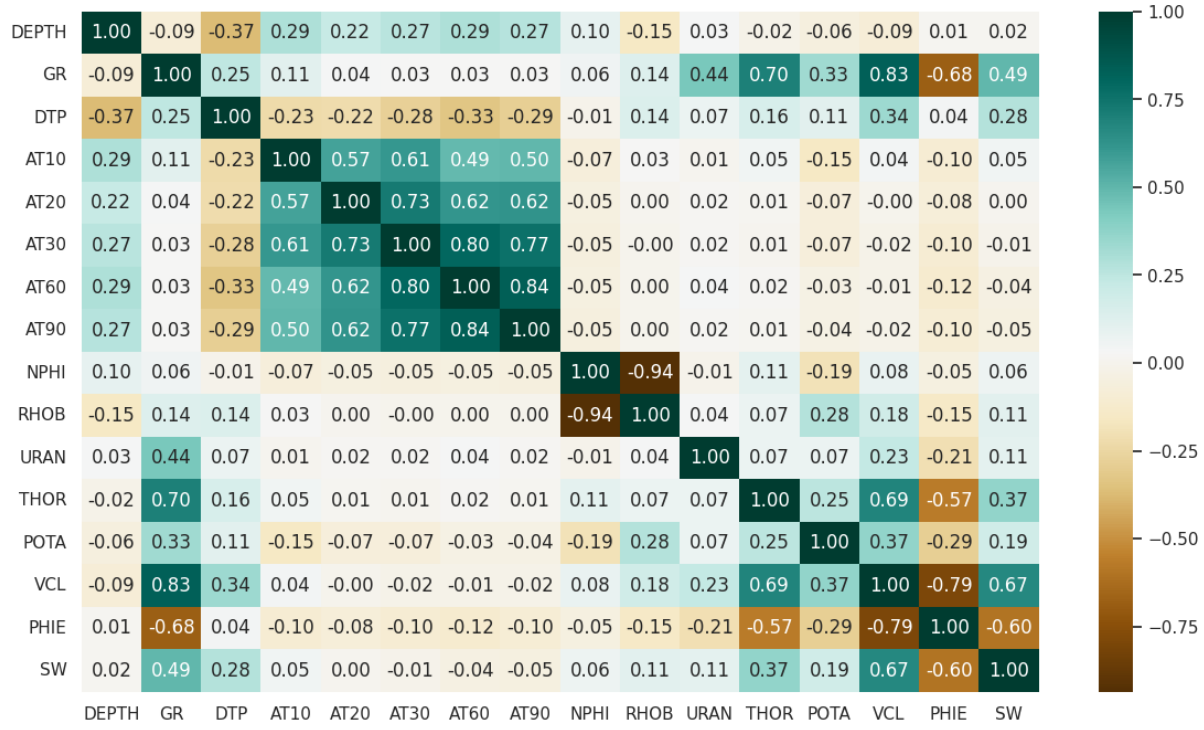


Figure 5.5: Heatmap

The heatmap enables the confirmation of data coherence by visually highlighting correlations between variables. This visualization supports our geoscientific domain expertise, allowing us to validate expected relationships (e.g., inverse correlation between porosity and clay volume) and detect any inconsistencies that may indicate data quality issues.

5.2.7 Detecting Outliers

To visually identify potential outliers in the dataset, boxplots were generated for each numerical variable using Seaborn. A boxplot summarizes the distribution of a feature by displaying five key statistics: the minimum, first quartile $Q1$, median, third quartile $Q3$, and maximum. The interquartile range ($IQR = Q3 - Q1$) defines the middle 50% of the data. Values lying outside 1.5 times the IQR from the lower $Q1$ or upper $Q3$ quartile are considered outliers and are plotted as individual points beyond the "whiskers" of the box. This visual tool provides a straightforward method to assess whether a variable exhibits skewness or contains extreme values. Identifying these outliers is essential, as they can bias statistical measures or degrade the performance of predictive models. For instance, parameters such as `AT20` or `RHOB` might display notable outliers, either due to measurement anomalies or geological variability.

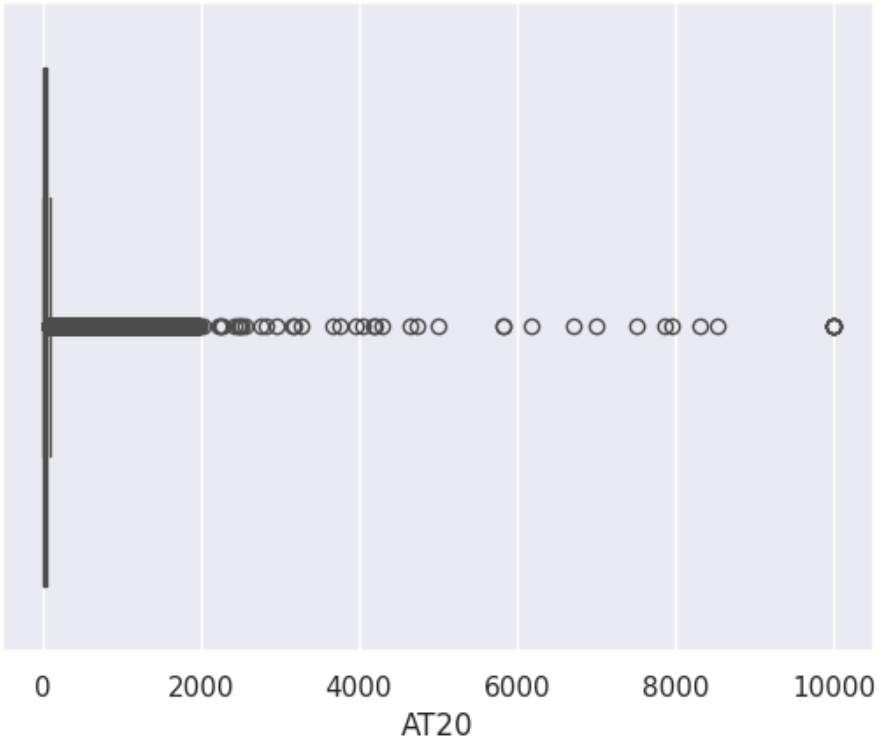


Figure 5.6: AT20 Boxplot close-up

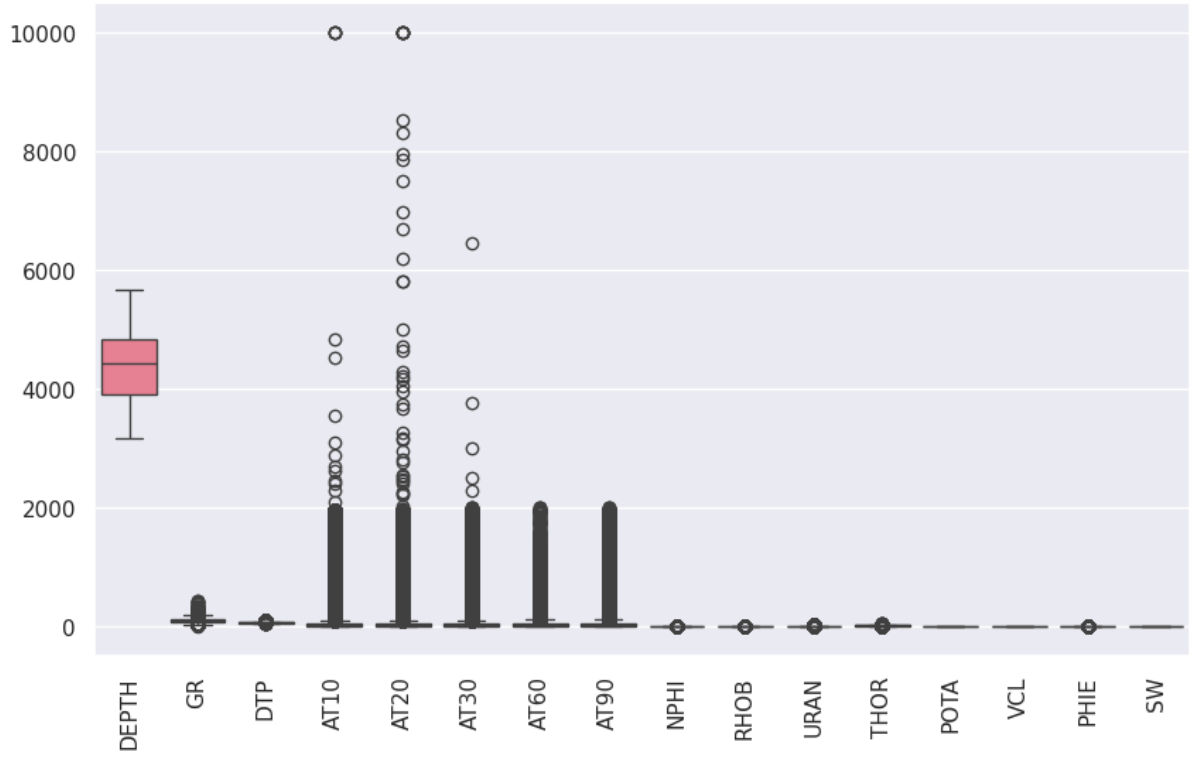


Figure 5.7: General Boxplot

5.3 Summary Statistics

The following table represents basic statistical summaries generated for each of the columns, and includes : count of the data points, mean, standard deviation (std), quartiles (25%,

50%, 75%), min and max values.

Listing 5.2: Summary Statistics

```
# Display summary statistics
summary_stats = df.describe()
print(summary_stats)
```

Table 5.3: Descriptive Statistics Of The Data.

Stat	DEPTH	GR	DTP	AT10	AT20	AT30	AT60	AT90	NPHI	RHOB	URAN	THOR	POTA	VCL	PHIE	SW
Count	88762	88762	88762	88762	88762	88762	88762	88762	88762	88762	88762	88762	88762	88762	88762	88762
Mean	4396.56	105.98	66.84	101.23	61.59	57.38	50.76	58.63	0.3402	2.4993	3.23	17.05	0.99	0.616	0.027	0.745
Std	571.13	31.66	6.36	338.36	211.63	155.99	103.87	144.44	0.5666	0.5865	2.07	4.57	1.01	0.321	0.044	0.363
Min	3178.73	16.24	48.57	0.23	0.18	0.18	0.16	0.14	-0.011	-1.948	-1.01	-2.63	-0.09	0.000	0.000	0.001
25%	3901.29	86.95	62.31	12.15	11.68	11.69	12.09	12.26	0.1382	2.5652	1.93	14.38	0.02	0.375	0.000	0.379
50%	4440.94	110.61	65.46	23.40	23.08	23.20	23.82	24.42	0.2068	2.6886	3.16	17.72	0.65	0.712	0.001	1.000
75%	4834.28	125.69	70.45	49.67	48.88	49.98	52.21	54.14	0.2820	2.7203	4.19	20.17	1.96	0.890	0.041	1.000
Max	5671.87	443.78	109.40	10000.00	10000.00	6455.08	2000.00	2000.00	3.3322	3.6709	40.31	45.43	4.57	1.000	0.272	1.000

These statistics were useful to detect inconsistencies or implausible values (e.g., negative values for resistivity or porosity above 1.0).

Table 5.4: Descriptive Statistics of The Data *(highlighting outliers and non-physical values)*

Stat	DEPTH	GR	DTP	AT10	AT20	AT30	AT60	AT90	NPHI	RHOB	URAN	THOR	POTA	VCL	PHIE	SW
Count	88762	88762	88762	88762	88762	88762	88762	88762	88762	88762	88762	88762	88762	88762	88762	88762
Mean	4396.56	105.98	66.84	101.23	61.59	57.38	50.76	58.63	0.3402	2.4993	3.23	17.05	0.99	0.6156	0.0268	0.7454
Std	571.13	31.66	6.36	338.36	211.63	155.99	103.87	144.44	0.5666	0.5865	2.07	4.57	1.01	0.3206	0.0439	0.3635
Min	3178.73	16.24	48.57	0.23	0.18	0.18	0.16	0.14	-0.011	-1.95	-1.01	-2.63	-0.09	0.0000	0.0000	0.0013
25%	3901.29	86.95	62.31	12.15	11.68	11.69	12.09	12.26	0.1382	2.5652	1.93	14.38	0.0213	0.3746	0.0000	0.3793
50%	4440.94	110.61	65.46	23.40	23.08	23.20	23.82	24.42	0.2068	2.6886	3.16	17.72	0.6513	0.7115	0.0010	1.0000
75%	4834.28	125.69	70.45	49.67	48.88	49.98	52.21	54.14	0.2820	2.7203	4.19	20.17	1.9600	0.8905	0.0412	1.0000
Max	5671.87	443.78	109.40	10000.00	10000.00	6455.08	2000.00	2000.00	3.3322	3.6709	40.31	45.43	4.567	1.0000	0.2722	1.0000

Summary statistics revealed a consistent sample size across all features, indicating no missing rows at this stage. However, extreme values were identified in several attributes. Notably, the attenuation logs (AT10, AT20, AT60, and AT90) exhibited implausibly high values (e.g., 10,000 or 1999.999), likely representing fill values. Additionally, negative readings were observed in physical properties such as neutron porosity (NPHI), bulk density (RHOB), and radioactive elements (URAN, THOR, POTA), which are physically unrealistic and indicative of either measurement errors or placeholder values. These outliers were flagged and treated as missing values. The effective porosity (PHIE) showed very low overall values, suggesting either poor reservoir quality or incorrect scaling.

5.4 Data Cleaning and Preprocessing

The cleaning and preprocessing steps were implemented in Python using `pandas`, `numpy`, and `scipy` libraries, and are summarized below.

5.4.1 Handling Missing and Non-Physical Values

The initial step consists in identifying and quantifying missing values in each column. This involves comparing the number of missing entries to the total number of rows in the

dataset. Based on the resulting proportion, an informed decision is made on whether to discard, impute, or retain the affected records, depending on their potential impact on the analysis.

Table 5.5: Missing values per feature after final cleaning

Variable	Number of Missing Values
VCL	2405
PHIE	2405
SW	2405
Well	0
DEPTH	0
GR	0
DTP	0
AT10	0
AT20	0
AT30	0
AT60	0
AT90	0
NPHI	0
RHOB	0
URAN	0
THOR	0
POTA	0

Many logging tools return default values when measurements are unreliable. These include:

- Fill values (placeholders for missing data) such as `-999.25`, `-999`, `10000.0`, `1999.999`
- Negative values in physical parameters like `RHOB`, `NPHI`, and spectral logs (`URAN`, `THOR`, `POTA`), which are not physically meaningful

These values were treated as missing (`NaN`) and removed from the dataset.

Listing 5.3: Replacing Fill Values and Negative Measurements

```
fill_values = [10000.0, 1999.999, -999.25, -999.0]
df.replace(fill_values, np.nan, inplace=True)

columns_to_check_negatives = ['NPHI', 'RHOB', 'URAN', 'THOR', 'POTA']
for col in columns_to_check_negatives:
    df.loc[df[col] < 0, col] = np.nan
```

A summary of missing values was computed after cleaning:

Listing 5.4: Missing Values Report

```
missing_counts = df.isnull().sum()
missing_percent = 100 * missing_counts / len(df)

missing_df = pd.DataFrame({
    'Missing Values': missing_counts,
    'Percentage (%)': missing_percent.round(2)
})

missing_df = missing_df[missing_df['Missing Values'] > 0].sort_values(by='Miss
print(missing_df)
```

Example: After cleaning, `RHOB` had **0.01%** missing values, and `URAN` had **0.03%**. Table 5.6 summarizes the missing value statistics for all features.

Table 5.6: Number of missing values after cleaning

Variable	Nb of Missing Values	Percentage (%)
SW	2405	2.71
PHIE	2405	2.71
VCL	2405	2.71
URAN	31	0.03
NPHI	25	0.03
THOR	16	0.02
POTA	15	0.02
RHOB	8	0.01
AT20	6	0.01
AT10	4	0.00
Well	0	0.00
AT90	0	0.00
AT60	0	0.00
DEPTH	0	0.00
GR	0	0.00
AT30	0	0.00
DTP	0	0.00

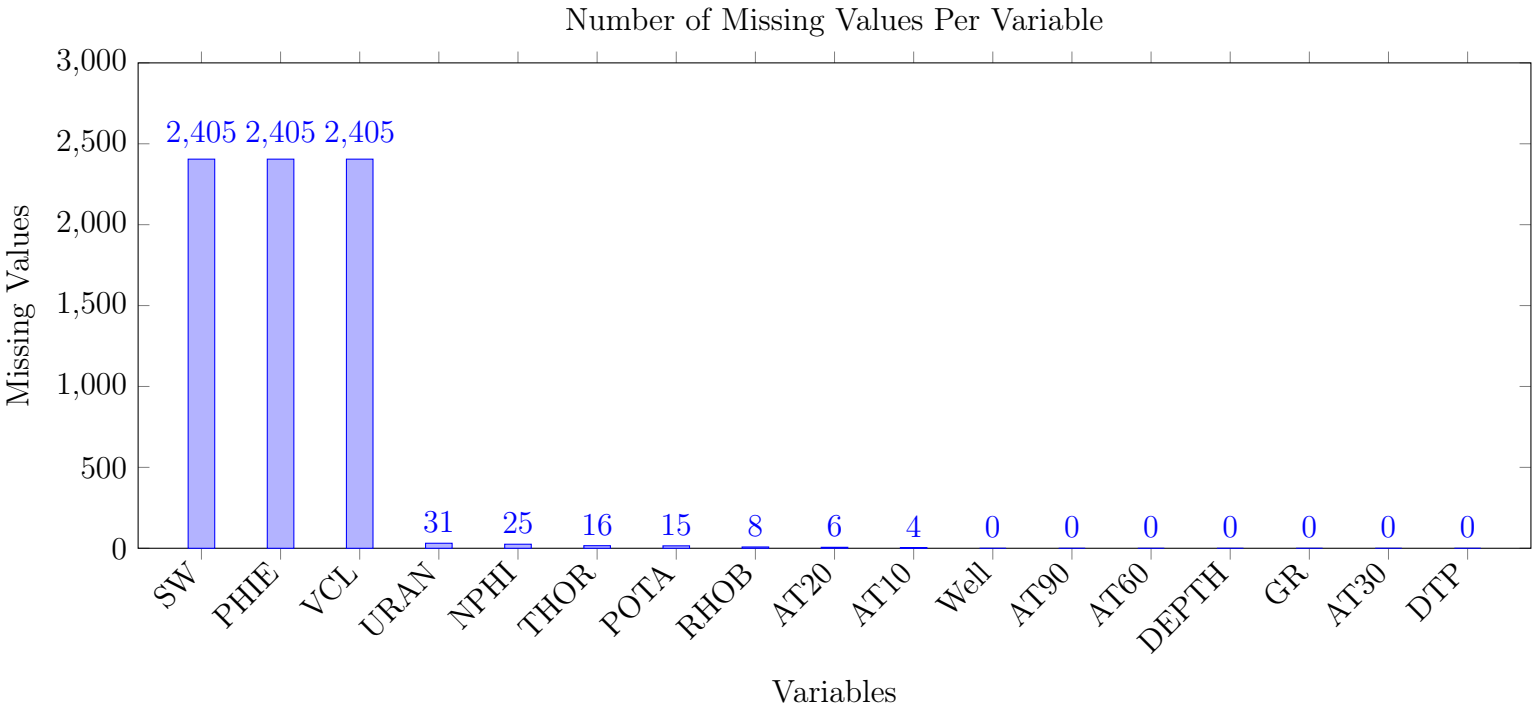


Figure 5.8: Visualization of missing values per variable, after cleaning

5.4.2 Variable Transformation

5.4.2.1 Skewness and Kurtosis of Numerical Features

To better understand the shape of the distributions of the numerical features, we compute their **skewness** (asymmetry) and **kurtosis** (tailedness). High skewness values indicate asymmetric distributions, while high kurtosis values reflect heavy tails or outliers.

- **Skewness** indicates the asymmetry of the distribution:
  - A value near 0 implies a symmetric distribution.
  - Positive skew indicates a long tail to the right.
  - Negative skew indicates a long tail to the left.
- **Kurtosis** measures the tailedness of the distribution:
  - **A value of 3**: corresponds to a normal distribution.
  - **Greater than 3**: heavy tails (outliers more likely).
  - **Less than 3:** light tails.

The following Python code summarizes the skewness and kurtosis of all numerical columns in the cleaned dataset:

Listing 5.5: Computation of Skewness and Kurtosis for Numerical Variables

```
# Skewness and kurtosis
skew_kurt = pd.DataFrame({
    'Skewness': df[numerical_cols].skew(),
    'Kurtosis': df[numerical_cols].kurtosis()
})

print(skew_kurt)
```

The results are summarized in Table 5.7.

Table 5.7: Skewness and Kurtosis of Numerical Variables

Feature	Skewness	Kurtosis
DEPTH	0.016	-0.935
GR	-0.084	1.744
DTP	0.835	0.556
AT10	5.940	56.318
AT20	17.094	541.777
AT30	9.267	120.761
AT60	8.867	116.943
AT90	8.552	91.795
NPHI	3.756	12.587
RHOB	-3.528	11.302
URAN	2.601	23.460
THOR	-0.497	0.259
POTA	0.469	-1.221
VCL	-0.644	-0.891
PHIE	1.908	3.306
SW	-0.918	-0.920

Each feature exhibits a distinct statistical distribution, such as symmetry, skewness, or heavy tails. As a result, a one-size-fits-all transformation approach is often inadequate. Table 5.8 summarizes the skewness types and appropriate transformations for selected features.

Table 5.8: Recommended transformations for selected features based on skewness

Feature	Skewed Type	Best Fit Transform
PHIE	Moderate right skew	Log or Box-Cox
AT20	Extreme right skew	Log (safer than Box-Cox for heavy-tailed data)
RHOB	Left skewed	Cube root (works for negatives too)

Following the recommended processes, log, Box-Cox, and cube-root transformations were applied:

Listing 5.6: Transformations of `PHIE`, `AT20`, and `RHOB`

```
from scipy.stats import boxcox

min_phie = df['PHIE'].min()
shift = 1e-5 if min_phie > 0 else abs(min_phie) + 1e-5
phie_shifted = df['PHIE'] + shift

df['PHIE_log'] = np.log1p(df['PHIE'])
df['AT20_log'] = np.log1p(df['AT20'])

phie_shifted_finite = phie_shifted.dropna()
phie_bc_transformed_values, fitted_lambda = boxcox(phie_shifted_finite)
df['PHIE_bc'] = np.nan
df.loc[phie_shifted_finite.index, 'PHIE_bc'] = phie_bc_transformed_values

df['RHOB_sym'] = np.cbrt(df['RHOB'])
```

5.4.2.2 Visual Comparison of Original vs Transformed Variables

Distributions before and after transformation were compared using histograms:

Listing 5.7: Histogram Plots of Original vs Transformed Features

```
import seaborn as sns
import matplotlib.pyplot as plt

plot_pairs = [
    ('PHIE', 'PHIE_log', 'PHIE: Original vs Log'),
    ('AT20', 'AT20_log', 'AT20: Original vs Log'),
    ('RHOB', 'RHOB_sym', 'RHOB: Original vs Cube Root'),
]

for original, transformed, title in plot_pairs:
    plt.figure(figsize=(12, 4))

    plt.subplot(1, 2, 1)
    sns.histplot(df[original].dropna(), kde=True, bins=50)
    plt.title(f'Original {original}')

    plt.subplot(1, 2, 2)
    sns.histplot(df[transformed].dropna(), kde=True, bins=50)
    plt.title(f'Transformed: {transformed}')
```

```
    plt.suptitle(title, fontsize=14)
    plt.tight_layout(rect=[0, 0, 1, 0.95])
    plt.show()
```

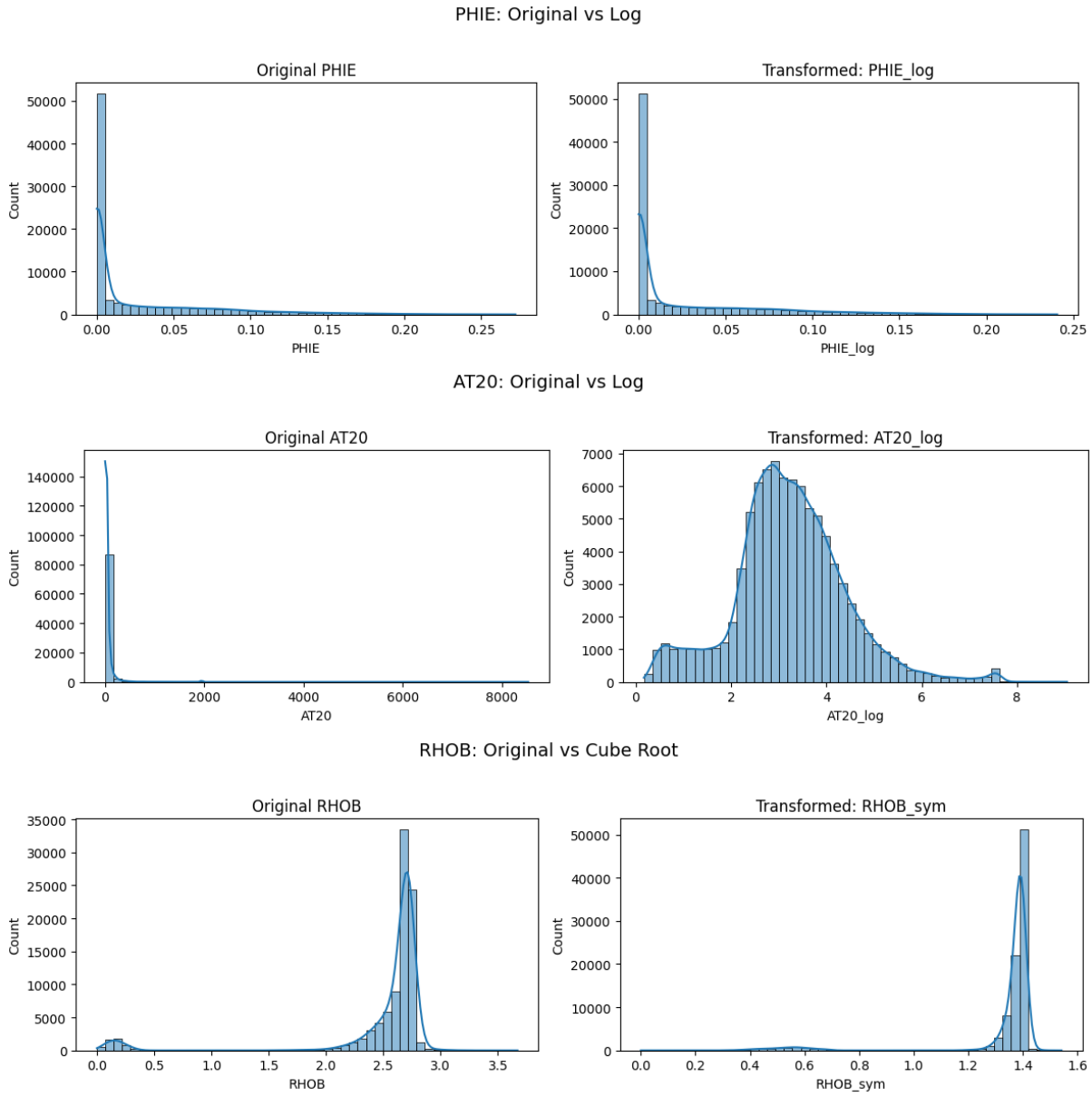


Figure 5.9: Original VS. Transformed Data

5.4.2.3 Final Cleanup

Finally, all rows containing remaining `NaN` values were removed to ensure consistency during model training. Given that the maximum number of rows affected by missing values was 2,405 out of a total of 91,167 (2.71%), the loss of data was considered negligible :

Listing 5.8: Drop Missing Values

```
df.dropna(inplace=True)
print(df.isnull().sum().sort_values(ascending=False))
```

Table 5.9: Number of missing values after dropping rows with `NaN`

Variable	Number of Missing Values
Well	0
DEPTH	0
GR	0
DTP	0
AT10	0
AT20	0
AT30	0
AT60	0
AT90	0
NPHI	0
RHOB	0
URAN	0
THOR	0
POTA	0
VCL	0
PHIE	0
SW	0

5.5 Training and Testing Split

The data is divided into two separate Excel sheets, each one has a distinct purpose in the machine learning Process:

5.5.1 Training Dataset:

This sheet contains data of 15 wells (which represents 78.95%) of the database and is used to train and calibrate the machine learning models.
The goal is to enable the models to learn the underlying relationships between the well log inputs and the target petrophysical properties.

5.5.2 Testing Dataset:

This subset envelopes data of 4 wells (which represents 21.05%) that were not part of the training process. It is used to evaluate the model's performance and to test its ability to generalize to new, unseen data.
By assessing the model on different wells, we simulate a real-world scenario and verify whether the model can be applied to other wells with similar characteristics.

5.6 Conclusion:

The dataset has been thoroughly explored and meticulously cleaned, bringing it to the best possible state given its initial quality. Various data issues –such as invalid fill values,

negative measurements, and pronounced skewness–were identified and addressed through appropriate imputation techniques, filtering, and feature transformations (including logarithmic, Box-Cox, and cube root methods). Although certain crucial measurements such as pressure, bit size, and caliper data were missing, the remaining dataset retains a rich array of petrophysical information. The variables preserved after cleaning offer sufficient depth and reliability to support the development of accurate predictive models. However, according to these results, a preliminary challenge was identified: a significant imbalance in the distributions of both water saturation (S_w) and effective porosity ($PHIE$). Recognizing this early on is essential, as it directly informs model design, training strategies, and evaluation metrics. Overall, this careful data preparation lays a robust foundation for meaningful and interpretable machine learning outcomes.

Chapter 6

Pipeline Steps

*(**Objective:** Explain the full machine learning workflow used to predict petrophysical parameters specifically water saturation (S_w), clay volume (V_{cl}), and effective porosity ($PHIE$) and, outline each essential stage of the pipeline, including data preprocessing, feature engineering, and model training.)*

6.1 Preprocessing

The data preprocessing procedures were previously carried out and described in detail in the Exploratory Data Analysis **(EDA)**, Chapter Five (Exploratory Data Analysis (EDA)). This step included cleaning the raw well log dataset by removing null or invalid values, correcting inconsistent formats, and validating the measurements to ensure data integrity. The resulting dataset was used as the foundation for building and training the machine learning models.

6.2 Feature Engineering

After data cleaning, a selection of relevant geophysical and petrophysical logs was made to serve as input features for the machine learning models.

Table 6.1: Selected input features and predicted output parameters

Input Features	**Output Parameters**
Gamma Ray (GR) Sonic Transit Time (DTP) Resistivity Measurements: AT10, AT20, AT30, AT60, AT90 Neutron Porosity (NPHI) Bulk Density (RHOB) Natural Radioactivity Logs: Uranium (URAN), Thorium (THOR), Potassium (POTA)	Water Saturation (S_w) Volume of Clay (V_{cl}) Effective Porosity ($PHIE$)

Then, the dataset was split into training and testing subsets, allowing the evaluation

of model performance while ensuring that the test set remains unseen during training. Subsequently, the training dataset was further divided into training and validation sets using a 70: 30 ratio to optimize model development and prevent overfitting.

6.3 Scaling & Transformation

Data Loading: Datasets in Excel format are imported using the `pandas.read_excel()` function. These datasets include several input features and one or more output targets.

Feature Selection and Scaling: A subset of relevant features is selected, and the data is normalized using the `RobustScaler` from `Scikit-learn`. This scaling method transforms each feature x to reduce the influence of outliers using the following formula:

$$x' = \frac{x - \text{median}(x)}{\text{IQR}(x)}$$

where median(x) is the median of the feature and IQR(x) is the interquartile range. Due to the presence of outliers and skewed distributions in petrophysical features such as porosity and spectral gamma ray measurements, the `RobustScaler` was selected for feature scaling. Unlike the `StandardScaler`, which centers the data using the mean and scales it to unit variance, the `RobustScaler` relies on the median and interquartile range IQR, making it significantly less sensitive to extreme values. This is particularly important in petrophysical datasets, where measurement anomalies and tool limitations can lead to unrealistic spikes or heavy-tailed distributions. By using `RobustScaler`, the scaling process preserves the integrity of the data and improves the stability and performance of subsequent machine learning models.

6.4 Model Fitting

6.4.1 Linear Regression (LR)

A model was employed as a baseline model for predicting petrophysical parameters, namely water saturation (S_w), volume of clay (V_{cl}), and effective porosity ($PHIE$). The model was trained using `Scikit-learn's LinearRegression` class with default hyperparameter settings.
The linear regression model learns a linear mapping between the input features and the target variable based on the following equation:

$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n$$

- $\hat{y}$: Predicted value (S_w, V_{cl}, or $PHIE$)
- x_i: Input features (e.g., GR, DTP, RHOB, etc.)

- β_i: Learned coefficients for each feature
- β_0: Intercept term

6.4.2 XGBoost

The XGBoost model used in this study is a gradient boosting decision tree model optimized for regression tasks. Its architecture relies on an ensemble of decision trees, where each tree attempts to correct the residual errors of the previous ones.

Architecture & Model Parameters

The following hyperparameters were used to configure the XGBoost model. Each plays a specific role in controlling the learning process, model complexity, and generalization performance:

- **n_estimators**
  Defines the number of boosting rounds; each round builds a new tree to reduce previous prediction errors.
- **max_depth**
  Sets the maximum depth of individual trees; controls model complexity and helps prevent overfitting.
- **colsample_bytree**
  Specifies the fraction of features randomly selected for each tree; improves generalization and reduces overfitting.
- **subsample**
  Indicates the fraction of training samples used per boosting round; adds regularization and enhances robustness.
- **random_state**
  Ensures reproducibility of results by setting a fixed random seed.

Table 6.2: XGBoost Architecture and Model Parameters

Parameter	Value
n_estimators	100
max_depth	6
colsample_bytree	0.8
subsample	0.8
random_state	42

Training parameters

Key parameters such as boosting rounds and learning rate define the learning dynamics and convergence speed of the XGBoost model.

- **Boosting Rounds**
  This refers to the number of iterations the XGBoost algorithm performs to sequentially build decision trees. In each round, the model focuses on correcting the errors made by the previous trees.

- **Learning Rate**
  The learning rate controls how much each new tree contributes to the final prediction.

Table 6.3: XGBoost Training Parameters

Parameter	Value
Training Strategy	100 boosting rounds
Learning Rate	0.05

Optimization & Regularization Settings

XGBoost uses a method called gradient boosting, where it builds decision trees one after another to improve predictions.Parameters like subsample and colsample_bytree help by randomly selecting data and features, which makes the model more robust.

- **Gradient Boosting**
  It builds a model sequentially, where each new decision tree corrects the errors made by the previous ones. This is done by minimizing a loss function using gradient descent.

- **Subsample**
  Uses a random portion of the training data for each tree. This helps prevent overfitting by introducing randomness and diversity.

- **Colsample_Bytree**
  Randomly selects a subset of features for each tree. Like subsample, this reduces overfitting and increases model robustness.

Table 6.4: XGBoost Optimization and Regularization Settings

Parameter	Value
Boosting Strategy	Gradient Boosting
Regularization Method	subsample and colsample_bytree
colsample_bytree	0.8
subsample	0.8

Hyperparameter Tuning: Grid Search vs Random Search

Optimizing hyperparameters is a key step in improving the performance of an XGBoost model. Two used strategies for this task are **Grid Search** and **Random Search**.

- **Grid Search**:
  - Exhaustively tests all possible combinations from a predefined grid of hyperparameter values.
  - Guarantees that the best combination (within the grid) is evaluated.
  - However, it becomes computationally expensive when the search space is large or includes many parameters to test.
- **Random Search**:
  - Randomly samples a fixed number of combinations from the full hyperparameter space.
  - Significantly faster than Grid Search and often sufficient to find near-optimal settings.
  - Recommended when time or computational resources are limited.

6.4.3 Hyperparameter Tuning Strategy

To optimize the performance of our XGBoost models, we employed a two-step hyperparameter tuning approach combining both **Randomized Search** and **Grid Search**.

We began with a **RandomizedSearchCV** to explore a broad range of hyperparameter combinations efficiently. This method samples a fixed number of parameter settings from specified distributions, making it particularly useful for identifying promising regions in the hyperparameter space in a limited time and computational resources.

Once the most influential parameters and their approximate ranges were identified, we refined our search using a more focused **GridSearchCV**. In this second step, we performed an exhaustive search over a narrower grid centered around the best-performing candidates from the randomized search.

This two-phase strategy allowed us to balance exploration and precision: leveraging the speed of random search to detect good zones, followed by the exhaustive nature of grid search to fine-tune model performance.

This approach enabled us to efficiently converge toward a promising model architecture without the need to evaluate possible hyperparameter combination.

Listing 6.1: Hyperparameter tuning using GridSearchCV for XGBoost

```
from sklearn.model_selection import GridSearchCV
from xgboost import XGBRegressor

# Define the grid of hyperparameters to search
param_grid = {
    'n_estimators': [50, 100],            # Number of boosting rounds
    'max_depth': [3, 6],                  # Maximum depth of a tree
    'learning_rate': [0.01, 0.05, 0.1],   # Step size shrinkage
    'subsample': [0.8, 1.0],              # Subsample ratio of the training ins
    'colsample_bytree': [0.8, 1.0]        # Subsample ratio of columns when cor
}

# Initialize the XGBoost regressor
xgb = XGBRegressor(random_state=42)

# Set up the GridSearchCV
grid_search = GridSearchCV(
    estimator=xgb,
    param_grid=param_grid,
    scoring='neg_mean_absolute_error',  # Evaluation metric (can be changed to
    cv=3,                               # 3-fold cross-validation
    verbose=1,                          # Print progress messages
    n_jobs=-1                           # Use all available CPU cores
)

# Fit the model on training data
grid_search.fit(X_train_reg, y_train_reg)
```

GridSearchCV was performed over 48 parameter combinations using a 3-fold cross-validation, resulting in a total of 144 fits. The best model was obtained with the following hyperparameters: `colsample_bytree = 1.0`, `learning_rate = 0.05`, `max_depth = 6`, `n_estimators = 100`, and `subsample = 0.8`.

Feature Importance in XGBoost

XGBoost, being a tree-based ensemble method, inherently supports the estimation of feature importance during training. This importance reflects the contribution of each input feature in predicting the target variable, and can be computed using different strategies:

- **Weight (Frequency):** The number of times a feature is used to split the data across all boosting trees. A higher count implies that the feature is more frequently selected, though not necessarily more informative.

- **Gain:** The average improvement in the loss function brought by splits using the feature. This is often considered the most relevant measure of importance, as it directly reflects predictive power.

- **Cover:** The number of observations affected by a particular feature when it is used to split data. It is weighted by the number of data points passing through those splits.

Feature importance in XGBoost can be visualized using the `plot_importance` function from the `xgboost` package:

Listing 6.2: Plotting feature importance based on gain

```
xgb.plot_importance(model, importance_type='gain')
```

Which shows :

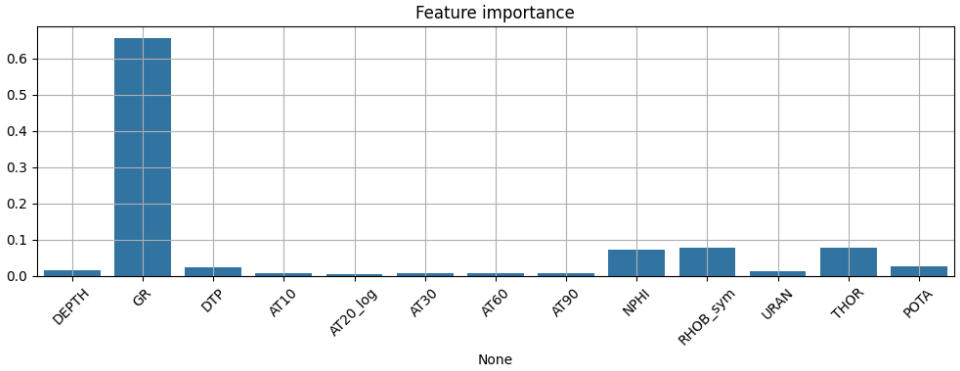


Figure 6.1: V_{cl} Feature Importance

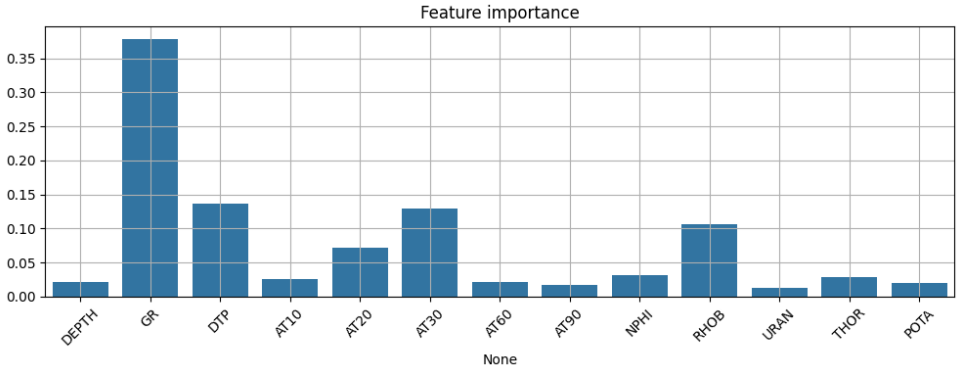


Figure 6.2: $PHIE$ Feature Importance

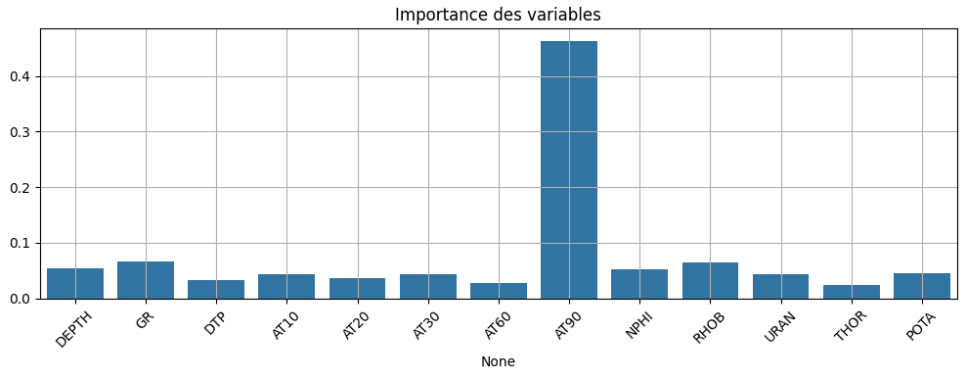


Figure 6.3: S_w Feature Importance.

This analysis helps in understanding the internal decision logic of the model and in performing feature selection for optimization or interpretability purposes.

Feature importance is available in various other models, particularly tree-based algorithms such as Random Forest, LightGBM, and CatBoost.

6.4.4 Deep Learning (MLP & CNN)

Architecture & Model Parameters

In a deep learning model, the architecture defines the overall structure and flow of data through the network.
It typically consists of three main types of layers and activation functions:

- **Input Layer**
  This is the first layer of the model, where data is introduced into the network. Each neuron in the input layer corresponds to one feature in the dataset neurons.

- **Hidden Layers**
  These are the intermediate layers between the input and output layers. They are composed of neurons that apply transformations to the input data using weights, biases, and activation functions. The number of hidden layers and neurons per layer determines the depth and capacity of the model.

- **Output Layer**
  This layer produces the final predictions. Its size depends on the number of output variables. For regression tasks such as predicting water saturation (S_w), clay volume (V_{cl}), or effective porosity ($PHIE$)

- **ReLU Activation Function**
  The `Rectified Linear Unit (ReLU)` is a widely used activation function in deep learning models. It outputs zero for any negative input and returns the input directly if it is positive. This function introduces non-linearity into the network while maintaining computational efficiency, helping models learn complex patterns

without vanishing gradient issues, based on the following equation:

$$\text{ReLU}(z) = \max(0, z)$$

Where:

- z: Linear output before applying the sigmoid, typically computed as $z = \mathbf{w}^\top \mathbf{x} + b$.
- $\mathbf{x}$: Input feature(s) to the neuron.
- $\mathbf{w}$: The learned weights.
- $\mathbf{b}$: The bias

- **Sigmoid Activation Function**
  The `Sigmoid function maps` input values to a range between 0 and 1. It is especially useful in binary classification problems or in the output layer when probabilities are needed using the update rule:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Where:

- z: Linear output before applying the sigmoid, typically computed as $z = \mathbf{w}^\top \mathbf{x} + b$.

Propagation Mechanisms

- **Forward Propagation**
  It is the process where the input data passes through the layers of the neural network. At each layer, the data is transformed using weights, biases, and activation functions. This continues until the model produces a prediction. The output of each layer l is computed by:

$$z^{(l)} = W^{(l)} a^{(l-1)} + b^{(l)}$$

$$a^{(l)} = \phi(z^{(l)})$$

where:

- $W^{(l)}$ and $b^{(l)}$ are the weights and biases,
- $a^{(l-1)}$ is the activation from the previous layer,
- ϕ is the activation function (ReLU or sigmoid).

- **Backpropagation**
  comes after the output is generated. It compares the predicted output to the actual

target and calculates the error. Then, it moves backward through the network, updating the weights using optimization algorithms like gradient descent as followed:

$$\frac{\partial \mathcal{L}}{\partial W^{(l)}} = \frac{\partial \mathcal{L}}{\partial a^{(l)}} \cdot \frac{\partial a^{(l)}}{\partial z^{(l)}} \cdot \frac{\partial z^{(l)}}{\partial W^{(l)}}$$

We update the weights by subtracting the learning rate times the gradient.

$$W^{(l)} \leftarrow W^{(l)} - \eta \frac{\partial \mathcal{L}}{\partial W^{(l)}}$$

Where:

- η is the learning rate.

Training parameters

- **Epochs**
  An epoch represents one complete pass through the entire training dataset. During each epoch, the model updates its weights based on the error it makes. Using more epochs allows the model to learn more deeply, but too many can lead to overfitting.

- **Batch Size**
  Batch size refers to the number of samples the model processes before updating its weights, this helps speed up training and stabilize the optimization process.

- **Validation Split**
  The validation split determines the portion of the training data that is set aside for validation. A value of 0.3 means 30% of the training data is used to evaluate the model's performance during training, helping to detect overfitting or underfitting.

- **Loss Function (Mean Squared Error - MSE)** The `loss function` measures how well the model's predictions match the actual values. Mean Squared Error calculates the average squared difference between predicted and actual values, penalizing larger errors more. It is commonly used for regression problems like predicting S_w, V_{cl}, and $PHIE$.

Optimization and Regularization

To ensure efficient and stable training of the deep learning model, several optimization and regularization strategies were implemented. These include the use of an adaptive optimizer, a carefully selected learning rate, and a checkpoint mechanism to preserve the best-performing model.

- **Optimizer**
  This algorithm helps the model learn by adjusting weights to reduce error. `RMSprop`

is good at handling changes in the learning process by adapting how fast or slow the model learns.

- **Learning Rate**
  This controls how big each step is when the model updates itself.

- **Checkpoint Callback**
  This saves the best version of the model during training, so if the performance worsens later, the best one is kept and used.

Multilayer Perceptron (MLP)

The development process followed the stages below, using `Python` and `TensorFlow/Keras frameworks`

1. **Architecture & Model Parameters**
   The Multilayer Perceptron (MLP) used as a simple yet effective feedforward neural network.
   The architecture is composed of three fully connected layers ,and in this approach, three Architectures were used to choose the best combinaison:

Table 6.5: MLP Architecture with 16 Neurons per Hidden Layer

Layer Index	Layer Type	Neurons	Activation
1	Dense	16	ReLU
2	Dense	16	ReLU
3	Dense	1	Sigmoid

Table 6.6: MLP Architecture with 32 Neurons per Hidden Layer

Layer Index	Layer Type	Neurons	Activation
1	Dense	32	ReLU
2	Dense	32	ReLU
3	Dense	1	Sigmoid

Table 6.7: MLP Architecture with 64 Neurons per Hidden Layer

Layer Index	Layer Type	Neurons	Activation
1	Dense	64	ReLU
2	Dense	64	ReLU
3	Dense	1	Sigmoid

The `ReLU function` outputs the input directly if it is positive, and zero if it is negative. It makes the model train faster and helps solve the vanishing gradient problem.
The sigmoid function is used in the output layer to constrain the output between 0 and 1, aligning with the expected range of V_{cl} values.

2. **Training parameters:**
   During the training phase of the MLP model, several key hyperparameters were defined to control the learning process.
   The main hyperparameters used in this configuration are:

Table 6.8: Training Hyperparameters for the MLP Model

Parameter	Value
Epochs	100
Batch Size	32
Validation Split	0.3
Loss Function	Mean Squared Error (MSE)

3. **Optimization and Regularization Settings**
   To enhance training, optimization and regularization techniques were applied, including the `RMSprop optimizer`, a `learning rate`, and a `checkpoint callback` to retain the best model.

Table 6.9: Training Configuration Parameters

Parameter	Value
Optimizer	RMSprop
Learning Rate	0.001
Checkpoint Callback	Enabled

Convolutional Neural Network (CNN)

1. **Architecture and Model Parameters:**
   The model is based on a one-dimensional convolutional neural network (Conv1D), designed to process well log data treated as sequences of features, and extract high-level patterns relevant for regression tasks.

Table 6.10: Summary of Conv1D model architecture

Layer Type	Configuration	Details
Input	Shape = (1, 14)	1 timestep, 14 features
Conv1D	Filters = 32, Kernel size = 1	Activation = ReLU
Batch Normalization	–	–
Conv1D	Filters = 32, Kernel size = 1	Activation = ReLU
Batch Normalization	–	–
Conv1D	Filters = 32, Kernel size = 1	Activation = ReLU
Batch Normalization	–	–
Global Max Pooling1D	–	Reduces spatial dimensions
Dense	Units = 32	Activation = ReLU
Dropout	Rate = 0.3	Regularization
Dense	Units = 32	Activation = ReLU
Output	Units = 1	Linear activation (for regression)

2. **Training Parameters:**
   This table outlines the essential settings used during the training process chose according to the tuning, including the choice of optimizer, learning rate, and loss/metric functions used to evaluate model performance.

Table 6.11: Training configuration

Parameter	**Value**
Loss Function	Mean Squared Error (MSE)
Optimizer	Adam
Learning Rate	0.001
Evaluation Metric	Mean Absolute Error (MAE)

3. **Optimization and Regularization Techniques:**
   To ensure both fast convergence and generalization to unseen data, the model integrates several optimization and regularization strategies, as shown below.

Table 6.12: Regularization and optimization components

Technique	**Purpose**
Batch Normalization	Stabilizes and accelerates training by normalizing intermediate activations
Dropout	Prevents overfitting by randomly dropping units during training (rate = 0.3)
Global Max Pooling	Reduces dimensionality and captures dominant features across time steps

6.5 Prediction

After training the models, predictions were conducted on a separate test dataset consisting of four wells, where each well was evaluated individually to preserve geological coherence and better observe the model's behaviour across different reservoir conditions.

The input features were normalized using the same `RobustScaler` applied during training for consistency.
Then, the trained model predicted the target variable (V_{cl}, $PHIE$, S_w) at each depth point in the well.
For each well, results were visualized through:

- A depth-wise plot comparing measured and predicted V_{cl}, $PHIE$, S_w values for each well.
- A comparison plot was generated showing predicted vs. measured V_{cl}, $PHIE$, S_w values for the whole test dataset.
- A summary table showing the evaluation metric of the models.

6.6 Evaluation Metrics

Model performance is evaluated using the following metrics on a test dataset:

1. **Mean Absolute Error (MAE):**
   It measures the average absolute difference between the actual values and the predicted values. A lower MAE indicates that the predictions are, on average, closer to the true values, since it provides a linear score without squaring the errors, it treats all individual differences equally.

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|$$

   Where:

   - N is the total number of samples,
   - y_i is the actual value of the i-th sample,
   - $\hat{y}_i$ is the predicted value of the i-th sample,
   - $\bar{y}$ is the mean of all actual values, computed as $\bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$.

2. **Mean Squared Error (MSE):** It calculates the average squared difference between the actual values and the predicted values. By squaring the errors, MSE penalizes larger errors more heavily than smaller ones, making it more sensitive to outliers than MAE. A lower MSE indicates that the model's predictions are generally closer to the true values.

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

   Where:

   - N: the total number of samples
   - y_i: the actual (true) value for the i-th sample
   - $\hat{y}_i$: the predicted value for the i-th sample

3. **Coefficient of Determination (R^2):**
   The coefficient of determination, denoted as R^2, measures how well a regression model explains the variance in the target variable. It indicates the proportion of the total variation in the actual values that is captured by the model's predictions. An R^2 value of 1 means the model perfectly predicts the data, while a value of 0 indicates that the model does no better than simply predicting the mean of the target values. Negative values suggest that the model performs worse than this basic baseline. Overall, a higher R^2reflects a better fit between the predicted and actual values.

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2}$$

Where:

- N is the total number of samples,
- y_i is the actual value,
- $\hat{y}_i$ is the predicted value,
- $\bar{y}$ is the mean of the actual values.

6.7 Limitations of Each Model in the Context of Petrophysical Parameter Prediction

6.7.1 Linear Regression (LR)

- Assumes linearity: Cannot model complex or nonlinear relationships between features and target variables (e.g., PHIE, SW).
- Sensitive to multicollinearity: Performance degrades when input features are highly correlated.
- Limited flexibility: Struggles with feature interactions and cannot capture hierarchical or conditional dependencies.
- Poor handling of noisy or skewed data: Especially problematic for targets like SW that often have saturation effects (e.g., many values near 1).

6.7.2 Multilayer Perceptron (MLP)

- Requires careful tuning: Model performance is highly dependent on architecture (number of layers/neurons), learning rate, and regularization.
- Prone to overfitting: Especially with small datasets or insufficient regularization.
- Less interpretable: Difficult to understand how input features influence the output.
- Computationally expensive: Training deep MLPs can be resource-intensive and slower compared to tree-based models.

6.7.3 XGBoost

- Sensitive to hyperparameters: Requires careful tuning of tree depth, learning rate, number of estimators, etc.
- Can struggle with extrapolation: Tree-based models do not predict well outside the range of the training data.

- May not capture subtle spatial or sequential patterns: Unlike CNNs or RNNs, XGBoost lacks built-in structure for handling ordered or spatial data.
- Performance may plateau: If the dataset has strong noise or if critical nonlinearities are not captured by the available features.

6.7.4 1D Convolutional Neural Network (CNN 1D)

- Requires large amounts of data: CNNs are data-hungry and may underperform with limited or sparse datasets.
- Sensitive to window size and filter configuration: Poor parameter choices can lead to ineffective feature extraction.
- Less intuitive feature interpretability: The internal convolutional filters are difficult to interpret in a geological context.
- Higher computational cost: Training CNNs, even in 1D, typically requires more time and resources compared to models like LR or XGBoost.

6.8 Conclusion

The workflow of this study was carefully structured to progressively explore and evaluate a range of modelling techniques, from the most interpretable to the most powerful. We began with classical regression models to establish a baseline of performance and to understand the linear relationships between the input well log data and the target petrophysical parameters. This initial step allowed us to interpret the basic correlations and set a reference for comparison.

Following this, we introduced a Multi-Layer Perceptron (MLP), a simple form of neural network capable of modelling non-linear relationships. The MLP served as a natural next step, providing greater flexibility in capturing complex interactions between features without significant computational cost.

To improve accuracy and leverage feature interactions more effectively, we then employed the XGBoost algorithm — a gradient boosting framework known for its robustness and high performance on structured data. XGBoost also provided additional interpretability through feature importance scores, which guided further analysis and model refinement.

Finally, we applied one-dimensional Convolutional Neural Networks (1D CNNs), which are particularly suited to sequential data like well logs. These models allowed us to exploit the spatial continuity in the log measurements, capturing local patterns and trends that previous models might have overlooked.

This structured progression (from simple to more complex and domain-adapted models) ensured both interpretability and performance, while providing insights at each stage of the modelling pipeline.

Chapter 7

Predicted Petrophysical Parameter Results and Discussions

*(**Objective:** Present and analyse the predicted petrophysical parameters (S_w, $PHIE$, and V_{cl}) and evaluates their accuracy using visualizations and statistical metrics.)*

7.1 Simulation Results For Volume of Clay (V_{cl})

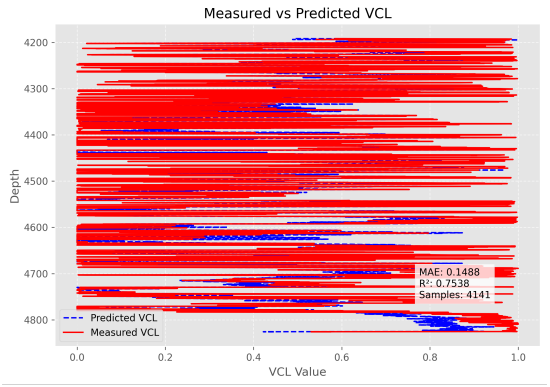
7.1.1 Linear Regression (LR) Results

The model was trained using a set of petrophysical features, and its performance was evaluated on the test dataset using standard regression metrics, including Mean Absolute Error (MAE), Mean Squared Error (MSE), and R^2 score.
The results obtained from the Linear Regression (LR) model for predicting the volume of clay V_{cl} are presented below:

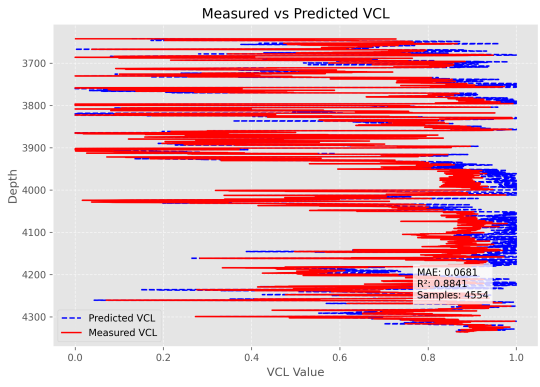
Table 7.1: Evaluation Metrics per Well

Well	MAE	R^2	MSE	Samples
Well 1	0.1498	0.7531	0.0333	4141
Well 2	0.0719	0.8702	0.0075	4554
Well 3	0.0718	0.8758	0.0082	5112
Well 4	0.0890	0.8939	0.0118	6142
All Wells	**0.0933**	**0.8585**	**0.0144**	**19949**

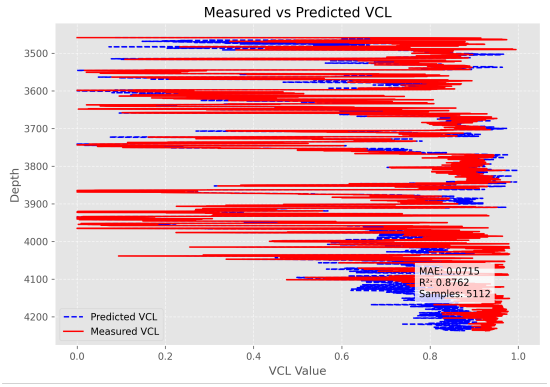
The following plots illustrate the predictive performance of this model across the different test wells:



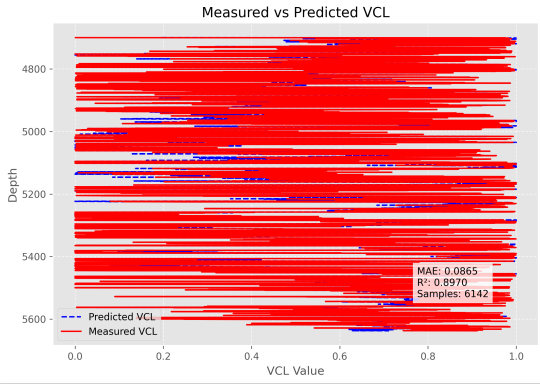
(a) Prediction for Well 1



(b) Prediction for Well 2



(c) Prediction for Well 3



(d) Prediction for Well 4

Figure 7.1: Prediction performance of Linear regression model for each test well.

The following plot presents the comparison between measured and predicted V_{cl} values for the entire test dataset:

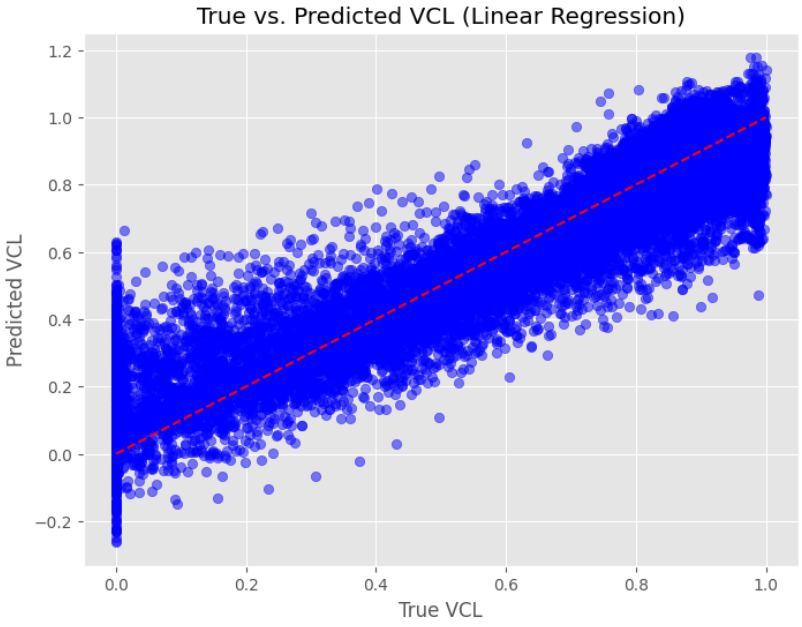


Figure 7.2: Measured vs Predicted V_{cl} for the test set using the Linear Regression

The Linear Regression (LR) performed relatively well in predicting clay volume V_{cl}, this is primarily because V_{cl} tends to exhibit a strong linear relationship with input features, most notably the gamma ray (GR) log, which is widely used as a direct indicator of shale content in formations.

7.1.2 Multilayer Perceptron (MLP) Results

Three MLP models with different hidden layer sizes (64, 32, and 16 neurons) were trained on the same petrophysical dataset to evaluate how the MLP network size affects training performance and the predictions on the testing dataset.

Predictions Evaluations

To evaluate the generalization capability of the trained MLP models, we tested them on a separate dataset consisting of four wells that were not seen during training. This test set allows us to assess the model's ability to predict petrophysical parameters on new, unseen data.
The performance of each model was compared using three key regression metrics:
Mean Absolute Error (MAE), Mean Squared Error (MSE), and R^2 score.

1. **With 64 Neurons per Hidden Layer**

Table 7.2: Summary of test performance for each well.

Well	MAE	MSE	R^2	Samples
Well 1	0.1198	0.0264	0.8040	4161
Well 2	0.1245	0.0240	0.5868	4554
Well 3	0.0809	0.0140	0.7884	5112
Well 4	0.1241	0.0322	0.7096	6151
All Wells	**0.1122**	**0.0245**	**0.7593**	**19978**

2. **With 32 Neurons per Hidden Layer**

Table 7.3: Summary of test performance for each well

Well	MAE	MSE	R^2	Samples
Well 1	0.1042	0.0195	0.8557	4161
Well 2	0.1222	0.0227	0.6094	4554
Well 3	0.0781	0.0122	0.8156	5112
Well 4	0.1360	0.0344	0.6892	6151
All Wells	**0.1114**	**0.0230**	**0.7742**	**19978**

3. **With 16 Neurons per Hidden Layer**

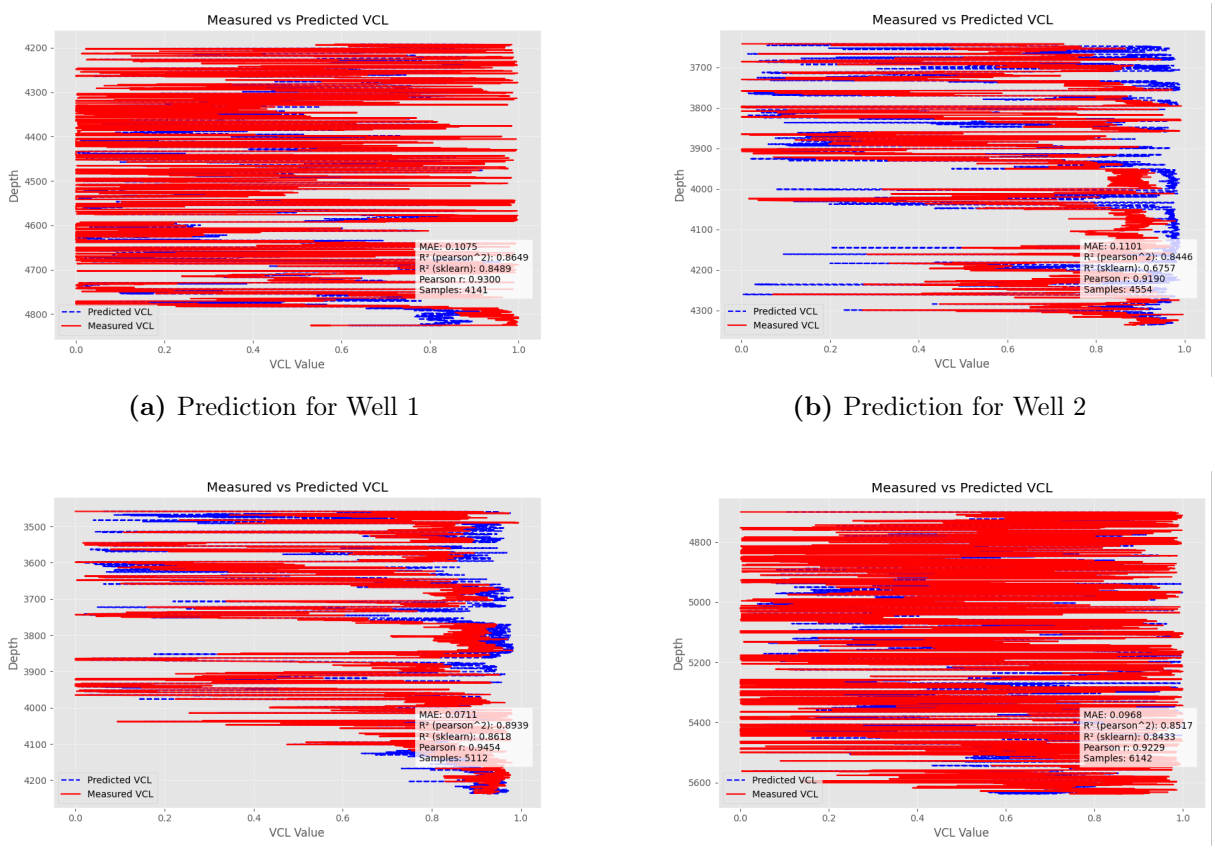
Table 7.4: Summary of test performance for each well.

Well	MAE	MSE	R^2	Samples
Well 1	0.0968	0.0174	0.8433	6142
Well 2	0.1075	0.0204	0.8489	4141
Well 3	0.1101	0.0188	0.6757	4554
Well 4	0.0711	0.0092	0.8618	5112
All Wells	**0.0955**	**0.0162**	**0.8322**	**19949**

Best MLP Model

By comparing the test results obtained on the four unseen wells, it is clear that the MLP model with 16 neurons per hidden layer demonstrates the best performance. This architecture achieves the lowest MAE and MSE while maintaining high R^2 scores across most wells, confirming its strong generalization ability.

The following plots illustrate the predictive performance of this model across the different test wells:



(a) Prediction for Well 1

(b) Prediction for Well 2

(c) Prediction for Well 3

(d) Prediction for Well 4

Figure 7.3: Prediction performance of best MLP model for each test well.

The following plot presents the comparison between measured and predicted (V_{cl} values for the entire test dataset, using the best-performing MLP model:

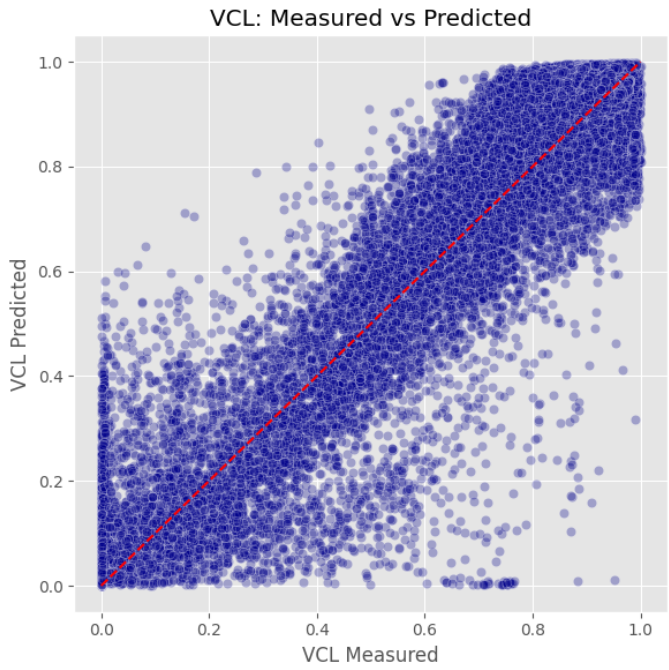


Figure 7.4: Measured vs Predicted (V_{cl}) for the test set using the best MLP model

Even though the MLP model with 16 neurons in each hidden layer performed well, Linear Regression (LR) achieved slightly better results. This is likely due to the relatively simple and linear nature of the clay volume (V_{cl}) prediction task, where a straightforward model like LR was sufficient to capture the strong correlations, particularly with features like gamma ray (GR).

After evaluating the performance of the MLP models, we opted for the XGBoost (Extreme Gradient Boosting) algorithm to predict petrophysical parameters. XGBoost is a powerful and efficient method that combines the predictive strength of decision trees with gradient boosting optimization.

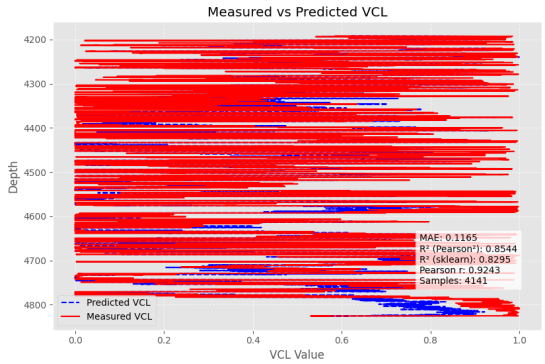
7.1.3 XGBoost Results

Compared to a neural network, XGBoost requires fewer computational resources and is particularly effective at capturing linear and moderately nonlinear relationships between input features and target variables.
The following results present the performance of the XGBoost model in predicting the target variable across each test well, evaluated using standard metrics:

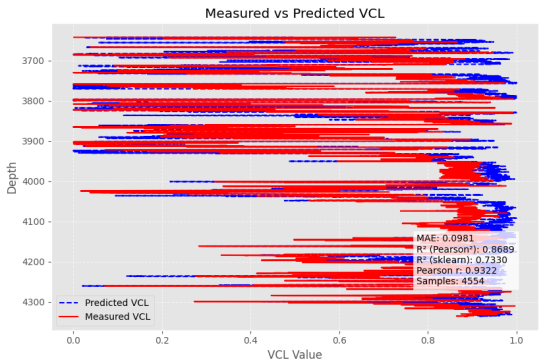
Table 7.5: Performance Metrics per Well (Sorted by Sample Count)

Well	MAE	MSE	R^2	Samples
Well 1	0.1175	0.0231	0.8291	4141
Well 2	0.0982	0.0155	0.7325	4554
Well 3	0.0666	0.0094	0.8582	5112
Well 4	0.1051	0.0179	0.8385	6142
All Wells	**0.0962**	**0.1275**	**0.8400**	**19949**

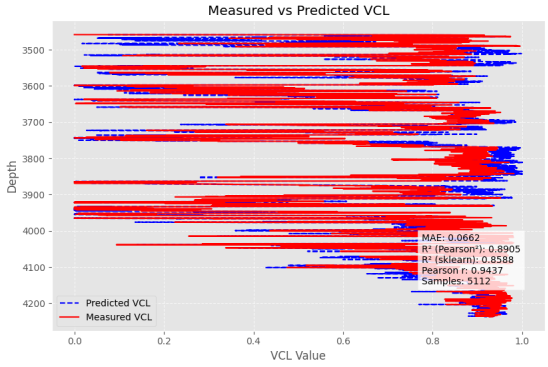
The following plots illustrate the predictive performance of this model across the different test wells:



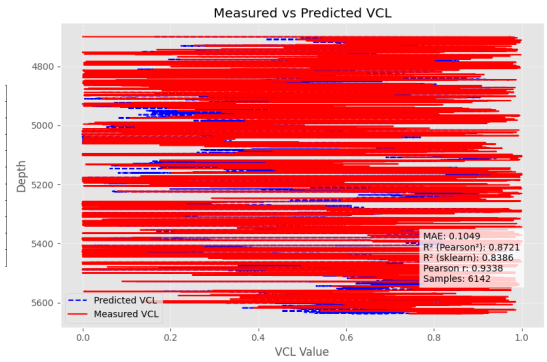
(a) Prediction for Well 1



(b) Prediction for Well 2



(c) Prediction for Well 3



(d) Prediction for Well 4

Figure 7.5: Prediction performance of the XGBoost model for each test well.

The following plot presents the comparison between measured and predicted V_{cl} values for the entire test dataset:

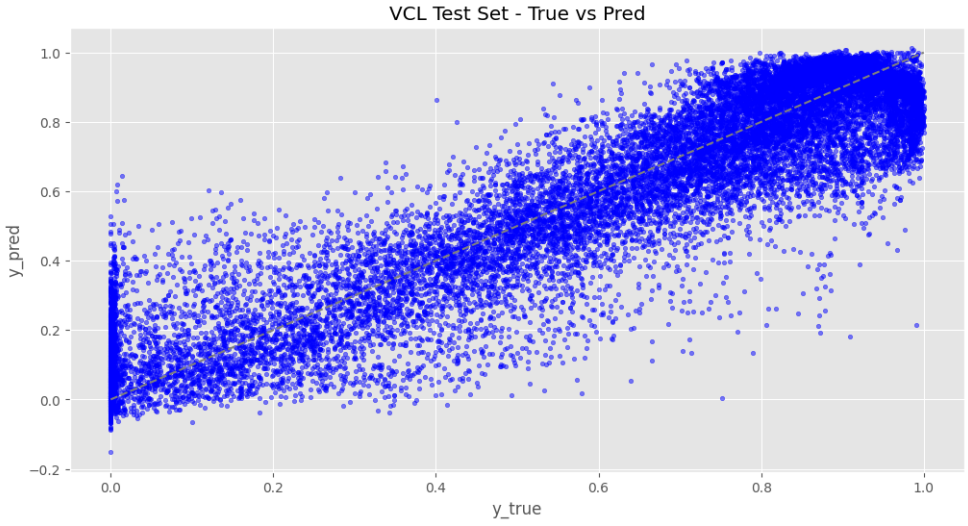


Figure 7.6: Measured vs Predicted V_{cl} for the test set using XGBoost

The XGBoost model performed well compared to the MLP models, particularly due to its direct regression approach and its ability to efficiently handle feature interactions without requiring deep architectures.

7.1.4 CNN Results

Compared to gradient boosting, Convolutional Neural Networks (CNNs) are more computationally demanding but offer powerful capabilities to learn complex and non-linear spatial relationships from structured data. In this work, CNNs were tested as an alternative approach for predicting the volume of clay (V_{cl}) from well log measurements. Two architectures were explored: one using a kernel size of 1 with 32 filters, and another using a kernel size of 3 with 16 filters.

Architecture 1

The following results correspond to the architecture with 32 filters and a kernel size of 1. The model's performance was evaluated on a well-by-well basis using standard metrics.

Well	**MAE**	**MSE**	R^2
Well 1	0.0859	0.0120	0.8489
Well 2	0.0484	0.0056	0.9366
Well 3	0.0365	0.0029	0.9644
Well 4	0.1268	0.0291	0.5818
All Wells	**0.0773**	**0.0135**	**0.8502**

Table 7.6: Prediction metrics per well using the optimized CNN model (Kernel size = 1, Filters = 32)

The CNN model demonstrated competitive performance, achieving an overall R^2 score of 0.85, comparable to that of XGBoost. However, higher variance was observed across individual wells, suggesting that CNNs may benefit from further tuning or regularization strategies to improve generalization across different reservoir conditions.

Architecture 2

A second convolutional architecture was explored using a kernel size of 3 and 16 filters. This design was motivated by the hypothesis that a slightly wider receptive field could help the network better capture local spatial dependencies in the well log data.

The table below summarizes the model's predictive performance across the different test wells using standard evaluation metrics.

Well	MAE	MSE	R^2
Well 1	0.0671	0.0070	0.9120
Well 2	0.0619	0.0054	0.9395
Well 3	0.0617	0.0050	0.9396
Well 4	0.1062	0.0201	0.7109
All Wells	**0.0766**	**0.0101**	**0.8873**

Table 7.7: Prediction metrics per well using CNN model (Kernel size = 3, Filters = 16)

While the average performance was slightly lower than the previous architecture for well 3, the overall R^2 score remains strong at 0.887. These results suggest that both architectures are viable, though the choice of kernel size and filter count can lead to trade-offs in generalization across different geological settings.

Architecture Comparison and Synthesis

Two convolutional neural network (CNN) architectures were evaluated for water saturation (S_w) prediction:

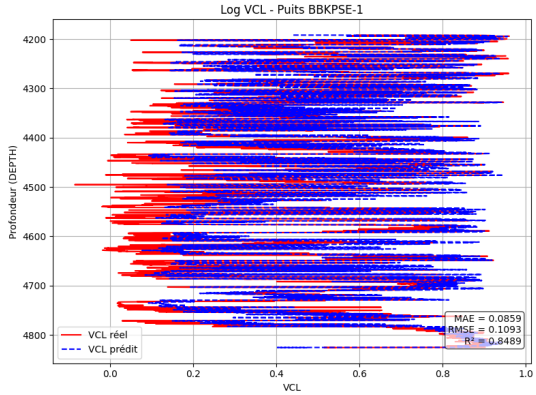
- **Architecture 1:** 32 filters with a kernel size of 1
- **Architecture 2:** 16 filters with a kernel size of 3

Both models demonstrated strong predictive performance, with global R^2 scores exceeding 0.85. Architecture 2 achieved the best overall performance, with a global $R^2 = 0.8873$ compared to 0.8502 for Architecture 1, and lower average errors (MAE and MSE).

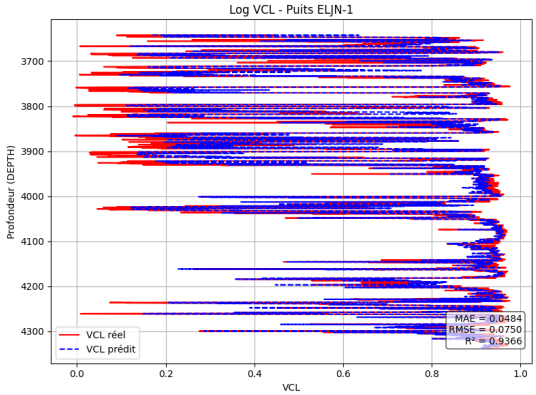
Architecture 1 exhibits a noticeable drop in Well 4 ($R^2 = 0.5818$) compared to its relatively consistent predictions of the other wells. Hence, architecture 2 obviously outperformed Architecture 1 in three individual wells, including the challenging Well 4 ($R^2 = 0.7109$ vs. 0.5818 for the first), indicating better generalization and robustness, even under varying geological conditions. The larger kernel size may have helped the model better capture local spatial dependencies in the well log data.

Given its superior accuracy both globally and per well, **Architecture 2 (kernel size = 3, filters = 16)** was selected for final deployment.

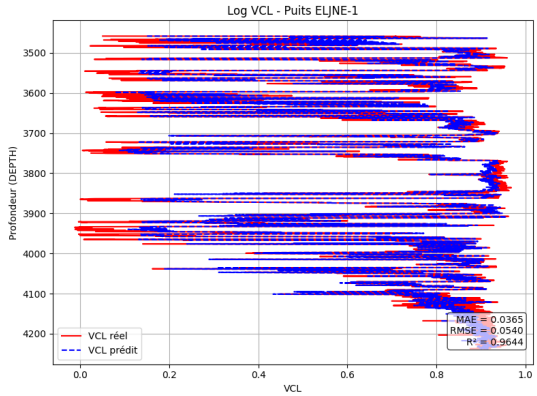
The following plots illustrate the predictive performance of this model across the different test wells:



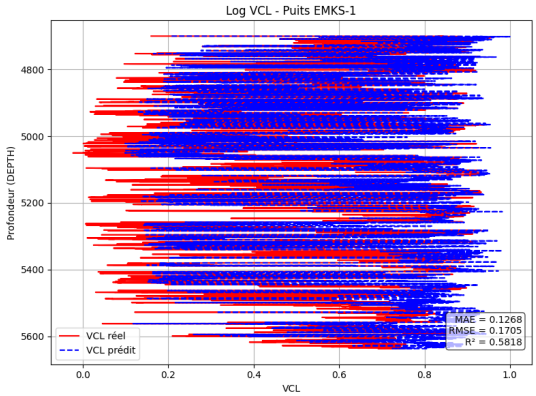
(a) Prediction for Well 1



(b) Prediction for Well 2

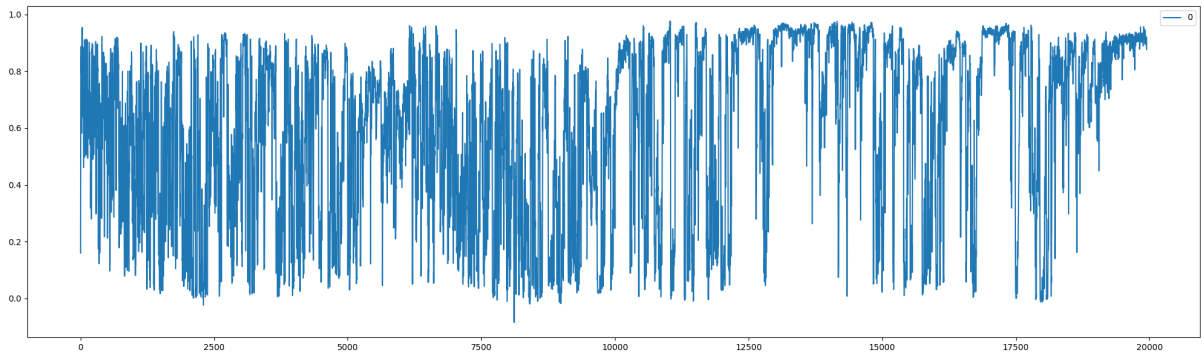


(c) Prediction for Well 3

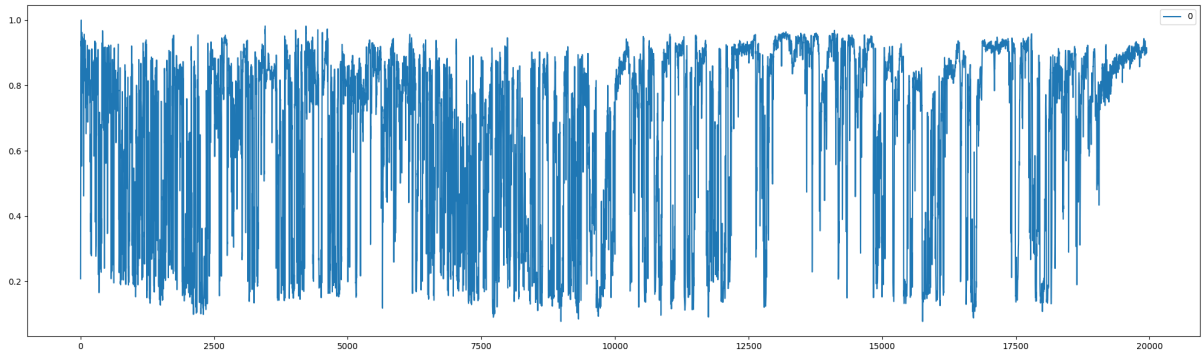


(d) Prediction for Well 4

Figure 7.7: Prediction performance of the CNN model for each test well.



(a) Actual V_{cl}



(b) Predicted V_{cl}

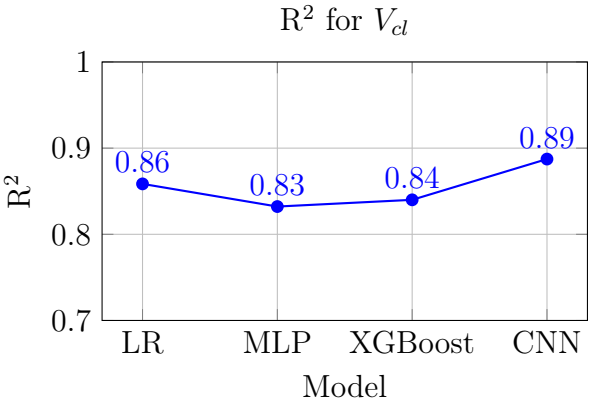
Figure 7.8: V_{cl} predicted vs. Actual

7.1.5 Comparative Study (V_{cl})

The prediction of the volume of clay (V_{cl}) has shown promising results across all models used in this study, suggesting that V_{cl} is the easiest petrophysical parameter to predict among those examined. This is primarily attributed to the strong linear relationship between V_{cl} and the gamma ray (GR) log, which serves as a highly informative input feature.

Table 7.8: Metrics for V_{cl}

Model	MAE	MSE	R^2
LR	0.0933	0.0144	0.8585
MLP	0.0955	0.0162	0.8322
XGBoost	0.0962	0.1275	0.8400
CNN	0.0766	0.0101	0.8873

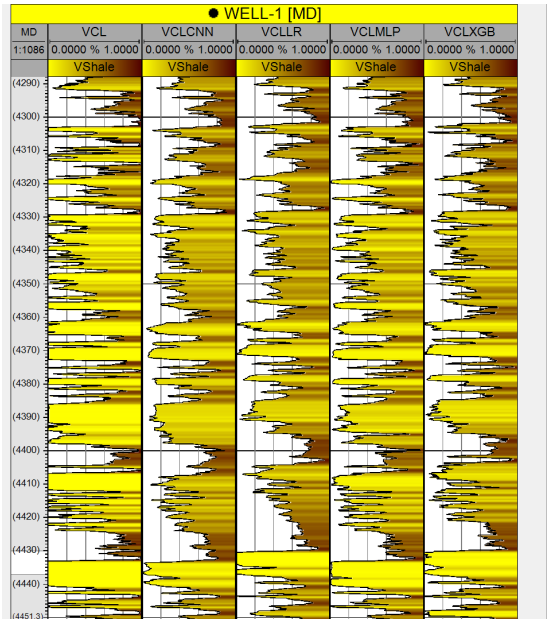


Key observations include:

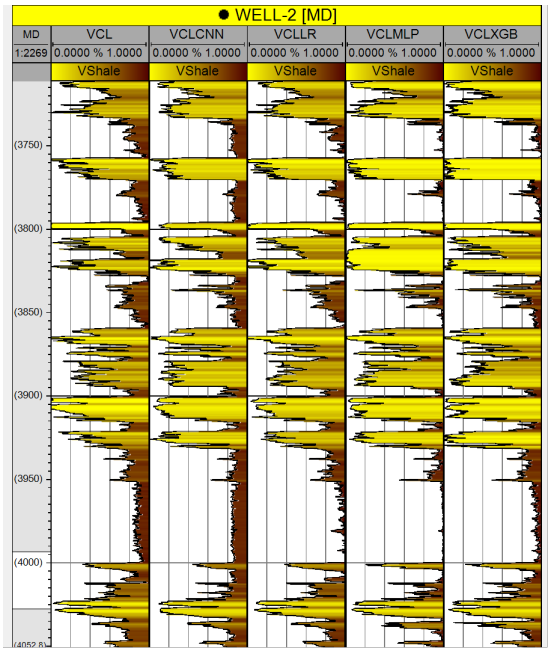
- Linear Regression (LR) performs very well ($R^2 = 0.8585$), reflecting the strong linear relationship between GR and V_{cl}.

- MLP and XGBoost models offer decent performance but do not significantly surpass LR, indicating limited added value from their complexity in this case.
- CNN achieves the best performance ($R^2 = 0.8873$), leveraging its deep structure to capture both linear and subtle nonlinear patterns, as well as spatial depth-dependent features.

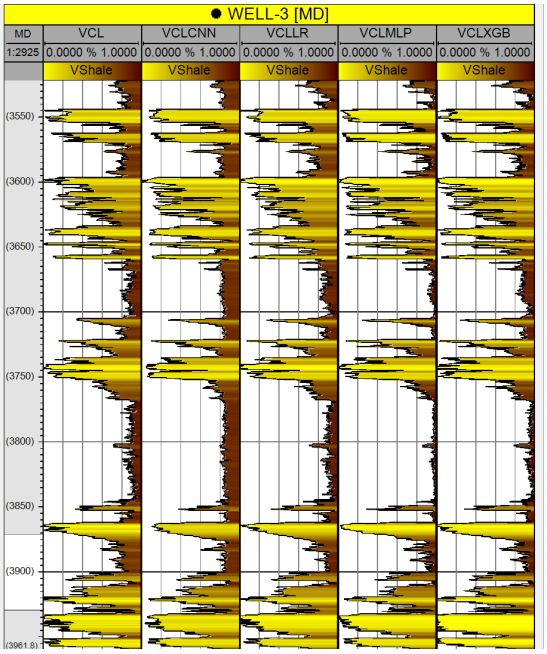
This confirms that while simpler models suffice for V_{cl}, deep learning can still enhance prediction accuracy and this can be observed more clearly in the log interpretation of each well presented below:



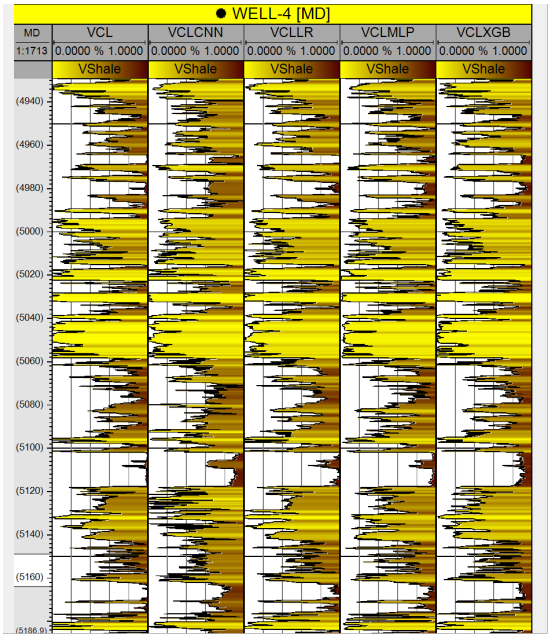
(a) Well 1 – V_{cl} Prediction



(b) Well 2 – V_{cl} Prediction



(c) Well 3 – V_{cl} Prediction



(d) Well 4 – V_{cl} Prediction

Figure 7.9: Log comparisons and V_{cl} predictions for selected wells

7.2 Simulation Results For Effective Porosity ($PHIE$)

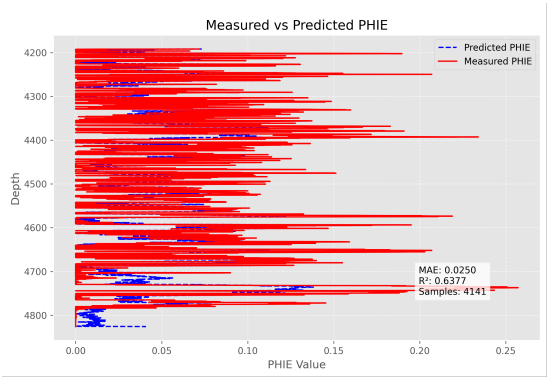
7.2.1 Linear Regression (LR) Results

The results obtained from the Linear Regression (LR) model for predicting the effective porosity ($PHIE$) are presented below:

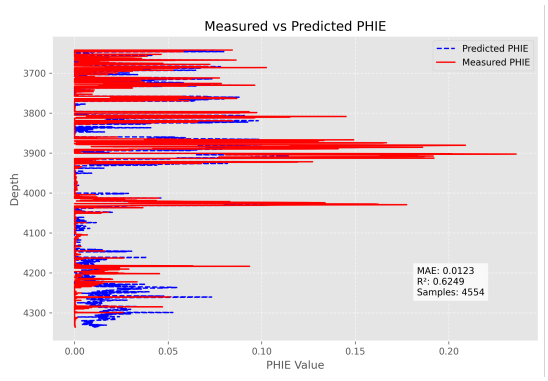
Table 7.9: Metrics Summary per Well

Well	MAE	RMSE	R^2	Samples
Well 1	0.0250	0.6377	0.0011	4141
Well 2	0.0123	0.6249	0.0004	4554
Well 3	0.0195	0.5422	0.0006	5112
Well 4	0.0234	0.5099	0.0010	6142
All Wells	**0.0202**	**0.6144**	**0.0008**	**19949**

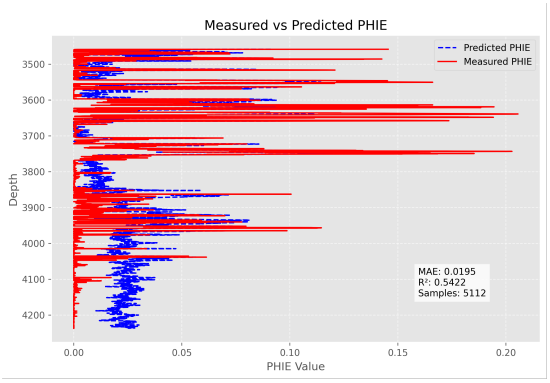
The following plots demonstrate how the model generalizes across different wells in the test set for predicting $PHIE$:



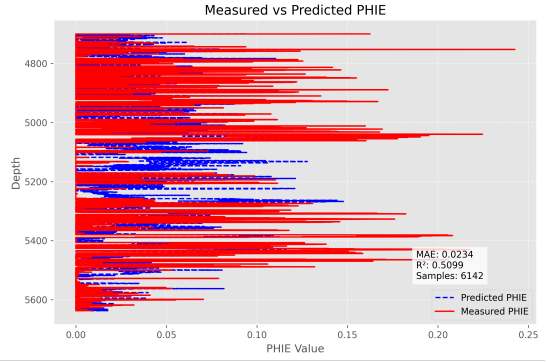
(a) Prediction for Well 1



(b) Prediction for Well 2



(c) Prediction for Well 3



(d) Prediction for Well 4

Figure 7.10: Prediction performance of Linear regression model for each test well.

The following plot presents the comparison between measured and predicted $PHIE$ values for the entire test dataset:

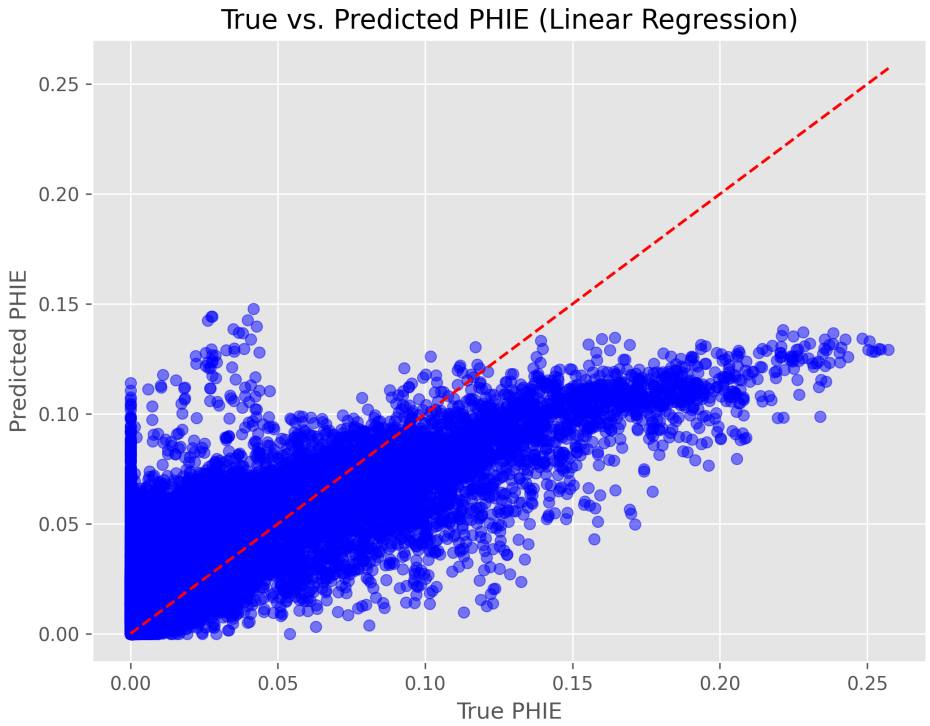


Figure 7.11: Measured vs Predicted $PHIE$ for the test set using Linear Regression

As we can observe from the overall performance metrics and the Measured vs. Predicted $PHIE$ plot, the Linear Regression (LR) model performed poorly in predicting effective porosity ($PHIE$). This is mainly because $PHIE$ is a more complex and nonlinear target compared to parameters like V_{cl}.
Its accurate estimation requires capturing intricate and multidimensional relationships among several input features something that a simple linear model is not capable of modeling effectively.
Consequently, more advanced and deeper learning architectures, such as MLPs, are better suited for generating accurate and reliable predictions of $PHIE$.

7.2.2 Multilayer Perceptron (MLP) Results

Predictions Evaluations

In this section, we evaluate the performance of the MLP models using standard regression metrics (MAE, MSE, and R^2) calculated for each well in the unseen test dataset. This evaluation aims to determine how well the models generalize the predicted $PHIE$ and whether they are capable of producing accurate predictions across different wells.

1. **With 64 Neurons per Hidden Layer**

Table 7.10: Summary of test performance for each well.

Well	MAE	MSE	R^2	Samples
Well 1	0.0142	0.0005	0.8294	4161
Well 2	0.0162	0.0011	0.0606	4554
Well 3	0.0110	0.0004	0.6640	5112
Well 4	0.0144	0.0008	0.6396	6151
All Wells	**0.0139**	**0.0007**	**0.6597**	**19978**

2. **With 32 Neurons per Hidden Layer**

Table 7.11: Summary of test performance for each well.

Well	MAE	MSE	R^2	Samples
Well 1	0.0159	0.0006	0.7919	4161
Well 2	0.0156	0.0010	0.1716	4554
Well 3	0.0113	0.0004	0.6746	5112
Well 4	0.0143	0.0007	0.6845	6151
All Wells	**0.0142**	**0.0007**	**0.6785**	**19978**

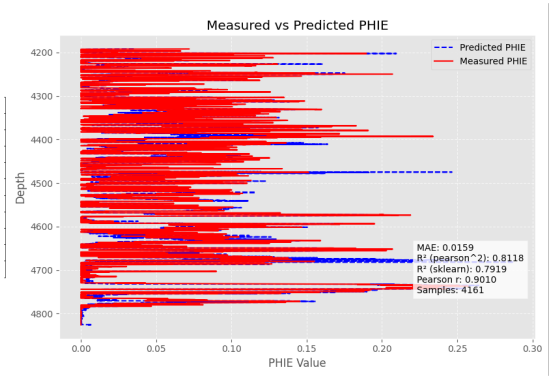
3. **With 16 Neurons per Hidden Layer**

Table 7.12: Summary of test performance for each well.

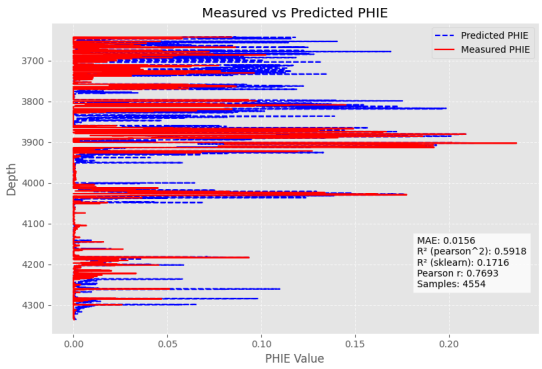
Well	MAE	MSE	R^2	Samples
Well 1	0.0149	0.0005	0.8297	4161
Well 2	0.0174	0.0013	-0.0444	4554
Well 3	0.0113	0.0004	0.6662	5112
Well 4	0.0151	0.0007	0.6940	6151
All Wells	**0.0146**	**0.0007**	**0.6635**	**19978**

Best MLP Model

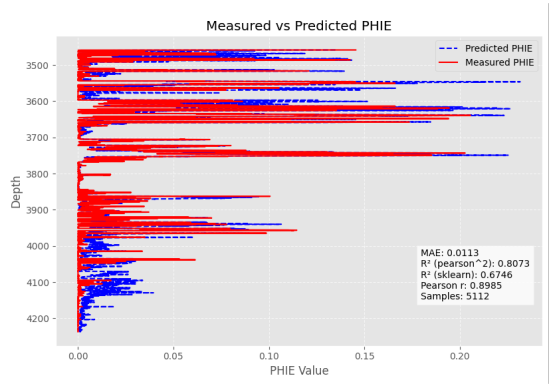
By comparing the test results obtained on the four unseen wells, we can confirm the conclusion drawn during the training evaluation: the MLP model with 32 neurons per hidden layer demonstrates the best performance in predicting $PHIE$. This architecture achieves the lowest MAE and MSE while maintaining high R^2 scores across most wells, confirming its strong generalization ability. The following plots demonstrate how the best MLP model generalizes across different wells in the test set for predicting $PHIE$:



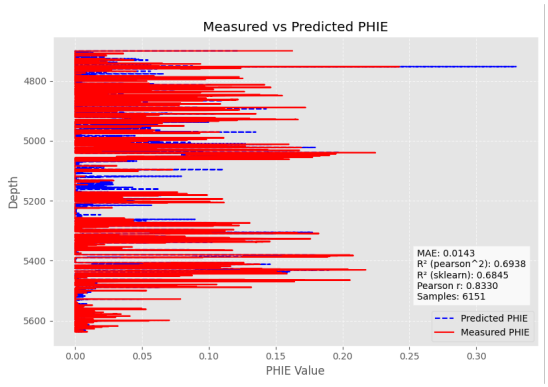
(a) Prediction for Well 1



(b) Prediction for Well 2



(c) Prediction for Well 3



(d) Prediction for Well 4

Figure 7.12: Prediction performance of Linear regression model for each test well.

The following plot presents the comparison between measured and predicted $PHIE$ values for the entire test dataset, using the best-performing MLP model.

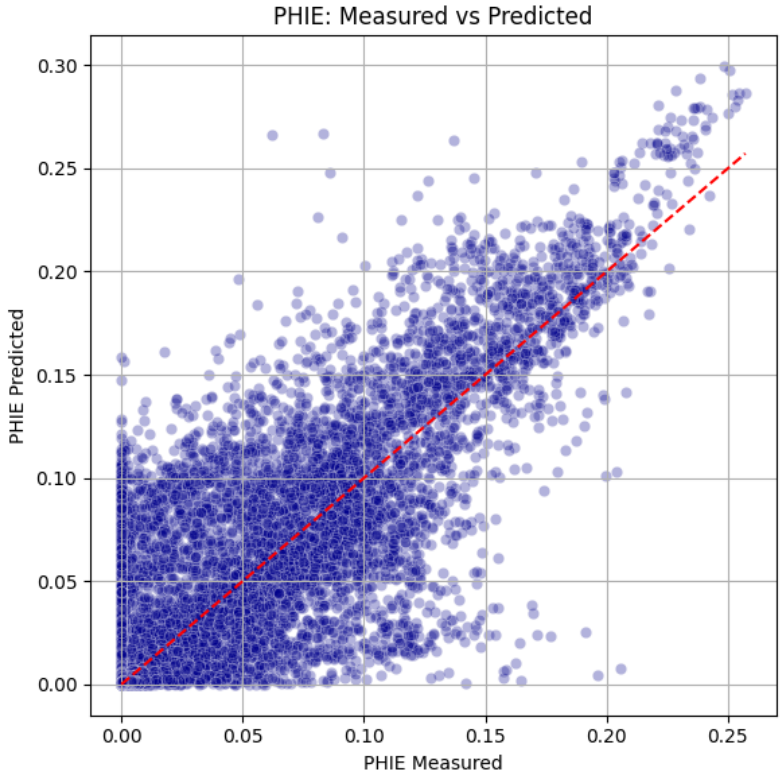


Figure 7.13: Measured vs Predicted $PHIE$ for the test set using the best MLP model

Comparing to LR, the MLP model achieved better results. Its layered structure, combined with the use of ReLU activation functions in the hidden layers and sigmoid activation in the output layer, allowed it to learn and model more complex feature interactions, which led to more accurate predictions of $PHIE$ across the test wells.

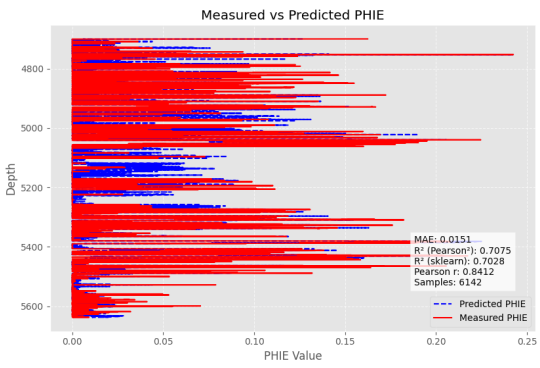
7.2.3 XGBoost Results

The following results present the performance of the XGBoost model in predicting the target variable $PHIE$ across each test well, evaluated using standard metrics:

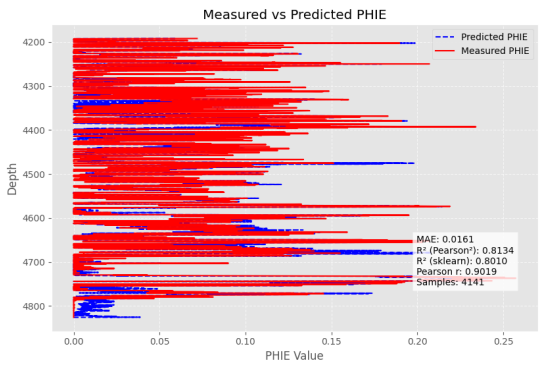
Table 7.13: Regression Performance Metrics Sorted by Sample Count (Ascending)

Well	MAE	MSE	R^2	Samples
Well 1	0.0163	0.0006	0.7997	4141
Well 2	0.0155	0.0009	0.2276	4554
Well 3	0.0110	0.0004	0.6410	5112
Well 4	0.0158	0.0006	0.6991	6142
All Wells	**0.0146**	**0.0255**	**0.6866**	**19949**

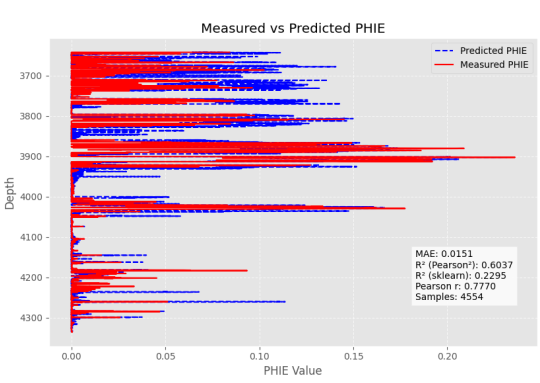
The following plots illustrate the predictive performance of this model across the different test wells:



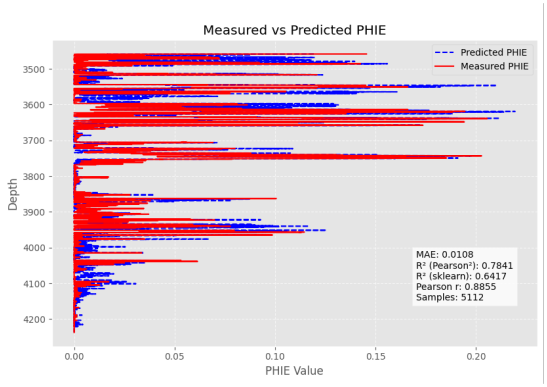
(a) Prediction for Well 1



(b) Prediction for Well 2



(c) Prediction for Well 3



(d) Prediction for Well 4

Figure 7.14: Prediction performance of XGBoost model for each test well.

The plot shown below presents the comparison between measured and predicted

$PHIE$ values for the entire test dataset:

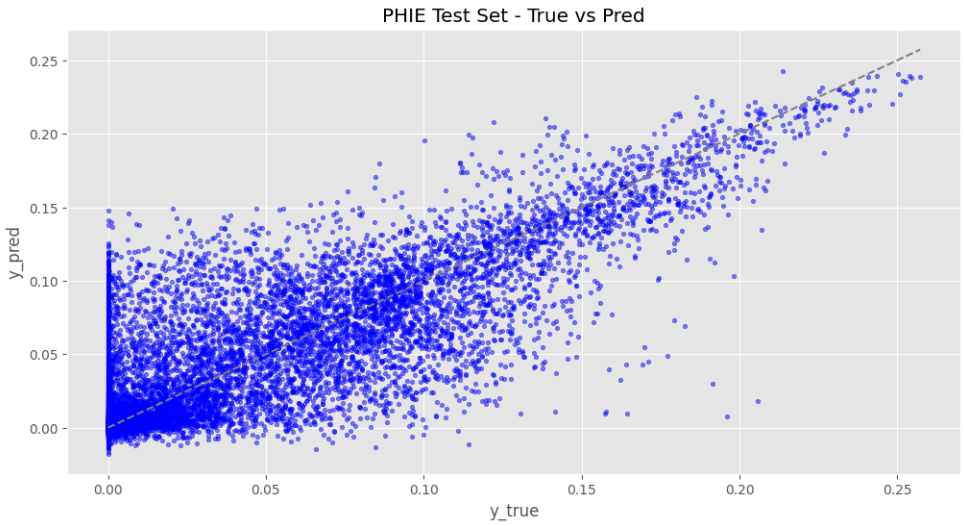


Figure 7.15: Measured vs Predicted $PHIE$ for the test set using XGBoost

Overall, the XGBoost model outperformed both Linear Regression and MLP, achieving better generalization across the test wells. This superior performance can be attributed to XGBoost's ability to efficiently handle complex feature interactions, its robustness against overfitting, and its flexible structure, which allows it to capture both linear and nonlinear patterns more effectively than traditional models.

7.2.4 CNN Results

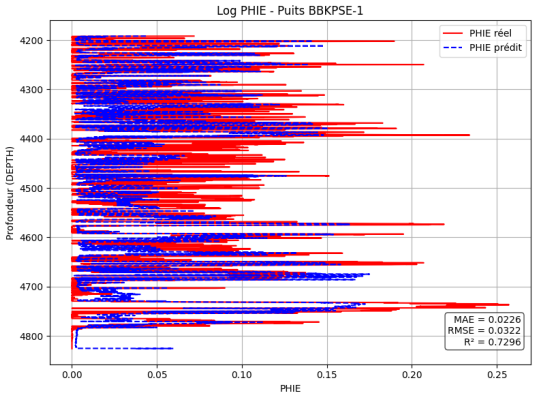
After evaluating the predictive performance of XGBoost for $PHIE$ estimation, we propose a shift to Convolutional Neural Networks (CNNs) because of its use of convolutional kernels to automatically extract local patterns, hierarchies of features, and spatial dependencies. The proposed CNN architecture consists of multiple 1D convolutional layers designed to extract spatial features from sequential log data, followed by fully connected layers for regression. Batch normalization and dropout were applied to improve generalization and training stability. The model was trained using the mean squared error loss and optimized with the Adam optimizer. Its compact yet effective design ensures a balance between performance and computational efficiency.

The results are shown below:

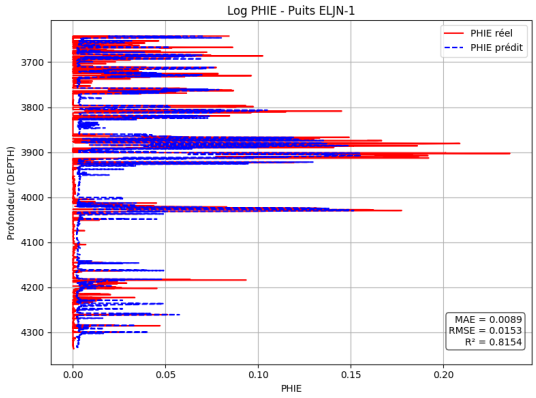
Table 7.14: Results by Well for $PHIE$ prediction using the optimized CNN model

Well	MAE	MSE	R^2
Well 1	0.0226	0.0010	0.6622
Well 2	0.0089	0.0002	0.8044
Well 3	0.0082	0.0002	0.8450
Well 4	0.0150	0.0006	0.7418
All Wells	**0.0134**	**0.0005**	**0.7651**

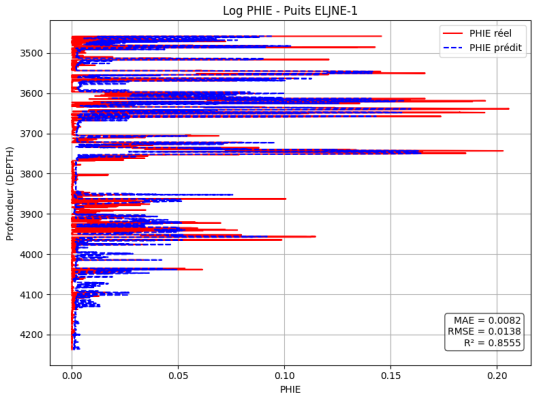
The following plots provide a visual representation of the model's predictive performance on the various test wells:



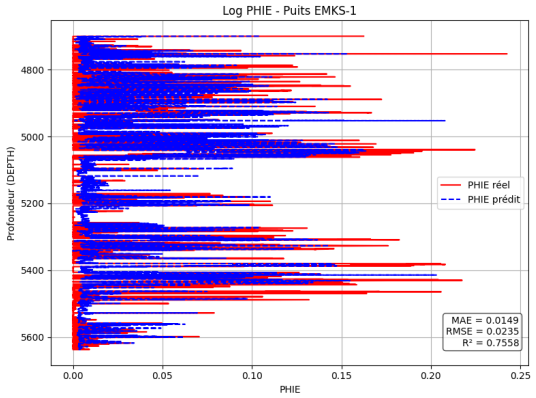
(a) Prediction for Well 1



(b) Prediction for Well 2



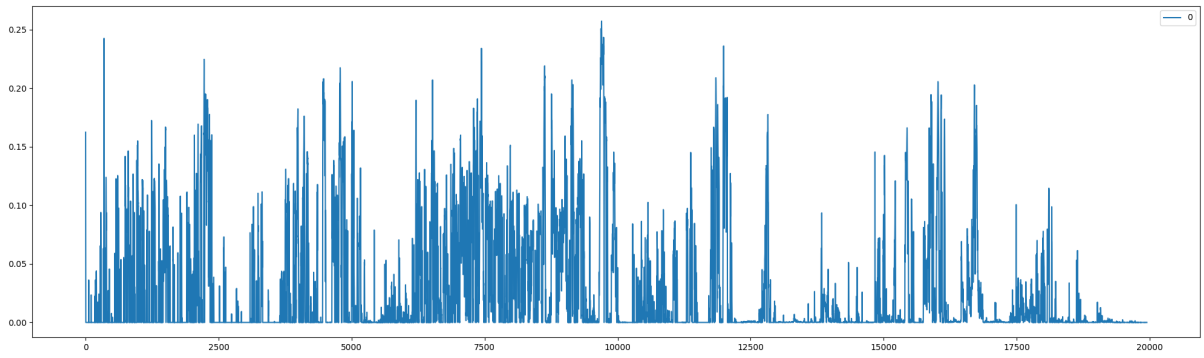
(c) Prediction for Well 3



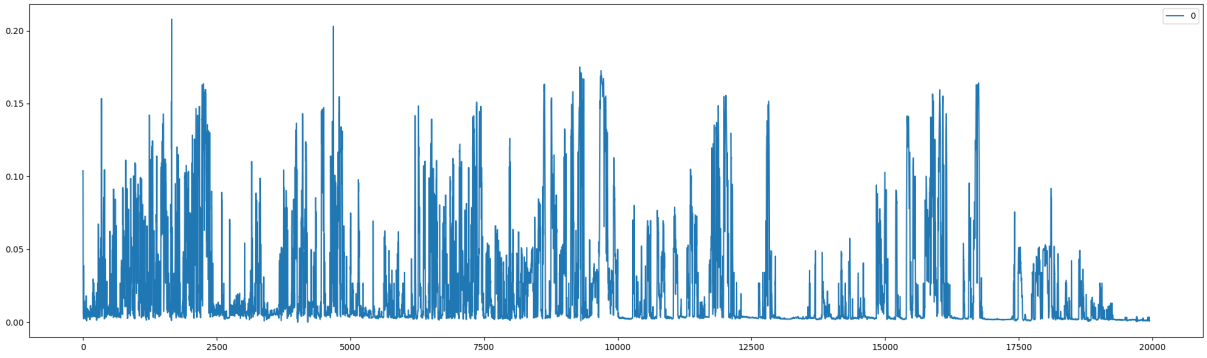
(d) Prediction for Well 4

Figure 7.16: Prediction performance of Linear Regression model for each test well on $PHIE$.

The next plots display the Measured and Predicted $PHIE$ values of the train dataset and test dataset, respectively:



(a) Actual $PHIE$



(b) Predicted $PHIE$

The results highlight the effectiveness of the optimized CNN model for predicting effective porosity $PHIE$. The overall performance is strong, with a global R^2 of 0.7651 and a low mean absolute error (MAE) of 0.0134, indicating good predictive accuracy across the wells.

Well 2 and **Well 3** show the best performance, with R^2 values of 0.8044 and 0.8450 respectively, along with very low error values. This suggests that the model successfully captured the underlying patterns in these wells, possibly due to more homogeneous lithology or higher-quality input data. **Well 4** also displays solid performance, achieving an R^2 of 0.7418.

In contrast, **Well 1** shows a slightly lower performance with an R^2 of 0.6622. While still acceptable, this drop might be explained by increased geological complexity, noisier measurements, or a feature distribution less represented during training.

Despite these slight variations, the overall results confirm the model's ability to generalize across wells while maintaining reliable predictive performance. These outcomes also suggest opportunities for further improvement through localized refinement.

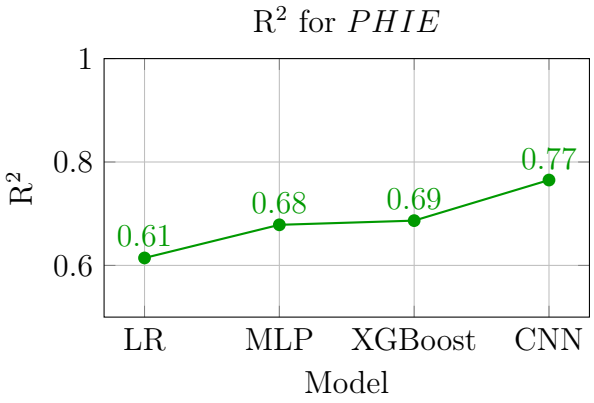
7.2.5 Comparative Study ($PHIE$)

It is evident that effective porosity ($PHIE$) is more difficult to predict than VCL because it depends on a combination of logs, including neutron, density, and sonic. The sonic log (DT) affects $PHIE$ since wave travel time changes with porosity, but it's also influenced by lithology and fluid type, which introduces variability. This makes the relationship

between inputs and $PHIE$ more complex and less directly linear, requiring more advanced models to capture it accurately.

Table 7.15: Metrics for $PHIE$

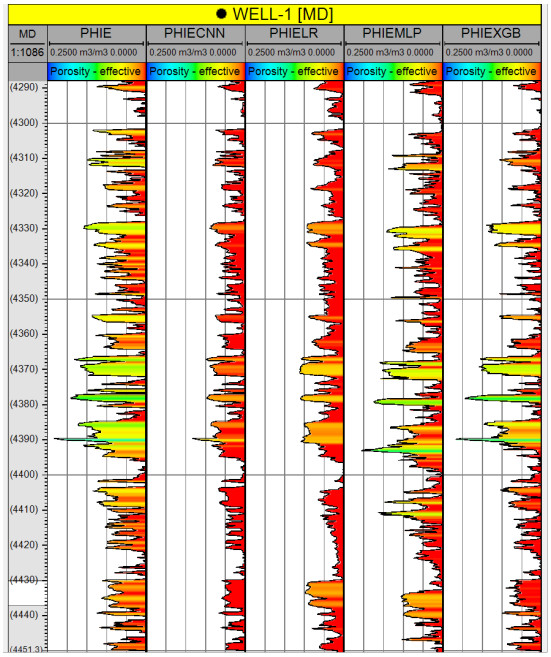
Model	MAE	MSE	R^2
LR	0.0202	0.0008	0.6144
MLP	0.0142	0.0007	0.6785
XGBoost	0.0146	0.0255	0.6866
CNN	0.0134	0.0005	0.7651



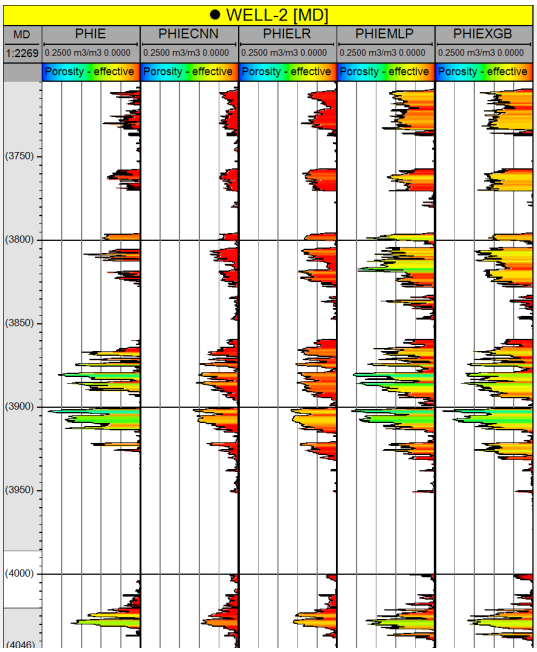
Based on the comparative plot and table, we can conclude that:

- Linear Regression (LR) yields modest results ($R^2 = 0.6144$), suggesting that $PHIE$ cannot be well predicted using linear relationships alone.
- MLP and XGBoost provide improved performance, capturing more complex, nonlinear patterns present in the data.
- CNN again performs best ($R^2 = 0.7651$), indicating its strength in modeling complex geological features and capturing fine-scale spatial dependencies.

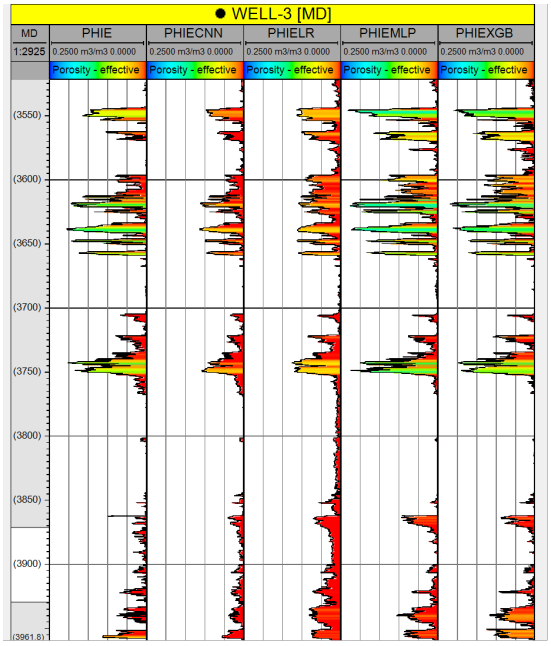
This is further illustrated in the following well log interpretations, where model predictions are displayed:



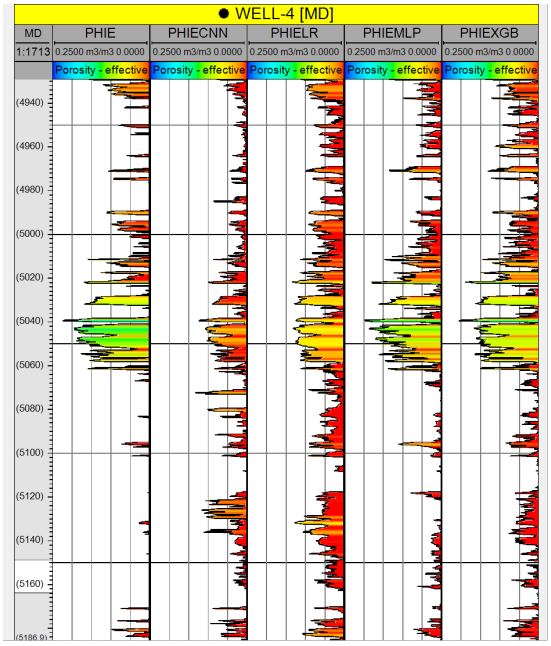
(a) Well 1 – *PHIE* Prediction



(b) Well 2 – *PHIE* Prediction



(c) Well 3 – *PHIE* Prediction



(d) Well 4 – *PHIE* Prediction

Figure 7.18: Log comparisons and *PHIE* predictions for selected wells

7.3 Simulation Results For Water Saturation (S_w))

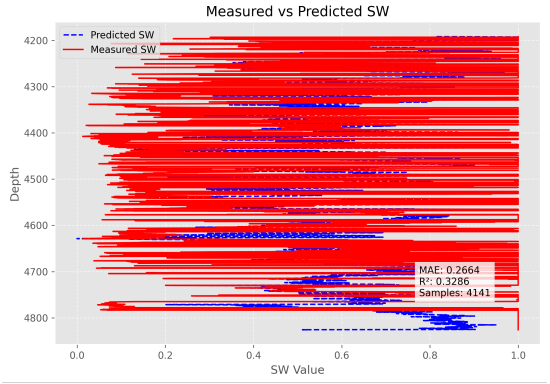
7.3.1 Linear Regression (LR) Results

The results obtained from the Linear Regression (LR) model for predicting Water Saturation (S_w) are illustrated below:

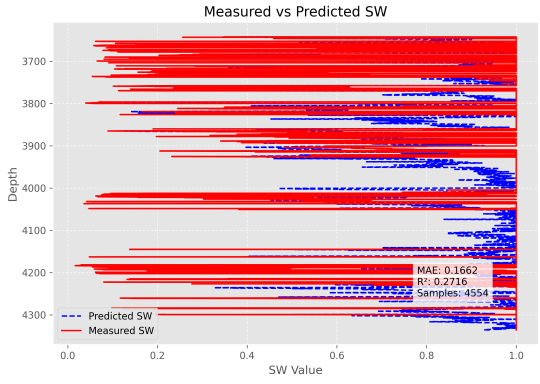
Table 7.16: Evaluation Metrics per Well

Well	MAE	RMSE	R^2	Samples
Well 1	0.2664	0.3106	0.3286	4141
Well 2	0.1662	0.2383	0.2716	4554
Well 3	0.1741	0.2302	0.3431	5112
Well 4	0.2669	0.3182	-0.1727	6142
All Wells	**0.2200**	**0.2790**	**0.2741**	**19949**

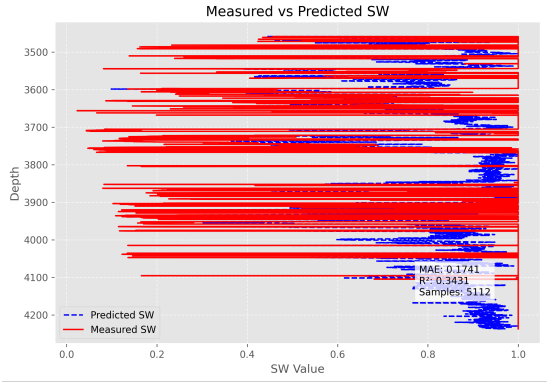
The following plots provide a visual representation of the model's predictive performance on the various test wells:



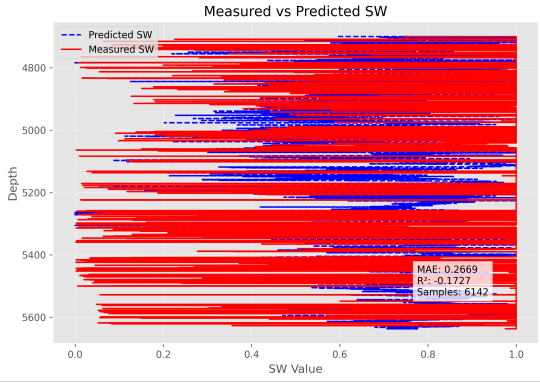
(a) Prediction for Well 1



(b) Prediction for Well 2



(c) Prediction for Well 3



(d) Prediction for Well 4

Figure 7.19: Prediction performance of Linear Regression model for each test well.

This plot illustrates the Measured vs Predicted S_w for the test set using LLinear Regression across the whole test dataset:

Figure 7.20: Measured vs Predicted S_w for the test set using Linear Regression

The prediction of water saturation (S_w) using Linear Regression (LR) was relatively poor. This is due to the complex nature of S_w, which is strongly influenced by resistivity-based measurements that are highly sensitive to formation conditions and can exhibit significant nonlinear behavior.

7.3.2 Multilayer Perceptron (MLP) Results

Predictions Evaluations

In this section, we evaluate the performance of the MLP models using standard regression metrics (MAE, MSE, and R^2) calculated for each well in the unseen test dataset for estimating S_w:

1. **With 64 Neurons per Hidden Layer**

Table 7.17: Model Evaluation Metrics per Well

Well	MAE	MSE	R^2	Samples
Well 1	0.1715	0.0791	0.4481	4161
Well 2	0.1536	0.1104	-0.4149	4554
Well 3	0.2208	0.1470	-0.8233	5112
Well 4	0.1554	0.1027	-0.1869	6151
All Wells	**0.1751**	**0.1109**	**-0.0343**	**19978**

2. **With 32 Neurons per Hidden Layer**

Table 7.18: Model Evaluation Metrics per Well

Well	MAE	MSE	R^2	Samples
Well 1	0.1668	0.0743	0.4813	4161
Well 2	0.1425	0.0950	-0.2173	4554
Well 3	0.1698	0.0936	-0.1611	5112
Well 4	0.2067	0.1354	-0.5654	6151
All Wells	**0.1743**	**0.1028**	**0.0412**	**19978**

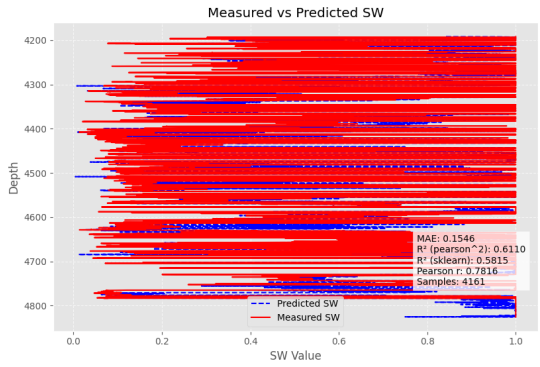
3. **With 16 Neurons per Hidden Layer**

Table 7.19: Model Evaluation Metrics per Well

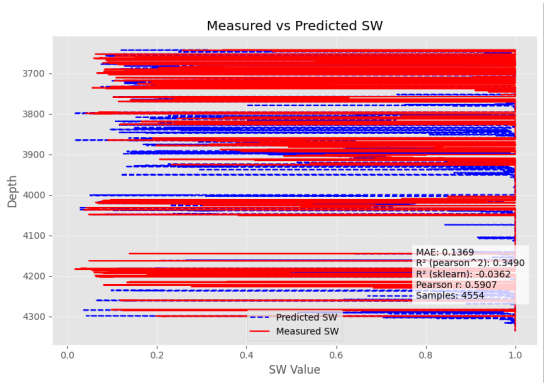
Well	MAE	MSE	R^2	Samples
Well 1	0.1546	0.0600	0.5815	4161
Well 2	0.1369	0.0808	-0.0362	4554
Well 3	0.1249	0.0604	0.2502	5112
Well 4	0.1570	0.0950	-0.0986	6151
All Wells	**0.1437**	**0.0756**	**0.2943**	**19978**

Best MLP Model

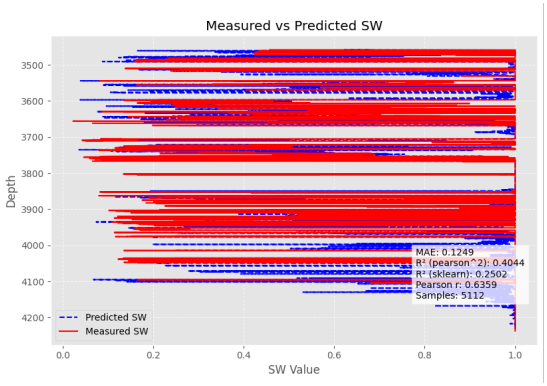
Based on the evaluation results, we conclude that the best-performing MLP architecture for predicting water saturation (S_w) is the one with 16 neurons in each hidden layer. Deeper models with more neurons tend to overfit the training data, especially when the number of informative features is limited or when the target distribution (often skewed with many values near 1) does not benefit from additional model capacity. The predictive accuracy of the best MLP model on each test well is illustrated in the following figures:



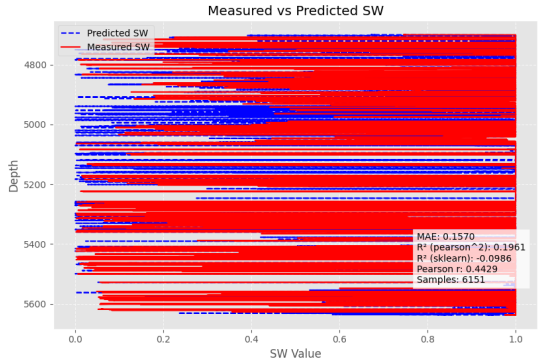
(a) Prediction for Well 1



(b) Prediction for Well 2



(c) Prediction for Well 3



(d) Prediction for Well 4

Figure 7.21: Prediction performance of the best MLP model for each test well.

The following plot presents the comparison between measured and predicted S_w values for the entire test dataset, using the best-performing MLP model:

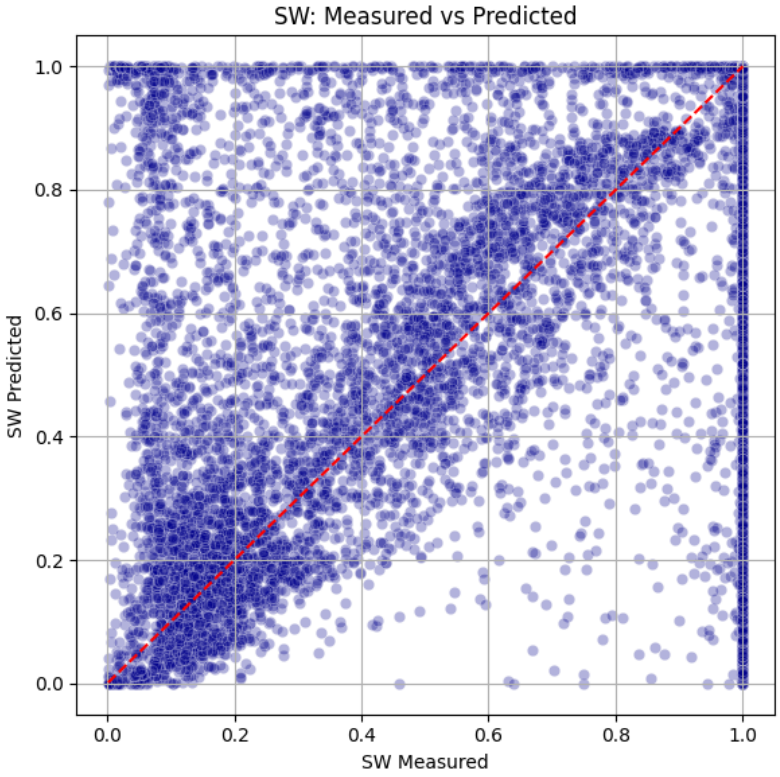


Figure 7.22: Measured vs Predicted S_w for the test set using the best MLP model

Predicting Water Saturation (S_w) is inherently more complex than predicting the other parameters due to its strong dependence on resistivity-based measurements, which tools are highly sensitive and often subject to measurement noise or environmental effects, which can introduce uncertainty into the logs.

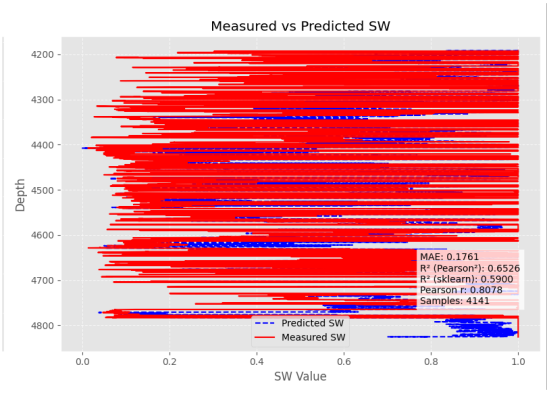
7.3.3 XGBoost Results

These results present the performance of the XGBoost model in predicting the target variable S_w across each test well, evaluated using standard metrics:

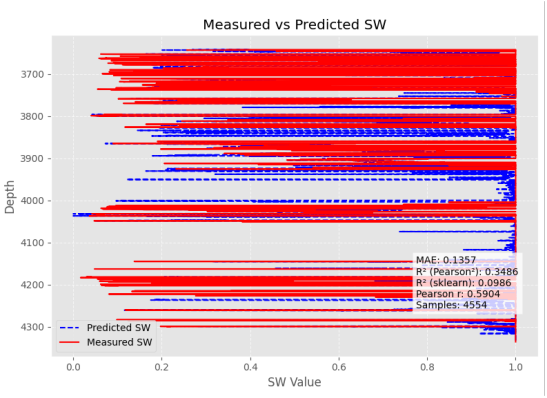
Table 7.20: Prediction Metrics per Well for XGBoost

Well	MAE	RMSE	R^2	Samples
Well 1	0.1766	0.0595	0.5890	4141
Well 2	0.1365	0.0710	0.0972	4554
Well 3	0.1396	0.0572	0.2869	5112
Well 4	0.1937	0.0719	0.2021	6122
All Wells	**0.1632**	**0.0653**	**0.4021**	**19949**

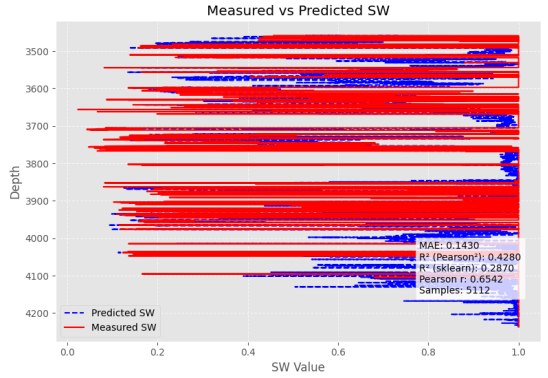
The graphs below illustrate how the model performed when applied to unseen data from the various test wells:



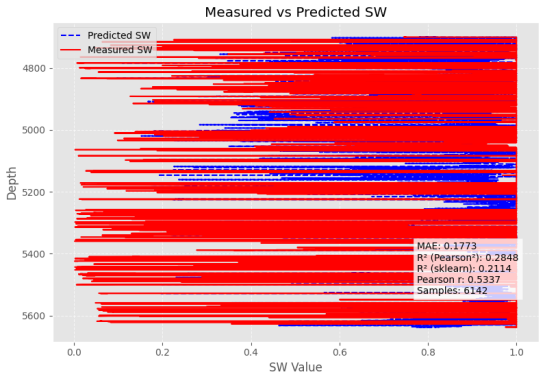
(a) Prediction for Well 1



(b) Prediction for Well 2



(c) Prediction for Well 3



(d) Prediction for Well 4

Figure 7.23: Prediction performance of the XGBoost model for each test well.

The following plot presents the comparison between measured and predicted S_w values for the entire test dataset, using XGBoost:

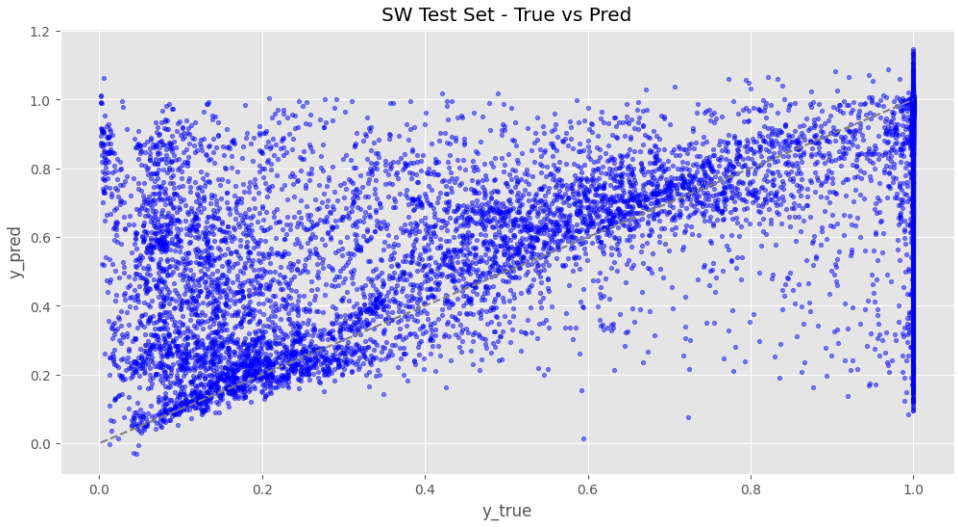


Figure 7.24: Measured vs. Predicted S_w for the test set using the XGBoost model

Compared to Linear Regression and MLP, the XGBoost model achieved better performance in predicting water saturation (S_w). This improvement can be attributed to XGBoost's ability to perform both regression and classification-like handling within its gradient boosting framework.

Additionally, XGBoost handles imbalanced or skewed data distributions more effectively, which is particularly relevant for S_w, where the dataset often contains numerous values close to 1.

However, despite the improved performance of XGBoost in predicting water saturation (S_w), the $\mathbf{R^2}$ score remains relatively low. This suggests that the model is still unable to fully capture the underlying complexity of S_w, which is heavily influenced by sensitive and nonlinear resistivity responses. Therefore, there is a need to explore more advanced approaches, such as Convolutional Neural Networks (CNNs), to enhance the model's ability to generalize and achieve more accurate predictions.

7.3.4 CNN Results

In this section, we present the results of the optimized 1D convolutional neural network (CNN) architecture applied to water saturation (S_w) prediction. The CNN model is composed of stacked 1D convolutional layers with a **kernel size of 1 and 32 filters**, followed by batch normalization, global max pooling, and fully connected dense layers with dropout for regularization.

Specifically, the network input shape is $(1, 14)$, corresponding to **14 features arranged as a single timestep** with multiple channels. This feature set includes the original logging measurements alongside predicted values of V_{cl} and $PHIE$ from previously trained models, simulating a sequential inference workflow.

The model was compiled with the Adam optimizer and trained to minimize the mean squared error (MSE) loss while monitoring mean absolute error (MAE) as an evaluation metric. Early stopping and checkpoint callbacks were applied to prevent overfitting, with a maximum of 150 epochs and a patience of 50 epochs based on validation MAE.

This dynamic pipeline—injecting predicted V_{cl} and $PHIE$ values as inputs for S_w prediction—reflects a realistic deployment scenario in petrophysical workflows. Strict separation of training and testing datasets was maintained throughout to avoid data leakage.

After prediction, the S_w values were **clipped** to the physically meaningful range $[0, 1]$, leveraging domain knowledge about saturation limits. This clipping step did not negatively impact overall model performance.

The final feature set for S_w prediction thus consisted of:

- Original log measurements: GR, AT10, AT20, AT30, AT60, AT90, DTP, NPHI, RHOB, URAN, THOR, POTA
- Predicted logs of V_{cl} and $PHIE$ from prior models

Evaluation on the test dataset showed robust predictive performance, with R^2 exceeding **0.81**, confirming the effectiveness of this staged hybrid CNN approach.

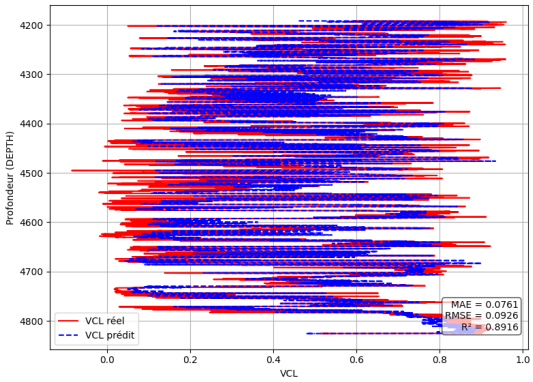
The following table summarizes the key prediction metrics—Mean Absolute Error (MAE), Mean Squared Error (MSE), and coefficient of determination (R^2)—evaluated

for each Well as well as for the combined dataset. These metrics provide insight into the accuracy and reliability of the model's performance across different wells.

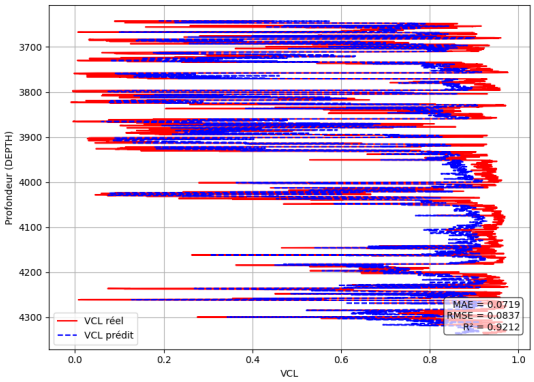
Table 7.21: Prediction Metrics by Well

Well	MAE	MSE	R^2
Well 1	0.0400	0.0046	0.9678
Well 2	0.0210	0.0028	0.9638
Well 3	0.0219	0.0027	0.9662
Well 4	0.1386	0.0560	0.3523
All Wells	0.0614	0.0195	0.8177

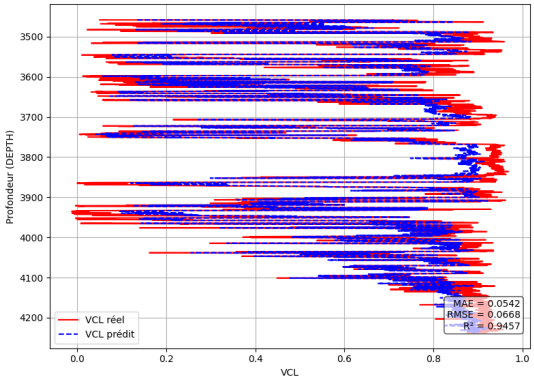
The following figure illustrates the model's predictive performance for each individual well, highlighting variations in accuracy and error metrics across different geological settings:



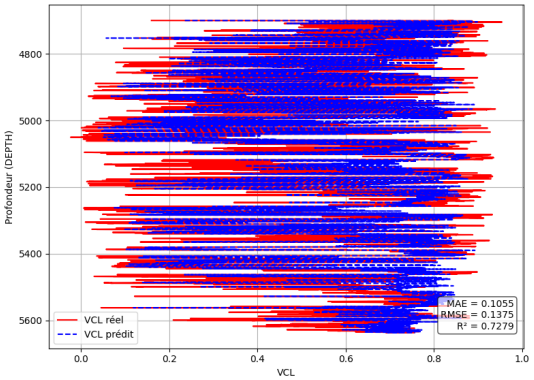
(a) Prediction for Well 1



(b) Prediction for Well 2



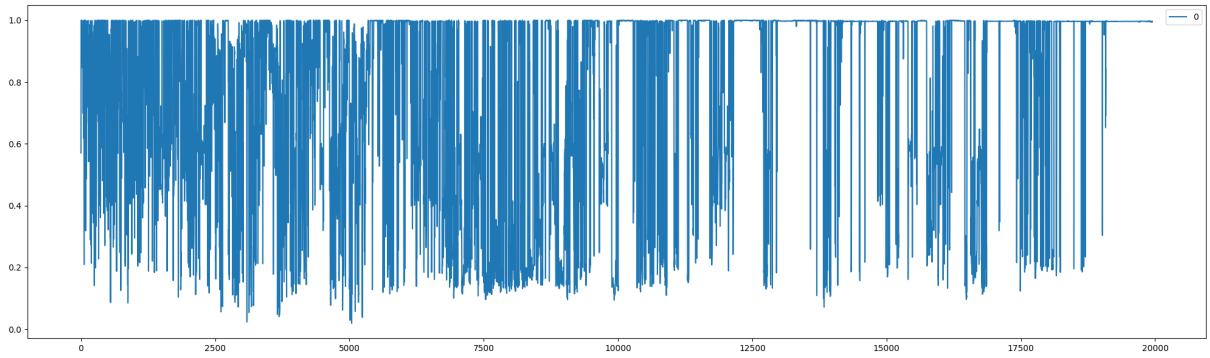
(c) Prediction for Well 3



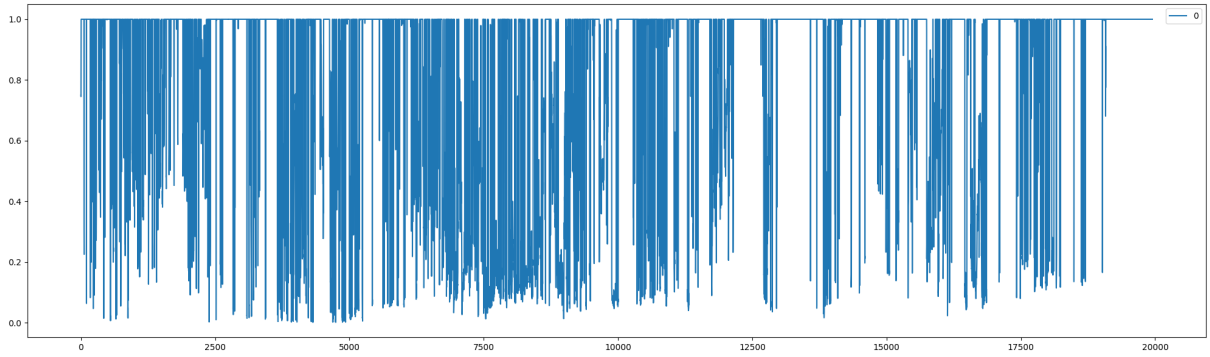
(d) Prediction for Well 4

Figure 7.25: Prediction vs. Measured performance of the CNN model for each test well for S_w.

The overall distribution of the true and predicted water saturation (S_w) values across all wells in the test dataset is displayed next. These visualizations provide an intuitive comparison of the model's predictive performance, highlighting its ability to capture the general trends and variability of S_w. While the true values represent the measured data, the predicted values demonstrate the CNN model's capacity to approximate these measurements, validating the model's practical application potential.



(a) Actual S_w



(b) Predicted S_w

Figure 7.26: Overall Predicted vs. Actual S_w Values

The results demonstrate strong predictive accuracy for Well 1, Well 2, and Well 3, with R^2 values exceeding 0.96 and low MAE and MSE, indicating effective modeling of petrophysical properties in these wells.

In contrast, Well 4 exhibits a notable decrease in performance, with a much lower R^2 of 0.35 and higher error metrics. This drop may be due to more complex geological variability, such as heterogeneous lithology, changes in fluid saturation, or poorer quality of logging data. Additionally, Well 4 could contain reservoir zones that differ significantly from the training set distribution, challenging the model's generalization capability.

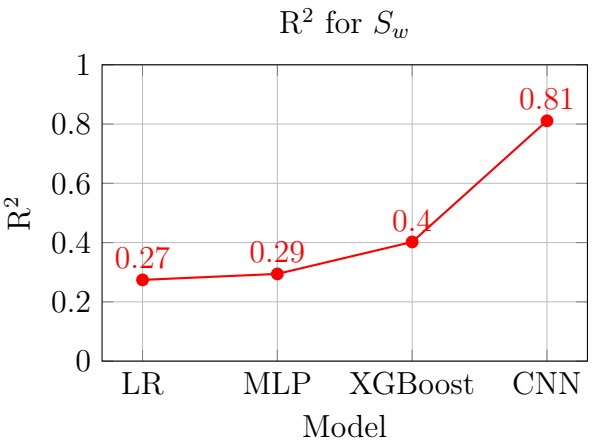
These observations suggest that while the model performs reliably in most wells, specialized treatment or model adaptation may be necessary to maintain accuracy in more geologically complex wells like Well 4.

7.3.5 Comparative Study (S_w)

It is observed that the progression of model performance in predicting water saturation (S_w), is more complicated since it depends on indirect factors such as porosity, resistivity, clay content, and saturation models (like Archie's equation), which varies significantly across formations.

Table 7.22: Metrics for S_w

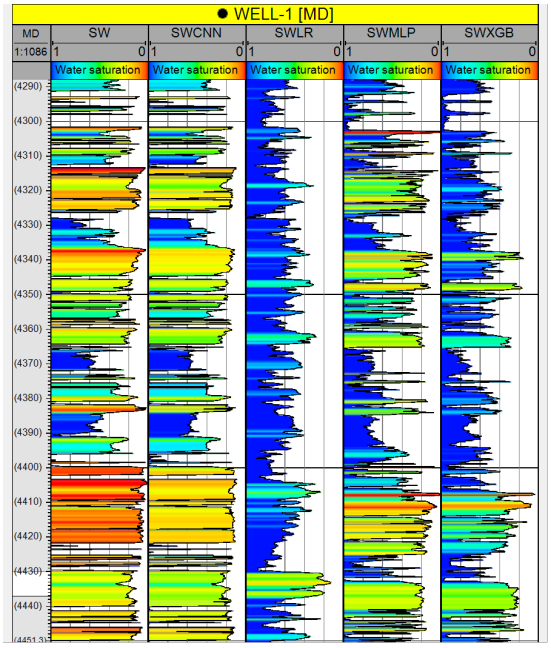
Model	MAE	MSE	R^2
LR	0.2200	0.0778	0.2741
MLP	0.1437	0.0756	0.2943
XGBoost	0.1632	0.0653	0.4021
CNN	0.0614	0.0195	0.8177



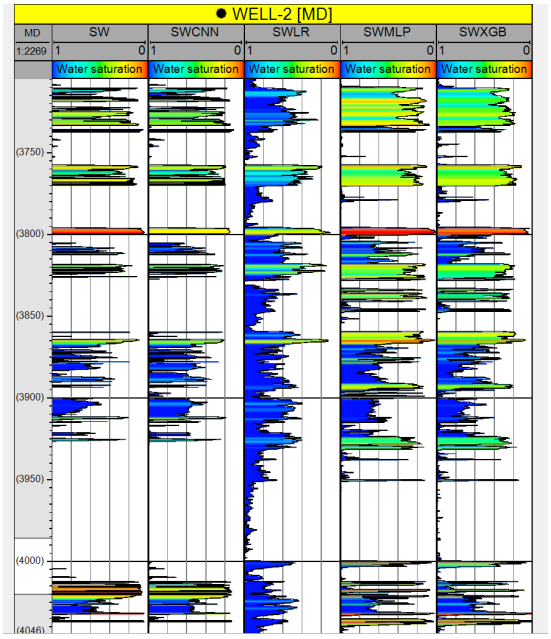
From the table of the summary matrix and the plot, the performance insights are as follows:

- Linear Regression (LR) performs poorly ($R^2 = 0.2741$), confirming the complex, non-linear nature of water saturation behavior.
- MLP and XGBoost offer small improvements but still suffer from relatively low R^2 values, indicating difficulty in capturing the necessary features from the base input alone.
- CNN using additional features (V_{cl}, $PHIE$) shows a major improvement ($R^2 = 0.8177$), highlighting the importance of including V_{cl} and $PHIE$ as input features and the ability of CNNs to handle complex, multi-feature interactions.

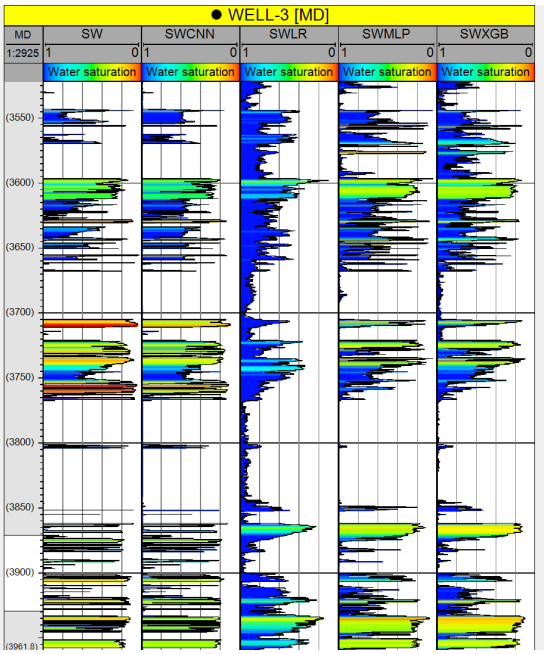
This is further illustrated through the following well log sections:



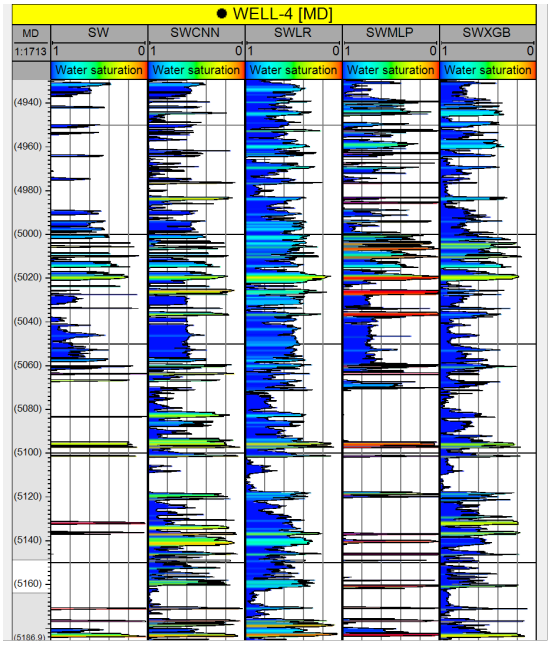
(a) Well 1 – S_w Prediction



(b) Well 2 – S_w Prediction



(c) Well 3 – S_w Prediction



(d) Well 4 – S_w Prediction

Figure 7.27: Log comparisons and S_w predictions for selected wells

7.4 Conclusion

Across all three target variables—clay volume (V_{cl}), effective porosity ($PHIE$), and water saturation (S_w)—the Convolutional Neural Network (CNN) model achieved the best performance when compared to Linear Regression (LR), Multi-Layer Perceptron (MLP), and XGBoost. For V_{cl}, CNN yielded the lowest MAE (0.0766) and highest R^2 (0.8873), outperforming other models in both accuracy and reliability. In the case of $PHIE$, CNN again led with an R^2 of 0.7651 and the smallest error metrics. Most notably, for S_w prediction—a particularly complex task—CNN significantly outperformed the rest, achieving an R^2 of 0.8177 compared to only 0.4021 with XGBoost, despite using additional features (V_{cl} and $PHIE$) as inputs. This superiority stems from CNN's ability to extract local patterns and spatial dependencies within the well log data, allowing for more nuanced and robust learning than traditional methods. These findings clearly establish CNN as the most capable and generalizable model for petrophysical prediction in this study.

Pilot Well Log

The aim of these predictions is to shift from human-based interpretations and calculations, to a more automised and intelligent approach. Here is the pilot logging of our final work, that can be implemented *in-situ* for real-time predictions, where the main well logs (such as Gamma Ray, Density, and Neutron Porosity) are displayed alongside the predicted petrophysical parameters: S_w, $PHIE$, and V_{cl}.

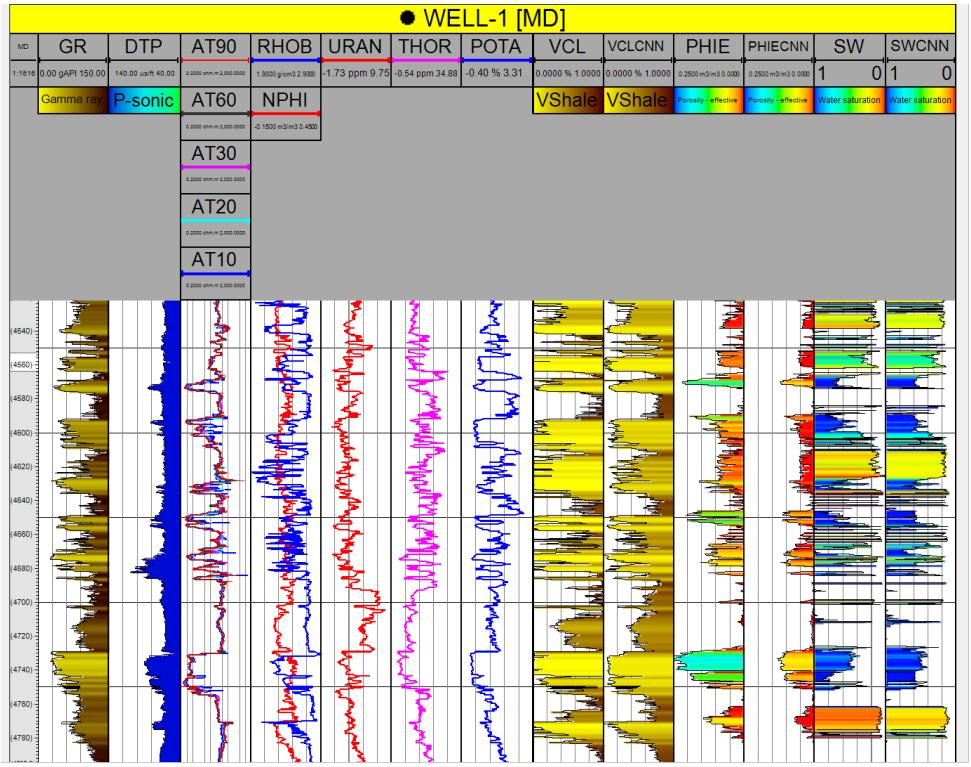


Figure 7.28: Pilot Logging Display with Input Logs and Predicted Petrophysical Parameters (S_w, $PHIE$, V_{cl})

General Conclusion

The purpose of this study focused on the prediction of petrophysical properties using machine learning techniques, with the aim of improving the characterization of hydrocarbon reservoirs despite many challenges faced with the data related to oil and gas exploration.

We performed a detailed exploratory data analysis (EDA) to filter and transform the raw dataset. Then, we selected, developed, and evaluated several machine learning models suitable for the problem, including XGBoost and neural networks for predicting effective porosity $PHIE$, volume of clay V_{cl}, and water saturation S_w. These models were carefully assessed using metrics, such as R^2 and MSE, to select the best performing ones for further deployment.

However, the best retained model consists of the conventional neural network trained with a custom Focal Loss function, whose performance achieved R^2 scores above **0.76**, **0.88**, and **0.81** for $PHIE$, V_{cl}, and S_w prediction respectively, demonstrating an ability to handle class imbalance and noisy data.

One of the major limitations was the obvious imbalance in the data distribution, leading to a bias in the model. Despite this imperfection, the richness of logs provided enough information to adjust the models for producing meaningful predictions.

Future projects related to this research work could include integrating drilling parameters such as penetration rate and torque, which could further improve the model performance. Additionally, applying these models to a wider range of wells would help assess generalization, or exploring advanced machine learning algorithms that allow the integration of physical concepts to train the model based on strong logical reasoning.

In conclusion, this modest study demonstrates that even with input well log data, reliable predictions of key petrophysical parameters are achievable with great interest through careful data preparation and judicious use of machine learning techniques. These models enable promising real-time predictions during Logging While Drilling (LWD) or Wireline Logging, likely to revolutionize the characterization of hydrocarbon reservoirs by providing faithful and valuable information.

Bibliography

[1] Oxford English Dictionary. *"petrophysics (n.),"*. https://doi.org/10.1093/OED/1133649562. 2023.

[2] Steve Cannon. *Petrophysics: A Practical Guide*. Accessed via ProQuest Ebook Central. John Wiley & Sons, Incorporated, 2015. URL: https://ebookcentral.proquest.com/lib/ecolenati/detail.action?docID=7104024.

[3] G. E. Archie. "Introduction to Petrophysics of Reservoir Rocks". In: *AAPG Bulletin* 34.5 (1950), pp. 943–961.

[4] E. C. Thomas. "50th Anniversary of the Archie Equation: Archie Left More Than Just an Equation". In: *The Log Analyst* (May 1992). Invited paper; includes slides from the presentation at Shell Oil Company.

[5] Society of Petroleum Engineers. *Petrophysics Free*. FreeView content licensed by SPE. Petrowiki content subtype. Jan. 2025. URL: https://doi.org/10.2118/PW0706.

[6] Yuan Zhe Ma. "Uncertainty Analysis in Reservoir Characterization and Management". In: *AAPG Memoir 96* 96 (Dec. 2011), pp. 1–15.

[7] Bahareh Mirghaed, Abolfazl Dehghan Monfared, and Ali Ranjbar. "Enhanced petrophysical evaluation through machine learning and well logging data in an Iranian oil field". In: *Scientific Reports* 14 (Nov. 2024). DOI: 10.1038/s41598-024-80362-w.

[8] J. F. Bristow. "Real-time Formation Evaluation for Optimal Decision Making While Drilling: Examples from the Southern North Sea". In: *Geological Applications of Well Logs*. Ed. by M. Lovell and N. Parkinson. Vol. 13. AAPG Methods in Exploration. 2002, pp. 1–13.

[9] Ao Li et al. "From Streamline to Pathline: Visualizing Particle Trajectories Under Changing Velocity Fields". In: *SPE Journal* 29 (2024), pp. 3801–3812. DOI: 10.2118/215088-PA. URL: https://doi.org/10.2118/215088-PA.

[10] Yasir Shahzad. *Evaluation of the Enhance Oil Recovery (EOR) Potential of Water and Gas Injection in Oil Shale Reservoirs*. Tech. rep. Available at SSRN. SSRN, Feb. 2014. DOI: 10.2139/ssrn.2478558. URL: https://ssrn.com/abstract=2478558.

[11] F. Hormozzade Ghalati et al. "Petrophysical analysis of geothermal systems at Mount Meager, southwestern British Columbia (part of NTS 092J)". In: *Geoscience BC Summary of Activities 2023*. Report 2024-01. Geoscience BC, 2024, pp. 93–100.

[12] Shuvajit Bhattacharya et al. "Integrated Petrophysical Studies for Subsurface Carbon Sequestration". In: *Proceedings of the SPWLA 64th Annual Logging Symposium*. Lake Conroe, Texas, USA, June 2023. DOI: 10.30632/SPWLA-2023-0003. URL: https://doi.org/10.30632/SPWLA-2023-0003.

[13] Leonid Buryakovsky et al. *Fundamentals of the Petrophysics of Oil and Gas Reservoirs*. Salem, Massachusetts: Scrivener Publishing LLC, 2012. ISBN: 9781118354894.

[14] Djebbar Tiab and Erle C. Donaldson. *Petrophysics: Theory and Practice of Measuring Reservoir Rock and Fluid Transport Properties*. 2nd ed. Oxford: Gulf Professional Publishing, 2004, p. 123.

[15] C.L.G. Amorim et al. "Effect of clay-water interactions on clay swelling by X-ray diffraction". In: *Nuclear Instruments and Methods in Physics Research. Section A, Accelerators, Spectrometers, Detectors and Associated Equipment* 580.1 (Sept. 2007), p. 768–770. ISSN: 0168-9002. DOI: 10.1016/j.nima.2007.05.103.

[16] Harpreet Singh and Jianchao Cai. "Chapter 6 - Permeability of Fractured Shale and Two-Phase Relative Permeability in Fractures". In: *Petrophysical Characterization and Fluids Transport in Unconventional Reservoirs*. Ed. by Jianchao Cai and Xiangyun Hu. Elsevier, 2019, pp. 105–132. ISBN: 978-0-12-816698-7. DOI: https://doi.org/10.1016/B978-0-12-816698-7.00006-1. URL: https://www.sciencedirect.com/science/article/pii/B9780128166987000061.

[17] Morris Muskat and Milan W. Meres. "The Flow of Heterogeneous Fluids Through Porous Media". In: *Physics* 7.9 (1936), pp. 346–363. DOI: 10.1063/1.1745403. URL: https://doi.org/10.1063/1.1745403.

[18] G. E. Archie. "The Electrical Resistivity Log as an Aid in Determining Some Reservoir Characteristics". In: *Transactions of the AIME* 146.01 (1942), pp. 54–62. DOI: 10.2118/942054-G.

[19] A. M. Mohamad and G. M. Hamada. "Determination techniques of Archie's parameters: a, m and n in heterogeneous reservoirs". In: *Journal of Geophysics and Engineering* 14.6 (Dec. 2017), pp. 1358–1367. DOI: 10.1088/1742-2140/aa805c. URL: https://doi.org/10.1088/1742-2140/aa805c.

[20] Schlumberger. "Wettability and Its Effect on Oil Recovery". In: *Oilfield Review* 28.2 (2016). Available at: https://www.slb.com/resource-library/article/opr/wettability-and-its-effect-on-oil-recovery, pp. 4–19.

[21] John H. Doveton. *Principles of Mathematical Petrophysics*. Accessed via ProQuest. Amsterdam: Elsevier, 2014. ISBN: 9780444634117.

[22] Djebbar Tiab and Erle C. Donaldson. *Petrophysics: Theory and Practice of Measuring Reservoir Rock and Fluid Transport Properties*. 4th ed. Boston: Gulf Professional Publishing, 2015. ISBN: 9780128031889.

[23] Malcolm Rider and Martin Kennedy. *The Geological Interpretation of Well Logs*. 3rd ed. Sutherland, UK: Rider-French Consulting, 2011. ISBN: 9780954190685.

[24] George Asquith et al., eds. *Basic Well Log Analysis*. Vol. 16. AAPG Methods in Exploration. ISBN print: 0891816674. Tulsa, OK: American Association of Petroleum Geologists, 2004. ISBN: 9781629810492. DOI: 10.1306/Mth16823.

[25] M. R. J. Wyllie, A. R. Gregory, and L. W. Gardner. "Elastic Wave Velocities in Heterogeneous and Porous Media". In: *Geophysics* 21.1 (1956), pp. 41–70. DOI: 10.1190/1.1438217.

[26] Wellsite Inc. *Mudlog 8.8.0*. https://wellsight.com/software/mudlog-8-8-0. Mudlogging software, accessed: 2023-08-15. 2023.

[27] AAPG Wiki Contributors. *Density-neutron log porosity*. 2023. URL: https://wiki.aapg.org/Density-neutron_log_porosity.

[28] u/ThrowAwaySoIL. *Anyone make cool stuff out of rock cores?* Posted on *r/Geotech*. 2023. URL: https://www.reddit.com/r/Geotech/comments/13941dk/anyone_make_cool_stuff_out_of_rock_cores/.

[29] U.S. Geological Survey. *Photomicrograph of a rock thin section showing a gabbroic inclusion*. https://www.usgs.gov/index.php/media/images/photomicrograph-rock-thin-section-gabbroic-inclusion. Image. 2016.

[30] Ruidong Qin et al. "Petrophysical parameters prediction and uncertainty analysis in tight sandstone reservoirs using Bayesian inversion method". In: *Journal of Natural Gas Science and Engineering* 55 (2018), pp. 431–443. ISSN: 1875-5100. DOI: 10.1016/j.jngse.2018.04.031. URL: https://www.sciencedirect.com/science/article/pii/S1875510018301872.

[31] Said Hassaan et al. "Real-Time Prediction of Petrophysical Properties Using Machine Learning Based on Drilling Parameters". In: *ACS Omega* 9.15 (2024), pp. 17066–17075. DOI: 10.1021/acsomega.3c08795. URL: https://doi.org/10.1021/acsomega.3c08795.

[32] Chris Smith. *The History of Artificial Intelligence*. 2006.

[33] *Topics in Artificial Intelligence*. https://www.ibm.com/think/topics/artificial-intelligence. Accessed: 2025-05-31.

[34] Hussein Belyadi. *Machine Learning Guide for Oil and Gas Using Python*. Gulf Professional Publishing, 2021.

[35] IBM. *Artificial Intelligence*. https://www.ibm.com/think/topics/artificial-intelligence. Accessed: 2025-06-27. n.d.

[36] Dan Shewan. *9 Applications of Machine Learning from Day-to-Day Life*. https://www.wordstream.com/blog/ws/2017/07/28/machine-learning-applications. Accessed: 2025-06-27. 2017.

[37] *Topics in Supervised Learning*. https://www.ibm.com/think/topics/supervised-learning. Accessed: 2025-05-31.

[38] Godwin Ola, Jackson Tyler, and Jordan Nelson. "Introduction to Supervised Learning". In: (Mar. 2025).

[39] Farnoush Farhadi. "Learning Activation Functions in Deep Neural Networks". In: (Dec. 2017).

[40] Farnoush Farhadi. "Learning Activation Functions in Deep Neural Networks". ProQuest Dissertations & Theses, Publication No. 10957109. PhD thesis. École Polytechnique, Montréal (Canada), 2017.

[41] *Topics in Logistic Regression*. https://www.ibm.com/think/topics/logistic-regression. Accessed: 2025-05-31.

[42] *Topics in Unsupervised Learning*. https://www.ibm.com/think/topics/unsupervised-learning. Accessed: 2025-05-31.

[43] Noor Saud Abd and Kamel Karoui. "The Importance of the Clustering Model to Detect New Types of Intrusion in Data Traffic". In: (Nov. 2024).

[44] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. 2nd ed. MIT Press, 2015.

[45] IBM. *Reinforcement Learning*. https://www.ibm.com/think/topics/reinforcement-learning. Accessed: 2025-06-27. n.d.

[46] Ganesh Viswanathan et al. In: *World Journal of Advanced Engineering Technology and Sciences* (Mar. 2025).

[47] GeeksforGeeks. *Neural Networks Architecture [Image]*. https://media.geeksforgeeks.org/wp-content/cdn-uploads/20230602113310/Neural-Networks-Architecture.png. Accessed: 2025-06-27. 2023.

[48] Yuxiong Sun. "A Stock Price Prediction Model Based on MLP". In: (Apr. 2025).

[49] Ashutosh Hathidara and Lalit Pandey. "Implementing an Artificial Quantum Perceptron". In: (Jan. 2025).

[50] GeeksforGeeks. *What is Perceptron? The Simplest Artificial Neural Network*. https://www.geeksforgeeks.org/what-is-perceptron-the-simplest-artificial-neural-network/. Accessed: 2025-06-27. n.d.

[51] Emma Oye and Rebekah Lucas. "Convolutional Neural Networks (CNNs)". In: (Dec. 2024).

[52] Carlos Martinez and Nicole Robinson. "Extensive Review and Comparison of CNN and GAN". In: (Dec. 2024).

[53] Marjan Qazvini. "Forecasting Mortality in the Middle-Aged and Older Population of England: A 1D-CNN Approach". In: (Oct. 2024).

[54] R. Boiger et al. "Direct Mineral Content Prediction from Drill Core Images via Transfer Learning". In: *arXiv* (2024).

[55] J. C. Teixeira et al. “Neural Machine Translation of Seismic Waves for Petrophysical Inversion”. In: (2024).

[56] “Machine Learning-Based Prediction of Well Logs Guided by Rock Physics and Its Interpretation”. In: *Sensors* 25.3 (2025), p. 836.

[57] H. Shang et al. “Lithofacies Prediction from Well Log Data Using Deep Learning”. In: *MDPI* (2024).

[58] J. Ullah et al. “RockDNet: Deep Learning Approach for Lithology Classification”. In: *MDPI* (2024).

[59] Ministry of Energy, Mines and Renewable Energies. *Energy Sector Overview*. Government of Algeria. Ministry of Energy, Algeria, 2010.

[60] G. Geleazzi. “Stratigraphy and Structural Evolution of the Berkine Basin”. In: *Petroleum Geoscience* 16.2 (2010). Hypothetical values, adjust if needed, pp. 123–135.

[61] World Energy Council. *World Energy Outlook*. WEC Publication. World Energy Council, 2007.

[62] IBM Editorial Team. *What is Exploratory Data Analysis (EDA)?* Accessed: 2025-06-17. 2020. URL: https://www.ibm.com/think/topics/exploratory-data-analysis.

[63] Sonatrach. *Document Sonatrach*. Unpublished document. Accessed: 2025-06-27. n.d.