PONTIFÍCIA UNIVERSIDADE CATÓLICA
DO RIO DE JANEIRO

Ciro dos Santos Guimarães

Permeability Predictions Using Borehole Logs and Well Testing Data: A Machine Learning Approach

Dissertação de Mestrado

Dissertation presented to the Programa de Pós-graduação em Engenharia Mecânica of PUC-Rio in partial fullfillment of the requirements for the degree of Mestre em Engenharia Mecânica

Advisor: Prof. Ivan F. M. Menezes

Co-Advisor: Prof. Hélio C. V. Lopes

Rio de Janeiro
March 2021

PONTIFÍCIA UNIVERSIDADE CATÓLICA
DO RIO DE JANEIRO

Ciro dos Santos Guimarães

Permeability Predictions Using Borehole Logs and Well Testing Data: A Machine Learning Approach

Dissertation presented to the Programa de Pós-graduação em Engenharia Mecânica of PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Engenharia Mecânica. Approved by the Examination Committee.

Prof. Ivan F. M. Menezes
Advisor
Departamento de Engenharia Mecânica – PUC-Rio

Prof. Hélio C. V. Lopes
Co-Advisor
Departamento de Informática – PUC-Rio

Dr. Priscila Magalhães Ribeiro
PETROBRAS - Petróleo Brasileiro S.A.

Dr. Rafael Jesus de Moraes
PETROBRAS - Petróleo Brasileiro S.A.

Prof. Abelardo Borges Barreto Jr.
Departamento de Engenharia Mecânica – PUC-Rio

Rio de Janeiro, 12th of March, 2021

Ciro dos Santos Guimarães

The author graduated in Mechanical Engineering from PUC-Rio – 2011. The author joined Petrobras in 2012, as a Petroleum Engineer, and holds a postgraduate degree in petroleum engineering from Petrobras Corporate University (2013, Brazil).

Bibliographic data

Guimarães, Ciro dos Santos

Permeability predictions using borehole logs and well testing data : a machine learning approach / Ciro dos Santos Guimarães ; advisor: Ivan F. M. Menezes ; co-advisor: Hélio C. V. Lopes. – 2021.
104 f. : il. color. ; 30 cm

Dissertação (mestrado)–Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Mecânica, 2021.
Inclui bibliografia

1. Engenharia Mecânica – Teses. 2. Permeabilidade. 3. Aprendizado de máquina. 4. Petrofísica. 5. Perfis de imagem. 6. Teste de formação. I. Menezes, Ivan Fábio Mota de. II. Lopes, Hélio Côrtes Vieira. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Mecânica. IV. Título.

CDD: 621

To my wife, Isabela,

And our loving son Benjamin.

Acknowledgements

First and foremost, I would like to thank my wife Isabela and our beautiful son Ben for their unwavering patience and unconditional love; my brother and father for inspiring me to embark on this journey; and my mother for being my resting place in times of trouble.

I would also like to thank my advisor Prof. Ivan Menezes, my co-advisor Prof. Helio Lopes, and Prof. Abelardo Borges Barreto Jr. for their guidance throughout the course of this work. A special thanks to Luiz Schirmer and Guilherme Schardong for sharing their invaluable knowledge and expertise in the field of data science.

I would like to thank Petrobras for giving me the opportunity to carry out this study and for providing the field data analyzed in this research. My deepest gratitude to my manager, Ana Paula Martins, for conceding time for me to dedicate myself to this work. Furthermore, I would like to recognize the support of my fellow colleagues at Petrobras who assisted me in many work-related matters so that I could dedicate time to this project. A special thank you to Christiane de Camargo, for all the support throughout my career.

My sincerest gratitude to Diane Viana for showing me that within my vulnerability shall come forth my strength.

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de NívelSuperior - Brasil (CAPES) - Finance Code 001.

Abstract

Guimarães, Ciro dos Santos; Menezes, Ivan Fábio Mota de (Advisor); Lopes, Hélio Côrtes Vieira (Co-Advisor). **Permeability Predictions Using Borehole Logs and Well Testing Data: A Machine Learning Approach.** Rio de Janeiro, 2021. 104p. Dissertação de Mestrado – Departamento de Engenharia Mecânica, Pontifícia Universidade Católica do Rio de Janeiro.

This work investigates the performance of intelligent models on the forecasting of permeability in heterogeneous reservoirs. Production logs are used to compute loss functions for regression in the algorithms' optimization process. A flow profile interpretation method is used to remove wellbore skin effects from the measured flow rate. Additionally, a segmentation technique is applied to high-resolution ultrasonic image logs which provide not only the image of mega and giga pore systems but also identify the permeable facies along the reservoir. The image segmentation jointly with other borehole logs provides the necessary input data for the models' training process. The estimations presented herein demonstrate the algorithms' ability to learn non-linear relationships between geological input variables and a reservoir dynamic data even if the actual physical relationship is complex and not known a priori. Though the preprocessing stages of the procedure involve some data interpretation expertise, the algorithms can easily be coded in any programming language, requiring no assumptions on physics in advance. The proposed procedure provides more accurate permeability curves than those obtained from conventional methods, which may fail to predict the permeability measured on drill stem tests (DSTs) conducted in dual-porosity reservoirs. The novelty of this work is to incorporate dynamic production logging (PL) data into the permeability estimation workflow using machine learning algorithms.

Keywords

Permeability; Machine Learning; Petrophysics; Image logs; Well-testing; Production logs;

Resumo

Guimarães, Ciro dos Santos; Menezes, Ivan Fábio Mota de (Advisor); Lopes, Hélio Côrtes Vieira (Co-Advisor). **Previsões de Permeabilidade Utilizando Perfis à Poço Aberto e Dados de Teste de Formação: Uma Abordagem com Aprendizado de Máquinas.** Rio de Janeiro, 2021. 104p. Dissertação de Mestrado – Departamento de Engenharia Mecânica, Pontifícia Universidade Católica do Rio de Janeiro.

Este trabalho investiga o desempenho de modelos inteligentes na previsão de permeabilidade de reservatórios heterogêneos. Perfis de produção são utilizados para computar funções-objetivo para regressão no processo de otimização dos algoritmos. Um método de interpretação de perfil de produção é usado para remover efeitos de *skin* das medições de vazão. Adicionalmente, uma técnica de segmentação é aplicada a perfis de imagem acústica de alta resolução que fornecem, não apenas a imagem do sistema de mega e giga poros, mas também identifica fácies permeáveis ao longo do reservatório. A segmentação da imagem junto com outros perfis a poço aberto fornece os atributos necessários para o processo de treinamento do modelo. As estimativas apresentadas neste trabalho demonstram a habilidade dos algoritmos em aprender relações não lineares entre as variáveis geológicas e os dados dinâmicos de reservatório, mesmo quando a própria relação física é complexa e desconhecida à priori. Apesar das etapas de pré-processamento envolverem experiência em interpretação de dados, os algoritmos podem ser facilmente implementados em qualquer linguagem de programação, não assumindo qualquer premissa física de antemão. O procedimento proposto fornece curvas de permeabilidades mais acuradas que aquelas obtidas a partir de métodos convencionais que muitas vezes falham em prever a permeabilidade medida em testes de formação (TFR) realizados em reservatórios de dupla-porosidade. A contribuição deste trabalho é incorporar os dados dinâmicos oriundos dos perfis de produção (PP) ao processo de estimativa de permeabilidade usando algoritmos de Machine Learning.

Palavras-chave

Permeabilidade; Aprendizado de Máquina; Petrofísica; Perfis de Imagem; Teste de Formação; Perfis de Produção

Table of Contents

List of Figures

List of Tables

List of Algorithms

Acronyms

2D = Two dimensional
API = American Petroleum Institute
ANN = Artificial Neural Network
COP = Constrained Optimization Problem
CV = Cross-Validation
DST = Drill Stem Test
GOR = Gas-Oil Ration
II = Injectivity Index
KKT = *Karush-Kuhn-Tucker*
LSHSIM = Locality Sensitive Hashing Simulation
MLP = Multilayer Perceptron
NMR = Nuclear Magnetic Resonance
PDF = Probability Density Function
PI = Productivity Index
PL = Production Logging
PLT = Production Logging Tool
PTA = Pressure Transient Analysis
QP = Quadratic Programming Problem
ReLu = Rectified Linear Unit
REV = Representative Elementary Volume
SVM = Support Vector Machine
SVR = Support Vector Regression

Nomenclature

Elements in vector space are represented in boldface letters: vectors are represented in lowercase (e.g., $\boldsymbol{x}$ and $\boldsymbol{y}$) and matrices in uppercase types (e.g., $\boldsymbol{K}$ and $\boldsymbol{Q}$).

α = Lagrangian multiplier
B = Set of indexes activating box constraint
b = Constant offset
Real-valued bias
$\boldsymbol{b}$ = Vector-valued bias
C = Regularization parameter
Cost function
C' = Gradient of the cost function with respect to the output
c_o = Oil compressibility
c_r = Rock compressibility
c_t = Total system compressibility
$\boldsymbol{d}$ = Multivariate search direction
δ = Backpropagation recursive variable
E_i = Exponential integral function
ε = Error tolerance
f = Objective function of constrained optimization problems
$\boldsymbol{f}$ = Vector-valued function that performs linear regressions
φ = Nonlinear mapping function
Activation function
ϕ = Porosity
$\boldsymbol{g}$ = Gradient of the Lagrangian function
γ = Euler-Mascheroni constant
h = Uniform formation thickness
$\boldsymbol{I}$ = Identity matrix
$\boldsymbol{j}$ = All-ones vector
$\boldsymbol{J}$ = Jacobian matrix
k = Kernel function
Effective permeability
Number of independent variables
K = Number of subsets in a K-Fold Cross-Validation
$\boldsymbol{K}$ = Quadratic positive semidefinite kernel matrix
$\mathcal{K}$ = Feature Vector Space
l = Width of hidden layer
λ = One-dimensional parameter search

Neural network learning rate

$\mathcal{L}$ = Lagrangian function

$Mat1$ = High permeable matrix class

$Mat2$ = Low permeable matrix class

$M1$ = Fraction of $Mat1$ class

$M2$ = Fraction of $Mat2$ class

M_z = Magnetization of atomic nuclei

M_0 = Maximum magnetization of atomic nuclei

μ = Fluid viscosity

Lagrangian multiplier

Distribution’s mean value

n = Number of training sets

Depth of the artificial neural network structure

$Pore$ = Large-scaled porosity class

P = Fraction of $Pore$ class

p = Pressure

p_w = Pressure measured at wellbore radius

p'_w = Pressure at wellbore for reservoir with formation damage

q_w = Constant flow rate at the wellbore

$\boldsymbol{Q}$ = Quadratic Positive Semidefinite Matrix

r = Radius

Radial direction

r_w = Wellbore radius

$\mathbb{R}$ = Real coordinate space

s = Skin factor

$\boldsymbol{sgn}$ = Vector-valued signum function

σ = Distribution’s standard deviation

t = Time

T_1 = Time needed to reach 63% of total magnetization

T_2 = Transverse relaxation time

θ = Angle between two elements in vector space

$\boldsymbol{x}$ = Vector-valued input

$\mathcal{X}$ = Input Space

ξ = Representation of target deviations higher than the hyperplane

$\bar{\xi}_i$ = Representation of target deviations lower than the hyperplane

y = Real-valued target output

$\hat{y}$ = Real-valued measured output

$\boldsymbol{y}$ = Vector-valued output

$\hat{\boldsymbol{y}}$ = Vector-valued measured output

$\boldsymbol{w}$ = Normal vector to a hyperplane

Weights connecting layer to a neuron of the subsequent layer

$\boldsymbol{W}$ = Set of weights

Operators

$\|\boldsymbol{x}\|$ = Euclidean Norm of a vector
$|x|$ = Modulus of a real number
$\langle \boldsymbol{x}, \boldsymbol{y} \rangle$ = Inner product of two vectors
$\circ$ = Hadamard (elementwise) product of two vectors

Superscripts

$\boldsymbol{x}^*$ = Optimum solution
$\bar{\boldsymbol{\xi}}$ = Elements with target deviations below the hyperplane
(j) = j^{th} layer of a multilayered reservoir system
m = Arbitrary dimension of a real coordinate space
p = Arbitrary dimension of a real coordinate space
t = Transpose of matrix

Subscripts

a = Representation of a skin-affected zone
BF = Bound fluid
FF = Free fluid
i = Initial conditions
Representation of elements in a training set
Representation of a single layer in a neural network structure
k = Representation of an arbitrary element in input space
p = Element without target deviation from the optimal hyperplane
TOT = Total
w = Wellbore

1
Introduction

The relationship between the velocity of fluid flowing through a porous medium and the hydraulic gradient under which it flows was originally observed by Darcy in the late 19th century. Since then, countless techniques have been developed with the objective of quantifying and estimating this relation in terms of a second-rank symmetric tensor known as permeability. *In-situ* measurements, especially those from non-dynamic sources, have failed to deliver general and repeatable results. The main reason for that is that these estimations are highly dependent on the state of the rock, fluid saturation distribution, and its levels of anisotropy - i.e., the scale of the medium under investigation.

This work focuses on full-field heterogeneous formations in which different scales of porosity are present. The objective herein is to provide an accurate upscaling of permeability deduced from petrophysical logs into well-testing dimensions using modern machine learning techniques. Ultrasonic image jointly with other borehole logs provide the necessary input features while production logs provide suitable targets for the modeling process. The procedures presented in this work deliver more accurate estimations than the ones obtained from conventional methods, which fail to predict permeability results from drill stem tests (DSTs) conducted in dual-porosity systems. Moreover, the use of data-driven methods allows us to develop intelligent models capable of accurately learning non-linear relationships between geological variables and dynamic behavior even when the actual physical relationship is not known a priori.

The effectiveness of the proposed techniques has also been compared to benchmark results, such as the classical Timur-Coats relation. In addition, a field case study is provided.

1.1.
Permeability Models and the Scale Problem

Permeability is arguably one of the most important flow parameters associated with subsurface production and injection (Ahmed *et al.*, 1991). Darcy (1856) described it as an intrinsic characteristic of a porous material, which determines how easily fluid can flow through it. According to Hubbert (1956) and later Bear (1968), whenever the flow of fluids through porous media is discussed, the concept of a continuum is implicitly introduced. The continuum consists of an idealization under which fluids can be modeled on a macroscopic scale, even though it is composed by molecules, on a microscopic one.

In this case, a Representative Elementary Volume (REV) is defined in which properties consist of a statistical average of their values in a molecular length scale. In his work, Haldorsen (1986) recommends that the REV be smaller than the entire flow domain and larger than the size of a single pore. The idea is to avoid large fluctuations of properties observed in the microscopic length scale and, at the same time, avoid the gradual changes of properties caused by heterogeneity. Additionally, the magnitude of the REV depends on the nature of property under consideration.

According to Bachmat and Bear (1985), any elementary volume may be selected as a REV and the selection in any particular case depends only on the model’s objectives. Haldorsen defines four conceptual scales of averaging volumes in porous media: microscopic, macroscopic, megascopic, and gigascopic [1].

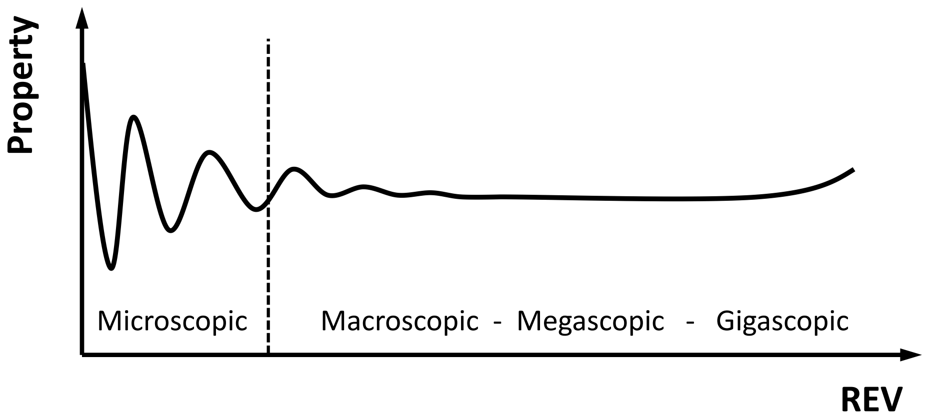

Figure 1.1: Variations of properties with REV (adapted from Bear, 1968).

---

[1] **Gigascopic** – Conceptual scale defined by Haldorsen (1986) for averaging volumes in porous media from seismic data and history matching.

As we move from the macroscopic to the gigascopic averaging, we lose detail in the representation of reality. Nonetheless, this may be considered acceptable depending on the stage of maturity of the field's development, and the level of risk one is willing to partake.

1.1.1.
Macroscopic Scale: rock samples and core plugs

The macroscopic scale represents the properties measured through rock samples and core plugs. According to Keelan (1972), conventional core analysis, especially in the early stages of field development, provides the most basic information required – lithology of the rock, type of fluid it contains, and its storage capacity. Although core analysis also allows the measurement of permeability in a controlled environment, the formation's deliverability can be considered no more than a rough estimation at this length scale. This is because, among many other reasons, the laboratory procedures for measuring permeability lack the means to reproduce the reservoir's relative permeability and overburden effects. This must be achieved through correlations.

1.1.2.
Megascopic Scale: petrophysical logs and well-testing

According to Collins *et al*. (1961), unique distributions of porosity do not exist. Therefore, to reach reliable estimates of average properties in porous media, the volume of sampling is considered exceptionally relevant. Haldorsen (1986) considers borehole logging to be the primary source of reservoir megascopic data. However, there are limitations to the volume it is capable of investigating. Furthermore, there is a need to correlate the properties deduced from wireline logs to their units measured through core analysis at the same depth.

Well-testing operations, on the other hand, can be considered a direct source of *in-situ* measurement of effective permeability, despite its interpretive nature. Additionally, it is easily capable of investigating thousands of feet into the reservoir, depending on the level of the system's transmissibility. Therefore, permeabilities measured through well-testing operations are truly an average over a vast volume, containing large heterogeneities in a megascopic scale.

1.1.3.
Gigascopic Scale: Seismic and History Matching

Intuitively, errors in average properties decrease as the investigated volume increases. Likewise, the need for a detailed representation of the reservoir in a megascopic scale decreases as the projects' maturity increases. At this time, seismic attributes and reservoir history matching begin to become more relevant. Conversely, rock samples, core plugs, petrophysical logs, and even well-testing data become luxuries in the advanced stages of reservoir modeling.

1.2.
Literature Review

In the megascopic scale of the permeability estimation workflow, Carman-Kozeny's relation constitutes the basis for several of the available techniques known today. This empirical relation is derived from Poiseuille's equation, which considers a laminar-viscous flow in straight-uniform non-communicating capillary tubes. Although it is a widespread relation, some of its parameters adopt varied values based on simplifying assumptions, which may affect the estimates' accuracy (Haro, 2006). Additionally, these predictions can become more challenging in carbonate formations due to the high levels of anisotropy caused by diagenesis and structural effects.

The heterogeneous nature leads to additional uncertainties in the reservoir's initial development plan (Fraga *et al.*, 2014). Hydraulic connectivity, sweep, oil displacement, and stimulation techniques are some of the main challenges faced in this scenario. As a result, sophisticated petrophysical and engineering methods are made necessary to better understand of both geological and dynamic aspects of the system and ultimately enhance the field's development strategy at early stages.

Timur (1968) combined Carman-Kozeny's relation with the model proposed by Korringa *et al.* (1962) for the relaxation mechanism of protons of a hydrogenous liquid in a porous medium. Through laboratory evidence, it was proved that nuclear magnetism logs could be successfully employed for matrix permeability predictions in sandstone reservoirs. However, there are limitations to open-hole logging in vuggy and fractured intervals. Additionally, core plugs cannot accurately respond in the presence of such structures (Sullivan, 2007). The work of Menezes de Jesus *et al*. (2017) addressed the issues concerning estimations for dual-porosity systems. The authors presented a regression method based on high-resolution ultrasonic image logs to estimate permeabilities in these scenarios.

Previous research has shown that the information contained in open hole logs can be used to predict dynamic behavior using Carman-Kozeny's relation as the foundation for feature-based methods (Timur, 1968). However successful, these methods rely on meticulous handling of features, which require some assumptions on the physics model in advance. Compared to feature-based learning, *Artificial Neural Networks* (ANNs), as well as *Support Vector Machines* (SVMs), can be considered more powerful tools for problems where the forward model has not yet been determined (Tian and Horne, 2017).

Many authors have investigated the ANN's applicability in petrophysical studies (e.g., Wong *et al*., 2000; Mohaghegh, 2005; Lim, 2005; Li and Misra, 2018; Li *et al*., 2019). Li and Misra (2018) were able to successfully synthesize nuclear magnetic resonance (NMR) T2 distributions by using deep neural networks. Because of their high pattern recognition capability, these intelligent models can be considered far superior to other regular statistical methods. The ANN's ability to reproduce non-linear behavior allows it to succeed in complex tasks such as porosity and permeability estimations (Helle *et al*., 2001).

In the pursuit of an algorithm with an even more promising generalization performance than the ANNs at permeability predictions, Al-anazi and Gates (2010) proposed the use of the SVM's adaptation to regression problems - Support Vector Regression (SVR). One of the main advantages of SVR is its compatibility with kernel methods to perform nonlinear regressions. This feature reduces the dependency between the computational complexity and the dimensionality of the input space. Thus, high levels of accuracy at low computational costs can be expected.

Despite these algorithms' capacities of generalizing functions to high volumes of data, the definition of the correct output measurement is critical. It directly impacts the scale upon which one wishes to perform their predictions. Ahmed *et al*. (1991) concluded that single-phase transient well-test data provide the best quantitative formation permeability if production logging (PL) data is available. Permeability, however, cannot be directly measured by production logging tools. Instead, it can be determined from PL pressure and flow-rate measurements (Stewart *et al.*, 1981; Larsen, 1982; Kucuk *et al.*, 1986; Ehlig-Economides and Joseph, 1987, Galvao and Guimaraes, 2017).

Sullivan (2007) presented PL data integration with rock samples, borehole logs, and pressure transient analysis (PTA). The author also proposed a *Simplified Approach*, assuming uniform skin across the wellbore, where layer permeability is distributed vertically according to the flow profile observed in the PL data and then normalized to the flow capacity measured with PTA. The author concludes that potential errors caused by uncertainties in layer pressure and skin distribution can be significant. The method proposed by Galvao and Guimaraes (2017) provides a practical approach for decoupling individual layer permeabilities and skin factors from measured flowrate.

1.3.
Motivation

Despite the numerous amounts of research available on permeability predictions, few are the ones that attempt to find robust correlations with data that can be considered a true average of the property over a vast porous volume. Additionally, for reservoirs containing structures with larger scales of porosity, such as caves and open fractures, the conventional sources of measurements can be misleading. Intelligent methods become useful in this scenario because they allow not only the incorporation of a greater mass of data to the model but also ones from different origins and length of scales.

A broader look into the oil field’s full domain is necessary to overcome the carbonate formation’s complex and heterogeneous nature. Although reservoir simulation is considered the most suitable tool for understanding how inter-well variations affect the movement of fluids, it involves a strong and iterative human-based effort before reaching an acceptable level of predictive accuracy. This research focuses on studying the applicability of fast supervised learning algorithms for integrating data from different scales and sparsely acquired inside a field’s hydraulic borders.

1.4.
Research Objectives and Contributions

The objective of this work is to predict permeability curves from petrophysical logs. More specifically, it aims to reach consistent correlations between acoustic borehole images, NMR logs, and production logs without many assumptions on the physics model in advance. Two machine learning algorithms, namely Support Vector Regression (SVR) and Artificial Neural Network (ANN), are implemented and trained through supervised learning. Their performance is examined with respect to computational efficiency and overall accuracy.

The main contribution of this work is two-fold: first, to incorporate production logging data to the permeability estimation workflow, which is typically performed using general well-testing results; secondly, to investigate the applicability of machine learning algorithms to this procedure, which could ultimately help overcome the complex challenge of understanding the physical relation between geological features and dynamic behavior in heterogeneous environments. The results represent an important step not only for the study of permeability in the megascopic length scale but also for future efforts towards the upscaling of this property into a gigascopic scale of the reservoir. Both algorithms implemented in this work were structured as plug-ins, to be incorporated into a commercial log analysis software (Interactive Petrophysics, 2019).

1.5.
Dissertation Outline

This dissertation is divided into six chapters, the first one being this introduction. The second chapter looks at the acquisition and preprocessing of the input and output data. For the input, a segmentation technique is used to extract relevant information from 2D ultrasonic image logs. For the output, an analytical solution for the inverse problem of a heterogeneous multilayered reservoir provides the means to compute accurate permeability curves from production logs. This chapter also gives an overview on relevant data structures and how it is implemented in Python Language (van Rossum, 1995).

Chapter 3 gives an overview of Machine Learning’s general concepts of supervised learning. In particular, the two regression algorithms implemented in this research are explained in greater detail. Once more, the pseudocode structures implemented in Python Language are provided.

Chapter 4 presents two synthetic examples, where both algorithms are properly implemented. It will be shown that they are capable of accurately reproducing a predefined relation between borehole logs and a well’s dynamic deliverability. Results are compared to a commercial machine learning library.

Chapter 5 presents a field case study, where four well locations provide the necessary data to train a predictive model. It compares the performance of the two algorithms amongst themselves, as well as with a benchmark relation. The objective of this chapter is to examine consistency, computational efficiency, and accuracy of the various methods.

Chapter 6 summarizes the result of this research and presents some final remarks. Outline recommendations for further work are also presented in this chapter.

2
Reservoir Data Acquisition and Processing

The purpose of this chapter is to provide an overview of the physical and geological aspects of the features used in this work. Data acquisition, processing, and interpretation are thoroughly discussed herein. Additionally, to illustrate the impact of formation damage on permeability estimations, the mathematical formulation of a multi-layered reservoir problem is presented. Finally, a data-structuring routine is provided to supply the algorithms with the appropriate training information.

2.1.
Petrophysics

Determining which geological properties have the greatest influence on a well’s dynamic deliverability is challenging, and feature selection methods can be computationally expensive. For dual-porosity systems, permeability is fundamentally dependent on matrix porosity and fracture network geometry, i.e., secondary porosity (Menezes de Jesus *et al.*, 2017). This idea serves as a guideline to define proper sources for the independent variables used in this work.

Porosity is one of the most important rock properties in reservoir engineering. It is defined as the ratio of pore spaces to the bulk volume of the rock. The ratio of the interconnected pores to the total volume is further classified as effective porosity and represents spaces containing unconstrained fluids. The difference between total porosity and effective porosity represents the unconnected pore spaces occupied by constrained water known as clay-bound water.

Because of the intermolecular interactions between liquids and the solid surface of the rock, part of the effective porosity is also consumed by immovable fluids called bound fluids. The remaining part is occupied by fluids that can indeed flow, which are known as free fluids. Figure 2.1 shows a schematic representation of a rock sample’s pore spaces and the fluids they contain.

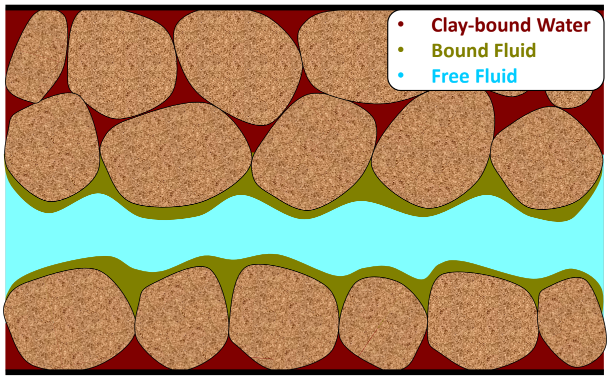

Figure 2.1: Classification of pore spaces and the fluids they contain: clay-bound water (brown), bound fluid (green), and free fluid (blue).

The bound fluid fraction is represented by ϕ_{BF}; the free fluid fraction is represented by ϕ_{FF}; total porosity is ϕ_{TOT}; and effective porosity is ϕ_{EF}. Note that $\phi_{EF} = \phi_{BF} + \phi_{FF}$. Permeability predictions will depend on the quantification of these spaces and fluid fractions, especially on the quantification of free fluid.

2.1.1.
Nuclear Magnetic Resonance

One of the many available tools for *in-situ* matrix porosity measurements is the NMR. The same principle involved in clinical imaging can be replicated to porous media evaluation. Initially, this technology was used by the industry for petrophysical laboratory research but soon was adapted into a downhole logging tool. Although it can be considered a useful resource for fluid property assessment, today, the NMR is best known for its capacity of measuring the size of pores and the fluids that they contain.

The NMR logging tool locally induces a constant magnetic field into the formation, followed by a perpendicular pulsed disturbance of radiofrequency energy. Each time one of these disturbances is ceased, the tool measures the response emitted by the spinning protons contained within the fluid-filled pore spaces. Signals are expressed in terms of the total time that the system takes to magnetize and then naturally demagnetize as the atomic nuclei return to their vertically aligned state. According to Bloch (1946), the vertical magnetization of the atomic nuclei is given by

$$M_z(t) = M_0\left(1 - e^{-\frac{t}{T_1}}\right), \tag{2.1}$$

where M_0 is the maximum magnetization provided by an induced constant field, t is the time that the protons are exposed to this magnetic field, and T_1 is the time it takes to reach approximately 63% of the maximum magnetization.

Between each disturbance, a receiver measures the response in the plane perpendicular to the direction to which protons were previously aligned, i.e., perpendicular to the direction of the constant magnetic field. This provides a distribution of an exponentially decaying signal known as the transverse relaxation time T_2. Both T_1 and T_2 can be interpreted for pore-size and fluid properties. However, total porosity can be directly derived from the amplitude response of the first disturbing pulse.

Assuming that mobility is proportional to pore size, bound fluid and free fluid can be obtained from the T_2 distribution. By defining a cutoff value above which fluids can flow, the distribution can be classified into movable and immovable fluids, as presented by Figure 2.2. An example of a typical NMR porosity log is presented in Figure 2.3.

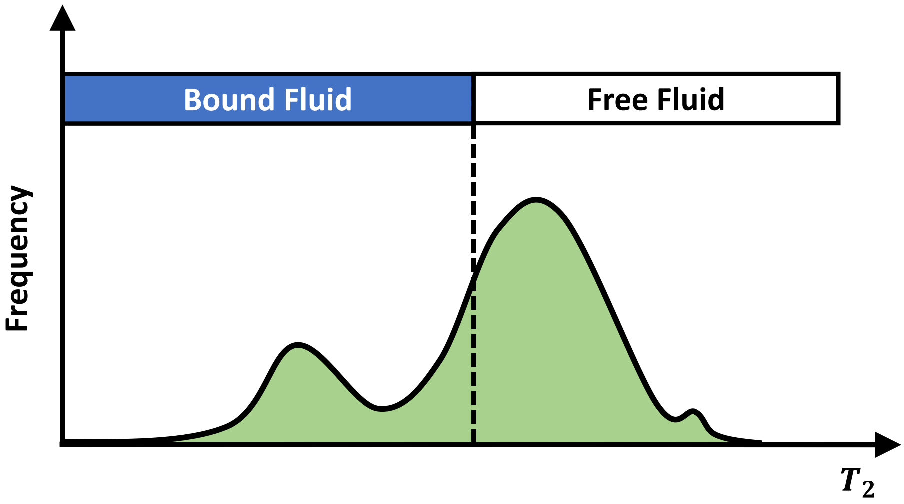

Figure 2.2: Distribution of the transverse relaxation time ($\boldsymbol{T_2}$) and the cutoff used to classify porosity into movable and immovable components.

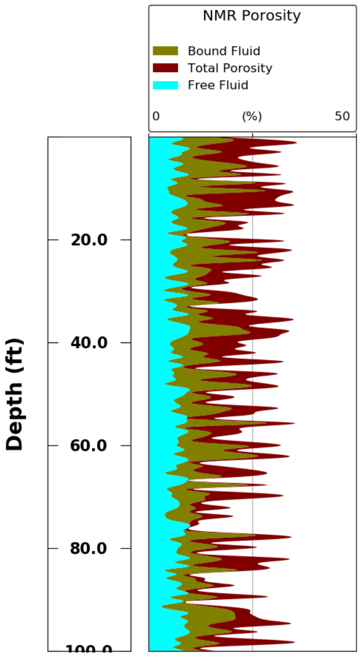

Figure 2.3: Example of a nuclear magnetic resonance porosity log.

2.1.2.
Ultrasonic Image

According to Rosa *et al*. (2006), secondary porosity is caused by geological processes that occur after rock sedimentation, such as tension, compression, or even chemical dissolution. These processes can produce caves, sinkholes, fractures, and other highly connected spaces with dimensions significantly larger than matrix porosity. Secondary porosities are responsible for the permeability spikes observed in carbonate drilled wells.

The NMR usually fails to thoroughly evaluate structures with larger porosity scales because of its limited depth of investigation and resolution. Because it operates by measuring the electromagnetic response of fluids' atomic nuclei, it can often misinterpret drilling artifacts as geological features. On the other hand, ultrasonic image logs are considered the ideal tools in this scenario, not only because of their high resolution but also because of their 360° coverage of the wellbore. According to Menger (1994), there is a strong relationship between the amplitudes measured by ultrasonic imaging and the pore arrangement within the reservoir. Figure 2.4 presents an example of an image acquired on the heterogeneous environment of a carbonate formation.

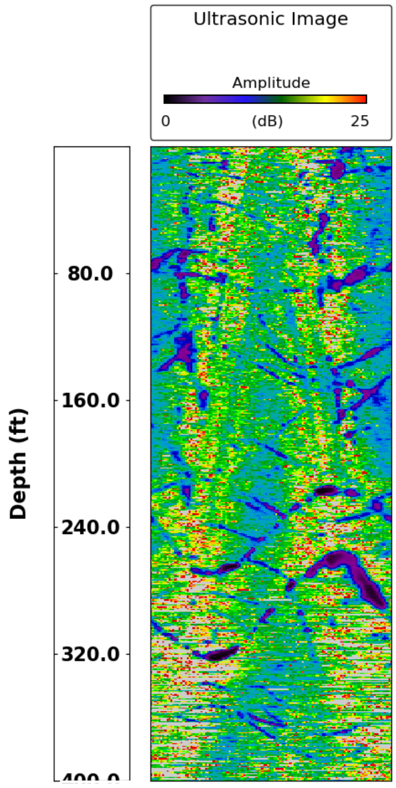

Figure 2.4: Ultrasonic image log in carbonate formation with indications of fractures and dissolutions.

The imaging logging tool operates by emitting pulsed acoustic energy inside the well and measuring both the amplitude and travel time of the reflected signal. Irregularities caused by secondary porosities tend to attenuate this reflected energy and therefore are easily detected in contrast with the smooth surface of the borehole wall. The same effect can be caused by induced irregularities such as breakouts and other deformities in wellbore geometry, which, unlike fractures, vugs, and cavernous structures, do not contribute to the system's overall permeability. The tool's 360° coverage can be considered one of its most valuable features because it allows log analysts to distinguish natural artifacts from the induced ones.

Ultrasonic imaging has a strong interpretive nature. Even its preprocessing stages require some expertise on the environment in which it has been acquired. Although many algorithms today are being developed to automize this process through intelligent methods, the role of the analyst has been hereby preserved. Therefore, in this work, image corrections and interpretations are performed by a specialist and are then used as inputs for the modeling processing.

In their work, Menezes de Jesus *et al.* (2017) use a segmentation technique to quantify the image attributes. By determining thresholds (usually two or three), the authors classify the amplitude distribution in terms of porosity sizes. As a result, the segmentation technique delivers discrete classification as opposed to continuous numerical amplitude values. Figure 2.5 presents an example of the segmentation technique using a single threshold value.

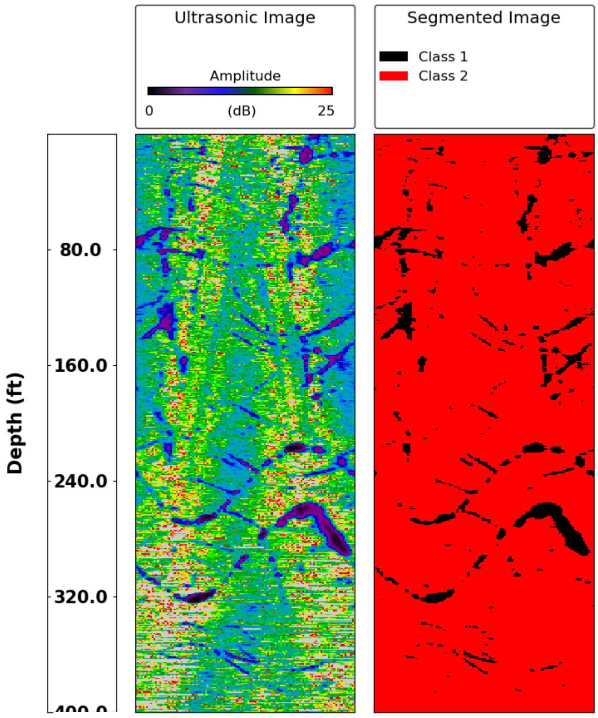

Figure 2.5: Ultrasonic image log and its segmented 2D image.

2.2.
Well Testing

The definition of a model’s output target is as critical as the definition of its geological input variables. Like any other reservoir property, permeability estimations are sensitive to the elementary volume of investigation. Well testing operations can measure an average permeability over a vast portion of the reservoir. Although its interpretation depends on narrowing down the possible solutions to an inverse problem of pressure-diffusion, well testing is known to be one of the most reliable tools for assessing reservoir’s dynamic conditions.

This is an on-site operation that consists of measuring and evaluating the change in pressure and temperature caused by a controlled disturbance in the state of equilibrium of a well’s production or injection. Preferably, flow rates are controlled and measured at the surface, while downhole gauges measure pressure and temperature data immediately above the reservoir.

2.2.1.
Pressure-Diffusion and the Transient-State

The pressure-diffusion equation for radial flow in porous media can be obtained by combining the principle of mass conservation with Darcy's law and the equations of state, and is given by

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial p}{\partial r}\right)=\frac{\phi\mu c_t}{k}\frac{\partial p}{\partial t}, \tag{2.2}$$

where ϕ, μ, k, and c_t represent the reservoir porosity, fluid viscosity, effective permeability, and fluid-saturated rock compressibility. Furthermore, this derivation is based on the assumptions that the porous medium is isotropic and homogeneous; that the flow is single-phase, isothermal, and strictly horizontal; gravity and capillary effects are negligible; fluids are slightly compressible; general properties are constant; and the wellbore is vertical and fully penetrates the reservoir.

By considering an undisturbed initial condition, $p(r,0)=p_i$; an infinite acting boundary condition, $p_{r\to\infty}=p_i$; and a line-source approximation of the well producing or injecting at a constant flow rate, q_w, the transient-state response can be derived from eq. (2.2) as

$$p(r,t)=p_i-C_2\frac{q_w\mu}{kh}\left[\frac{1}{2}E_i\left(\frac{\phi\mu c_t r^2}{4C_1 kt}\right)\right], \tag{2.3}$$

where h is the uniform formation thickness, C_1 and C_2 are unit conversion factors (see Appendix A), and E_i is the exponential integral function.

According to Abramowitz and Stegun (1964), for small values of the argument, the exponential integral function E_i can be approximated by the natural log function, as

$$E_i(-x)\cong\ln(\gamma x), \tag{2.4}$$

where $\gamma=e^{0.5722}=1{,}78108$. Thus, eq. (2.3) can be rewritten as

$$p_w(t) = p(r_w, t) = p_i - \frac{1}{2} C_2 \frac{q_w \mu}{kh} ln\left(\frac{4C_1 kt}{\gamma \phi \mu c_t r_w^2}\right), \tag{2.5}$$

where r_w is the radius of the wellbore.

2.2.2.
The Skin Factor

On a practical note, the reservoir previously depicted can suffer an impairment known as formation damage. It consists of an unintended restriction to the flow of fluids in the near-wellbore vicinity, inevitably caused by the very own existence of the well. Van Everdingen (1953) and Hurst (1953) proposed a mathematical model to explain this effect and ultimately incorporate it into the pressure-diffusion equations.

The physical concept consists of two radially concentric zones with uniform thickness and unequal permeabilities. The schematic model is presented in Figure 2.6.

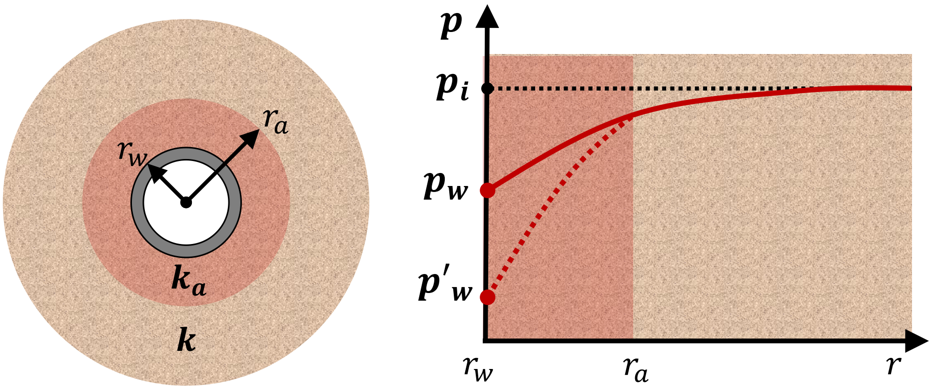

Figure 2.6: Illustration of the reservoir's skin-affected zone and pressure profile.

For a given elapsed time t_1, the pressure response at the interface of the two concentric regions can be derived from eq. (2.5), as

$$p(r_a, t_1) = p_a = p_i - \frac{1}{2} C_2 \frac{q_w \mu}{k_a h} ln\left(\frac{4C_1 k_a t}{\gamma \phi \mu c_t r_a^2}\right). \tag{2.6}$$

Likewise, the pressure response at the wellbore, for the same elapsed time, is given by

$$p'_w = p_i - \frac{1}{2} C_2 \frac{q_w \mu}{4\pi k_a h} ln\left(\frac{4C_1 k_a t}{\gamma \phi \mu c_t r_w^2}\right). \quad (2.7)$$

Subtracting eq. (2.7) from (2.6), we get

$$p_a - p'_w = C_2 \frac{q_w \mu}{k_a h} ln\left(\frac{r_a}{r_w}\right). \quad (2.8)$$

Considering the ideal scenario of an unaffected region in the vicinity of the wellbore, we have

$$p_a - p_w = C_2 \frac{q_w \mu}{kh} ln\left(\frac{r_a}{r_w}\right). \quad (2.9)$$

Subtracting eq. (2.9) from eq. (2.8), we get

$$\frac{1}{C_2} \frac{kh}{q_w \mu} \left(p_w - p'_w\right) = \left(\frac{k}{k_a} - 1\right) ln\left(\frac{r_a}{r_w}\right). \quad (2.10)$$

Eq. (2.10) was originally presented by Hawkins (1956) to show that the radius and permeability of the affected zone (*skin*) are related by a dimensionless pressure difference defined as skin factor. In other words,

$$s = \frac{1}{C_2} \frac{kh}{q_w \mu} \left(p_w - p'_w\right) = \left(\frac{k}{k_a} - 1\right) ln\left(\frac{r_a}{r_w}\right). \quad (2.11)$$

Note that if the skin's permeability is less than that of the formation, $s > 0$; otherwise, $s < 0$.

Van Everdingen (1953) and Hurst (1953) treated the case of positive skin as a zone of infinitesimal radius around the wellbore. This allowed both authors to incorporate skin to pressure-diffusion equations. For the case of the transient-state response, the pressure measured at the wellbore can be written as

$$p_w(t) = p_i - \frac{1}{2} C_2 \frac{q_w \mu}{kh} \left[ln\left(\frac{4C_1 kt}{\gamma \phi \mu c_t r_w^2}\right) + 2s\right]. \quad (2.12)$$

Hurst *et al.* (1969) later proved the applicability of this solution to the case of negative skin factors.

2.2.3.
Production Logging and Multilayered Reservoirs

Among other applications, the production logging tool (PLT) is used to diagnose production or injection problems and evaluate flow rate distribution in multilayered reservoirs. It is a wireline logging tool that comprises a variety of gauges and is usually requested during well testing operations. The tool's flowmeter is of particular interest for this work because it measures a fluid's axial flowing velocity inside the wellbore, which can easily be converted into flow rate.

The PLT's flowmeters are composed of spinners or turbines that rotate at a velocity compatible with the average speed of the fluid passing through it. When the tool runs through an interval where there is inflow from or into the reservoir, a deflection can be observed in the measured velocity, as shown in Figure 2.7.

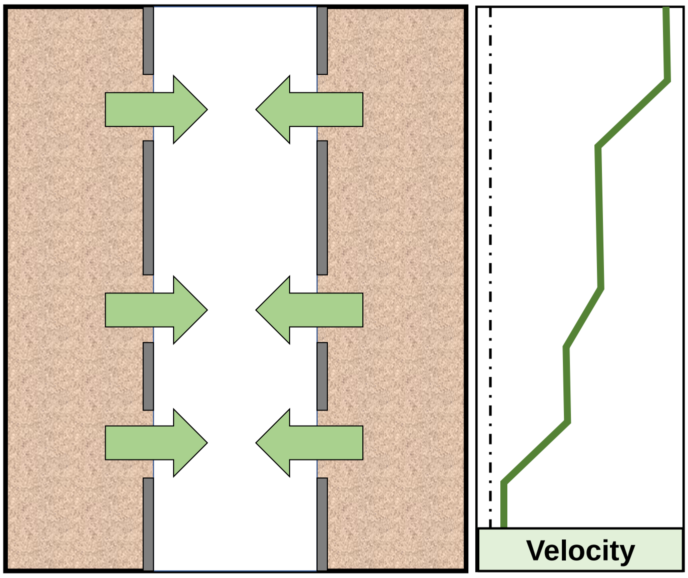

Figure 2.7: Production logging tool's flowmeter response.

Galvao and Guimaraes (2017) proposed a practical method to decouple permeability and skin from a production log's measured flow rates. Their approach considers a heterogeneous reservoir composed of several infinite acting, homogeneous, and isothermal layers with no formation crossflow; a vertically drilled well penetrating all the layers; and no friction loss inside the wellbore.

The key to the method's success lies in the operational recommendations provided by the authors. These recommendations allow the PLT to capture transient behaviors in layer flow rates, even when the system's total flow rate remains unchanged on the surface. Based on these considerations, eq. (2.12) can be written individually for any layer of a multilayered reservoir as

$$p_w^{(j)}(t) = p_i^{(j)} - \frac{1}{2} C_2 \frac{q_w^{(j)}(t)\,\mu}{k^{(j)} h^{(j)}} \left[ln\left(\frac{4C_1 k^{(j)} t}{\gamma \phi^{(j)} \mu c_t r_w^2} \right) + 2s^{(j)} \right], \tag{2.13}$$

where $j = [1,2,3,\dots,N]$ represents each layer. Note that $p_w^{(j)}$ and $p_i^{(j)}$ can be measured by any gauge, including the ones located in the PLT string; $q_w^{(j)}$ can be derived from the velocity measured during production logging; $\phi^{(j)}$ and $h^{(j)}$ are normally evaluated during borehole logging; and finally, μ and c_t are considered known properties for interpretation. It is important to point out that the relationship between layer flow rate and skin is not by any means negligible.

By combining layer flow rates and pressure transients at two distinct moments, the problem of determining individual skin factor and permeability is reduced to a couple of logarithmic equations with two unknown variables for each layer, i.e.,

$$p_w^{(j)}(t_1) = p_i^{(j)} - \frac{1}{2} C_2 \frac{q_w^{(j)}(t_1)\,\mu}{k^{(j)} h^{(j)}} \left[ln\left(\frac{4C_1 k^{(j)}\, t_1}{\gamma \phi^{(j)} \mu c_t r_w^2} \right) + 2s^{(j)} \right], \tag{2.14}$$

$$p_w^{(j)}(t_2) = p_i^{(j)} - \frac{1}{2} C_2 \frac{q_w^{(j)}(t_2)\,\mu}{k^{(j)} h^{(j)}} \left[ln\left(\frac{4C_1 k^{(j)}\, t_2}{\gamma \phi^{(j)} \mu c_t r_w^2} \right) + 2s^{(j)} \right]. \tag{2.15}$$

By adopting a few reasonable hypotheses, the method previously described converts the inverse problem of pressure-diffusion into a direct one and ultimately allows the computation of continuous permeability curves, free of skin effects, from production logging profiles.

2.3.
Data Structures

A standard layout has been adopted for the input and output datasets throughout this work to efficiently compare the performances of different predictive methods. Data structuring helps minimize the influence of the preprocessing stages of any implementation on the final results. Therefore, every algorithm presented hereafter, whether implemented from scratch or extracted from a commercial library, uses datasets that have undergone the same preprocessing procedures.

2.3.1.
Input Data

The input data, or the independent variables, consist of the geological measurements obtained from NMR and ultrasonic image logs. As mentioned before, this work does not attempt to optimize the interpretation of either one. Rather, it focuses on correlating its processed information. Therefore, every numerical computation carried out in the following chapters assume that procedures such as eccentricity corrections, interpolations, and other data preparation routines have been previously conducted.

Matrix porosity is represented by the following NMR-derived curves: free fluid (ϕ_{FF}), bound fluid (ϕ_{BF}), and total porosity (ϕ_{TOT}). All three measurements are naturally normalized because of the very property they represent, i.e., the ratio of volume occupied by a specific fluid to the bulk volume of the rock. These variables are arranged in an array-like data structure. This consists of a collection of values identified by an index, which corresponds to its specific depth of reference in the wellbore.

Ultrasonic image logs are manipulated to provide algorithms with data structures similar to the NMR ones. The idea is to simplify the entries fed to the models and ultimately reduce the impact of noise and other outliers on their performances. After the image segmentation procedure has been applied, the fraction curves of each class are extracted. This results in multiple array-like curves with complementary values that correspond each class’s fractions at a given depth. Figure 2.8 presents an example of a preprocessed image log, a two-class segmented version, and its corresponding fraction curves.

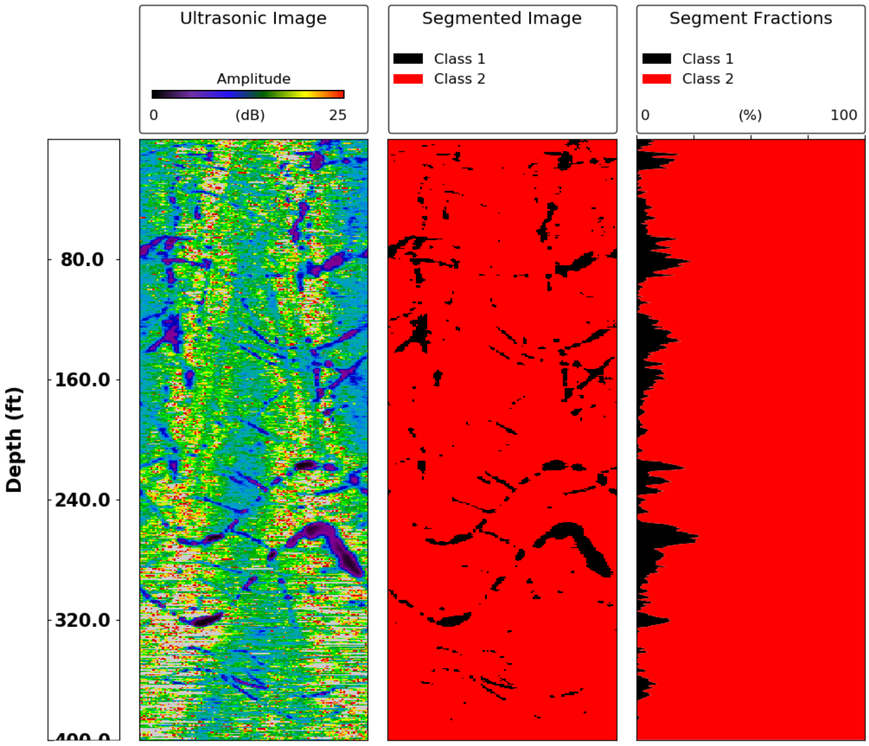

Figure 2.8: Ultrasonic image log, its segmentation, and fraction curves.

Indeed, working with one-dimensional arrays instead of the two-dimensional amplitude distribution simplifies the entire process. For example, the fraction curves presented in Figure 2.8 will feed our models with information on how much matrix and secondary porosity contributions are expected. However, on the other hand, this eliminates the possibility of correlating the geometry of the secondary porosity structure to the well's deliverability. In other words, fractures, dissolutions, and other cavernous structures are considered indistinguishable in the course of this work. This is, unarguably, a point for improvement in future studies.

Note that the ultrasonic image logging tool has a superior resolution compared to the NMR. Hence, the multiple fraction curves derived from an ultrasonic image also present higher resolution. To guarantee a point-to-point correspondence between matrix porosity and secondary porosity measurements, a cubic interpolation is carried out on the former.

2.3.2.
Target Output

The target output data, i.e., the dependent variables, consist of the effective permeability curves generated from production logs. Like with input features, the optimization of the method used to generate permeability from production logging data is out of the scope of this work. In fact, other available methods could be used (e.g., Stewart *et al.*, 1981; Ehlig-Economides and Joseph, 1987; Del Rey *et al.*, 2009) without affecting the overall methodology of this work. However, to preserve the idea of upscaling petrophysical predictions into well-testing dimensions, it is of utmost importance that the permeability target information be derived from production logs. The method proposed by Galvao and Guimaraes (2017) was adopted because it presents a practical approach for data acquisition without undermining the heterogeneous nature of skin factor distribution on carbonate formations.

The effective permeability curve is also arranged in an array-like data structure. The NMR porosity and fraction curves need to be cropped in order to match the permeability interval in size, which is normally more restrictive than those of borehole logs. Because production logging flow profiles present resolutions inferior to the ultrasonic image, it is also necessary to interpolate the permeability array to achieve point-to-point correspondence with the input features. Figure 2.9 presents the data preparation routine used in the following chapters.

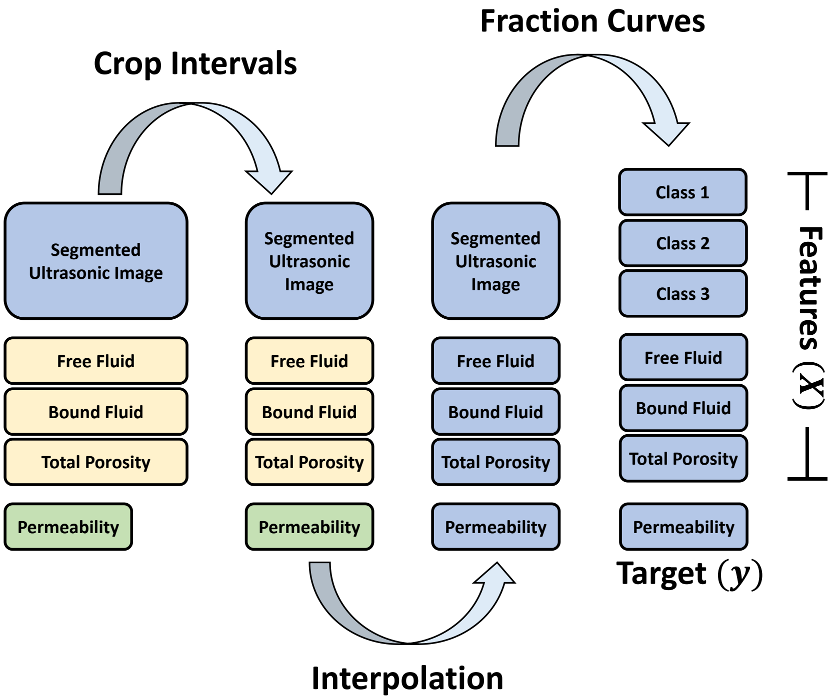

Figure 2.9: Data preparation routine: colors represent different data resolutions and the box dimensions represent the sizes of the intervals evaluated through each source.

3
Fundamentals on Supervised Learning Algorithms

This chapter gives an overview of important concepts of machine learning theory. Detailed descriptions of two supervised learning techniques are shown, namely the Support Vector Regressions and the Artificial Neural Networks – presenting the mathematical foundation for implementing each of their algorithms. At the end of this chapter, complete pseudocodes for each of the techniques are provided.

3.1.
Overview on Machine Learning Theory

One of the fundamental objectives in learning is to generalize observations to never-before-seen data. In other words, learning consists of finding similarities between examples and new problems. According to Schölkopf and Smola (2002), another important goal in learning is to place intuitive arguments into a mathematical framework.

Similarity can be defined as a measure of comparison between two elements in a vector space, to which a single number is assigned for evaluation. Among all possible measures, the inner product between two vectors is particularly appealing. Geometrically, it represents the projection of a vector onto another of unit length and is defined by

$$\langle \boldsymbol{x}, \boldsymbol{x}' \rangle = \|\boldsymbol{x}\| \, \|\boldsymbol{x}'\| \cos\theta, \tag{3.1}$$

where θ is the angle between $\boldsymbol{x}$ and $\boldsymbol{x}'$.

Inner products allow distances, directions, and lengths to be computed. Thus, almost any geometrical construction can be written in terms of this measure. Since an observation can be made up of any object, it is necessary first to represent it in a vector space, or more specifically, in an inner product space. Even if the observation already does exist in this domain, it may be convenient to change its representation to make it more compatible with the similarity measure. Therefore, the first step to learning is embedding input data into the measure's vector space, which will be conventionally named from now on as feature space $\mathcal{K}$.

A standard strategy in machine learning algorithms is to define a function $\varphi: \mathcal{X} \rightarrow \mathcal{K}$ that maps data from the input domain ($\mathcal{X} \in \mathbb{R}^m$) into a feature space of higher dimension ($\mathcal{K} \subset \mathbb{R}^{m+p}$). This can be very useful for solving nonlinear problems through linear solutions. A nonlinear decision boundary is defined in the original input domain for every linear definition in the higher-dimensional space, as shown in Figure 3.1.

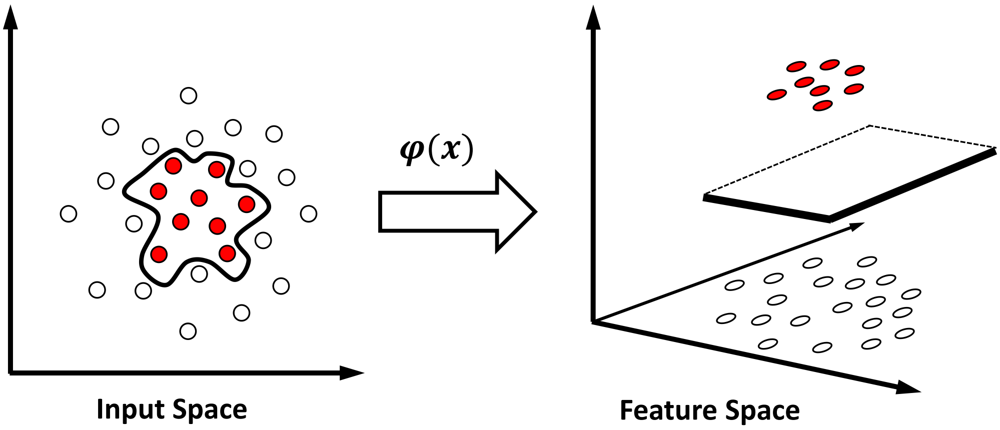

Figure 3.1: Mapping from input space into higher-dimension feature space for data classification.

3.1.1.
Supervised Learning

Machine learning problems can be classified according to the nature of the data available to solve them. The purpose of this work fits the *Supervised Learning* category because both input and output information are used. In essence, the objective of the algorithms presented herein is to produce an approximation of an unknown mapping function solely by analyzing the elements from its feature space and their correspondence in the codomain. According to Lyaqini *et al.* (2020), the relationship between supervised learning and inverse problems resides in the latter's reformulation into a quadratic optimization problem.

Supervised learning can be further categorized in terms of its outputs, i.e., regression or classification. What separates the two is that in the former, output targets are real-valued numbers rather than a binary representation of a class. Since this work aims to estimate a reservoir's permeability distribution, efforts will be focused on understanding regression problems in greater detail.

3.1.2.
Overfitting and Underfitting

An algorithm's generalization capacity is defined by its ability to perform on previously unseen data. Some techniques treat this issue implicitly (e.g., support vector machines) while others need to separate part of the dataset to evaluate it explicitly (e.g., artificial neural networks). Nonetheless, in both cases, the model's generalization capacity is sensitive to the degree of freedom carried by the mechanism used to treat nonlinearities.

Normally, predictive models are conceived and improved by minimizing the dissimilarities between their outputs and measured data. This minimization of error constitutes the basic principle for supervised learning solutions. However, the effectiveness of this procedure does not necessarily imply good generalizations. In other words, the objective of any technique should be an overall balance between its performance on new data and learning examples.

Figure 3.2 presents three alternative models used to explain the same dataset. The first one to the left shows neither a good fit to the training set nor a good generalization capacity (*underfitting*); the third alternative to the right performs well on the training set, however, is not expected to do so on new data (*overfitting*); the model shown in the middle appears to hold an even balance between error minimization and generalization performance.

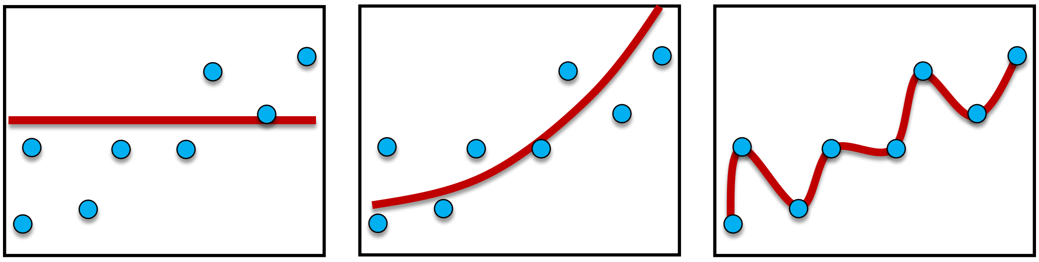

Figure 3.2: Illustration of underfitting and overfitting

3.2.
Support Vector Learning

Vapnik and Chervonenkis (1964) proposed an algorithm based on optimal hyperplanes to solve linearly separable problems, namely the Support Vector Machines (SVM). Boser *et al.* (1993) later extended this solution to nonlinear problems by adopting the concept of *kernel perceptron*, originally presented by Aizerman *et al.* (1964). The *kernel perceptron*, most commonly known as the *kernel trick*, expresses the inner product between two vectors in feature space $\mathcal{K}$ in terms of a function k, described as

$$k(\boldsymbol{x}, \boldsymbol{x}') = \langle \boldsymbol{\varphi}(\boldsymbol{x}), \boldsymbol{\varphi}(\boldsymbol{x}') \rangle. \tag{3.2}$$

The kernel trick allows the use of *Vapnik-Chervonenkis* solution on nonlinear problems without determining the mapping function φ, explicitly. A list of the most commonly used kernel functions is presented in Appendix C.1.

The generalization of SVM to regression problems is known as SVR. It will be shown in this chapter that this technique accepts a certain amount of error in the construction of an optimal-hyperplane-based model. The SVR's threshold towards errors makes it an appropriate tool for permeability estimations in the megascopic scale, given that in many cases, the relevance in predictions lies in its variance and not in the value itself.

3.1.1.
Optimal Hyperplane

A hyperplane is the generalization of the two-dimensional line or the three-dimensional subsurface plane into a higher-dimensional space. Vapnik and Chervonenkis (1991) proposed the existence of a unique subsurface that maximizes the margin of separation between itself and any training point for classification problems. Schölkopf and Smola (2002) presented theoretical arguments supporting hyperplane's generalization performance. It can be mathematically expressed in terms of an inner product as

$$\langle \boldsymbol{w}, \boldsymbol{x_k} \rangle + b = 0, \tag{3.3}$$

where $\boldsymbol{w}$ and $\boldsymbol{x_k} \in \mathcal{K},\ b \in \mathbb{R},\ \boldsymbol{w}$ represents a normal vector to the subsurface, and b is a constant offset. Naturally, any point in feature space that does not belong to the optimal hyperplane can be written as

$$|\langle \boldsymbol{w}, \boldsymbol{x_k} \rangle + b| > 0. \tag{3.4}$$

Let $\boldsymbol{\varphi}(\boldsymbol{x_i})$ be the representation of $\boldsymbol{x_i}$ ($\boldsymbol{x_i} \in \mathcal{X}$) in feature space. For a set of training points $\{\boldsymbol{x_i},\ i = 1,2,3, \dots, n\}$, where n represents the total number of points, $\boldsymbol{w}$ and b can be rescaled so that the ones closest to the hyperplane are written as

$$|\langle \boldsymbol{w}, \boldsymbol{\varphi}(\boldsymbol{x_i}) \rangle + b| = 1. \tag{3.5}$$

Without loss of generality, two points from the subsurfaces that arise from eq. (3.5) can be written as

$$\langle \boldsymbol{w}, \boldsymbol{\varphi}(\boldsymbol{x_1}) \rangle + b = 1 \tag{3.6}$$

and

$$\langle \boldsymbol{w}, \boldsymbol{\varphi}(\boldsymbol{x_2}) \rangle + b = -1. \tag{3.7}$$

Subtracting eq. (3.7) from eq. (3.6), we obtain

$$\langle \boldsymbol{w}, \boldsymbol{\varphi}(\boldsymbol{x_1}) - \boldsymbol{\varphi}(\boldsymbol{x_2}) \rangle = 2. \tag{3.8}$$

Finally, dividing eq. (3.8) by $||\boldsymbol{w}||$, we get

$$\langle \frac{\boldsymbol{w}}{||\boldsymbol{w}||}, \varphi(\boldsymbol{x_1}) - \varphi(\boldsymbol{x_2}) \rangle = \frac{2}{||\boldsymbol{w}||}. \tag{3.9}$$

From eq. (3.9), the distance (margin) between the closest point and the optimal hyperplane equals $1/||\boldsymbol{w}||$. Therefore, to maximize the margin, $||\boldsymbol{w}||$ needs to be minimized. This understanding is important to set up the constrained optimization problem of the SVM or SVR.

3.1.2.

Constrained Optimization Problem (COP)

Vapnik (1995) presented the use of an ε-insensitive loss function to build an optimal hyperplane for regression. He proposed an algorithm that balances complexity and an ε tolerance towards errors. The complexity of the algorithm is associated with the maximum margin mentioned before. Thus, the constrained optimization problem (COP) proposed by Vapnik (1995) can be written, in its primal form, as

$$\underset{\boldsymbol{w}}{\text{minimize}}\ f = \frac{1}{2}\|\boldsymbol{w}\|^2 \tag{3.10}$$

$$\text{subject to}\ \ |y_i - \langle \boldsymbol{w}, \boldsymbol{\varphi}(\boldsymbol{x_i})\rangle - b| \le \varepsilon,\ \ \text{for } i = 1,2,\dots,n, \tag{3.11}$$

where $y_i \in \mathbb{R}$ and n represents the total number of training points.

The subtlety of this formulation lies in its objective function. While simple linear regression seeks to minimize error, Vapnik's formulation proposes the minimization of complexity. In this case, the error is incorporated into the COP's constraint. Note that this allows the generalization issue to be implicitly incorporated into problem.

When dealing with measured data, solutions may not exist. Indeed, a perfect fit is unlikely, and violations of the constraint are bound to occur. Therefore, another hyperparameter is introduced to serve the points that fall outside the margin, as shown in Figure 3.3.

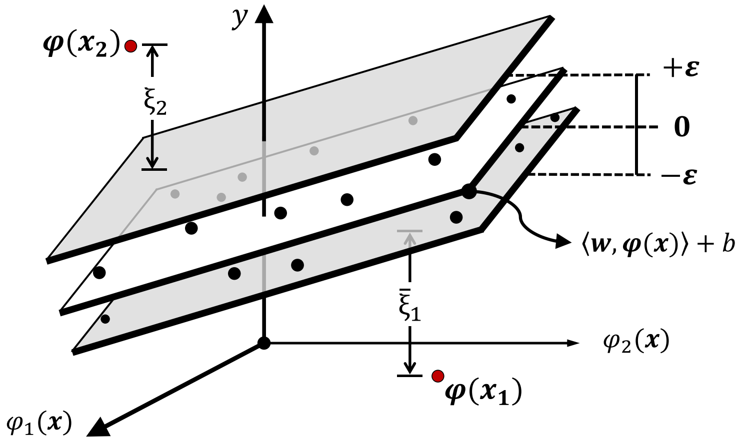

Figure 3.3: Classical illustration of SVR's conceptual formulation

The real-valued slack variables ξ_i and $\bar{\xi}_i$ represent the deviations of targets higher and lower than the hyperplane, respectively. The COP, also known as the *primal problem*, can now be rewritten as

$$\underset{\boldsymbol{w}}{\text{minimize}}\ f = \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{n}\left(\xi_i + \bar{\xi}_i\right) \tag{3.12}$$

$$\text{subject to}\quad y_i - \langle \boldsymbol{w}, \boldsymbol{\varphi}(\boldsymbol{x_i})\rangle - b \le \varepsilon + \xi_i, \tag{3.13}$$

$$\langle \boldsymbol{w}, \boldsymbol{\varphi}(\boldsymbol{x_i})\rangle + b - y_i \le \varepsilon + \bar{\xi}_i, \tag{3.14}$$

$$\xi_i, \bar{\xi}_i \ge 0, \tag{3.15}$$

where C represents a regularization parameter responsible for determining the trade-off between complexity and error. It is important to point out that errors within threshold ε imply slack variables equal to zero.

At this point, it is convenient to introduce the Method of Lagrange Multipliers. Lagrange multipliers are variables that represent the objective function's sensitivity to constraints. The Lagrangian function $\mathcal{L}$ is defined as

$$\begin{aligned}
&\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\alpha}, \boldsymbol{\bar{\alpha}}, \boldsymbol{\mu}, \boldsymbol{\bar{\mu}}) = \\
&\quad = \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{n}\left(\xi_i + \bar{\xi}_i\right) \\
&\quad - \sum_{i=1}^{n}\alpha_i\left(\varepsilon + \xi_i + \langle \boldsymbol{w}, \varphi(\boldsymbol{x_i})\rangle + b - y_i\right) \\
&\quad - \sum_{i=1}^{n}\bar{\alpha}_i\left(\varepsilon + \bar{\xi}_i + y_i - \langle \boldsymbol{w}, \varphi(\boldsymbol{x_i})\rangle - b\right) \\
&\quad - \sum_{i=1}^{n}\left(\mu_i\xi_i + \bar{\mu}_i\bar{\xi}_i\right),
\end{aligned} \tag{3.16}$$

where $\alpha_i, \bar{\alpha}_i, \mu_i$, and $\bar{\mu}_i$ are the multipliers associated with each of the constraints.

Vanderplaat (1984) demonstrated that eq. (3.16) has a saddle point defined at $\mathcal{L}(\boldsymbol{w}^*, b^*, \boldsymbol{\alpha}^*, \boldsymbol{\bar{\alpha}}^*, \boldsymbol{\mu}^*, \boldsymbol{\bar{\mu}}^*)$, which consists of the minimum of the Lagrangian function with respect to the parameters that define the hyperplane ($\boldsymbol{w}$, b, $\boldsymbol{\xi}$, and $\boldsymbol{\bar{\xi}}$), and the maximum with respect to the multipliers ($\boldsymbol{\alpha}, \bar{\boldsymbol{\alpha}}, \boldsymbol{\mu}$, and $\bar{\boldsymbol{\mu}}$). If it were possible to know the Lagrangian multipliers that correspond to this saddle point in advance, the COP would be reduced to a single unconstrained optimization problem. This approach can be appealing because, in many cases, finding the optimum multipliers is, indeed, a much simpler problem to solve.

By considering the *Karush-Kuhn-Tucker* (*KKT*) first-order complementary conditions (see Appendix B), a new COP, known as the *dual problem*, can be formulated as

$$\begin{aligned} \underset{\alpha_i,\bar{\alpha}_i,\mu_i,\bar{\mu}_i}{\text{maximize}}\ \mathcal{L} = & \\ &= \frac{1}{2}\|\boldsymbol{w}^*\|^2 + C\sum_{i=1}^{n}\left(\xi_i + \bar{\xi}_i\right) \\ &- \sum_{i=1}^{n}\alpha_i\left(\varepsilon + \xi_i + \langle \boldsymbol{w}^*, \varphi(\boldsymbol{x_i})\rangle + b^* - y_i\right) \\ &- \sum_{i=1}^{n}\bar{\alpha}_i\left(\varepsilon + \bar{\xi}_i + y_i - \langle \boldsymbol{w}^*, \varphi(\boldsymbol{x_i})\rangle - b^*\right) \\ &- \sum_{i=1}^{n}\left(\mu_i\xi_i + \bar{\mu}_i\bar{\xi}_i\right) \end{aligned} \tag{3.17}$$

$$\text{subject to} \quad \alpha_i, \bar{\alpha}_i, \mu_i, \bar{\mu}_i \geq 0. \tag{3.18}$$

Considering the saddle point condition of $\mathcal{L}$, the following equations can be derived from eq. (3.16):

$$\nabla_{\boldsymbol{w}}\mathcal{L}\left(\boldsymbol{w}^*, b^*, \boldsymbol{\xi}^*, \bar{\boldsymbol{\xi}}^*\right) = \boldsymbol{w}^* - \sum_{i=1}^{n}(\bar{\alpha}_i - \alpha_i)\,\varphi(\boldsymbol{x_i}) = 0 \tag{3.19}$$

$$\nabla_{b}\mathcal{L}\left(\boldsymbol{w}^*, b^*, \boldsymbol{\xi}^*, \bar{\boldsymbol{\xi}}^*\right) = \sum_{i=1}^{n}(\alpha_i - \bar{\alpha}_i) = 0 \tag{3.20}$$

$$\nabla_{\xi_i}\mathcal{L}\left(\boldsymbol{w}^*, b^*, \xi_i^*, \bar{\boldsymbol{\xi}}_i^*\right) = n\,C - \alpha_i - \mu_i = 0 \tag{3.21}$$

From eq. (3.19), $\|\boldsymbol{w}^*\|^2$ can be expressed in terms of the multipliers as

$$\|\boldsymbol{w}^*\|^2 = \langle \boldsymbol{w}^*, \boldsymbol{w}^* \rangle = \sum_{i,j=1}^{n} (\bar{\alpha}_i - \alpha_i)\,(\bar{\alpha}_j - \alpha_j)\langle \varphi(\boldsymbol{x_i}), \varphi(\boldsymbol{x_j})\rangle. \tag{3.22}$$

The substitution of eq. (3.20), (3.21), and (3.22) in eq. (3.17) and (3.18) allows the *dual problem* to be rewritten as

$$\begin{aligned} \underset{\boldsymbol{\alpha_i},\bar{\alpha}_i}{\text{maximize}}\ \mathcal{L} = &-\frac{1}{2}\sum_{i,j=1}^{n} (\bar{\alpha}_i - \alpha_i)\,(\bar{\alpha}_j - \alpha_j)\langle \boldsymbol{\varphi}(\boldsymbol{x_i}), \boldsymbol{\varphi}(\boldsymbol{x_j})\rangle \\ &- \varepsilon\sum_{i=1}^{n}(\bar{\alpha}_i + \alpha_i) + \sum_{i=1}^{n} y_i(\bar{\alpha}_i - \alpha_i), \end{aligned} \tag{3.23}$$

$$\text{subject to} \quad \sum_{i=1}^{n}(\alpha_i - \bar{\alpha}_i) = 0 \text{ and } \alpha_i, \bar{\alpha}_i \in [0, C], \tag{3.24}$$

where $\boldsymbol{\alpha}$ and $\bar{\boldsymbol{\alpha}}$ are the problem's support vectors. Note that $\mathcal{L}$ can be expressed without knowing the mapping function φ, simply by substituting the term $\langle \boldsymbol{\varphi}(\boldsymbol{x_i}), \boldsymbol{\varphi}(\boldsymbol{x_j})\rangle$ by the kernel function $k(\boldsymbol{x_i}, \boldsymbol{x_j})$ presented in eq. (3.2).

By combining eq. (3.19) with the equation of the hyperplane, the linear regression estimates in input space $\mathcal{X}$, can be expressed as

$$y(\boldsymbol{x}) = b + \sum_{i=1}^{n}(\bar{\alpha}_i - \alpha_i)\langle \boldsymbol{\varphi}(\boldsymbol{x_i}), \boldsymbol{\varphi}(\boldsymbol{x})\rangle. \tag{3.25}$$

Once again, the inner product $\langle \boldsymbol{\varphi}(\boldsymbol{x_i}), \boldsymbol{\varphi}(\boldsymbol{x})\rangle$ can be substituted by the kernel function $k(\boldsymbol{x_i}, \boldsymbol{x})$. Additionally, the offset b can be computed by considering eq. (3.13) or (3.14) as equalities. This is only possible when the corresponding slack variables ξ_i or $\bar{\xi}_i$ are zero. Eq. (3.25) can be rewritten as

$$y(\boldsymbol{x}) - y_p = \sum_{i=1}^{n} (\bar{\alpha}_i - \alpha_i)\left[k(\boldsymbol{x_i}, \boldsymbol{x}) - k\left(\boldsymbol{x_i}, \boldsymbol{x_p}\right)\right], \qquad \text{for } \xi_p = 0. \tag{3.26}$$

3.1.2.
Dual Problem Solution

Both dual and primal versions of the COP are called Quadratic Programming Problems (QP) because the objective functions are quadratic, and the constraints are linear. According to Bottou and Lin (2007), the solution to the dual formulation is computationally more efficient not only because the constraints are simpler but also because they match the variables in number. Generically, QPs are defined by

$$\underset{\boldsymbol{x}}{\text{minimize}} \;\; q = \frac{1}{2}\boldsymbol{x^t Q x} + \boldsymbol{f^t x} \tag{3.27}$$

$$\text{subject to} \quad \boldsymbol{A^t x} = \boldsymbol{a}, \tag{3.28}$$

$$\boldsymbol{B^t x} \geq \boldsymbol{b}, \tag{3.29}$$

where $\boldsymbol{Q}$ is a quadratic positive semidefinite matrix.

By defining two new variables $\beta_i = (\bar{\alpha}_i - \alpha_i)$ and $\lambda_i = (\bar{\alpha}_i + \alpha_i)$, and introducing the kernel function $k\left(\boldsymbol{x_i}, \boldsymbol{x_j}\right) = \langle \boldsymbol{\varphi}(\boldsymbol{x_i}), \boldsymbol{\varphi}(\boldsymbol{x})\rangle$, the dual formulation can be rewritten as

$$\underset{\boldsymbol{\beta_i}}{\text{maximize}} \;\; \mathcal{L} = -\frac{1}{2}\sum_{i,j=1}^{n} \beta_i \beta_j k(\boldsymbol{x_i}, \boldsymbol{x_j}) - \varepsilon \sum_{i=1}^{n} \lambda_i + \sum_{i=1}^{n} y_i \beta_i \tag{3.30}$$

$$\text{subject to} \;\; \sum_{i=1}^{n} \beta_i = 0 \text{ and } |\beta_i| \leq C. \tag{3.31}$$

The *KKT* complementary condition states that the product between a Lagrange multiplier and its constraint is equal to zero for the optimum solution (see eq. (B.9)). Thus, for the constraints presented in eq. (3.13) and (3.14), it can be written that

$$\alpha_i^*(\varepsilon + \xi_i + \langle \boldsymbol{w}^*, \varphi(\boldsymbol{x_i})\rangle + b^* - y_i) = 0, \tag{3.32}$$

$$\bar{\alpha}_i^*\left(\varepsilon + \bar{\xi}_i + y_i - \langle \boldsymbol{w}^*, \varphi(\boldsymbol{x_i})\rangle - b^*\right) = 0. \tag{3.33}$$

From this set of equations, it can be inferred that α_i and $\bar{\alpha}_i$ can never be simultaneously nonzero. Additionally, with the *KKT* positivity condition (see B.7), it can be deduced that

$$\lambda_i = -\beta_i, \qquad \text{for } \beta_i < 0, \tag{3.34}$$

and

$$\lambda_i = \beta_i, \qquad \text{for } \beta_i > 0. \tag{3.35}$$

In other words, the classical quadratic form of description of the SVR constrained optimization problem is given by

$$\underset{\boldsymbol{\beta_i}}{\text{minimize}}\ -\mathcal{L} = \frac{1}{2}\boldsymbol{\beta}^t\boldsymbol{K}\boldsymbol{\beta} - \boldsymbol{y}^t\boldsymbol{\beta} + \varepsilon\sum_{i=1}^{n}|\beta_i| \tag{3.36}$$

$$\text{subject to } \ (\boldsymbol{j})^t\boldsymbol{\beta} = 0 \text{ and } -\mathrm{C} \le \beta_i \le C. \tag{3.37}$$

where $\boldsymbol{j}$ is an all-ones vector with the same dimension as $\boldsymbol{\beta}$ and $K_{i,j} = k(\boldsymbol{x_i}, \boldsymbol{x_j})$ is a quadratic positive semidefinite matrix.

The optimal solution can be reached through direction searches. This consists of slightly moving $\boldsymbol{\beta}$ along a direction $\boldsymbol{d} = (d_1, d_2, \dots, d_n)$ without violating the constraints. For each multi-dimensional direction, the following one-dimensional problem must be solved:

$$\underset{\boldsymbol{\lambda}}{\text{minimize}}\ -\mathcal{L}(\boldsymbol{\beta^i}), \qquad \text{for } \boldsymbol{\beta^i} = \boldsymbol{\beta^{i-1}} + \lambda.\boldsymbol{d^{i-1}} \tag{3.38}$$

Although there are many different types of multi-dimensional search algorithms available, gradient based methods are considered to be the most suitable for QPs, especially when associated with conjugate directions. Directions $\boldsymbol{d^i}$ and $\boldsymbol{d^j}$ are considered conjugate, if and only if

$$(\boldsymbol{d^i})^t \boldsymbol{K} \boldsymbol{d^j} = 0, \qquad \text{for } i \neq j. \tag{3.39}$$

Hestenes and Stiefel (1952) demonstrated that, for conjugate directions**,** the solution to the one-dimensional problem presented in eq. (3.38) is given by

$$\lambda^i = \frac{-(\boldsymbol{g^i})^t \boldsymbol{d^i}}{(\boldsymbol{d^i})^t \boldsymbol{K} \boldsymbol{d^i}}, \tag{3.40}$$

where $\boldsymbol{g^i}$ is the gradient of function $\mathcal{L}$ at a given point $\boldsymbol{\beta^i}$. The gradient can be computed by

$$\boldsymbol{g^i} = \boldsymbol{K}\boldsymbol{\beta^i} - \boldsymbol{y} + \varepsilon . \boldsymbol{sgn}(\boldsymbol{\beta^i}), \tag{3.41}$$

where $\boldsymbol{sgn}(\boldsymbol{\beta^i})$ is a vector-valued signum function that extracts the sign of the elements in $\boldsymbol{\beta^i}$. Additionally, search direction $\boldsymbol{d^i}$ can be recursively determined by projecting the gradient $\boldsymbol{g^i}$ to the subspace defined by eq. (3.39). Thus, $\boldsymbol{d^i}$ can be calculated as

$$\boldsymbol{d^i} = -\boldsymbol{g^i} + \frac{-(\boldsymbol{g^i})^t \boldsymbol{K} \boldsymbol{d^{i-1}}}{(\boldsymbol{d^{i-1}})^t \boldsymbol{K} \boldsymbol{d^{i-1}}} \boldsymbol{d^{i-1}}. \tag{3.42}$$

The first search direction $\boldsymbol{d^0}$ must be considered as $\boldsymbol{d^0} = -\boldsymbol{g^0}$.

Algorithm 3.1 presents the main steps to solving the unconstrained QP using conjugate gradient search.

Algorithm 3.1**:** Conjugate Gradient Search Algorithms

```
$\boldsymbol{x} \to input$;  $\boldsymbol{y} \to target$;  $\varepsilon \to tolerance$
define matrix $\boldsymbol{K}$, where $K_{ij} = \langle \boldsymbol{\varphi}(\boldsymbol{x_i}), \boldsymbol{\varphi}(\boldsymbol{x_j}) \rangle$

$\boldsymbol{\beta} = \boldsymbol{0}$;  $\boldsymbol{g} = \boldsymbol{K\beta} - \boldsymbol{y}$;  $\boldsymbol{d} = -\boldsymbol{g}$

loop 1: Until $\boldsymbol{g} = \boldsymbol{0}$
    $\lambda \leftarrow (\boldsymbol{g^t d})/(\boldsymbol{d^t K d})$
    $\boldsymbol{\beta} \leftarrow \boldsymbol{\beta} - \lambda.\boldsymbol{d}$
    $\boldsymbol{g} \leftarrow \boldsymbol{K\beta} - \boldsymbol{y} + \varepsilon.\boldsymbol{sgn}(\boldsymbol{\beta})$
    $\boldsymbol{d} \leftarrow -\boldsymbol{g} + \boldsymbol{d}.(-\boldsymbol{g^t K d})/(\boldsymbol{d^t K d})$
end loop 1
```

To solve a support vector regression, it is inevitable to consider the problem's constraints. Fortunately, the equality condition presented by eq. (3.37) can be easily met simply by projecting $\boldsymbol{d^i}$ to any generic vector $\boldsymbol{u}$ that belongs to the domain defined by $\boldsymbol{j}^t\boldsymbol{u} = 0$, i.e.,

$$\boldsymbol{d} \leftarrow \frac{\langle \boldsymbol{u}, \boldsymbol{d} \rangle}{\langle \boldsymbol{u}, \boldsymbol{u} \rangle}\boldsymbol{u}, \tag{3.43}$$

Inequality constraints, on the other hand, significantly increase the algorithm's complexity. The condition presented in eq. (3.37) is known as a box constraint and is usually dealt with by inserting the gradient projection step within an iterative loop that removes points activating it. According to Bottou and Lin (2007), algorithms spend most of their computing time searching for points that violate the box constraint. A suggestion on how to deal with the box constraint can be found in Algorithm 3.2.

Figure 3.4 illustrates the geometry of a hypothetical problem with only two support vectors α_1 and α_2. Note that the two-dimensional search is reduced to the single projection in the equality constraint domain in this particular case.

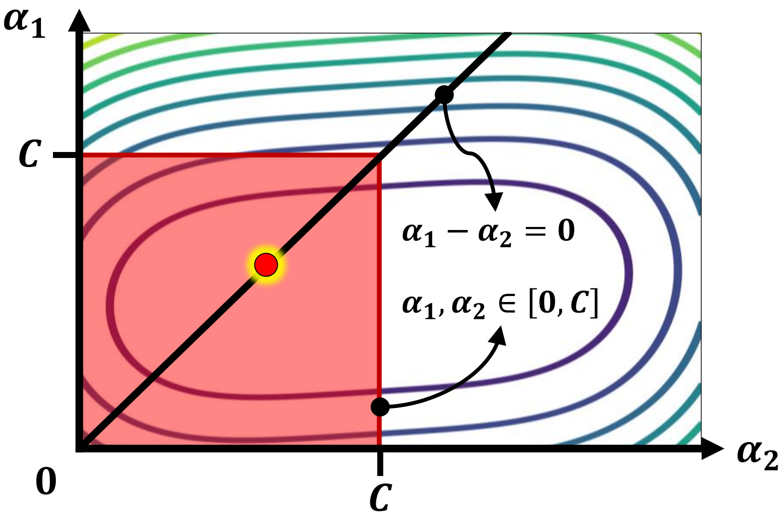

Figure 3.4: Illustration of the COP geometry in its Dual Form.

3.3.
Artificial Neural Network

Artificial Neural Networks (ANNs) can be considered the basis of the modern-day Machine Learning discipline. They represent a breakthrough for research in recent years. These computational models outperform by far other classical algorithms due to their capacity of generalizing functions to a high volume of data.

Like every other machine learning algorithm, the essence of this technique lies in its ability to extend linear solutions to nonlinear problems. As described in the previous section, support vector regressions overcome nonlinearities by using kernel functions to express the inner product between vectors in feature space $\mathcal{K}$. Although this strategy provides satisfactory generalizations, it implicitly results in very generic mapping functions $\varphi: \mathcal{X} \rightarrow \mathcal{K}$.

By relying on the computers' processing power today, artificial neural networks propose to learn these mapping functions, from generic to more specific ones. This gives the algorithm the flexibility to solve advanced physical problems. Hence, φ is parametrized to be eventually optimized with the other sets of parameters that perform linear regression $(\boldsymbol{w}, b)$. The resulting outcome of a neural network can be expressed as a function of these parameters, i.e.,

$$y(\boldsymbol{x}) = f(\boldsymbol{w}, b, \varphi). \tag{3.44}$$

3.3.1.
Network Design

Also known as Multilayer Perceptrons (MLPs), the ANNs are composed of interconnected computational nodes called neurons. These neurons work in a distributed manner to learn from the input to optimize the output (O'Shea and Nash, 2015). Ultimately, this intelligent model represented by eq. (3.44) can also be defined by the following surjective multifunction composition:

$$f(\boldsymbol{w}, b, \varphi) = \boldsymbol{f_n} \circ \boldsymbol{f_{n-1}} \circ \dots \boldsymbol{f_2} \circ \boldsymbol{f_1}(\boldsymbol{x}), \tag{3.45}$$

where $\boldsymbol{f_i}$ represent vector-valued functions that perform individual calculations on a group of neurons called hidden layer.

The term "hidden" stems from the fact that the outputs $\boldsymbol{y_i}$ of each of these functions $\boldsymbol{f_i}$ are not shown during the modeling process. The number of neurons that compose a single hidden layer determines its width (l_i), as the total number of hidden layers determines the model's depth (n). Figure 3.5 presents a generic illustration of a neural network design.

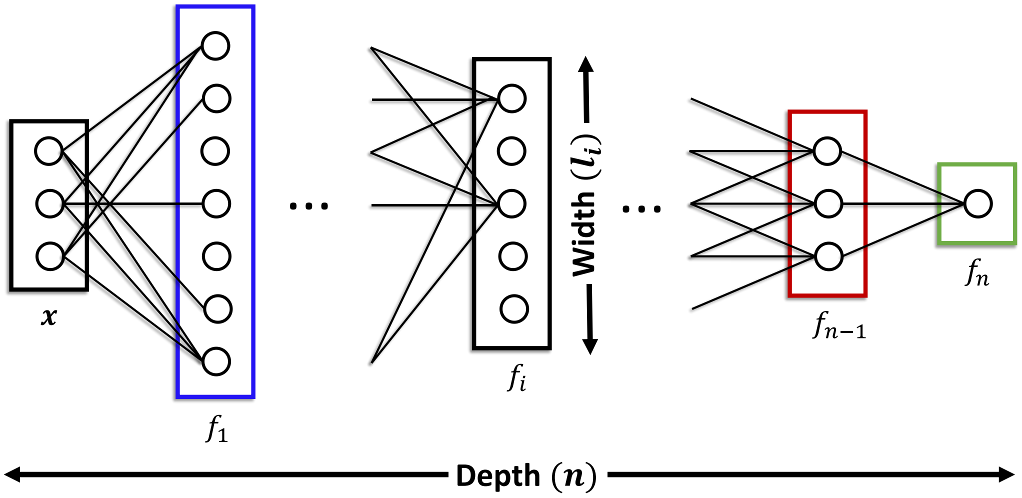

Figure 3.5: Illustration of a neural network design.

The nonlinear parametrization can be decoupled from $\boldsymbol{f_i}$ in the form of activation functions $\boldsymbol{\varphi_i}$. By intercalating the two, a nonlinear transformation is possible.

$$f(\boldsymbol{x}) = (\boldsymbol{\varphi_n} \circ \boldsymbol{f_n} \dots \boldsymbol{\varphi_2} \circ \boldsymbol{f_2} \circ \boldsymbol{\varphi_1} \circ \boldsymbol{f_1})(\boldsymbol{x}). \tag{3.46}$$

According to Goodfellow *et al.* (2016), a common principle in computer science is to build complicated systems from minimal components. Therefore, it is recommended to adopt nearly linear activation functions (e.g., rectified linear units) to build universal approximators. This strategy preserves the simplicity of linear optimization in the development of more complex solutions. Appendix C.2 presents the most used activation functions in ANN architectures.

3.3.2.
Forward Propagation

MLPs are known as feedforward models because the input information $\boldsymbol{x}$ is propagated throughout its structure until an output $\boldsymbol{y}$ is reached. This propagation is driven by linear regressions followed by nonlinear activations performed at each neuron. Figure 3.6 presents all the elements responsible for this propagation in the generic i^{th} layer of the network. For simplicity, the input of this layer has been represented by $\boldsymbol{x}$ and the output by $\boldsymbol{y}$.

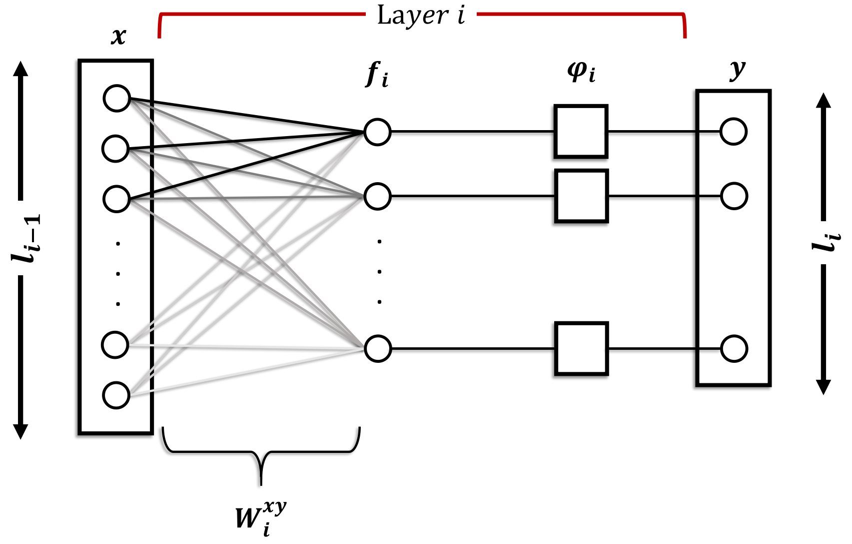

Figure 3.6: Elements of the $\boldsymbol{i^{th}}$ layer.

Note that the layer shown in Figure 3.6 contains a total of l_i neurons, while the one immediately before contains l_{i-1}. It is convenient to define this layer's set of weights as

$$W_i^{xy} = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1l_i} \\ w_{21} & w_{22} & \cdots & w_{2l_i} \\ \vdots & \vdots & \ddots & \vdots \\ w_{l_{i-1}1} & w_{l_{i-1}2} & \cdots & w_{l_{i-1}l_i} \end{bmatrix}_{l_{i-1}\times l_i} . \quad (3.47)$$

The linear and nonlinear vector-valued functions are respectively given by

$$\boldsymbol{f_i}(W_i, \boldsymbol{x}, \boldsymbol{b_i}) = (W_i)^t \boldsymbol{x} + \boldsymbol{b_i}, \quad (3.48)$$

$$\boldsymbol{y} = \boldsymbol{\varphi_i}\big(\boldsymbol{f_i}(W_i, \boldsymbol{x}, \boldsymbol{b_i})\big), \quad (3.49)$$

where $\boldsymbol{b_i}$ represents a vector-valued parameter known as *bias*. Eq. (3.48) correlates to the optimal hyperplane described in the previous section.

3.3.3.
Backpropagation

After the forward propagation has run its course, it is necessary to propagate the output back through the network. Backpropagation begins with calculating a cost function $C(\boldsymbol{w}, b, \boldsymbol{\varphi})$ and aims to minimize it by adjusting the network's parameters (Rumelhart *et al.*, 1986). This is accomplished by determining the gradients of C with respect to each element encountered along the process. The only difference between the SVR's gradient-based search and backpropagation is that in the latter, directions are determined sequentially by the network arrangement.

The gradient of the cost function with respect to each of the variables of $\boldsymbol{f_i}$ can be derived from eq. (3.48) by considering the chain rule of calculus, and are given by

$$\boldsymbol{\nabla_x}(C) = W_i . \boldsymbol{\nabla}_{\boldsymbol{f_i}}(C), \quad (3.50)$$

$$\boldsymbol{\nabla}_{W_i}(C) = \boldsymbol{x} . \left(\boldsymbol{\nabla}_{\boldsymbol{f_i}}(C)\right)^t . \quad (3.51)$$

$$\boldsymbol{\nabla}_{\boldsymbol{b_i}}(C) = I . \left(\boldsymbol{\nabla}_{\boldsymbol{f_i}}(C)\right)^t . \quad (3.52)$$

By repeating this process with eq. (3.49), the gradient of C with respect to $\boldsymbol{f_i}$ can be written as

$$\boldsymbol{\nabla}_{\boldsymbol{f_i}}(C) = \left(\mathrm{J}_{\varphi_i}\right)^t . \boldsymbol{\nabla}_{\boldsymbol{y}}(C), \tag{3.53}$$

where J_{φ_i} is the Jacobian matrix of $\boldsymbol{\varphi_i}$ at $\boldsymbol{f}(W_i, \boldsymbol{x}, \boldsymbol{b_i})$. Given that $\boldsymbol{\varphi_i}$ has a bijective correspondence to $\boldsymbol{f_i}$, its Jacobian matrix is considered diagonal and, therefore, is symmetric. Thus, J_{φ_i} can be written as

$$\left(\mathrm{J}_{\varphi_i}\right)^t = \mathrm{J}_{\varphi_i} = \begin{bmatrix} \frac{\partial \boldsymbol{\varphi_1}}{\partial \boldsymbol{f_1}} & 0 & \cdots & 0 \\ 0 & \frac{\partial \boldsymbol{\varphi_2}}{\partial \boldsymbol{f_2}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{\partial \boldsymbol{\varphi_{l_i}}}{\partial \boldsymbol{f_{l_i}}} \end{bmatrix}_{l_i \times l_i} . \tag{3.54}$$

Combining eq. (3.54), (3.53), and (3.50), and then left-multiplying the result by $\mathrm{J}_{\varphi_{i-1}}$, we get

$$\mathrm{J}_{\varphi_{i-1}} \boldsymbol{\nabla}_{\boldsymbol{x}}(C) = \mathrm{J}_{\varphi_{i-1}} W_i . \mathrm{J}_{\varphi_i} . \boldsymbol{\nabla}_{\boldsymbol{y}}(C). \tag{3.55}$$

Thus, it is convenient to define a recursive variable as

$$\boldsymbol{\delta_i} = \mathrm{J}_{\varphi_i} . \boldsymbol{\nabla}_{\boldsymbol{y}}(C). \tag{3.56}$$

Since every input is another layer's output, except for the first and last elements, eq. (3.55) can be rewritten as

$$\delta_{i-1} = \mathrm{J}_{\varphi_{i-1}} . W_i . \delta_i . \tag{3.57}$$

The substitution of eq. (3.53) to eq. (3.51) yields the gradient of the cost function with respect to the layer's weights in terms of $\boldsymbol{\delta_i}$ and is given by

$$\boldsymbol{\nabla}_{W_i}(C) = \boldsymbol{x}.\left(\left(\mathrm{J}_{\varphi_i}\right)^t.\boldsymbol{\nabla}_{\boldsymbol{y}}(C)\right)^t = \boldsymbol{x}.\boldsymbol{\delta}_{\boldsymbol{i}}^{\boldsymbol{t}}\,. \tag{3.58}$$

Generically, this gradient can be written for any layer as

$$\boldsymbol{\nabla}_{W_i}(C) = \boldsymbol{y}_{\boldsymbol{i-1}}.\boldsymbol{\delta}_{\boldsymbol{i}}^{\boldsymbol{t}}\,. \tag{3.59}$$

Following the same idea, the gradient of C with respect to $\boldsymbol{b_i}$ presented in eq. (3.52) can be written as

$$\boldsymbol{\nabla}_{\boldsymbol{b}_i}(C) = I.\left(\left(\mathrm{J}_{\varphi_i}\right)^t.\boldsymbol{\nabla}_{\boldsymbol{y}}(C)\right)^t = \boldsymbol{\delta}_{\boldsymbol{i}}^{\boldsymbol{t}}\,. \tag{3.60}$$

To determine the initial condition of the recursive formulation, it is necessary to understand the network's output. Figure 3.7 presents an illustration of the last layer of a single output ANN.

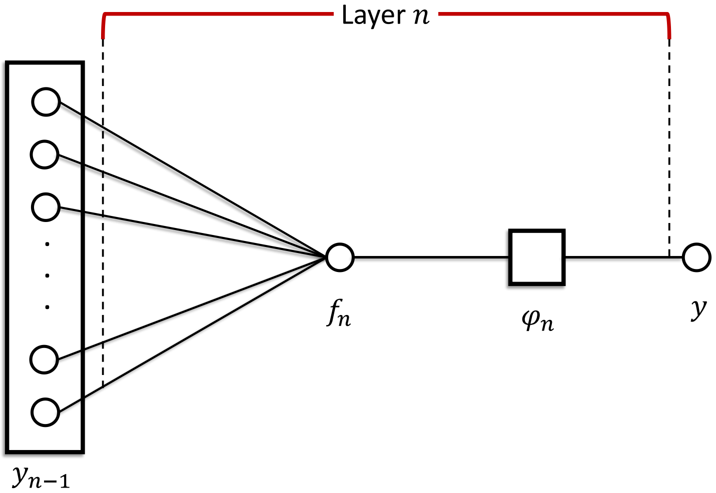

Figure 3.7: Elements of the $\boldsymbol{n^{th}}$ layer.

The initial condition can be expressed, in terms of the recursive variable, as

$$\boldsymbol{\delta}_{\boldsymbol{n}} = \mathrm{J}_{\varphi_n}.\boldsymbol{\nabla}_{\boldsymbol{y}}(C) = \mathrm{J}_{\varphi_n}C', \tag{3.61}$$

where C' is the gradient of the cost function with respect to $\boldsymbol{y}$. This can be determined analytically, depending only on the definition of the cost function C. As an example, considering the cost function as half of the square error of the output, we get

$$C' = \boldsymbol{\nabla}_{\boldsymbol{y}} \left[\frac{1}{2} (\widehat{\boldsymbol{y}} - \boldsymbol{y})^2 \right] = \boldsymbol{y} - \widehat{\boldsymbol{y}}, \tag{3.62}$$

where $\widehat{\boldsymbol{y}}$ is the measured target output.

To update W_i and $\boldsymbol{b_i}$, a learning rate λ must be chosen. Unlike SVRs, backpropagation does not rely on an exact line search algorithm to determine the value of λ, so this must be done deliberately. Low learning rates imply computationally expensive optimizations while large ones may not converge. As backpropagation regresses through the layers, parameters are updated continuously by

$$W_i \leftarrow W_i - \lambda \boldsymbol{\nabla}_{W_i}(C), \tag{3.63}$$

$$\boldsymbol{b_i} \leftarrow \boldsymbol{b_i} - \lambda \boldsymbol{\nabla}_{\boldsymbol{b_i}}(C). \tag{3.64}$$

Finally, when the input layer is reached, the forward propagation is recommenced. A single cycle of forward and backward propagation is called *epoch.* In practice, epochs are performed on batches of data rather than the entire set.

3.3.4.
Cross-Validation

The parametrization of the mapping function provides the ANNs with the flexibility necessary to approximate advanced nonlinear correlations. In return, overfitting becomes a real concern. Unlike SVRs, the minimization of the model's complexity is not intrinsic to the formulation of the optimization problem. In this case, a procedure called Cross-validation (CV) is used to evaluate the model's generalization potential.

The idea behind this procedure is to separate a part of the labeled data to test the performance of a model that must have been trained on the remaining set. Ultimately, overfitting and underfitting are avoided by comparing the progression of the cost function with each epoch for both sets. Figure 3.8 presents an example of this assessment, indicating the optimal number of epochs.

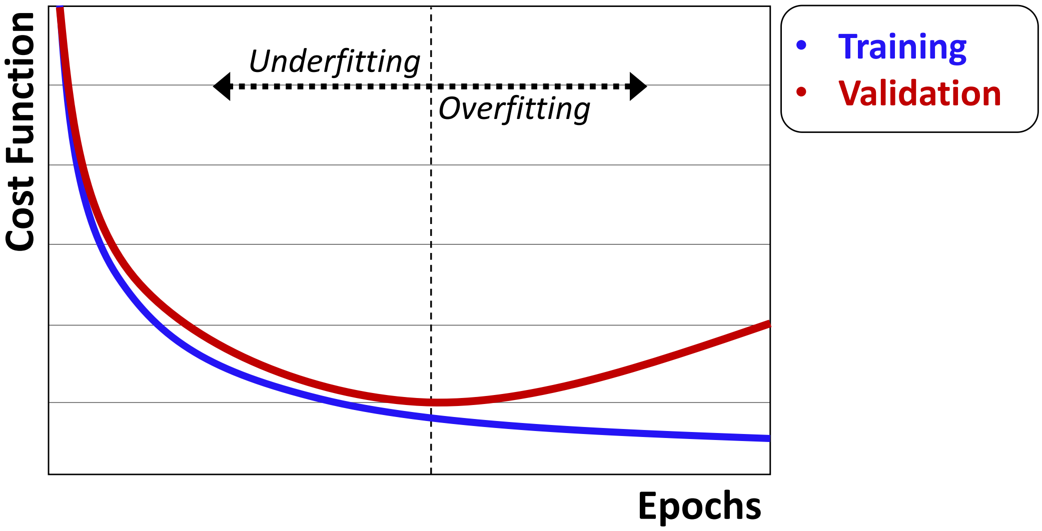

Figure 3.8: Illustration of cross-validation analysis.

The ANNs designed in this work adopt the K-Fold Cross-Validation (CV) method for training. It consists of dividing the data into K subsets and applying the cross-validation procedure a K number of times. In each time, a different subset is used for validation while the other $(K-1)$ sets are used for training. Performance is evaluated as an average over all the results. This CV method significantly reduces the risk of choosing inappropriate or biased training sets.

3.4.
Pseudocodes

The pseudocodes for implementing the SVR and the ANN are presented in Algorithm 3.2 and Algorithm 3.3, respectively.

Algorithm 3.2**:** Pseudocode for support vector regression .

```
$\boldsymbol{x} \rightarrow input;$  $\boldsymbol{y} \rightarrow target;$  $\varepsilon \rightarrow tolerance;$
$C \rightarrow Regularization\ Parameter$
define matrix $\boldsymbol{K},$ where $K_{ij} = \langle \boldsymbol{\varphi}(\boldsymbol{x_i}), \boldsymbol{\varphi}(\boldsymbol{x_j}) \rangle$

$\boldsymbol{\beta} = \boldsymbol{0};$
$B = \{1,2, \dots, N\}$

loop 1: Until $\boldsymbol{\beta}$ is unchanged
    $\boldsymbol{g} \leftarrow \boldsymbol{K\beta} - \boldsymbol{y} + \varepsilon\, \boldsymbol{sgn}(\boldsymbol{\beta})$

    loop 2: Until $B$ is unchanged
        $\boldsymbol{d} \leftarrow -\boldsymbol{g} + \boldsymbol{d}(-\boldsymbol{g}^{\mathbf{t}}\boldsymbol{Kd})/(\boldsymbol{d}^{\mathbf{t}}\boldsymbol{Kd})$
        $\boldsymbol{d} \leftarrow \boldsymbol{u}(\boldsymbol{u}^{\mathbf{t}}\boldsymbol{d})/(\boldsymbol{u}^{\mathbf{t}}\boldsymbol{u})$ **,** where $\boldsymbol{j}^{\mathbf{t}}\boldsymbol{u} = 0$
        $d_k = 0 \ \ (k \notin B)$
        $B \leftarrow B - \{k \in B \mid (\beta_k = C$ and $d_k > 0)$ or $(\beta_k = -C$ and $d_k < 0)\ \}$
    end loop 2

    $\lambda \leftarrow (\boldsymbol{g}^{\mathbf{t}}\boldsymbol{d})/(\boldsymbol{d}^{\mathbf{t}}\boldsymbol{Kd})$
    $\boldsymbol{\beta} \leftarrow \boldsymbol{\beta} - \lambda\boldsymbol{d}$
end loop 1

choose: $x_p \in \boldsymbol{x}$ and $y_p \in \boldsymbol{y} \ \mid \beta_p = 0$
$y(\hat{x}) = y_p + \sum_{i=1}^{N} \beta_i \left[ \langle \boldsymbol{\varphi}(\boldsymbol{x_i}), \boldsymbol{\varphi}(\hat{\boldsymbol{x}}) \rangle - \langle \varphi(x_i), \varphi(x_p) \rangle \right]$
```

Algorithm 3.3**:** Pseudocode for the artificial neural network.

```
$\hat{\boldsymbol{x}} \rightarrow input$;  $\hat{\boldsymbol{y}} \rightarrow target$;

$W = \{\boldsymbol{W_1}, \boldsymbol{W_2}, \dots, \boldsymbol{W_N}\}$;    $b = \{\boldsymbol{b_1}, \boldsymbol{b_2}, \dots, \boldsymbol{b_N}\}$

loop 1
        $\boldsymbol{y} = \hat{\boldsymbol{x}}$
        loop 2: for $i = 1$ to $N$
            $\boldsymbol{f_i} = (W_i)^t \boldsymbol{y} + \boldsymbol{b_i}$
            $\boldsymbol{\varphi_i} = \text{activation}(\boldsymbol{f_i})$
            $\boldsymbol{y} = \boldsymbol{\varphi_i}$
        end loop 2

        $E = \text{cost}(\boldsymbol{y}, \hat{\boldsymbol{y}})$
        $\boldsymbol{dE} = \boldsymbol{\nabla}_{\mathbf{y}} E$
        Perform Cross-Validation

        $\boldsymbol{\delta} = \varphi_N \circ \boldsymbol{dE}$
        loop 3: for $i = N$ down to 1
            $\boldsymbol{\nabla W_i} = (\boldsymbol{f_i})(\boldsymbol{\delta})$
            $\boldsymbol{W_i} \leftarrow \boldsymbol{W_i} - \lambda . \boldsymbol{\nabla W_i}$
            $\boldsymbol{b_i} \leftarrow \boldsymbol{b_i} - \lambda . \boldsymbol{\delta}$
            $\boldsymbol{\delta} \leftarrow \boldsymbol{\varphi_{i-1}} \circ (\boldsymbol{W_i})(\boldsymbol{\delta})$
        end loop 3

end loop 1
```

4
Performance of the Algorithms on Synthetic Data

In this chapter, different synthetic examples are designed to test the implementations of the pseudocodes presented in Chapter 3. Performances are evaluated based on their levels of accuracy, generalization capacity, number of iterations, and the time required to train. Additionally, the algorithms are confronted by the results of a free software machine learning library for the Python programming language — *Scikit-learn* (Pedregosa *et al*., 2011).

First, the methodology used to generate geological features synthetically is introduced, and important considerations on the procedure are provided. Subsequently, random correlations between porosity and permeability are used to generate target outputs. This chapter's ultimate objective is to show that artificial neural networks and support vector regressions can be considered suitable tools for modeling and predicting, especially when the physics is not fully understood *a priori*.

4.1.
Generating Synthetic Instances

According to Mitchell (1997), machine learning is the study of computer algorithms that thrive on data-driven experiences. Based on his definition, it is reasonable to consider dataset instances as an indispensable part of the field. Yet, privacy concerns and other data protection issues can become real obstacles for the methodological development of machine learning models. High-quality synthetic data can be leveraged to facilitate this process.

Normally, to manufacture dataset instances, the procedure must be carefully selected. It must comprise a large enough variability to capture the approximate behavior of the real system. However, in this case, the main purpose of artificial data synthesis and augmentation is to test the algorithms' implementation and evaluate their performances. Therefore, greater efforts have been deployed to accurately reproduce the synthetic ultrasonic image feature, which requires more strenuous work than simply generating data from probability density functions (PDFs).

4.1.1.
Ultrasonic Image Logs

To synthetically generate the high-resolution ultrasonic image logs, a multiple-point geostatistics method called Locality Sensitive Hashing Simulation (LSHSIM) has been applied. In their work, Moura *et al.* (2017) use LSHSIM to register all possible patterns of an input image in a hash table. This enables the fast retrieval of similar image patterns, quickly allowing several different realizations to be created from one single training image.

An ultrasonic high-resolution (0.508 × 0.543 cm) image log with 180 amplitude measurements by 100 feet of depth has been used as a seed to create different realizations. This single seed has been carefully selected to provide the LSHSIM method with enough variability to reproduce real geological features. Figure 4.1 presents an example of five different realizations generated from this single seed.

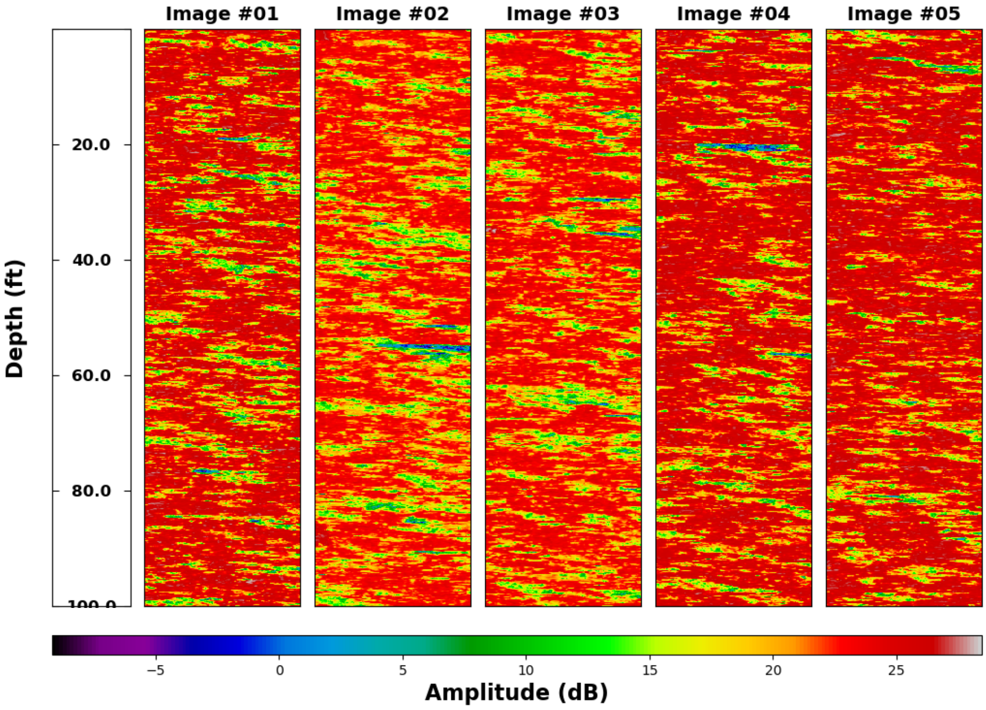

Figure 4.1: Synthetic image logs generated from LSHSIM.

After creating one hundred different images, segmentation has been applied using the same amplitude criterium for all instances. For the present purpose, simple thresholding has provided satisfactory results. In image processing, this is considered the simplest form of segmentation, where single real values are assigned to amplitude ranges. Figure 4.2 presents the results of this procedure on the same images shown in Figure 4.1.

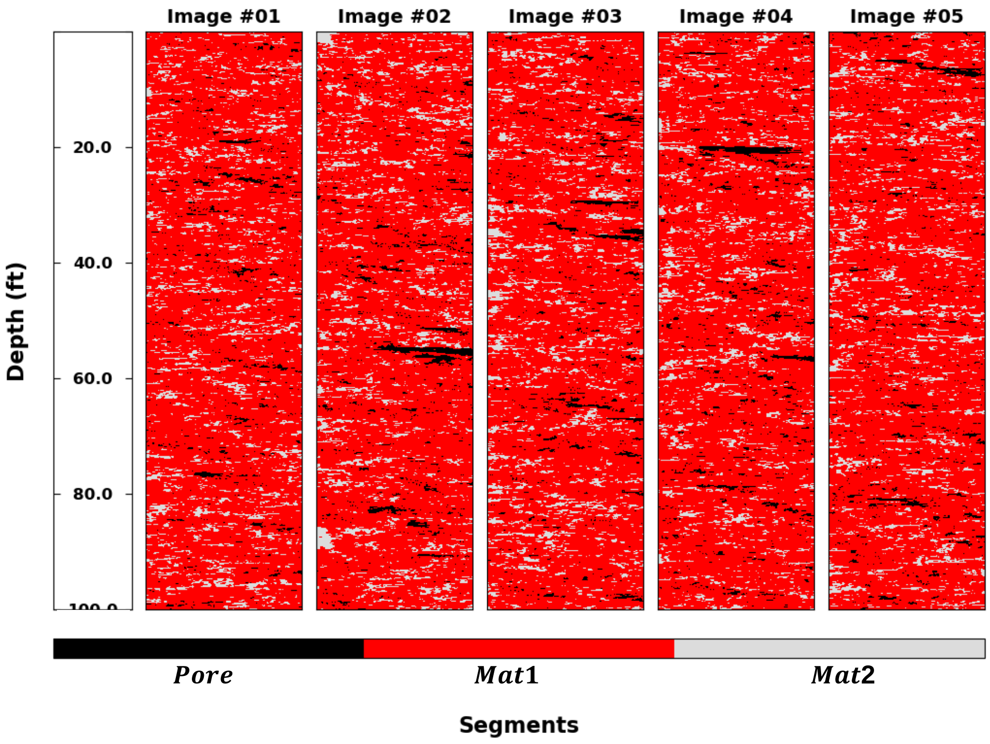

Figure 4.2: Synthetic segmented image logs.

Note that instances have been divided into three classes. The *Pore* class contains the lowest amplitude signals and represents structures with pore sizes greater than image resolution (e.g., fractures and vugs). *Mat1* and *Mat2* represent high and low matrix permeability, respectively.

4.1.2.
Nuclear Magnetic Resonance Logs

The synthetic NMR porosity measurements need to be consistent with their corresponding ultrasonic image logs. For example, rock dissolution and other karstified structures cause drilling fluid to invade the porous media. The subsequent alteration in hydrogen concentration is expected to affect the NMR total porosity and bound fluid quantifications.

Therefore, random samples of porosities have been drawn from a normal distribution where mean values (μ) and standard deviations (σ) are chosen according to the segments extracted from the image logs. Hence, before manufacturing the synthetic NMR data, fraction curves for each image class have been computed using the following equations:

$$P = \frac{Pore}{Pore + Mat1 + Mat2}, \tag{4.1}$$

$$M1 = \frac{Mat1}{Pore + Mat1 + Mat2}, \tag{4.2}$$

$$M2 = \frac{Mat2}{Pore + Mat1 + Mat2}. \tag{4.3}$$

Table 4.1 through Table 4.3 present the PDF parameters that are used to generate artificial NMR according to the extracted segments. These parameters have been chosen according to the author’s experience with carbonate formations. Nonetheless, they do not interfere with the result as long as consistently applied. Additionally, the natural order of the synthetic measurements must be respected: $\phi_{FF} + \phi_{BF} \leq \phi_{TOT} \leq 1$.

Table 4.1: PDF Parameters for synthetic NMR: $\boldsymbol{P > M_1, M_2}$.

Porosity	$\boldsymbol{\mu}$	$\boldsymbol{\sigma}$
ϕ_{FF}	0.03	0.03
ϕ_{BF}	0.5	0.1
ϕ_{TOT}	0.5	0.1

Table 4.2: PDF Parameters for synthetic NMR: $\boldsymbol{M_1} > \boldsymbol{P}, \boldsymbol{M_2}$.

Porosity	$\boldsymbol{\mu}$	$\boldsymbol{\sigma}$
ϕ_{FF}	0.14	0.08
ϕ_{BF}	0.14	0.08
ϕ_{TOT}	0.15	0.10

Table 4.3: PDF Parameters for synthetic NMR: $\boldsymbol{M_2} > \boldsymbol{P}, \boldsymbol{M_1}$.

Porosity	$\boldsymbol{\mu}$	$\boldsymbol{\sigma}$
ϕ_{FF}	0.03	0.03
ϕ_{BF}	0.03	0.02
ϕ_{TOT}	0.05	0.05

The difference between the ultrasonic image log resolution and the NMR resolution has been accounted for. Cubic interpolation is performed on the latter to guarantee that features have the same dimension for the modeling process.

Different case studies have been performed using random realizations of these manufactured geological features. Each study uses a different methodology to generate its permeability target outputs.

4.2.
Performance Evaluation Metrics

Before presenting the synthetic case studies, it is important to define the performance evaluation metrics. Case Study 1 is evaluated in terms of the root mean-squared error (RMSE) and the R-squared metrics. The former is a measure of the predictions' average deviation from the target output values. It is the square root of the mean squared variance of residuals and is computed by

$$RMSE(\boldsymbol{y}, \widehat{\boldsymbol{y}}) = \sqrt{\frac{\sum(y_i - \hat{y}_i)^2}{n}}, \tag{4.4}$$

where n is the total number of observations, and $\boldsymbol{y}$ and $\widehat{\boldsymbol{y}}$ represent model and target output, respectively.

R-squared, on the other hand, measures the correlation between the dependent and independent variables. It indicates how much of the output variance is explained by the regression model. For example, R-squared equal to 100% means that all the variability is explained by the regression. Conversely, negative values mean that the null hypothesis, i.e., the mean observed value, outperforms the resulting regression model. In this work, the R-squared metric is applied to the area under the permeability curve, to provide a predictability indicator of the system's total flow capacity. R-squared is defined as

$$R^2 = 1 - \frac{\sum\left(S_i - \hat{S}_i\right)^2}{\sum(S_i - \bar{S})^2}, \tag{4.5}$$

where $\bar{S}$ is the area under the mean observed value of the target output curve. $\boldsymbol{S}$ and $\widehat{\boldsymbol{S}}$ represent the areas under the model and target output curves, respectively.

For case study 2, the adjusted R-squared metric is introduced. This metric enables the assessment of the correlation's reliability for different amounts of features. The adjusted R-squared metric is defined as

$$Adjusted\ R^2 = 1 - \frac{(1 - R^2)(n - 1)}{n - k - 1}, \tag{4.6}$$

where k is the number of independent variables and n is number of observations.

This correlation evaluates exactly how much the input features are contributing to the model's overall performance. For example, if the adjusted R-squared correlations remain the same or even decreases as the number of variables increases, it means unnecessary data is being used to perform regression. In the worst-case scenario, this can result in inaccurate performance assessments.

4.3.
Synthetic Case Study 1

The purpose of this example is to show that ANN models and SVRs are capable of approximating non-linear multivariate functions. In their work, Menezes de Jesus *et al.* (2017) used an equation with a Timur-Coates structure to correlate porosity to permeability and generate a predictive model from parameter tuning. A similar equation has been used in this example to generate permeability outputs and is given by

$$k = A \left(\frac{\phi_{FF}}{\phi_{BF}}\right)^{a_1} \phi_{TOT}^{a_2}\, M_1 + B \left(\frac{\phi_{FF}}{\phi_{BF}}\right)^{b_1} \phi_{TOT}^{b_2}\, M_2 + CP\,, \tag{4.7}$$

where typical tuning parameters for carbonate reservoirs have been used $(A = 15000\, mD;\ a_1 = 1; a_2 = 2; B = 11350\, mD;\ b_1 = 2; b_2 = 3; C = 20000\, mD)$. Figure 4.3 presents the graphic representation of the geological features and target output (instance #3).

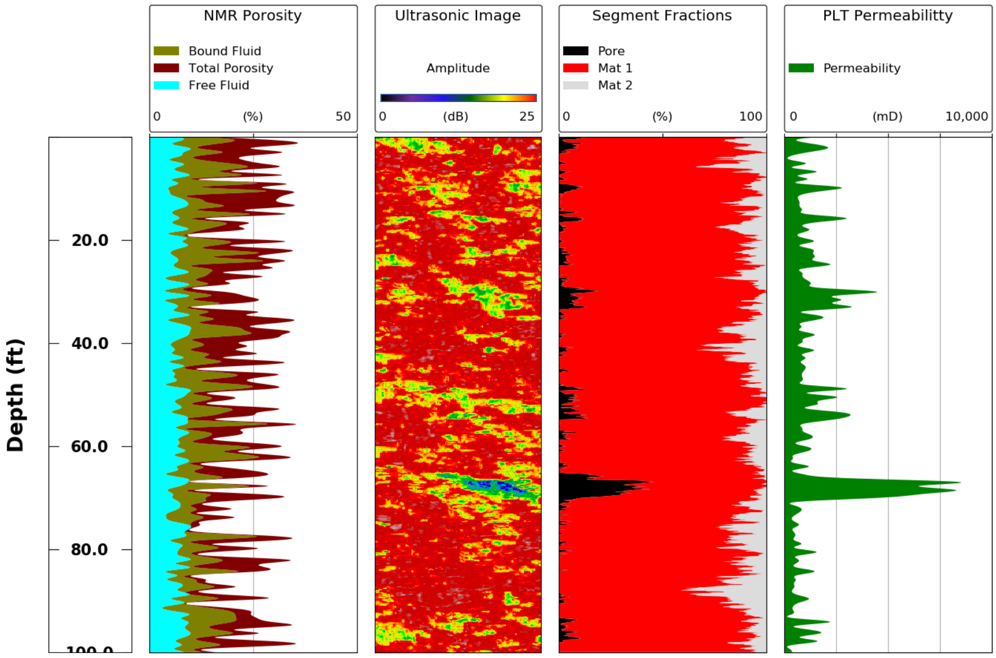

Figure 4.3: Synthetic case study 1 - features and target output (instance #03).

4.3.1.
Case Study 1: Support Vector Regressions

The pseudocode for the SVR algorithm presented in the previous chapter (see Algorithm 3.2) has been properly implemented in the Python programming language (van Rossum, 1995). The ε tolerance parameter, which defines the region in the output domain not accounted for by the optimization loss function, has been set to $\varepsilon = 0.1$. During the training process, different kernel functions have been experimented with. However, the one that delivered the overall best result is a polynomial kernel given by

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \left(\boldsymbol{x}_i^T \boldsymbol{x}_j + 10\right)^5, \tag{4.8}$$

where $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ represent two of the six selected features. This was expected because permeability has been synthetically generated from the polynomial function presented by eq. (4.7).

Instance #05 has been selected for training, and the remaining ninety-nine instances have been used for blind testing. The idea is to simulate a real situation where a model is trained with the dataset from one location and then applied on another. Figure 4.4 shows the graphic result of a test performed on instance #54. To better visualize the result, only a 30-foot segment of the entire interval is presented. Table 4.4 provides the accuracy indicator that resulted from training (instance #05) and the average indicator for the blind tests.

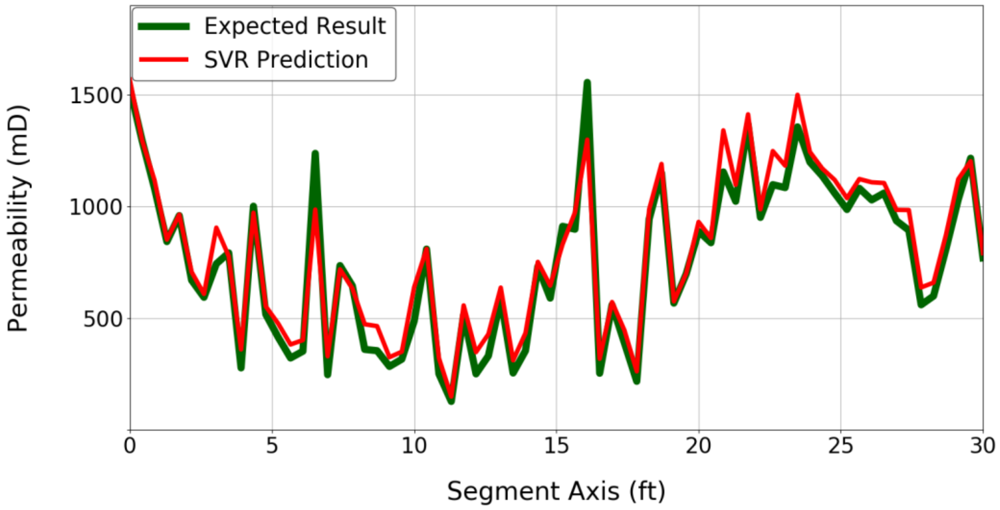

Figure 4.4: Synthetic case study 1 - SVR: permeability predictions vs expected result for the 30-foot segment of instance #54.

Table 4.4: Synthetic Case Study 1 - SVR: accuracy indicators.

	Performance (RMSE)	Cumulative Flow Capacity (R2)
Training Set (instance #05)	101.42	98.28%
Testing Set (average)	104.00	93.94 %

4.3.2.
Case Study 1: Artificial Neural Network

The pseudocode for the ANN presented in the previous chapter (see Algorithm 3.3) has also been implemented in the Python programming language. The network design consists of six input neurons, two hidden layers, and a single output neuron. The first layer is made up of ten nodes and the second layer, twenty. A Rectified Linear Unit (ReLu) function has been adopted for every single activation, as recommended by Goodfellow *et al*. (2016). It is worth mentioning that this design has been obtained after experimenting with different configurations. A simple heuristic approach has been adopted where an ideal architecture is built on a minimum number of components.

Because noise has not been added to the synthetic target output, validation error decreases monotonically, and therefore cross-validation has been overlooked. The same training instance used in the SVR application (instance #05) has been selected herein. The remaining ninety-nine synthetic datasets have been used for blind testing. Once again, the test performed on the same 30-foot segment of instance #54 is presented in Figure 4.5. Table 4.5 provides the accuracy indicator resulted from training and average indicator for the blind tests.

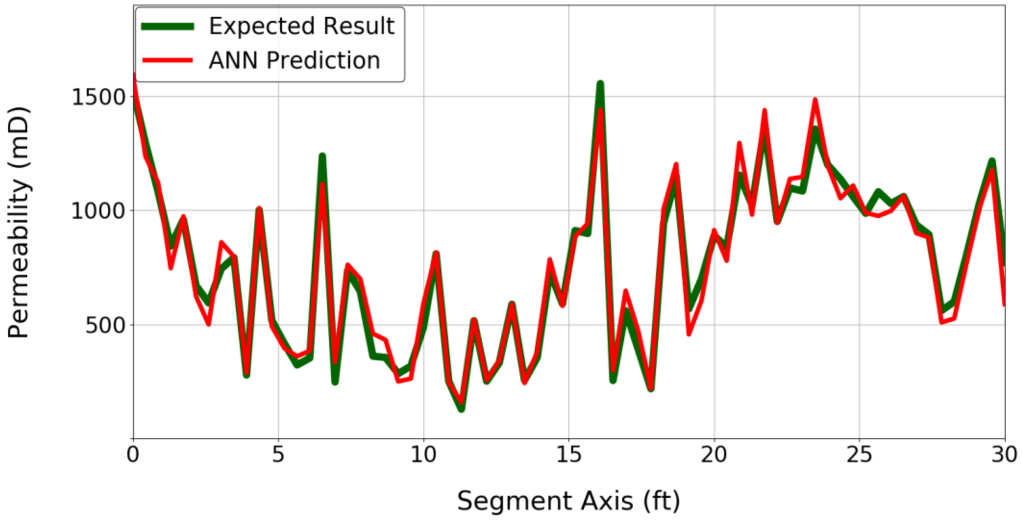

Figure 4.5: Synthetic case study 1 - ANN: permeability predictions vs expected result for the 30-foot segment of instance #54.

Table 4.5: Case study 1 - ANN: accuracy indicators.

	Performance (RMSE)	**Cumulative Flow Capacity (R2)**
Training Set (instance #05)	79.38	99.18 %
Testing Set (average)	90.36	95.29 %

4.3.3.
Case Study 1: Timur-Coates Relation

To demonstrate the importance of feature selection, the results are compared to the classic Timur-Coates relation. Note that this analysis aims neither to evaluate computational efficiency nor to test implementation but to show how sensitive predictions really are to feature datasets. The Timur-Coates equation does not account for image logging data and therefore, is expected to underperform the other algorithms in this case study. The relation is defined by

$$k = a \left(\frac{\phi_{FF}}{\phi_{BF}}\right)^{b} (\phi_{TOT})^{c} , \tag{4.9}$$

where, a, b, and c are the parameters that need to be adjusted to optimize predictions.

For consistency, the same instance has been selected for training (instance #05) and the remaining ninety-nine to assess generalization. In this example, training consists of minimizing the error between the model output and the synthetically generated permeability by slightly changing parameters a, b, and c. A gradient-based method similar to the one presented in Algorithm 3.1 has been used to solve this unconstrained optimization problem.

Once again, the graphic result for the blind test performed on the 30-foot segment of instance #54 is presented in Figure 4.6. The grey filled area represents the *Pore* class fraction, which is the only term in eq. (4.7) that is not multiplied by an NMR porosity measurement. Table 4.6 presents the same accuracy indicators as the previous applications.

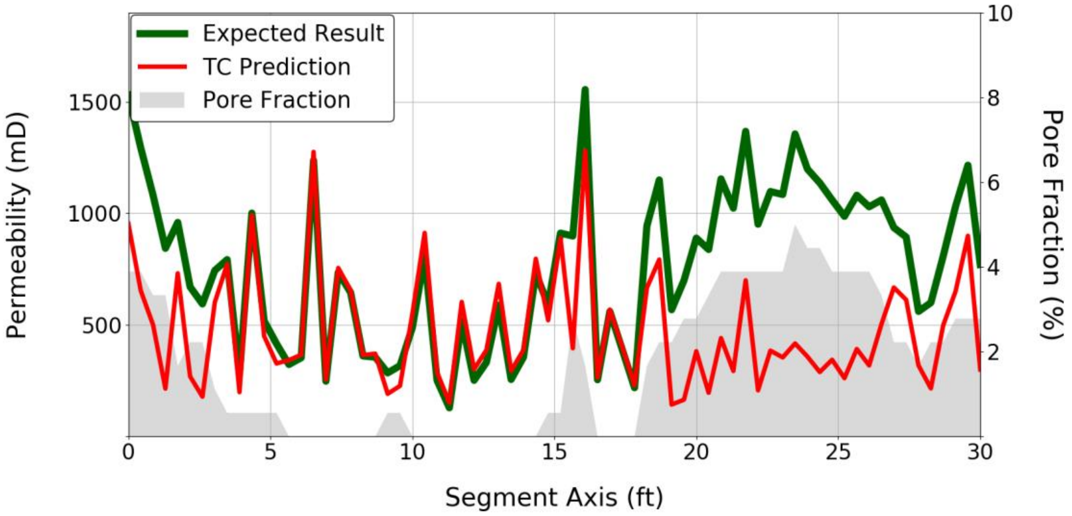

Figure 4.6: Synthetic case study 1 - ANN: permeability predictions vs expected result for the 30-foot segment of instance #54.

Table 4.6: Case study 1 - Benchmarking: accuracy indicators.

	Performance (RMSE)	**Cumulative Flow Capacity (R2)**
Training Set (instance #05)	1159.95	76.54 %
Testing Set (average)	1341.72	- 42.94 %

Naturally, the Timur-Coates model provides a better fit in regions where the *Pore* class is less present. This result could have been foreseen by simply analyzing the solution from which permeability was derived. However, the relevance of this example lies in the fact that mispredictions often occur in real situations where the log analysts fail to properly select feature datasets. The following example attempts to simulate this scenario, thoroughly.

4.4.
Synthetic Case Study 2

For this case study, permeability has been generated using an untrained neural network with a random set of weights, biases, and activation functions. A larger number of layers has been used compared to the ANNs that are built to perform predictions. The objective of this example is to simulate a situation where the physical relation between porosity and permeability is unknown. Figure 4.7 presents the graphic representation of the geological features and target outputs used for this example (instance #3).

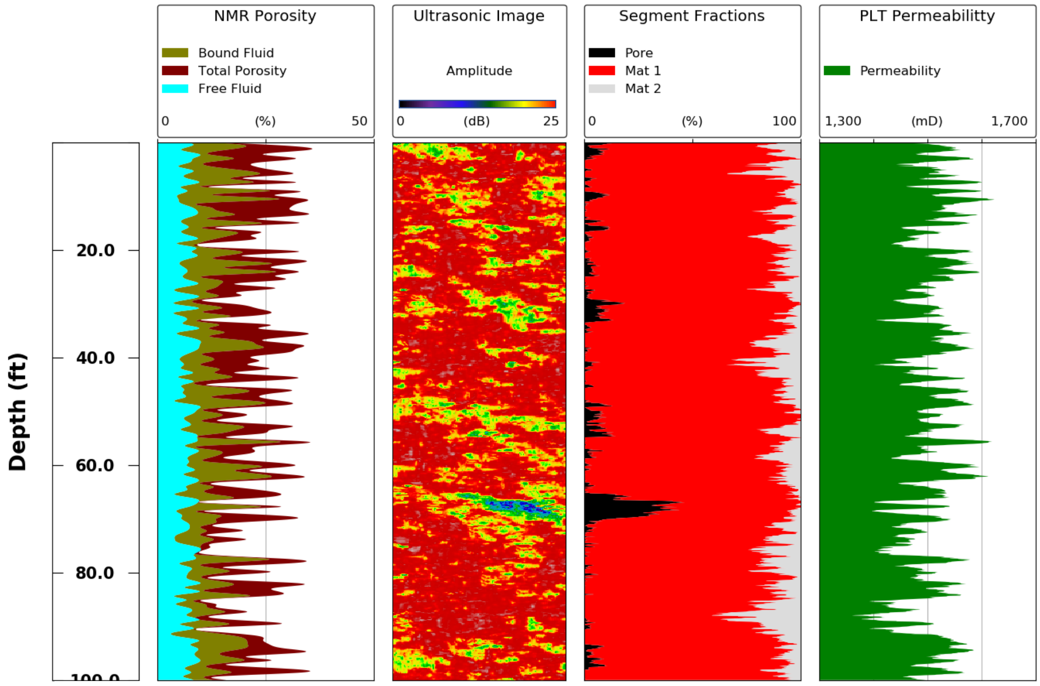

Figure 4.7: Synthetic case study 2 - features and target output (instance #03).

Compared to the previous case study, the permeability range is limited, and its outline is visually less dependent on the image's *Pore* segment. This is not by any means intentional, nor does it accurately reproduce real reservoir behavior. However, this shows randomness in the target output generation process, which is the exact intent of this synthetic case study.

For each algorithm, different feature selections have been tested and compared: feature selection (a) considers only NMR porosity; (b) considers only the synthetic image log; (c) combines NMR measurements with the *Pore* segment from the image log; (d) considers all the available features.

4.4.1.
Case Study 2: Support Vector Regressions

The same SVR kernel function and parameters of the previous example have been adopted herein. Accordingly, the same instance #05 has been selected for training and the remaining ninety-nine to test the model's generalization capacity. Figure 4.8 presents the graphic results of blind tests performed on instance #88, for each feature selection. Once again, to better visualize the result, the first 30-foot segment of the entire interval is presented. Table 4.7 provides the accuracy indicators obtained from training (instance #05) and Table 4.8 presents the average indicators for the blind tests.

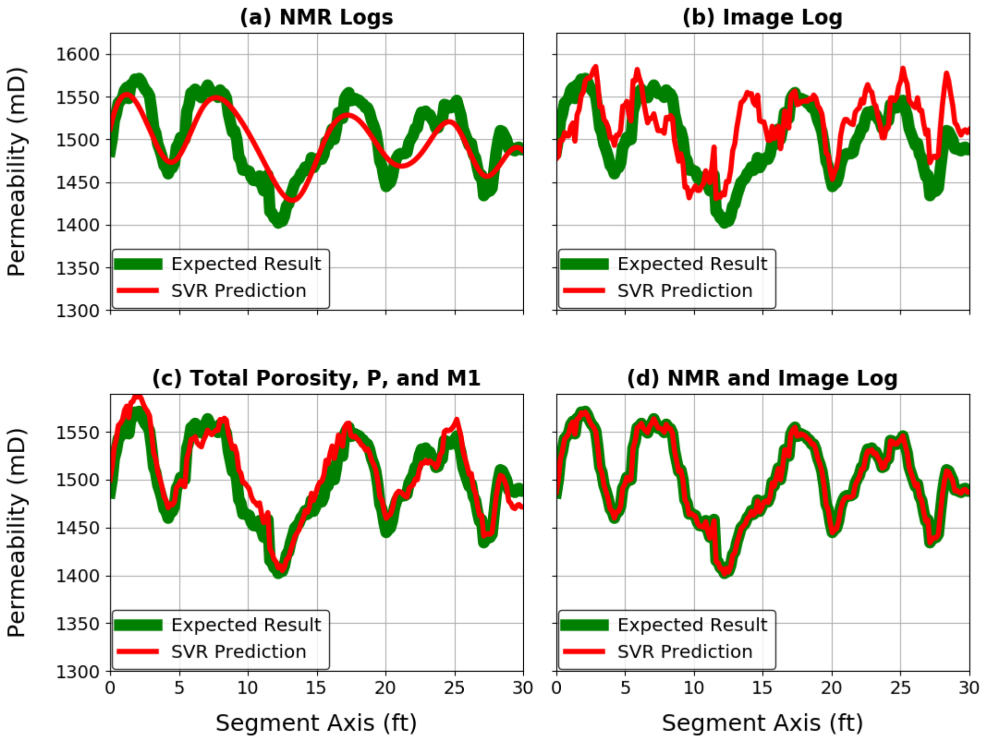

Figure 4.8: Synthetic case study 2 - SVR: permeability predictions vs expected result for the first 30-foot segment of instance #88.

Table 4.7: Case study 2 - SVR: accuracy indicators of the training set.

Features	**Performance (RMSE)**	**Cumulative Flow Capacity (Adjusted R2)**
(a) NMR (3)	23.32	68.57 %
(b) Image (3)	36.38	23.45 %
(c) NMR + *Pore* (4)	13.43	89.57 %
(d) NMR + Image (6)	0.75	99.97%

Table 4.8: Case study 2 - SVR: average indicators of the testing sets.

Features	**Performance (RMSE)**	**Cumulative Flow Capacity (Adjusted R2)**
(a) NMR (3)	27.74	44.71 %
(b) Image (3)	47.86	-
(c) NMR + *Pore* (4)	13.29	86.16 %
(d) NMR + Image (6)	0.84	99.96%

4.4.2.
Case Study 2: Artificial Neural Network

The ANN structure of Case Study 1 has also been applied in this example. For the sake of comparison, the same training instance used in the SVR application has been selected herein (instance #05). The remaining ninety-nine synthetic datasets have been used to test the model's generalization capacity. For consistency, blind tests performed on the first 30-foot segment of instance number 88 are presented in Figure 4.9, for each feature selection. Table 4.9 provides the performance indicators for the training set (instance #05) and Table 4.10 presents the average indicators for the blind tests.

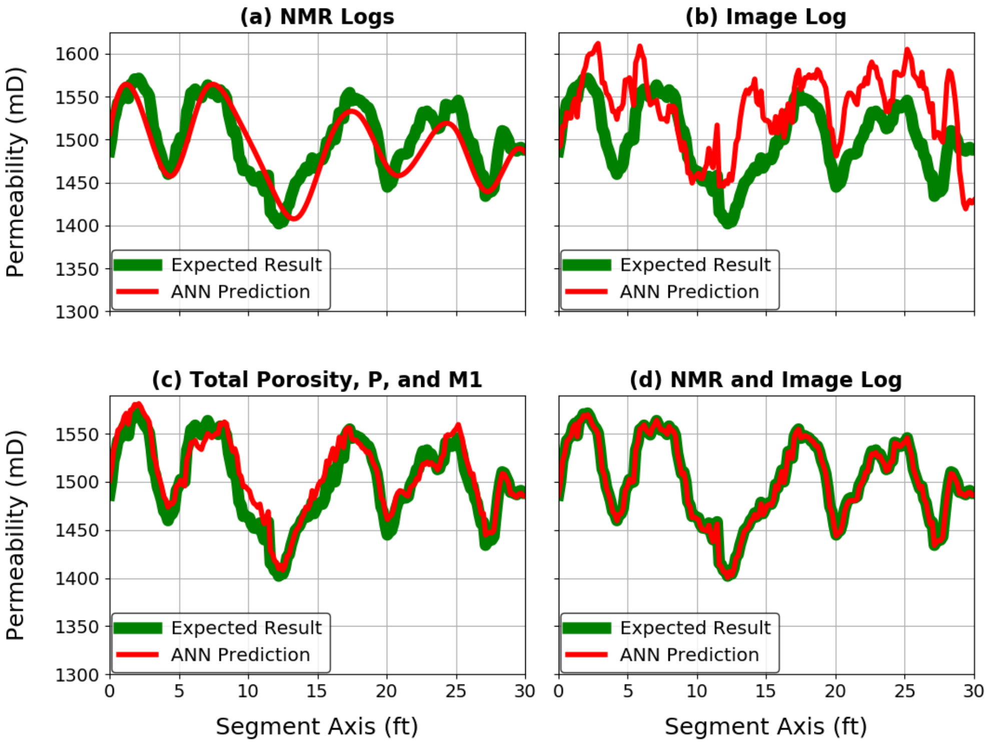

Figure 4.9: Synthetic case study 2 - ANN: permeability predictions vs expected result for the first 30-foot segment of instance #88.

Table 4.9: Case study 2 - ANN: accuracy indicators of the training set.

Features	Performance (RMSE)	Cumulative Flow Capacity (Adjusted R2)
(a) NMR (3)	23.40	68.35 %
(b) Image (3)	50.25	-
(c) NMR + *Pore* (4)	12.89	90.39 %
(d) NMR + Image (6)	1.29	99.90 %

Table 4.10: Case study 2 - ANN: average indicators of the testing sets.

Features	Performance (RMSE)	Cumulative Flow Capacity (Adjusted R2)
(a) NMR (3)	29.64	47.92 %
(b) Image (3)	61.33	-
(c) NMR + *Pore* (4)	12.85	84.64 %
(d) NMR + Image (6)	1.31	99.92 %

4.5.
Discussions and Comparisons

In Case Study 1, all the observed points lie close to the regression models. This is because noise has not been applied to the synthetic data and feature selections have not yet been experimented with in this example. Indeed, both evaluation metrics deliver corresponding results. It is important to point out that the convergence criteria for each technique are not under analysis, nor has a limit been established for the number of iterations.

In Case Study 2, different feature selections have been tested. While the minor differences between Table 4.8 and Table 4.10 indicate variations in the performances of each machine-learning technique, the adjusted R-squared metric indicates that the accuracy in permeability predictions depend on both NMR and image logs for this case study. NMR, however, seems to have a greater impact on the overall result. Nonetheless, experiments combining the image's *Pore* segments with NMR porosity showed significant improvements and good predictability.

The implemented codes are compared to commercial machine learning algorithms. For simplicity, the dataset of Case Study 1 has been adopted. Table 4.11 presents the R^2 indicators, as well as the number of iterations and computation time, for each application.

Table 4.11: Comparing algorithms - synthetic case study 1.

Application	R2	Iterations/Epochs	Time
ANN Implementation	99.39 %	5,000	34.45 (s)
ANN *Scikit-learn*	99.99 %	992	4.40 (s)
SVR Implementation	98.97 %	700	2.05 (s)
SVR *Scikit-learn*	99.99 %	500	< 1.0 (s)

From these results only, it would be inappropriate to state that the ANN has a better generalization capacity than the SVR, given that both algorithms presented high performances and that convergence criteria have not been analyzed. The field example presented in the following chapter is better suited for comparing the performances of both algorithms. However, the SVR did present a much faster convergence for both the implemented codes and the algorithms obtained from the *Scikit-learn* library (Pedregosa, 2011). Note that the high accuracy levels found in this chapter are not expected on field data. Predictions can be severely affected by noise and other uncertainties, such as tool offsets and depth errors. Neither were accounted for in these synthetic examples.

5
Field Case Study

This chapter presents a field example. Four well locations from distinct Brazilian offshore carbonate reservoirs have been selected to evaluate and compare the performances of the machine learning techniques depicted throughout this work. By experimenting with different data partitioning strategies, the impact of feature variability is assessed. Thus, this study is conducted by alternating training subjects and consistently leaving out one location for “blind tests”.

Locations have been carefully selected based on their similarities with respect to fluid and rock property, lithology, and diagenesis (Table 5.1). Furthermore, ultrasonic image, NMR, and production or injection logs have all been properly acquired. To minimize the influence of other uncontrolled features, locations have been chosen such that tool assemblies and service companies remain the same. Unlike the synthetic case study, unforeseen variables and noise are bound to occur, inevitably affecting performance accuracies.

Table 5.1: Fluid and rock properties, lithology, and diagenesis.

Property	**Well A**	**Well B**	**Well C**	**Well D**
μ_o (cP)	1.6	2.5	0.1	1.1
FVF	1.3	1.3	1.0	1.6
API	28	22	33	27
GOR	65	130	>1000	235
c_r ($\times 10^{-6}$ psi^{-1})	2.53	2.11	2.11	3.52
Lithology	Microbial Rift	Microbial Rift	Microbial Rift	Microbial Rift

Another point of attention is the interpretation process. To reduce human subjectivity, the dataset for all four wells have been analyzed by the same log specialist and well test interpreter. More importantly, like in the synthetic example, the same criterium has been adopted for the ultrasonic image log segmentation, where three classes have been determined: the *Pore* class represents structures created through diagenetic and superimposed deformational process, i.e., secondary porosity; the *Mat* 2 class which presents the highest amplitude values is interpreted as an almost impermeable matrix; and the *Mat* 1 class that exhibits intermediate amplitude values represents the reservoir's permeable matrix. Figure 5.1 presents the amplitude distribution histograms for each one of the four locations.

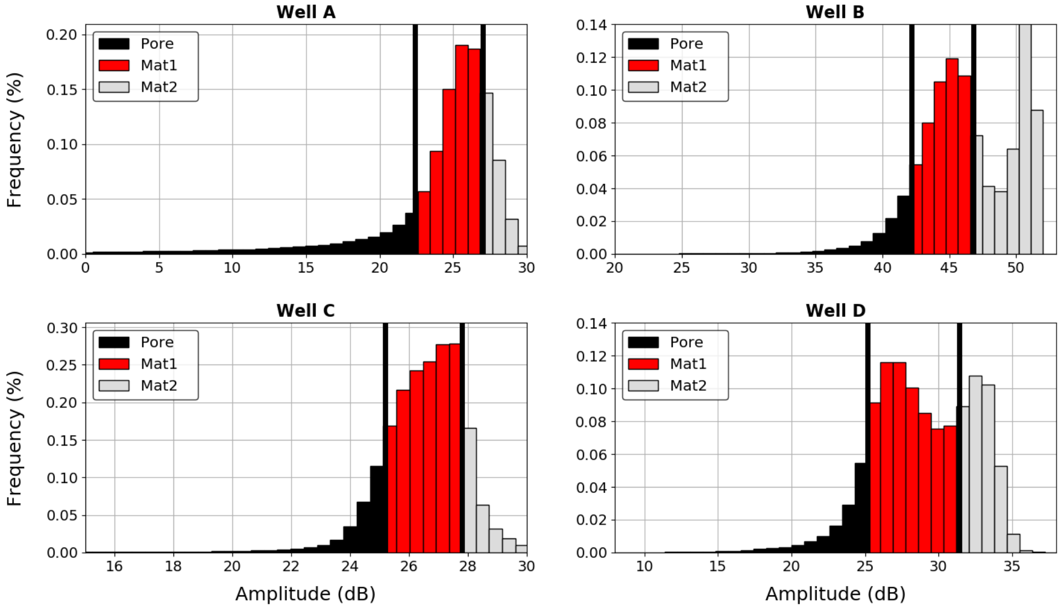

Figure 5.1: Amplitude distribution histograms for each well.

Non-geological artefacts like breakouts and drilling induced fractures have been segmented into a separate group and removed from the entire process.

Some redundancy is expected of the matrix properties extracted from the image and NMR logs. On the other hand, the secondary porosity's contribution to the deliverability prediction is solely dependent on the detailed representation of the *Pore* class. Note that the amplitude limits of the histograms are determined iteratively until a reasonable classification is reached. This allows different threshold values to be defined for each well, or even for specific segments of the same well.

5.1.
Locations and Data Details

The oil reserves found in the carbonate reservoirs of the Brazilian coast are among the most important discoveries of the last decades. According to Moczydlower *et al.* (2012), these reservoirs are a result of the mechanisms of trapping, binding, and cementation of sedimentary grains by microorganisms, known as cyanobacteria. As previously mentioned, the diagenesis and structural effects that occurred over geological time led to heterogeneous formations with large ranges of porosity. This example attempts to represent this heterogeneity by presenting locations with productivities varying from moderate to high degrees. The next subsection provides a detailed description of each location in terms of reservoir quality and data acquisition.

5.1.1.
Well A

Well A is the pioneer well of its field. It is located in a geological transfer zone, intensely affected by faults and fractures. This provides the necessary conditions for the percolation of hydrothermal fluids over time, which ultimately contributes to the process of dissolution and precipitation of minerals. In other words, Well A was drilled in a propitious environment for the existence of karsts and other cavernous structures. Additionally, the reservoir’s depositional process in structural highs conditioned a uniform distribution of microbial stromatolite facies and, therefore, high matrix permeability are expected.

A conventional well testing operation was conducted in this highly productive vertical well prior to any production in the field. The test was preceded by a matrix acidizing stimulation treatment and a complete suite of surface and downhole measurements was used throughout the operation. Early-time PTA showed a total near-wellbore effective permeability of $2{,}250\ mD$ and a total skin factor of -1.65.

Fluid production occurred above bubble point pressure even at tester valve depth. Water cut was measured below 1% during the entire operation. Two flow profiles were acquired at distinct rates, which allowed a permeability curve to be obtained. Figure 5.2 presents the NMR log, the ultrasonic image and its segment fraction curves, and the PLT-derived permeability curve.

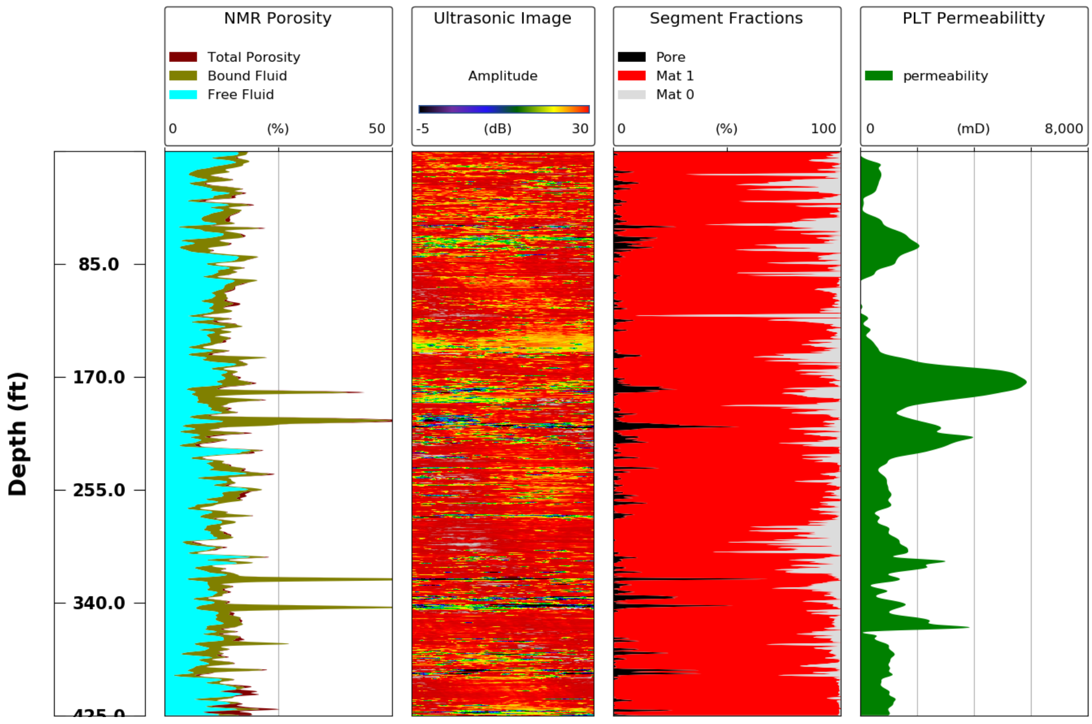

Figure 5.2: Input and output dataset for well A.

5.1.2.
Well B

Well B was one of the first wells inside of its ring fence to explore the limits and boundaries of the field. Unlike in Well A, it is not located in a severely active environment and therefore the diagenetic events consist basically of partial dolomitization of microbial structures, compression, and dissolution. While partial dolomitization promotes the obliteration of pores, dissolution contribute to increase secondary porosity. A high concentration of the latter would ultimately enhance the reservoir's transmissibility. However, with exception of the superior portion of the interval, the ultrasonic image log presented in Figure 5.3 does not indicate a remarkable presence of dissolution along the perforations. Additionally, depositional processes have caused an intercalation of grainstones and stromatolite with a rather thin grained distribution, which also leads to a low expectation of matrix permeability.

An injectivity test was performed in the same configuration as the previously described well testing operation. Its result exceeded every expectation in terms of the well's deliverability. Early-time PTA indicated a total near-wellbore effective permeability of $550\,mD$ and a total skin factor of -2.3. The PLT-derived permeability curve, also presented in Figure 5.3, suggests that the concentration of dissolution in the superior portion of the interval ultimately contributed to the well's unexpected performance.

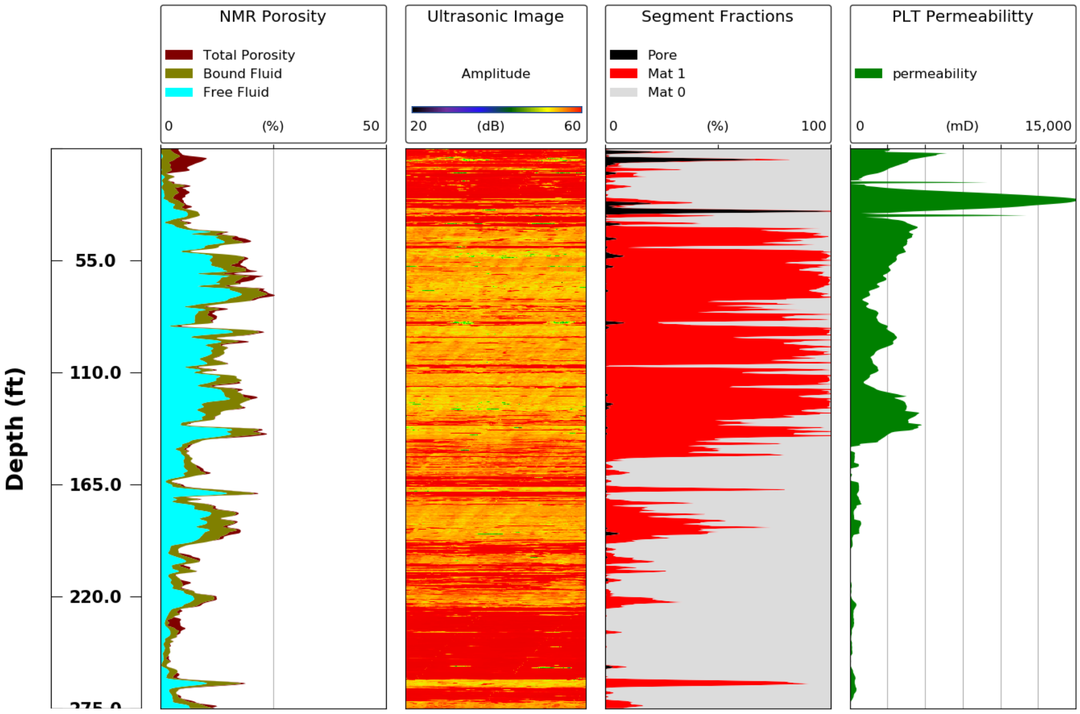

Figure 5.3: Input and output dataset for well B.

5.1.3.
Well C

Well C is the fifth well drilled inside of its field's ring fence. It is also located in a less tectonically active setting and therefore less affected by faults and fractures. Nonetheless, the field's pressure and temperature conditions have favored the diagenetic process of dissolution and dolomitization which can be identified through image logs in a less obvious manner. The reservoir's depositional process has also taken place in structural highs, however, its distribution of microbial stromatolite and grainstone facies are very widespread, and thus generating lower expectations of the reservoir's matrix potential.

A conventional well testing operation has also been conducted in an undisturbed environment of the field. The same description of the well testing conditions of Well A can be replicated herein. However, the early-time PTA showed a total near-wellbore effective permeability of 80 mD and a total skin factor of 0. Figure 5.4 presents the NMR log, the ultrasonic image and its segment fraction curves, and the PLT-derived permeability curve.

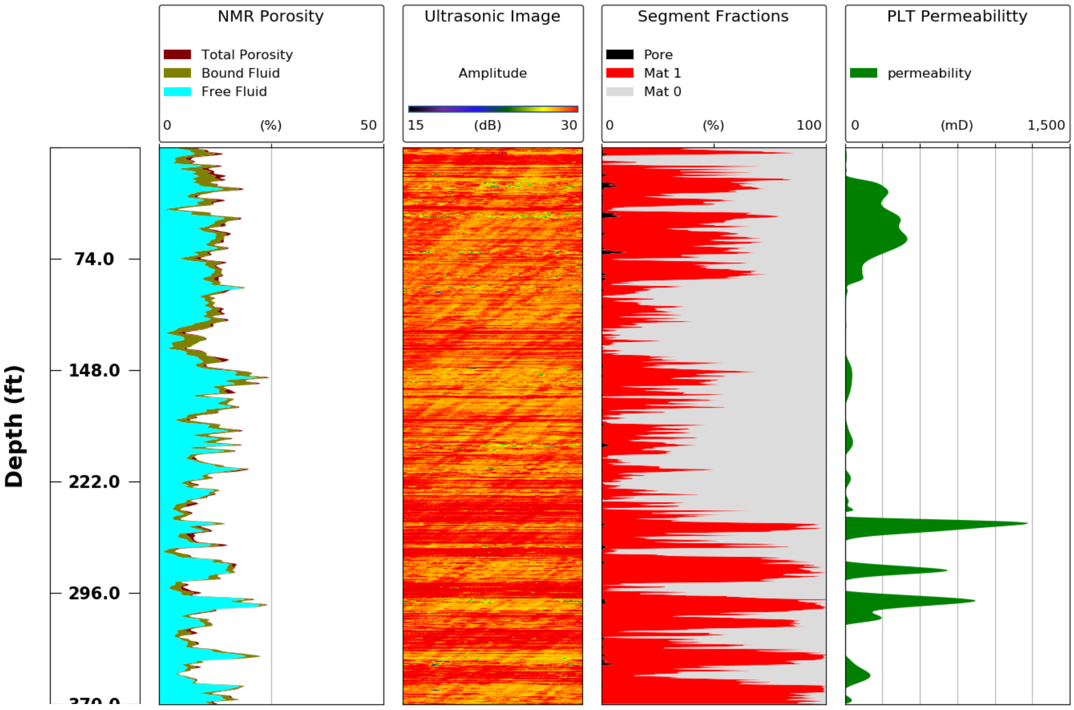

Figure 5.4: Input and output dataset for well C.

5.1.4. Well D

In terms of tectonic activities, Well D is known to be located in a setting similar to the location of Well A. Unfortunately, the borehole quality and geometry created great disturbances in the ultrasonic image log, making it difficult to interpret fractures and other secondary porosities along the permeable interval. Spiraling effects and oval-shaped geometries can be observed on the image log presented in the second track of Figure 5.5. This well was drilled in a structural low, where its depositional process conditioned a heterogeneous distribution of stromatolite and other thin grained facies.

A conventional well testing operation, like the ones previously described for wells A and C, was conducted. Early-time PTA indicated a total near-wellbore effective permeability of 260 mD and a total skin factor of -2.0. Like in Well B, The PLT-derived permeability curve suggests that this well's good performance is somehow related to secondary porosity structures, which the ultrasonic image log was unable to properly reveal.

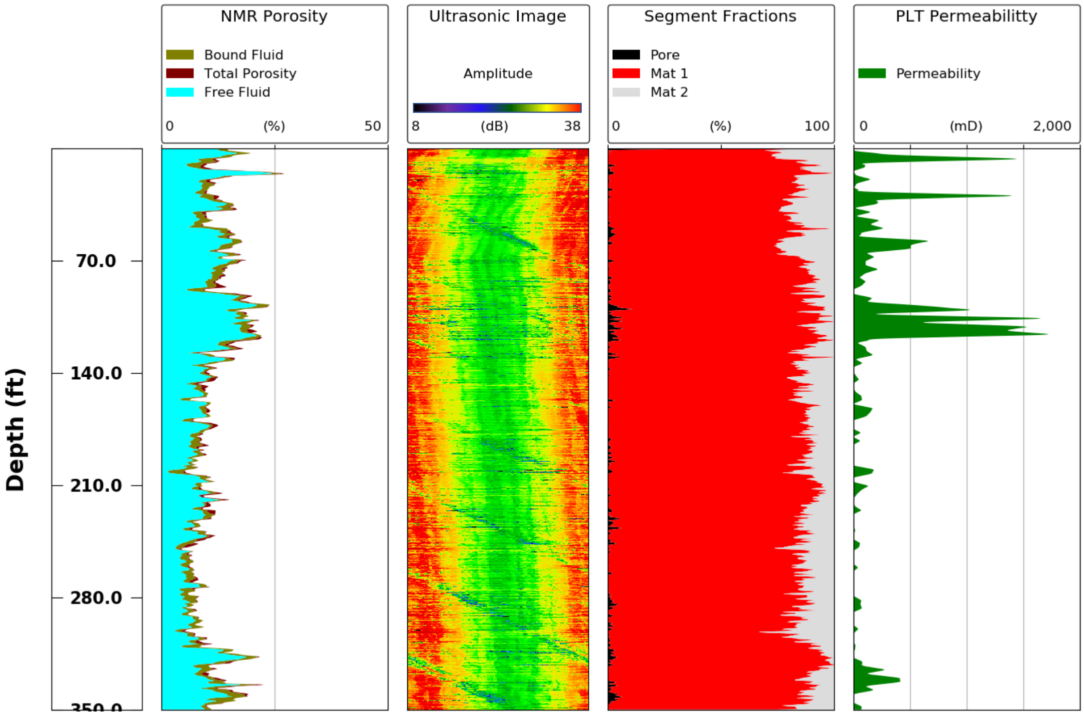

Figure 5.5: Input and output dataset for well D.

The main results of the well-testing and borehole logging operations are presented in Table 5.2. The deliverabilities of wells A, B, and D are presented in terms their productivity indexes (PI). For comparison, the deliverability of well C is presented in terms of its Injectivity Index (II) corrected by the reservoir's formation volume factor (FVF).

Table 5.2: Well logging and well testing results

Property	**Well A**	**Well B**	**Well C**	**Well D**
$h\ (ft)$	425	275	370	350
ϕ_{eq} (%)	9.5	10.2	9.3	10.5
$k_{eq}(mD)$	2,250	550	80	260
S_{eq}	-1.65	-2.3	0.0	-2.0
PI $(bbl/d/psi)$	525	-	105	65
II/FVF $(bbl/d/psi)$	-	50	-	-

5.2.
Blind Testing

Four blind tests have been arranged to demonstrate the ANNs' and SVRs' applicability on permeability predictions. In each test, datasets from three locations are used for training, leaving out the fourth location to evaluate performance. Throughout this example, results are compared to the classical Timur-Coats (TC) relation [see eq. (4.9)]. At the end of this chapter, averages of the indicators from the four blind tests are presented.

Both permeability and cumulative flow capacity curves have been analyzed. The latter is particularly appealing as it demonstrates the model's ability to predict total deliverability. Furthermore, it minimizes the high frequency noises caused by tool offsets and other minor disturbances, allowing a better visual assessment of the models' performances. The cumulative flow capacity curves are computed by integrating permeability over depth.

5.2.1.
Blind Test A

The first blind test was performed on well A after training the models with borehole logs and flow profiles obtained from wells B, C, and D. Figure 5.6 presents the permeability curves obtained with each technique. For visualization purposes, only a 200-foot segment of the entire interval is presented.

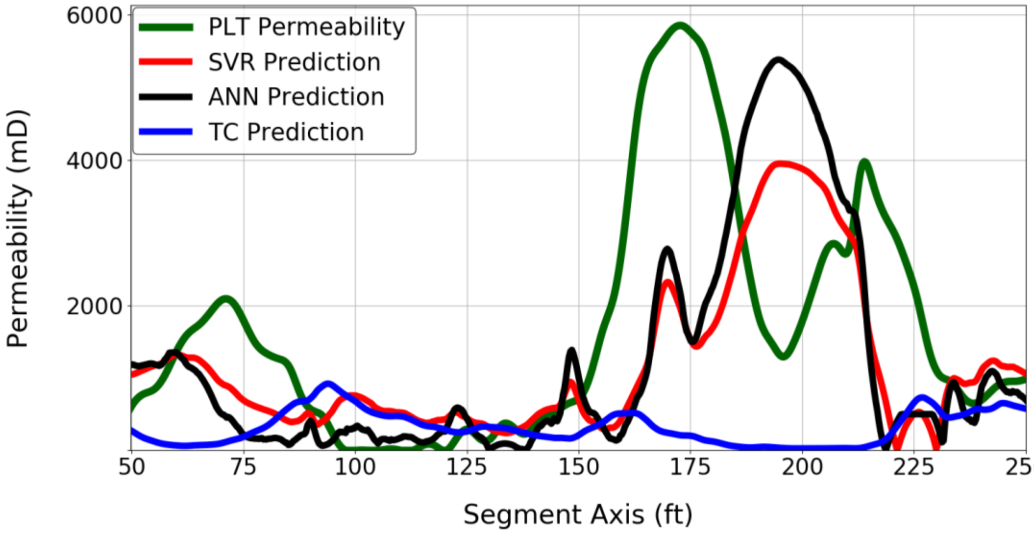

Figure 5.6: Permeability predictions on a 200-foot segment of well A, using wells B, C, and D for training.

Figure 5.7 presents the results in terms of cumulative flow capacity for the entire interval. Table 5.3 summarizes the indicators for both analyses and for all three predictive models (see section 4.2 for details on evaluation metrics).

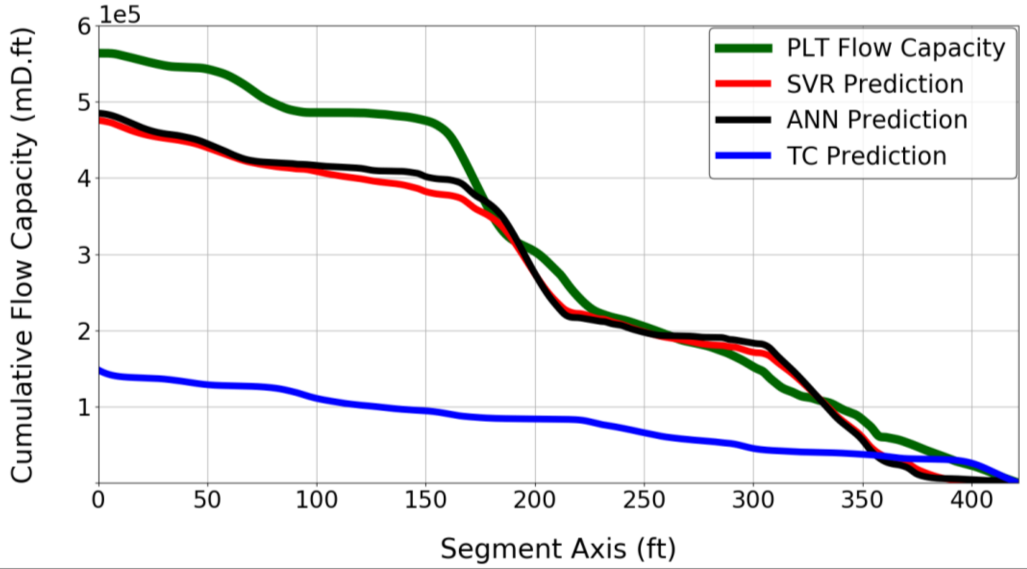

Figure 5.7: Cumulative flow capacity predictions for well A, using wells B, C, and D for training.

Table 5.3: Blind test A: performance indicators.

	Performance (RMSE)	**Cumulative Flow Capacity (R2)**
SVR	1256.92	98.19 %
ANN	1420.65	98.05 %
Timur-Coates	1669.89	45.84 %

The SVR algorithm considered a polynomial kernel of degree 5 and $\varepsilon = 0.1$. As for the ANN, the same design of the synthetic case studies has been adopted. The number of neurons for each layer has been optimized during a K-fold CV process in the training set.

Note how the TC relation fails to predict permeability in regions with high concentrations of secondary porosity (e.g., at 200 ft of depth). Nonetheless, the similar slope of the three curves in regions with low *Pore* class concentration indicates a successful modeling of matrix permeability.

5.2.2.
Blind Test B

The second blind test was performed on well B after training the models with the datasets from wells A, C, and D. Figure 5.8 presents the permeability curves obtained with each technique, Figure 5.9 presents the results in terms of cumulative flow capacity, and Table 5.4 summarizes the indicators for both analyses and for all three predictive models (see section 4.2 for details on evaluation metrics).

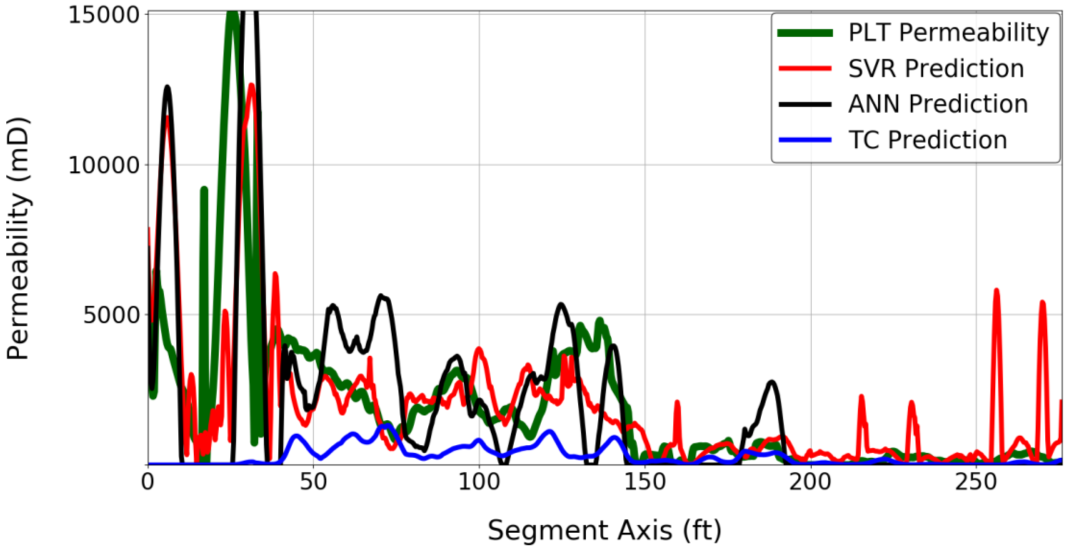

Figure 5.8: Permeability predictions for well B, using wells A, C, and D for training.

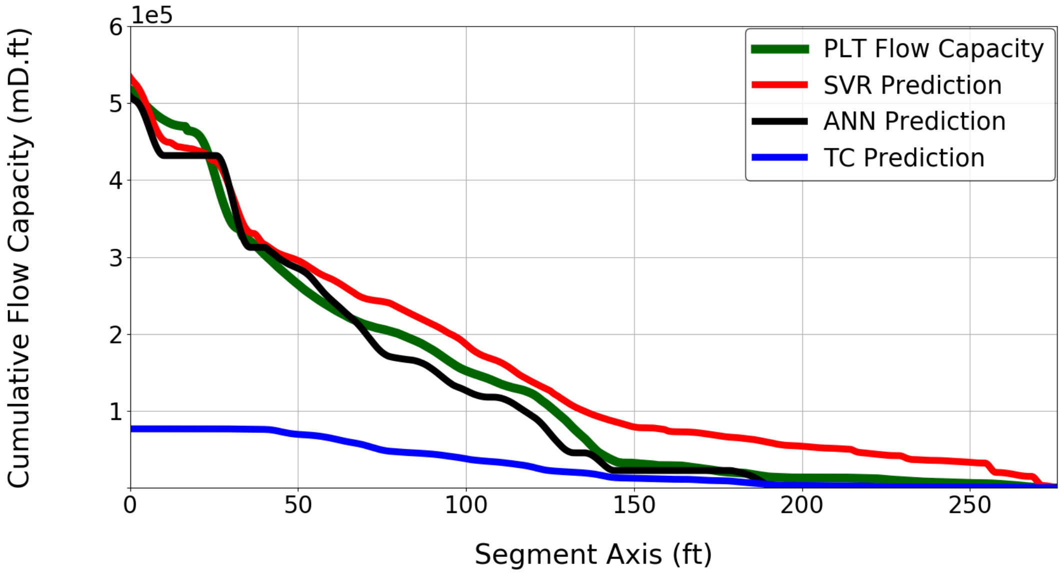

Figure 5.9: Cumulative flow capacity predictions for well B, using wells A, C, and D for training.

Table 5.4: Blind test B: performance indicators.

	Performance (RMSE)	**Cumulative Flow Capacity (R2)**
SVR	2255.94	99.88 %
ANN	2875.86	99.90 %
Timur-Coates	2958.94	27.42 %

The same SVR kernel function and ANN design of the previous test has been adopted in blind test B. Accordingly, parameters have been optimized during the training process. The result of this test supports the initial hypothesis that the fractures located in the superior portion of the interval contributed to this well’s unexpected performance during the injectivity test. Both SVR and ANN predicted high levels of transmissibility for the first 50-foot segment of the interval, while the TC relation failed to do so.

5.2.3.
Blind Test C

Blind test C was performed after training the models with the datasets from wells A, B, and D. Figure 5.10 presents the permeability curves obtained with each technique, Figure 5.11 presents the results in terms of the cumulative flow capacity, and Table 5.5 summarizes the performance indicators.

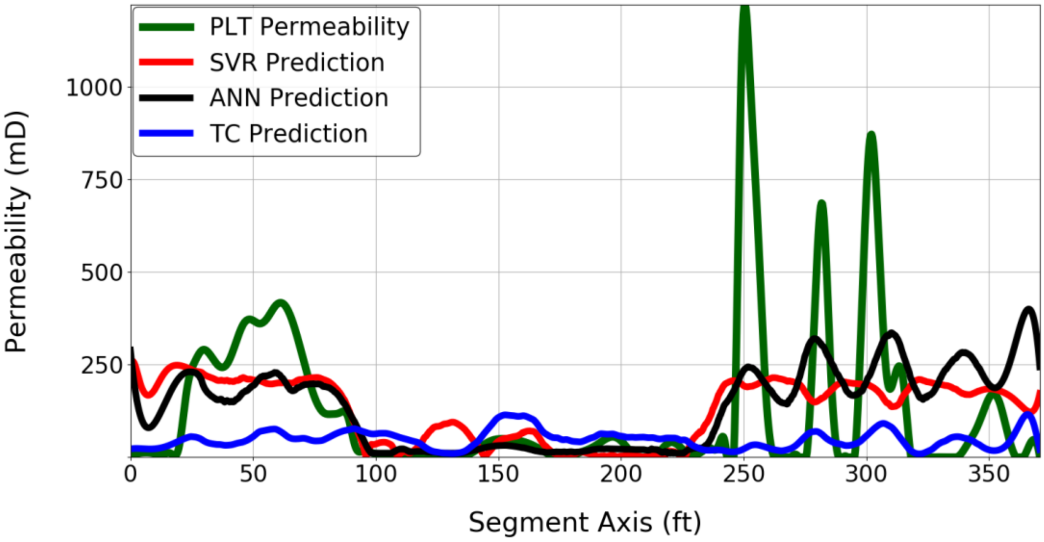

Figure 5.10: Permeability predictions for well C, using wells A, B, and D for training.

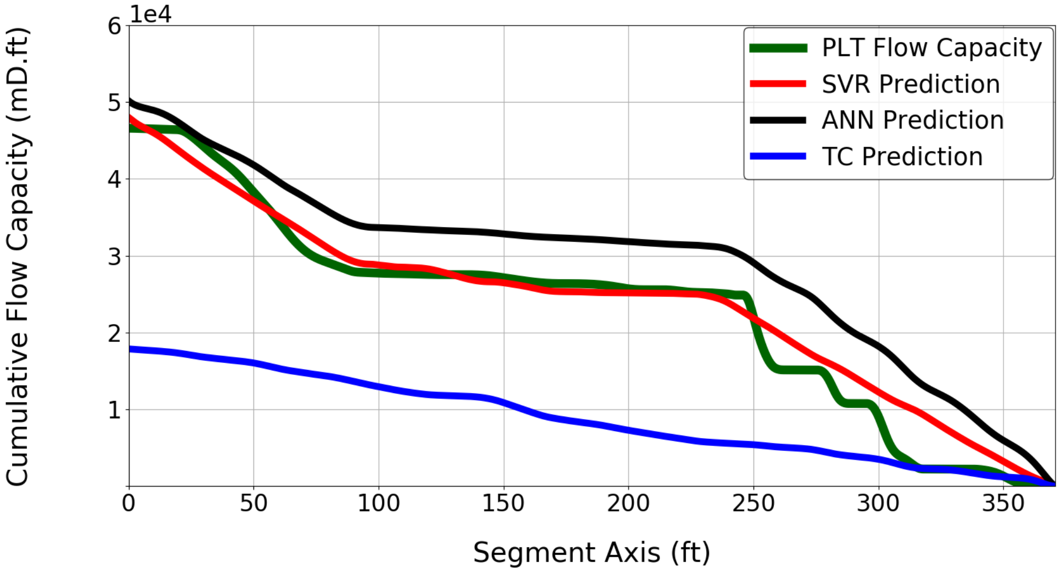

Figure 5.11: Cumulative flow capacity predictions for well C, using wells A, B, and D for training.

Table 5.5: Blind test C: performance indicators.

	Performance (RMSE)	**Cumulative Flow Capacity (R2)**
SVR	188.82	99.90 %
ANN	184.88	99.38 %
Timur-Coates	219.42	62.01%

Once again, the same SVR kernel function and ANN design were applied. Although higher performances were reached using the machine learning algorithms, the classical TC relation performed better than the previous tests. This could be explained by the low concentrations of secondary porosity found in this well's ultrasonic image log.

5.2.4.
Blind Test D

Blind test D was performed after training the models with the datasets from wells A, B, and C. Figure 5.12 presents the permeability curves obtained with each technique, Figure 5.13 presents the results in terms of the cumulative flow capacity, and Table 5.6 summarizes the performance indicators.

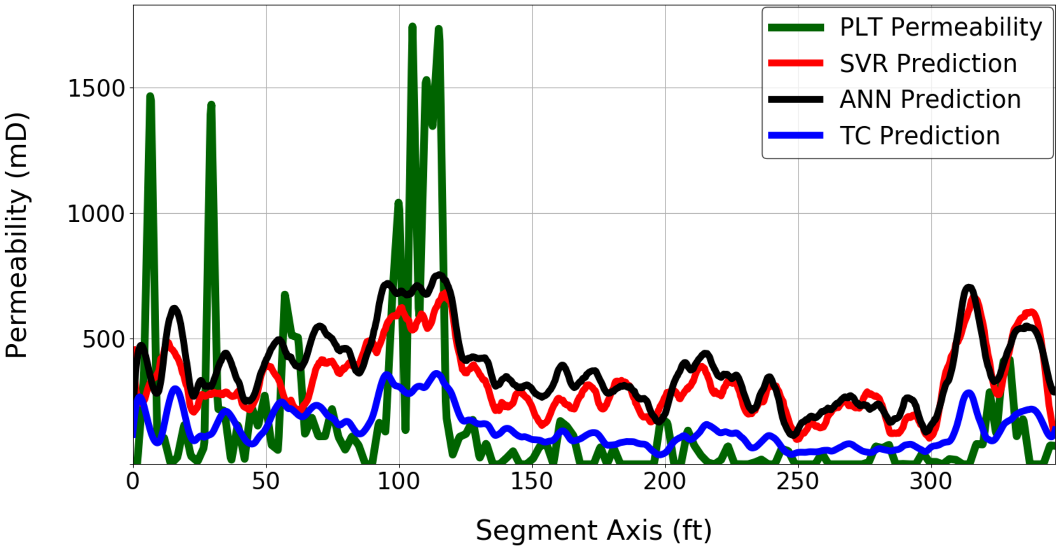

Figure 5.12: Permeability predictions for well D, using wells A, B, and C for training.

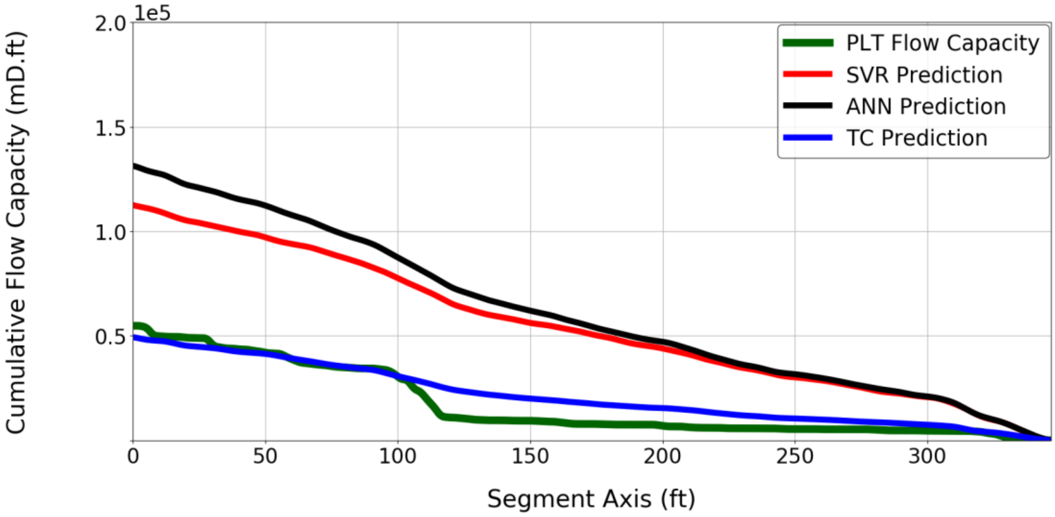

Figure 5.13: Cumulative flow capacity predictions for well D, using wells A, B, and C for training.

Table 5.6: Blind test D: performance indicators.

	Performance (RMSE)	**Cumulative Flow Capacity (R2)**
SVR	326.85	-11.59 %
ANN	346.77	-95.02 %
Timur-Coates	272.75	99.02%

For blind test D, simpler configurations for both the SVR and ANN methods were applied. The SVR model used a polynomial kernel of degree 3 and $\varepsilon = 0.1$, and the ANN used a single hidden layer of 20 nodes. This is because the models were fed better quality data during the training process.

Note how all three predictions present similar profiles at different magnitudes. A possible explanation for this, is that the fraction curves derived from the image log presents low variations for each of its three classes. Thus, predictions are more sensitive to the NMR data variations. The TC relation outperforms the machine learning algorithms for this test, probably because it was not influenced by the high levels of Mat 1 obtained from the poor-quality image log. The R-squared indicators reveal that the measured permeability's mean value outperforms both the SVR and the ANN algorithms.

5.3.
Field Case Study Conclusions

The comparative analysis of the three methods must be divided in two-fold: feature selection analysis and algorithm general performance analysis. The TC relation does not account for the structures revealed by the ultrasonic image log, and therefore cannot be directly compared to the intelligent machine learning methods that used all possible features. The TC relation, however, is considered a benchmark in permeability predictions from a petrophysics perspective. Its underperformance in this field case study reveals that a more complex model is required for heterogeneous carbonate formations.

On the other hand, the SVR and the ANN can be compared amongst themselves, in terms of their general performances. However, both algorithms performed similarly. Variations in RMSE values across each blind test indicate instability in precise point to point predictability. Nevertheless, a visual evaluation of their results indicates that both SVRs and ANNs are perfectly capable of predicting the reservoir's general behavior in a heterogeneous environment. This visual comparison of the algorithms is only possible because production logging data has been incorporated to the permeability estimation workflow. The cumulative flow capacity indicator reveals the models' ability to predict total deliverability from borehole logs.

6
Conclusions

Estimating permeability distribution from open-hole logs can provide important inputs for the geological modeling of the reservoir, especially when well-testing operations are not programmed to be performed. It can also have a great impact on acid job designs, which sometimes need to be executed before any dynamic data has been acquired.

This work evaluated two machine learning methods on the prediction of permeability curves in heterogeneous environments. By adopting ultrasonic image and NMR logs as sources for the models' input features, both matrix and secondary porosities have been accounted for along the process. Production logging data provided the permeability estimation workflow with the necessary means to reach well testing scales. Unequal skin distribution across the wellbore has been dealt with by adopting a multi-rate production log interpretation method to compute the algorithms' target outputs.

The mathematical foundation for each method has been thoroughly depicted throughout this work. Their algorithms were considered simple enough to be implemented on a desktop computer and did not require extremely high processing power. Implementations were verified by comparing their results with ones obtained from commercial machine learning algorithms, in two synthetic examples.

A field case study was presented, and both algorithms successfully predicted permeability curves for three of the four blind tests that were performed. This practical example revealed the importance of incorporating production logging data to the permeability estimation workflow, especially for heterogeneous carbonate formations.

6.1.
Main Contributions

This work attempts to contribute to the petrophysics' permeability estimation workflow, with the ambition that two of the most relevant methods of the machine learning discipline become a part of the community's toolbox for predicting reservoir dynamic behavior. The main contributions of this work are listed below:

1. Production logging data has been successfully included into the workflow;
2. Unequal skin distribution across the wellbore has been accounted for by adopting a multi-rate production logging interpretation method to compute the algorithms' target outputs;
3. The use of machine learning allows the modeling routine to be deployed without many assumptions on the physical relation between geological features and permeability, in advance;
4. Both matrix porosity and secondary porosity have been accurately modeled by adopting ultrasonic image logs and NMR logs as input features.

6.2.
Future Work

To extend the accomplishments of this work, the following aspects are considered promising subjects to be explored: (i) the use of convolutional neural networks to extract features from the 2D ultrasonic image logs; (ii) study of the effect of different secondary porosity structures in the measured permeability curves (e.g., fracture, vugs, and dissolutions); (iii) include seismic data and pressure history matching as independent variables of the permeability estimation workflow.

Machine-learning has been increasingly prominent in the field of formation evaluation. Its potential for permeability estimation problems has not been exhausted by any means. Other intelligent methods are yet to be explored with the ultimate objective of enhancing estimations' accuracies.

Bibliography

ABRAMOWITZ, M., STEGUN, I.A. Handbook of Mathematical Functions with Formulas, Graphs and Mathematical Tables. **National Bureau of Standards Applied Mathematics Series 55**, 1964.

AHMED, U., CRARY, S.F., AND COATES, G.R. Permeability Estimation: The Various Sorces and Their Interrelationships. **Journal of Petroleum Technology 43** (05): 578-587. doi:10.2118/19604-PA, 1991.

AIZERMAN, M.A., BRAVERMAN, E.M., AND ROZONOER, L.I. Theoretical foundations of the potential function method in pattern recognition learning. **Automation and Remote Control 25**, 821–837. 1964.

AL-ANAZI, A., GATES, I.D. A Support Vector Machine Algorithm to Classify Lithofacies and Model Permeability in Heterogeneous Reservoirs. **Engineering Geology 114**, (03-04): 267-277. doi: 10.1016/j.enggeo.2010.05.005, 2010.

ALOM, M.Z., TAHA, T.M., YAKOPCIC, C., WESTBERG, S., SIDIKE, P., NASRIN, M.S., ESESN, B.C.V, AWWAL, A.A.S., AND ASARI, V.K. The History Began from AlexNet: a Comprehensive Survey on Deep Learning Approaches. **arXiv preprint.** arXiv:1803.01164, 2018.

ARLOT, S., CELISSE, A. A Survey of Cross-Validation Procedures for Model Selection. **Statistics Surveys 04** (2010): 40-79. doi:10.1214/09-SS054, 2009.

BACHMAT, Y., AND BEAR, J.: The Concept and Size of an REV (Fundamentals of Transport Phenomena in Porous Media). *In*: **NATO Advanced Study Institute**, 1985.

BEAR, J. **Dynamics of Fluids In Porous Media**. New York: Elsevier, 1968.

BLOCH, F. Nuclear induction. **Physical Review 70**, (07-08): 460. doi:10.1103/PhysRev.70.460, 1946.

BOSER, B.E.; GUYON, I.M.; VAPNIK, V.N. A Training Algorithm for Optimal Margin Classifiers. *In:* **Proceedings of the fifth annual workshop on Computational learning theory – COLT**. doi:10.1145/130385.130401, *1992.*

BOTTOU, L., AND LIN, C.J. Support Vector Machine Solvers. **Large scale kernel machines 3,** (01): 301-320, 2007.

COLLINS, R.E. **Flow of Fluids Through Porous Materials**. Tulsa: PennWell Publishing Company, 1961.

DARCY, H. **Les Fontaines Publiques de la Ville de Dijon**. Paris: Victor Dalmont, 1856.

DEL REY, A.C., PERES, A.M, BARRETO, A.B., DE ALMEIDA, S.R., AND SOMBRA, C.L. Direct Permeability Estimation Using Production Log. *In*: **SPWLA 50th Annual Logging Symposium**. 2009.

EHLIG-ECONOMIDES, C. A., AND JOSEPH, J. A New Test for Determination of Individual Layer Properties in a Multilayered Reservoir. **SPE Formation Evaluation 2** (03): 261-283. SPE-14167-PA. doi:10.2118/14167-PA, 1987.

FRAGA, C.T., LARA, A.Q., PINTO, A.C., AND BRANCO, C.C. Challenges and Solutions to Develop Brazilian Pre-Salt Deepwater Fields. *In*: **World Petroleum Congress**. WPC-21-2084, 2014.

VAN EVERDINGEN, A.F. The Skin Effect and Its Influence on the Productive Capacity of a Well. **Journal of Petroleum Technology 5** (06): 171–176. doi: https://doi.org/10.2118/203-G, 1953

GALVAO, M.S.C., AND GUIMARAES, C. S. A New Method for Calculating Individual Layer Permeability and Skin in a Multilayered Reservoir Using Production Logging Data: The Delta Transient Method. *In*: **SPE Latin America and Caribbean Mature Fields Symposium**. doi:10.2118/184940-MS, 2017.

GOODFELLOW, I., YOSHUA, B., AND COURVILLE, A. **Deep learning**. MIT Press. http://www.deeplearningbook.org, 2016.

GUIMARAES, C.S., AND GALVAO, M.S.C. Application of the Delta Transient Method to Multi-Rate Tests: A Method for Calculating Individual Layer Permeability and Skin in a Multilayered Reservoir using Production Logging Data. *In*: **Offshore Technology Conference - OTC Brasil**. doi: https://doi.org/10.4043/28061-MS, 2017.

GUIMARAES, C.S., SCHIRMER, L., SCHARDONG, G., BARRETO, A.B., AND LOPES, H. Incorporating Dynamic Production-Logging Data to the Permeability-Estimation Workflow Using Machine Learning. **SPE Journal *25*.** 2765–2777. doi: 10.2118/198954-PA, 2020

HALDORSEN, H.H. Simulator Parameter Assignment and The Problem of Scale in Reservoir Engineering. **Reservoir characterization 6.** 293-340, 1986.

HARO, C.F. Permeability Modeling in Porous Media: Setting Archie and Carman-Kozeny Right. *In*: **SPE Europec/EAGE annual Conferences and Exhibition**.doi:10.2118/100200-MS, 2006.

HAWKINS, M.F. A Note on the Skin Effect. **Journal of Petroleum Technology 8**. 65–66. doi: 10.2118/732-G, 1956.

HE, J., AND MISRA, S. Generation of synthetic dielectric dispersion logs in organic-rich shale formations using neural-network models. **Geophysics, 84** (3): D117-D129. doi:10.1190/geo2017-0685.1, 2019.

HELLE, H.B., BHATT, A. AND URSIN, B. Porosity and Permeability Prediction from Wireline Logs using Artificial Neural Networks: a North Sea Case Study. **Geophysical Prospecting 49** (04): 431–444. doi:10.1046/j.1365-2478.2001.00271.x, 2001.

HURST, W. Establishment of the Skin Effect, and its Impediment to Fluid Flow into a Well Bore. **Petroleum Engineer**. B6-16. 1953.

HURST, W., CLARK, J.D., AND BRAUER, E.B. The Skin Effect in Producing Wells. **Journal of Petroleum Technology 21**. 1483–1489. doi: 10.2118/1854-PA, 1969

HESTENES, M.R., AND STIEFEL, E. Methods of Conjugate Gradients for Solving Linear Systems. **Journal of Research of the National Bureau of Standards 49**. (06): 409-436. doi: 10.6028/JRES.049.044,1952.

KEELAN, D.K. Core Analysis Techniques and Applications. *In*: **SPE Eastern Regional Meeting**. doi: https://doi.org/10.2118/4160-MS, 1972.

KORRINGA, J., SEEVERS, D.O., AND TORREY, H.C. Theory of Spin Pumping and Relaxation in Systems with a Low Concentration of Electron Spin Resonance Centers. **The Physical Review 127** (04): 1143-1150. doi:10.1103/PhysRev.127.1143, 1962.

KUCUK, F., KARAKAS, M., AND AYESTARAN, L. Well Testing and Analysis Techniques for Layered Reservoirs. **SPE Formation Evaluation 1** (04): 342-354. doi:10.2118/13081-PA, 1986.

LARSEN, L. Determination of Skin Factors and Flow Capacities of Individual Layers in Two-Layered Reservoirs. *In***: SPE Annual Technical Conference and Exhibition**. doi:10.2118/11138-MS, 1982.

LI, H., AND MISRA, S. Long short-term memory and variational autoencoder with convolutional neural networks for generating NMR T2 distributions. **IEEE Geoscience and Remote Sensing Letters 16** (2): 192-195. doi:10.1109/LGRS.2018.2872356, 2018.

LI, H., MISRA, S., AND HE, J. Neural network modeling of in situ fluid-filled pore size distributions in subsurface shale reservoirs under data constraints. **Neural Computing and Applications**: 1-13. doi:10.1007/s00521-019-04124-w, 2019.

LIM, J.S. Reservoir Properties Determination Using Fuzzy Logic and Neural Networks from Well Data in Offshore Korea. **Journal of Petroleum Science and Engineering 49** (03): 182-192. doi: 10.1016/j.petrol.2005.05.005, 2005.

LOYD'S REGISTER (LR). **Interactive Petrophysics** v4.6.2019.23403, Well Analysis Software. London: LR, 2019.

LYAQINI, S., QUAFAFOU, M., NACHAOUI, M., AND CHAKIB, A. Supervised learning as an inverse problem based on non-smooth loss function. **Knowledge and Information Systems:** 1-20. doi:10.1007/s11042-020-09900-8, 2020

MENEZES DE JESUS, C., COMPAN, A.L.M., SURMAS, R. Permeability Estimation Using Ultrasonic Borehole Image Logs in Dual-Porosity Carbonate Reservoirs. **Society of Petrophysicists and Well-Log Analysts 57** (06): 620-637, 2016.

MENGER, S. New Aspects of the Borehole Televiewer Decentralization Correction. **Society of Petrophysicists and Well-Log Analysts 35** (4): 14-20, 1994.

MISRA, S., AND HE, J. Shallow Neural Networks and Classification Methods for Approximating the Subsurface *In Situ* Fluid-Filled Pore Size Distribution. **Machine Learning for Subsurface Characterization**: 65, 2019.

MITCHELL, T.M. **Machine Learning**. New York: McGraw-Hill. ISBN: 978-0-07-042807-2, 1997.

MOCZYDLOWER, B., SALOMÃO, M.C., BRANCO, C.C., ROMEU, R.K., HOMEM, T.R., FREITAS, L.C., AND LIMA, H.A. Development of the Brazilian Pre-Salt Fields – When to Pay for Information and When to Pay for Flexibility. *In*: **SPE Latin America and Caribbean Petroleum Engineering Conference**. doi: https://doi.org/10.2118/152860-MS, 2012

MOGHADASI, L., RANAEE, E., INZOLI, F., AND GUADAGNINI, A. Petrophysical Well Log Analysis through Intelligent Methods. *In*: **SPE Bergen One Day Seminar**. doi:10.2118/185922-MS, 2017.

MOURA, P., LABER, E., LOPES, H., MESEJO, D., PAVANELLI, L., JARDIM, J., THIESEN, F., AND PUJOL, G. LSHSIM: A Locality Sensitive Hashing Based Method for Multiple-point Geostatistics. **Computers & Geosciences 107**. 49-60. doi: 10.1016/j.cageo.2017.06.013, 2017.

MOHAGHEGH, S.D. Recent Developments in Application of Artificial Intelligence in Petroleum Engineering. **Journal of Petroleum Technology 57** (04): 86-91. doi:10.2118/89033-JPT, 2005.

O'SHEA, K., AND NASH, R. An Introduction to Convolutional Neural Networks. **arXiv preprint** arXiv:1511.08458. https://arxiv.org/abs/1511.08458, 2015.

PEDREGOSA, F., VAROQUAUX, G., GRAMFORT, A., MICHEL, V., THIRION, B., GRISEL, O., BLONDEL, M., PRETTENHOFER, P., WEISS, R., DUBOURG, V., VANDERPLAS, J., PASSOS, A., COURNAPEAU, D., BRUCHER, M., PERROT, M., AND DUCHESNAY, E. Scikit-Learn: Machine Learning in Python. **Journal of Machine Learning Research 12**. 2825-2830. Arxiv: 1201.0490, 2011.

ROSA, A.J, CARVALHO, R.S., XAVIER, J.A.D. **Engenharia de Reservatórios de Petróleo**. Rio de Janeiro: Interciência. ISBN: 978-8-571-93135-0, 2006.

RUMELHART, D.E., HINTON, G.E., WILLIAMS, R.J. Learning Representations by Back-Propagating Errors. **Nature 323**. (6088): 533-536. doi: 10.1038/323533a0, 1986.

VAN ROSSUM, G. Python tutorial. **Technical Report CS-R9526**. Centrum voor Wiskunde en Informatica (CWI), 1995.

SCHÖLKOPF, B., AND SMOLA, A.J. **Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond**. London: MIT press. ISBN: 978-0-262-19475-4, 2002.

STEWART, G., WITTMANN, M. J., AND LEFEVRE, D. Well Performance Analysis: A Synergetic Approach to Dynamic Reservoir Description. *In*: **SPE Annual Technical Conference and Exhibition**. doi:10.2118/10209-MS, 1981.

SULLIVAN, M. J. Permeability From Production Logs-Method and Application. **Journal of Petroleum Technology 59** (07): 80-87. doi:10.2118/103486-JPT, 2007.

TIAN, C., AND HORNE, R.N. Recurrent Neural Networks for Permanent Downhole Gauge Data Analysis. *In*: **SPE Annual Technical Conference and Exhibition.** doi: 10.2118/187181-MS, 2017

TIMUR, A. Effective Porosity and Permeability of Sandstones Investigated Through Nuclear Magnetic Resonance Principles. *In*: **Annual Logging Symposium**. SPWLA-1968-K, 1968.

VANDERPLAATS, G.N., **Numerical Optimization Techniques for Engineering Design**. New York: McGraw Hill. 1984

VAPNIK, V., AND CHERVONENKIS, A. A Note on One Class of Perceptrons. **Automation and Remote Control 25**, 1964.

VAPNIK, V., AND CHERVONENKIS, A. The Necessary and Sufficient Conditions for Consistency in the Empirical Risk Minimization Method. **Pattern Recognition and Image Analysi**s **1**, (03): 283-305. 1991.

VAPNIK, V. **The Nature of Statistical Learning Theory**. New York: Springer. ISBN: 978-1-4757-3264-1, 1995.

WOLFE, P. A Duality Theorem for Non-Linear Programming. **Quarterly of Applied Mathematics 19.** (3): 239–244. doi: 10.1090/qam/135625, 1961.

WONG, P.M., JANG, M., CHO, S. AND GEDEON, T.D. Multiple Permeability Predictions Using an Observational Learning Algorithm. **Computers and Geosciences 26** (8): 907-913. doi:10.1016/S0098-3004(00)00026-1, 2000.

A
Adopted Unit System and Conversion Factors

Table A.1: Adopted Unit System: Oilfield Units

Quantity Name	Unit
Length	ft
Width	in
Mass	lb
Volume	bbl
Liquid Rate	bbl/d
Pressure	psi
Productivity Index (PI)	$bbl/d/psi$
Permeability	mD
Time	h
Viscosity	cP

Table A.2: Conversion Factors

Constant	SI	Oil Field
C_1	1.0	0.00026374
C_2	$1/2\pi$	141.2

B
Karush-Kuhn-Tucker Conditions

For the classical optimization problem defined by

$$\underset{\boldsymbol{x}}{\text{minimize}}\, f(\boldsymbol{x}) \tag{B.1}$$

$$\text{subject to} \quad g_i(\boldsymbol{x}) \leq 0, \tag{B.2}$$

$$h_j(\boldsymbol{x}) = 0, \tag{B.3}$$

where $i = \{1,2, \dots, n\}$ and $j = \{1,2, \dots, m\}$, the corresponding Lagrangian function is defined as

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\mu}, \boldsymbol{\lambda}) = f(x) + \sum_{i=1}^{n} \mu_i g_i(\boldsymbol{x}) + \sum_{j=1}^{m} \lambda_j h_j(\boldsymbol{x}). \tag{B.4}$$

The Karush-Kuhn-Tucker necessary conditions for optimality can be categorized into four groups: primal feasibility, positivity condition, stationary condition, and complementary condition.

B.1.
Primal Feasibility

$$g_i(\boldsymbol{x}^*) \leq 0 \tag{B.5}$$

$$h_j(\boldsymbol{x}^*) = 0 \tag{B.6}$$

B.2.
Positivity Condition

$$\mu_i^* \geq 0 \tag{B.7}$$

B.3.
Stationary Condition

$$\nabla_x f(\boldsymbol{x}^*) + \sum_{i=1}^{n} \mu_i^* \nabla g_i(\boldsymbol{x}^*) + \sum_{j=1}^{m} \lambda_j^* \nabla h_j(\boldsymbol{x}^*) = 0 \tag{B.8}$$

B.4.
Complementary Condition

$$\mu_i^* g_i(\boldsymbol{x}^*) = 0 \tag{B.9}$$

C
List of Functions for Incorporating Non-Linearities to Machine Learning Models

C.1.
Support Vector Regression: Kernel Functions

Table C.1: List of Examples of Kernel Functions

Kernel	$k(\boldsymbol{u}, \boldsymbol{v})$
Linear	$\boldsymbol{u}^{\boldsymbol{t}}\boldsymbol{v} + c$
Polynomial	$(\boldsymbol{u}^{\boldsymbol{t}}\boldsymbol{v} + c)^d$
Gaussian	$\exp\left(-\frac{\|\boldsymbol{u}-\boldsymbol{v}\|^2}{2\sigma^2}\right)$
Exponential	$\exp\left(-\frac{\|\boldsymbol{u}-\boldsymbol{v}\|}{2\sigma^2}\right)$
Hyperbolic Tangent (Sigmoid)	$\tanh(\boldsymbol{u}^{\boldsymbol{t}}\boldsymbol{v} + c)$
Power	$-\|\boldsymbol{u}-\boldsymbol{v}\|^d$

C.2.
Artificial Neural Network: Activation Functions

Table C.2: List of Examples of Activation Functions

Activation Function	$\varphi(x)$
ReLu	$f(x) = max(0, x)$
Logistic (Sigmoid)	$f(x) = \frac{1}{1+e^{-x}}$
Hyperbolic Tangent (Sigmoid)	$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$
Softsign	$f(x) = \left(\frac{x}{\lvert x\rvert + 1}\right)$
Exponential Linear Units	$f(x) = \begin{cases} x, & x > 0 \\ e^x, & x \leq 0 \end{cases}$