Pontifícia Universidade Católica
do Rio de Janeiro

Bruno Costa Pontes

Machine Learning Applications on Pressure and Temperature Data from Oil Well Sensors

Dissertação de Mestrado

Dissertation presented to the Programa de Pós–graduação em Mecânica, do Departamento de Mecânica of PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Mecânica.

Advisor : Prof. Márcio da Silveira Carvalho
Co-advisor: Dr. Jonatas dos Santos Grosman

Rio de Janeiro
September 2025

Pontifícia Universidade Católica
do Rio de Janeiro

Bruno Costa Pontes

Machine Learning Applications on Pressure and Temperature Data from Oil Well Sensors

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

Prof. Márcio da Silveira Carvalho
Advisor
Departamento de Engenharia Mecânica – PUC-Rio

Dr. Jonatas dos Santos Grosman
Co-advisor
Departamento de Informática – PUC-Rio

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

Dr. Emilio Jose Rocha Coutinho
PETROBRAS

Rio de Janeiro, September 16th, 2025

Bruno Costa Pontes

The author graduated in Aeronautical Engineering from ITA (Instituto Tecnológico de Aeronáutica) - 2010. 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

Costa Pontes, Bruno

Machine Learning Applications on Pressure and Temperature Data from Oil Well Sensors / Bruno Costa Pontes; advisor: Márcio da Silveira Carvalho; co-advisor: Jonatas dos Santos Grosman. – 2025.

101 f: il. color. ; 30 cm

Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Mecânica, 2025.

Inclui bibliografia

1. Engenharia Mecânica – Teses. 2. Aprendizado de máquina. 3. Aprendizado supervisionado. 4. Aprendizado não supervisionado. I. Carvalho, Márcio da Silveira. II. Grosman, Jonatas dos Santos. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Mecânica. IV. Título.

CDD: 621

Acknowledgments

I would like to express my deepest gratitude to my wife Lis and my daughter Marina for their unwavering patience, love, and support throughout my personal and academic journey.

To my mother, Pollenya, my father, Iran, and my brother, Victor, whose constant belief in me has been a profound source of inspiration.

I am sincerely thankful to my advisor, Prof. Márcio Carvalho, and my co-advisor, Prof. Jonatas Grosman, for their invaluable guidance and encouragement throughout this work.

My thanks also go to Pedro Nogueira and Santiago Toledo, my managers during this period, for granting me the time and flexibility needed to dedicate myself fully to this project. I am grateful to Lourenço Keuper and my colleagues at Petrobras, whose support with work-related matters allowed me to focus on this research.

I extend my appreciation to Emilio Coutinho and Prof. Abelardo Barreto Jr. for kindly serving on the examination committee and for their insightful contributions that enhanced the quality of this work.

To Priscila Ribeiro, thank you for your support at the beginning of this journey. I also thank Helon Ayala for providing the foundational mindset and technical framework essential for advancing in this field.

My sincere thanks to Petrobras for the opportunity to participate in the master's program and for providing the field data analyzed in this research.

I am grateful to the Department of Mechanical Engineering at PUC-RJ for their support during the development of this study.

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

Abstract

Costa Pontes, Bruno; Carvalho, Márcio da Silveira (Advisor); Grosman, Jonatas dos Santos (Co-Advisor). **Machine Learning Applications on Pressure and Temperature Data from Oil Well Sensors**. Rio de Janeiro, 2025. 101p. Dissertação de Mestrado – Departamento de Mecânica, Pontifícia Universidade Católica do Rio de Janeiro.

This master's thesis investigates the application of machine learning techniques to pressure and temperature data collected from permanent downhole sensors in multi-zone oil wells of the Brazilian Pre-salt carbonate reservoirs. These reservoirs feature complex architectures with intelligent completions that enable independent control and monitoring of multiple production zones, posing challenges for traditional physical modeling due to heterogeneous reservoir properties and interdependent flow dynamics.

The research addresses the gap in existing studies, which mostly focus on single-zone wells, by applying supervised and unsupervised learning methods to a comprehensive dataset obtained from selective production tests of two wells with similar multi-zone completions. Clustering algorithms were used to identify patterns related to valve configurations, which supported the development of classification models to accurately predict valve status from sensor data. Regression models, enhanced by segmenting data according to zone combinations, effectively estimated total oil production rates.

Model assessment across different wells made it possible to discuss robustness and generalizability, highlighting the importance of incorporating pressure- and time-dependent features, proper data normalization, and temporal cross-validation techniques. The results confirm that machine learning can successfully extract valuable information from complex sensor data, automate well monitoring, and improve decision-making processes in challenging multi-zone reservoir environments. This work contributes to advancing data-driven approaches for modern reservoir management, demonstrating their potential to complement and enhance traditional physical modeling in oil and gas production.

Keywords

Machine learning; Supervised learning; Unsupervised learning.

Resumo

Costa Pontes, Bruno; Carvalho, Márcio da Silveira; Grosman, Jonatas dos Santos. **Aplicação de aprendizado de máquina a dados de pressão e temperatura de sensores de poços de petróleo**. Rio de Janeiro, 2025. 101p. Dissertação de Mestrado – Departamento de Mecânica, Pontifícia Universidade Católica do Rio de Janeiro.

Esta dissertação de mestrado investiga a aplicação de técnicas de aprendizado de máquina em dados de pressão e temperatura coletados por sensores permanentes em poços multi-zona dos reservatórios carbonáticos do Pré-Sal brasileiro. Esses reservatórios possuem arquitetura complexa e são equipados com completação inteligente que permite o controle e monitoramento independentes das zonas de produção. Essa configuração apresenta desafios para modelagem física tradicional devido à heterogeneidade do reservatório e à dinâmica de fluxo interdependente das zonas.

A pesquisa aborda uma lacuna existente, uma vez que a maioria dos estudos foca em poços de zona única, aplicando métodos supervisionados e não supervisionados a um conjunto de dados obtidos em testes seletivos de produção de dois poços situados no mesmo reservatório e com completação similar. Algoritmos de clusterização foram utilizados para identificar padrões relacionados às configurações das válvulas, o que apoiou o desenvolvimento de modelos de classificação capazes de prever o status das válvulas a partir dos dados dos sensores. Modelos de regressão, aprimorados pela segmentação dos dados usando a classificação das combinações de zonas, estimaram efetivamente a vazão total de produção de óleo.

Avaliação dos modelos em diferentes poços permitiu discutir a robustez e capacidade de generalização, ressaltando a importância da incorporação de características dependentes da pressão e do tempo, normalização adequada dos dados e técnicas de validação temporal. Os resultados confirmam que o aprendizado de máquina pode extrair informação útil a partir de dados complexos de sensores, automatizar o monitoramento dos poços e aprimorar os processos de tomada de decisão em ambientes desafiadores de reservatórios complexos. Este trabalho contribui para o avanço da abordagem orientada a dados na gestão de reservatórios, demonstrando potencial para complementar e fortalecer a modelagem física tradicional na produção de óleo e gás.

Palavras-chave

Aprendizado de máquina; Aprendizado supervisionado; Aprendizado não supervisionado.

Table of contents

List of figures

List of tables

List of Abreviations

AI – Artificial Intelligence

CV – Cross-Validation

DTC – Decision Tree Classifier

DTR – Decision Tree Regression

GMM – Gaussian Mixture Models

GOR – Gas-Oil Ratio

ICV – Inflow Control Valve

II – Injectivity Index

k-NN – k-Nearest Neighbors

ML – Machine Learning

NPV – Net Present Value

PCA – Principal Component Analysis

PDG – Permanent Down-hole Gauges

PI – Productivity Index

PTA – Pressure Transient Analysis

RFC – Random Forest Classifier

SVC – Support Vector Classifier

SVM – Support Vector Machine

SVR – Support Vector Regression

t-SNE – t-distributed Stochastic Neighbor Embedding

VFM – Virtual Flow Metering

One of the chief lessons of history is that many of the things that we consider natural and eternal are, in fact, man-made and mutable.

Yuval Noah Harari, *Nexus: A Brief History of Information Networks from the Stone Age to AI.*

1
Introduction

1.1
Introduction

The Brazilian Pre-Salt province has emerged as one of the most prolific oil and gas exploration frontiers in the world, driven by the development of ultra-deep-water carbonate reservoirs. These reservoirs are characterized by substantial thickness and high oil productivity per well, which presents both opportunities and challenges for efficient resource management. As reservoir complexity increases, optimizing production becomes a central concern for operators. One of the key strategies to achieve this optimization is the subdivision of reservoirs into multiple production zones, enabled by advanced well completions known as intelligent completions (SCHNITZLER et al., 2021) and (SORTICA et al., 2023). These completions allow operators to control and monitor production from different zones independently, thereby enhancing recovery and operational flexibility.

A typical well configuration is schematically illustrated in Figure 1.1.

In Figure 1.1 S represents the sensors, the fluid exits the production zones

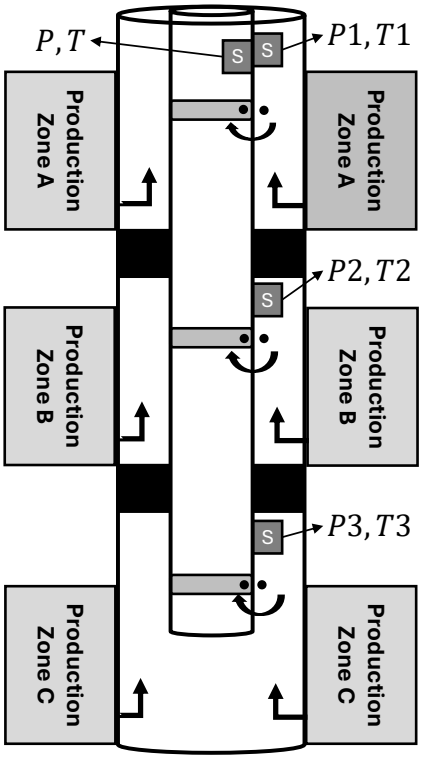

Figure 1.1: Typical Well Configuration with 3 Production Zones.

and passes through the Inflow Control Valves (ICVs), represented in gray, into the column.

Among the various parameters monitored to ensure optimal production, pressure and temperature stand out as fundamental indicators of reservoir and well behavior. Accurate and timely readings of these parameters are essential for understanding flow dynamics, diagnosing operational issues, and making informed decisions about well management. However, the development of reliable physical models to interpret sensor data and predict production behavior in such complex environments is often fraught with difficulties. The heterogeneous nature of carbonate reservoirs, coupled with the intricate flow patterns in multi-zone wells, can lead to physical models that are computationally intensive and sometimes unable to capture all relevant phenomena.

In recent years, the widespread deployment of permanent downhole sensors has generated large volumes of high-resolution pressure and temperature data. These sensors provide continuous monitoring of well conditions, offering a rich source of information that can potentially be used for improving reservoir management. However, the sheer volume and complexity of the data pose significant challenges for traditional analysis techniques, prompting the need for more advanced approaches capable of extracting actionable insights from sensor measurements.

Using machine learning, it is possible to uncover hidden relationships in sensor data, automate the interpretation of complex signals, and improve decision-making processes. Although physical modeling remains an important avenue of research, alternative approaches based on machine learning are increasingly being explored and developed to complement and, in some cases, surpass traditional methods.

This thesis focuses on the application of machine learning techniques to pressure and temperature sensor data from wells located in the Brazilian Pre-Salt. The objective is to demonstrate how these methods can be used to extract meaningful information, classify operational states, and predict production-related variables in a complex multi-zone environment. Other complementary approaches, including those based on well physical modeling, are being investigated in parallel by other research groups and are discussed in the literature review section. The results presented here aim to contribute to the growing body of knowledge at the intersection of digital technologies and oilfield operations, highlighting the transformative potential of machine learning for modern reservoir management.

1.2 Problem Description

The data set utilized in this work comprises real operational data collected during selective production tests of two wells within the same carbonate reservoir, both featuring similar intelligent completion designs. The data includes time series of pressure and temperature readings from all sensor locations, valve status records, and oil and gas production rates recorded at one-minute intervals.

Selective tests are especially valuable for machine learning applications because valve movements and operational states are systematically controlled and thoroughly documented, ensuring high data reliability. As a result, these periods of operation serve as the primary source of the training data set.

In Figures 1.2 and 1.3, the sensors locations are indicated as follows:

- No Number: Sensor in the tubing, measuring the total flow.
- 1: Sensor in the annulus of the upper zone.
- 2: Sensor in the annulus of the intermediate zone.
- 3: Sensor in the annulus of the lower zone.

For Well A (see Figure 1.2), the selective test covered all possible combinations of zones producing, allowing for a comprehensive data set that captures the interactions between simultaneous zone production. This provides a rich context for learning patterns associated with multi-zone flow behavior and valve status.

In contrast, Well B (see Figure 1.3) was tested by producing each zone individually. This approach yields clear data for the contribution of each zone to the overall production, facilitating the identification of zone-specific sensor responses and production characteristics.

1.3 Motivation

The data set derived from Brazilian Pre-Salt wells presents unique analytical challenges due to its multi-zone completion architecture and advanced sensor instrumentation. In this configuration, pressure and temperature readings from multiple zones are inherently interdependent, as each sensor is affected both by local reservoir conditions and by the combined flow dynamics of the entire well. This complexity complicates the interpretation of sensor data, making it difficult to isolate the influence of individual zones.

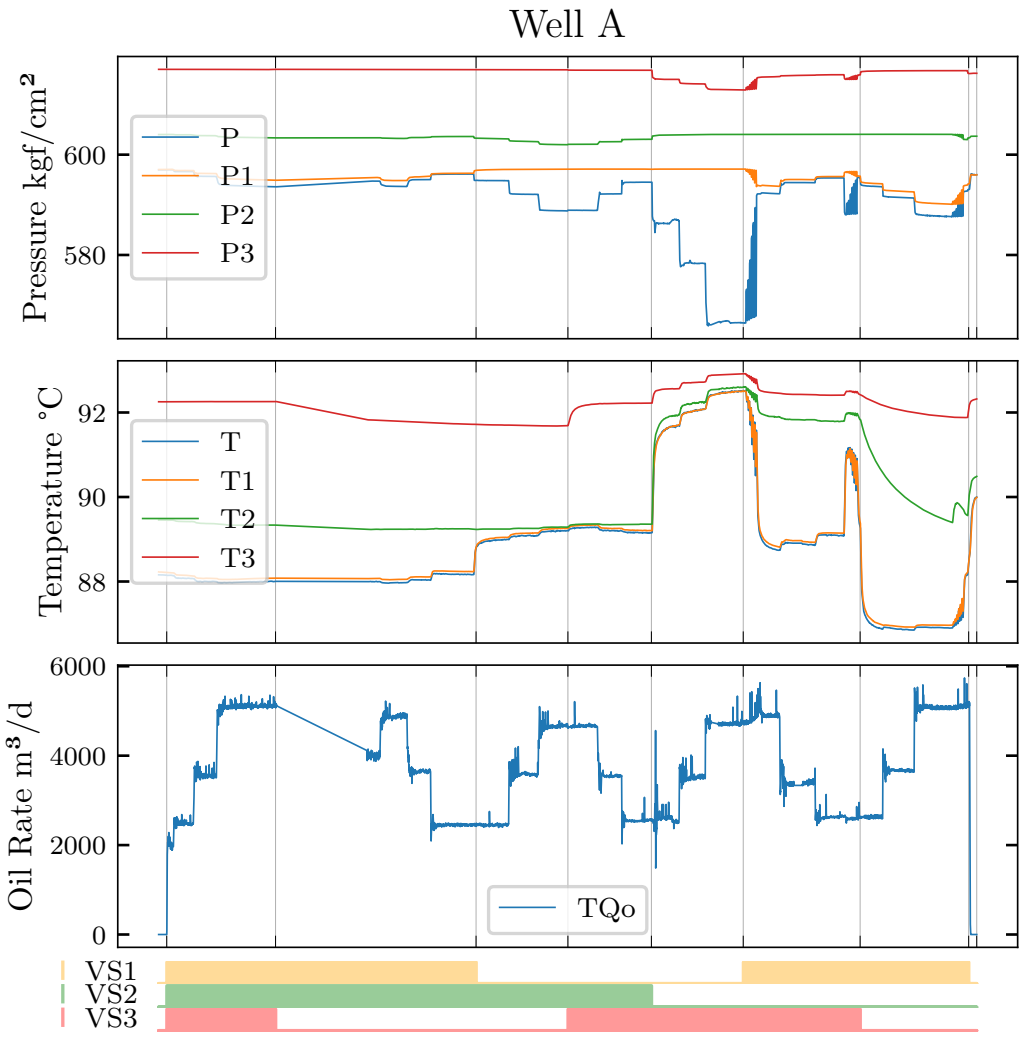

Figure 1.2: Pressure, Temperature, Flow Rate and Valve Status data set for Well A. The bottom part of the figure shows the actual status of valve openings in each zone represented by colored areas when the valves are open.

Additionally, the use of multiple Inflow Control Valves (ICVs) further increases operational complexity. Accurate identification of the status of each valve and understanding its direct impact on oil production are crucial for effective reservoir management. Dynamic interactions between zones, valves, and flow rates result in overlapping sensor signals and highly variable operational states.

Traditional modeling approaches struggle to address these intricacies, especially when attempting to predict valve status or allocate production to specific zones. Robust models are needed to navigate the variability and ambiguity introduced by commingled production and intelligent completions.

Although there is a substantial body of research on oil rate prediction using machine learning, most existing studies focus on wells with simple

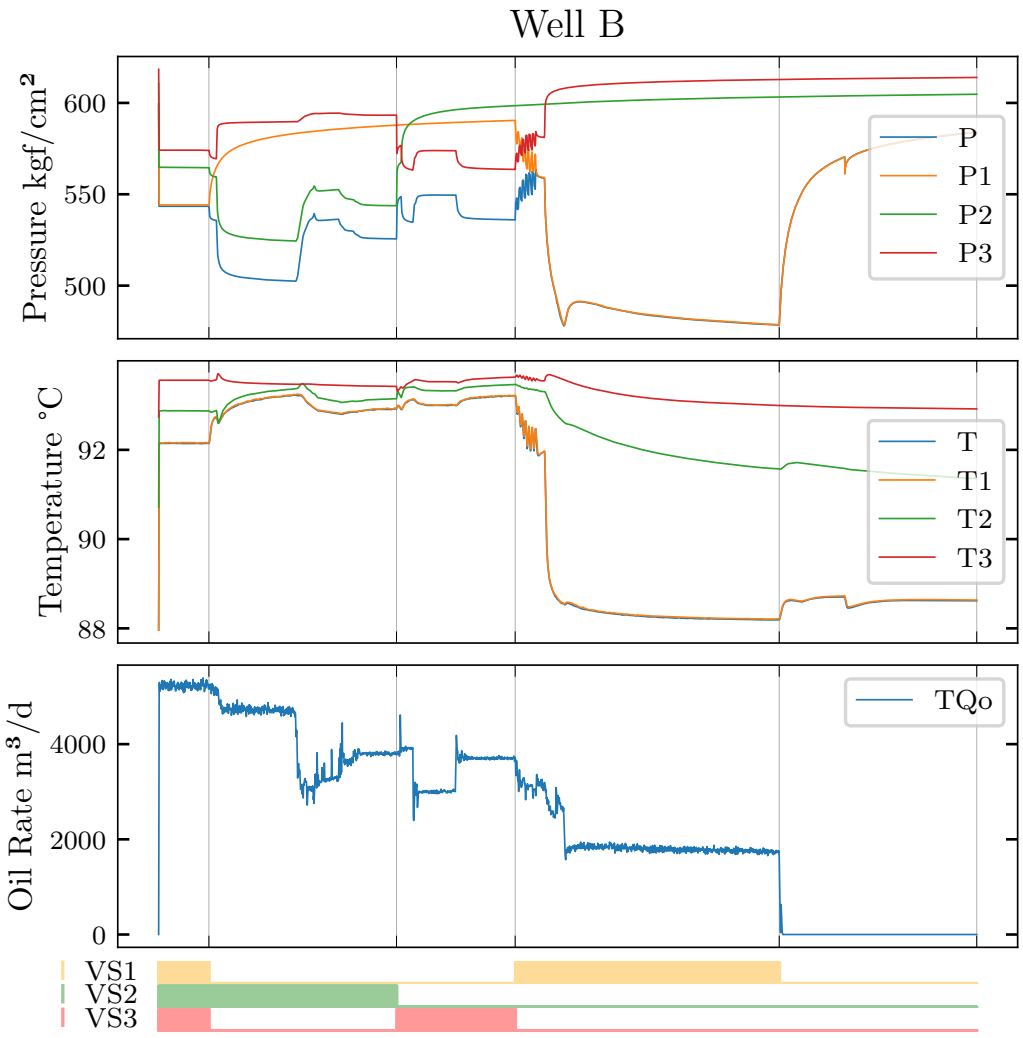

Figure 1.3: Pressure, Temperature, Flow Rate and Valve Status data set for Well B. The bottom part of the figure shows the actual status of valve openings in each zone represented by colored areas when the valves are open.

completions and single-zone production. These approaches do not account for the challenges associated with multi-zone wells, which are increasingly common in Pre-Salt operations. The complexity of commingled production in intelligent completions demands new methodologies capable of extracting meaningful information from more complicated data sets.

This research is motivated by the need to advance the application of machine learning in realistic, multi-zone scenarios. Using supervised and unsupervised learning techniques, this study aims to systematically extract valuable insights, classify operational states, and predict key production variables from complex sensor data. The goal is to bridge the gap between existing machine learning approaches and the demands of modern Pre-Salt reservoir management, contributing to the development of more effective data-driven solutions

for the oil and gas industry.

1.4 Research Objectives

The primary objective of this thesis is to explore and demonstrate the capabilities of machine learning methodologies when applied to complex, real-world sensor data from multi-zone oil wells in the Brazilian Pre-Salt, particularly those equipped with intelligent completions. This study aims to address the analytical challenges inherent in these advanced well architectures and to develop robust data-driven approaches for monitoring and optimizing oil production.

Within this framework, an initial objective was to perform production allocation among three distinct production zones using purely data-driven models. However, a significant challenge associated with this approach is the scarcity of labeled datasets, which constrains the ability of machine learning models to fully capture the underlying production dynamics. This limitation highlights the necessity of incorporating physical modeling techniques to improve the accuracy of allocation. Considering that another goal of this work was to investigate machine learning methods, the study focused on using the data readily available in the data set. As identified in the review of the literature, future research may overcome these challenges by adopting hybrid approaches that integrate data-driven techniques with physical modeling, potentially achieving more robust and accurate production allocation.

1.5 Dissertation Outline

This dissertation is organized into five chapters, each addressing a distinct aspect of the research and guiding the reader through the development and application of machine learning techniques to sensor data from multi-zone oil wells.

Chapter 1 – Introduction The introductory chapter sets the stage by presenting the context and motivation behind the research, describing the problem and analytical challenges associated with multi-zone intelligent completions, and outlining the main objectives of the study. It also introduces the data set used and highlights the relevance of applying machine learning to complex oilfield sensor data.

Chapter 2 – Literature Review This chapter provides an in-depth review of the relevant literature, focusing on well status identification, oil rate prediction, production allocation and subsurface characterization. It discusses existing approaches, their limitations in the context of multi-zone wells, and positions the current research within the broader field of machine learning applications in oil and gas production.

Chapter 3 – Methodology and Work Proposal The third chapter presents the foundational concepts of machine learning, including both supervised and unsupervised learning techniques. Details the specific methodological approach adopted in this study and justifies the selection of algorithms and data processing strategies. This chapter also describes the preparation of the data set, feature engineering, and the rationale behind the modeling choices for clustering, classification, and regression tasks.

Chapter 4 – Results and Discussion In this chapter, the results obtained from applying the developed methodology to the data set are presented and analyzed. The performance of clustering algorithms in identifying valve patterns, the effectiveness of classification models in predicting valve status, and the precision of regression models for oil rate estimation are discussed. The chapter also examines the impact of feature selection, data normalization, and temporal data handling on model performance and generalizability.

Chapter 5 – Conclusion The final chapter summarizes the key findings and contributions of the research. It reflects on the implications of the results for production optimization in multi-zone wells and provides recommendations for future research directions, including the potential integration of physical modeling, application of advanced deep learning architectures, and strategies for production allocation from each reservoir zone.

2
Literature Review

The oil and gas industry has always been at the forefront of technological innovation, leveraging advances in engineering and science to optimize exploration, production, and reservoir management. In recent years, the proliferation of digital sensors and the advent of big data analytics have opened new frontiers for operational efficiency and decision making.

(PóVOAS et al., 2025) provides an overview of the application of artificial intelligence (AI) in the oil and gas sector, highlighting applications in areas such as process optimization, predictive maintenance, reservoir management, image and video analysis, and operational safety.

(TARIQ et al., 2021) focuses on ML applications in petroleum engineering, such as prediction of well performance and the use of ML to accelerate oil reservoir simulations. It points to the challenge of access to field data, suggesting that oil companies should share their data to benefit from AI.

(BALAJI et al., 2018) highlights some ML techniques with specific applications and discusses the acceptance of data-driven methods in the oil industry, emphasizing the difficulty in valuing data-driven methods that are not part of the traditional engineering curriculum.

(SIRCAR et al., 2021) focuses on upstream applications and points out that AI solutions are not generic, they must be customized to the business context and database of each company that needs an internal team of specialists.

In the specific context of the use of ML in sensor data, previous articles highlighted well status identification, oil rate prediction, subsurface characterization, and production allocation as applications of interest. The next sections will detail the research on these topics and provide guidance for the applications developed in the thesis. The sections review a selection of recent articles that apply ML and data-driven approaches to reservoir engineering. Together, these works demonstrate the growing impact of ML, offering robust solutions for complex and dynamic production environments.

It is important to note that the majority of these works focus on wells with relatively simple completions, typically featuring only a single production zone. This contrasts with the context of the present thesis, which investigates a well with a more complex completion of three zones, presenting additional challenges and opportunities for the application of advanced ML methods to model and optimize commingled reservoir production.

2.1 Well Status Identification

Accurate identification of well status plays a crucial role in oil and gas production operations, as it enables the timely detection of spurious or unintended valve movements that could compromise the integrity of the system and operational safety. Furthermore, knowing the precise status of the valves serves as an important auxiliary variable in the estimation of the flow rate, directly affecting the reliability of virtual flow metering and other data-driven monitoring solutions (KADEM et al., 2024). By ensuring that the valve status is correctly identified and integrated into analysis frameworks, operators can significantly improve flow modeling accuracy and improve overall system control.

(KADEM et al., 2024) explores linear regression, logistic regression, random forest, decision tree, support vector machines, Gradient Boosting Machine and artificial neural networks to the problem of prediction of well status. The Random Forest algorithm was found to be the most accurate technique.

(KARIMI et al., 2025) uses multiple architectures of the One-Dimensional Convolutional Neural Network (1D CNN) and Extreme Gradient Boosting (XGBoost) together with hyperparameter tuning to obtain automated identification of the well status. The models studied demonstrated similar accuracy. The unification of data from multiple wells demonstrated sufficient accuracy to support the evaluation of new wells.

The (ALJUBRAN; HORNE, 2020) study explores the use of ML, specifically fully connected neural networks, to analyze and predict well performance based on various ICV settings to maximize well output. The study highlights the potential of reinforcement learning approaches that rely on real-time well feedback and production measurements to estimate well output under varying ICV settings.

The (ALJUBRAN; HORNE, 2021) work investigates a surrogate-based optimization algorithm designed to minimize the number of required field tests of ICV, predict the well performance for unseen combinations of ICV settings, and determine optimal ICV configurations in conjunction with the estimation of the net present value (NPV). The algorithm successfully predicts surface and subsurface flow profiles and optimizes the ICV settings, requiring only six and eleven field tests to achieve 80% and 90% R^2, respectively, for both surface and subsurface flow predictions.

2.2 Oil Rate Prediction

One prominent application of pressure and temperature sensor data is Virtual Flow Metering (VFM), where ML models estimate multiphase flow rates, such as oil, gas, and water, using sensor readings instead of direct measurement devices (BIKMUKHAMETOV; JäSCHKE, 2020b). Using continuous data streams from downhole and surface sensors, VFM enables real-time inference of production rates. This reduces dependence on expensive and often impractical hardware flow meters, allowing for scalable and cost-effective flow monitoring across extensive production networks. As a result, VFM enhances operational flexibility and supports more informed and timely decision making in reservoir management.

(BIKMUKHAMETOV; JäSCHKE, 2020a) highlights that while data-driven VFM offers significant advantages in speed and cost, future research should focus on improving robustness, uncertainty quantification, and integration of first-principles knowledge for hybrid modeling. The combination of dynamic state estimation, machine learning, and first-principles models holds promise to advance the accuracy of VFM, especially under transient flow conditions. However, challenges remain in implementing these advanced methods reliably in practical settings.

The (GRYZLOV et al., 2020) work explores virtual flow metering (VFM) using ML techniques, including dynamic mode decomposition and deep LSTM neural networks, to analyze pressure, temperature, choke position and ESP parameters. These methods enable the estimation of individual oil, gas and water flow rates from multiple reservoirs without the need for physical models, offering flexible and accurate production forecasting capabilities.

The (NEGASH; HIM, 2020) study proposes a unified physics and data-driven analytics workflow to reconstruct missing gas, oil, and water flow rates in reservoir data, evaluated using real field data from a North Sea reservoir. The approach integrates domain knowledge of fluid flow physics in porous media to generate new features for training ML models. Model validation is conducted using statistical tests and a novel physics-based validation involving material balance and pressure back-calculation.

The (GRYZLOV; SAFONOV; ARSALAN, 2022) article presents the application of continuous deep learning models, specifically latent Ordinary Differential Equation (ODE) models, to predict multiphase hydrocarbon production rates as time series. Despite requiring significant training time, Latent ODEs outperform other methods for forecasting multiphase flow rates in scenarios with irregular time steps due to their continuous nature, making them

particularly well-suited for production monitoring in dynamic reservoir environments.

In (NAGAO et al., 2023), a hybrid approach combines reduced physics models with neural networks to predict multiphase production rates and identify reservoir connectivity. By integrating routine injection and production data with pressure measurements, the method achieves robust forecasting performance without detailed physical modeling. Hybrid models consistently outperform purely data-driven ML methods, highlighting the value of incorporating physical insights.

In (TIAN; HORNE, 2017), the use of recurrent neural networks is studied to process PDG data. Operational variations introduce significant noise into PDG records. This noise is considered an inherent aspect of PDG data, which complicates analysis and interpretation. The work shows that RNNs present the noise tolerance and computational efficiency needed to deal with PDG data in practice.

The (TIAN; HORNE, 2019) work applies three different ML techniques to the interpretation of flow-rate, pressure, and temperature data from PDGs, aiming to develop robust methods for data interpretation and to explore the utility of temperature measurements. In general, ML models demonstrate the ability to model complex temperature and pressure data from PDGs, with KRR offering a balance between bias and variance, and the nonlinear relationship between flow rate and pressure being effectively captured through feature engineering.

Demonstrating the utility of deep learning, specifically LSTM networks, (WANG et al., 2021) work shows that flow rates can be accurately predicted from the downhole temperature and pressure data. The approach can be extended to separate flow rates from different reservoirs by training models on labeled data sets. In particular, LSTM models outperform other ML methods in predicting flow rates, supporting the adoption of advanced neural architectures for reservoir analysis.

The structured framework for the development of data-driven prediction models for the collection, processing and modeling of sensor data to estimate flow rates is described in (BIKMUKHAMETOV; JäSCHKE, 2020b) and (BELYADI, 2021). Feature engineering emerges as a key strategy, enabling the modeling of nonlinear relationships inherent in PDG data by constructing informative features from raw sensor measurements (LIU; HORNE, 2013). A significant insight drawn is the importance of incorporating temporal dynamics into model design (BIKMUKHAMETOV; JäSCHKE, 2020b). Taking into account time, either through dynamic features or by leveraging historical

measurements, data-driven approaches can better capture transient system behaviors and enhance prediction accuracy.

In addition, the literature highlights the promising role of temperature data, which can complement or, in some cases, substitute for pressure and flow rate measurements, expanding the scope and robustness of data-driven reservoir analysis (TIAN; HORNE, 2019).

The reviewed articles also offer important perspectives on the evaluation of flow rate regression models, detailing the use of robust metrics, considering mean squared error, mean absolute error, and R^2 in conjunction to assess prediction performance and quantify accuracy levels (OLAMIGOKE; ONYEALI, 2022).

2.3 Production Allocation

Another significant application of pressure and temperature sensor data, empowered by ML, is production allocation. In complex oil and gas fields, particularly those with combined production from multiple wells or reservoirs, accurately determining the contribution of each source to total production is a challenging task. Traditionally, production allocation relies on periodic well testing and manual calculations, which can be time consuming and prone to uncertainty.

Using continuous sensor data and advanced ML algorithms, it becomes possible to perform real-time or near-real-time allocation, inferring individual well or zone contributions based on observed changes in pressure, temperature, and other process variables. This not only improves the accuracy and operational efficiency of allocation, but also enhances reservoir management and financial reporting, as operators can more precisely track production performance and optimize resource utilization (KADEM et al., 2024).

The (DIASO et al., 2023) study presents the application of ML algorithms to analyze real-time data from pressure and temperature sensors, generating transient production rates for each reservoir. Using data-driven techniques, the allocation factors for production strings are determined without relying on explicit physical models. The results demonstrate a high level of confidence in the allocation output, indicating the practical viability of this approach for real-time reservoir management.

The (BARRETT et al., 2012) paper introduces an algorithm to determine gas flow rates from measured pressure and temperature profiles along the well, going beyond the reliance on physical models. Using a non-isothermal gas flow mathematical model, it accurately calculates the rate and thermal conductivity

profiles, with results showing good agreement between measured and computed rates, indicating potential for adaptation to ML applications.

The (MCCRACKEN; CHORNEYKO, 2006) work advocates for the use of permanent downhole pressure data to calculate bottomhole flow rates for each reservoir layer, applying Darcy's law. The method allows for accurate rate allocation without the need for physical models or extensive well interventions, contributing to improved reservoir management and operational efficiency.

The work (WU; HUMPHREY; LIAO, 2012) suggests the development of user-defined models and ML algorithms, grounded in real-time pressure and temperature sensor data, to analyze historical trends and correlations. The proposed methodology enables an effective rate allocation among different reservoirs in intelligent wells, bypassing the limitations of traditional physical models, and improving the precision of production monitoring.

2.4 Subsurface Characterization

Pressure and temperature data also offer information on reservoir compartmentalization. Subtle changes in sensor readings, correlated across multiple wells or locations, may indicate the presence of flow barriers, faults, or distinct reservoir compartments. ML techniques such as clustering, anomaly detection, and pattern recognition can help identify these subsurface features, supporting more accurate reservoir models and optimized field development strategies.

In (TIAN; HORNE, 2016) a formula for the connectivity between producers and injectors is developed and ML based multi-well testing is used as a validation of the connectivity estimated by the derived formula.

The (RAMCHARITAR; RAMDHANIE, 2021) study applies unsupervised pattern recognition techniques, such as hierarchical clustering, to legacy production data from individual well completions. The methodology enables the identification of distinct reservoir compartments based solely on production characteristics, eliminating the need for traditional geological workflows or physical models and facilitating effective field compartmentalization analysis.

3
Methodology and Work Proposal

In this chapter, the proposed methodology is described. The work aims to explore Machine Learning techniques in a real oil pressure and temperature sensor data set. The main steps are described, and details and considerations are discussed.

All the computational code written for this thesis was developed in Python version 3.11.0 [1].

3.1
Data Set Description

The data set utilized in this work comprises real operational data collected during selective production tests of two wells within the same carbonate reservoir, both featuring similar intelligent completion designs. The data includes time series of pressure and temperature readings from all sensor and backups, valve status records, and total oil and gas production rates recorded at one minute intervals. The sensors are located inside the tubing and in the annulus of each production zone, upper, intermediate, and lower.

3.2
Exploratory Analysis

Initially, a data cleaning and exploratory analysis was performed on the data set. To provide a comprehensive exploratory analysis of the data, the Python library YData-profiling [2] was used. YData-profiling is a package for data profiling that automates and standardizes the generation of detailed reports, complete with statistics and visualizations. This library provides a straightforward way to describe and analyze the correlations between the variables and to gain insight into the data set.

YData-profiling can also be used for an Exploratory Data Analysis on time-series data. It is useful for understanding the behavior of time-dependent variables regarding behaviors such as time plots, seasonality, trends, stationarity, and data gaps. Combined with the profiling reports, it is possible to compare the evolution and data behavior through time in terms of time-series-specific statistics. It also provides the identification of gaps in the time series caused by missing values or missing entries in the time index.

[1] https://www.python.org/downloads/release/python-3110/
[2] https://docs.profiling.ydata.ai/

An essential coefficient for analyzing correlations is the Pearson Correlation Coefficient (BELYADI, 2021). The coefficient quantifies the linear relationship between two continuous variables. it ranges from -1 to 1, a value close to 1 indicates a strong positive linear correlation, close to -1 indicates a strong negative linear correlation, and close to 0 suggests no linear correlation.

$$P_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} \tag{3-1}$$

where $cov(X,Y)$ is the covariance between the parameters X and Y, σ_X is the standard deviation of parameter X, and σ_Y is the standard deviation of parameter Y.

3.3 Machine Learning

In (BRUNTON; NOACK; KOUMOUTSAKOS, 2020), a general description of Machine Learning is presented. Machine Learning can be understood as a set of algorithms that extract patterns and information from data. The learning problem can be formulated as the process of estimating the associations between inputs, outputs, and parameters of a system. It is divided into three main branches:

- Supervised learning: Learning from labeled data by providing corrective information to the algorithm. If the target is a label, the problem is described as a classification, and if the target is a numerical value, the problem is described as a regression.
- Semi-supervised learning: Learning with partially labeled data or using rewards from the environment to learn (reinforcement learning).
- Unsupervised learning: Learning patterns without labeled training data. The main types of problems are dimensionality reduction and clustering.

3.4 Feature Engineering

Feature engineering is the process of using domain knowledge to create, modify, or select features (input variables) that enhance the performance of machine learning models. It involves transforming raw data into meaningful attributes that can better capture the underlying patterns relevant to the task at hand. In the context of small data sets, feature engineering becomes particularly crucial, as the limited amount of data can make it challenging for models to learn effectively. Small data sets are often prone to overfitting,

where models memorize the training data rather than generalize to new, unseen instances.

By carefully crafting features, such as creating interaction terms, encoding categorical variables, or aggregating data to capture trends, models can be provided with richer, more informative input that helps mitigate the risks of overfitting. Effective feature engineering can lead to improved model performance and robustness, making it a vital step in the machine learning pipeline, especially when working with small data sets where every piece of information counts.

In the following subsections, the features studied in the thesis are detailed.

3.4.1 Pressure and Oil Rate Relationship

In (AJAYI; FASASI; OKUNS, 2012), a simplified version of the analytical flow choke equation derived from principles of energy conservation is given:

$$q = C_d A \sqrt{2 \cdot \frac{\Delta P}{\rho}} \tag{3-2}$$

where q is the volumetric rate, A is the ICV cross-sectional area, ΔP is the pressure drop across ICV, ρ is the fluid density, and C_d is the discharge coefficient.

In (ALJUBRAN; HORNE, 2021), the decrease in pressure caused by flow through a constriction and the decrease in pressure caused by flow along the tubing are given by:

$$\Delta P = \Delta P_c + \Delta P_f = \frac{\rho_m q_m^2}{2 A_c^2 C_v^2} + 2f \frac{L}{D} \rho_m q_m^2 A_p^2 \tag{3-3}$$

where ΔP_c and ΔP_f represent the pressure decrease caused by cross-sectional constriction and lateral friction, respectively, q_m is the fluid mixture volumetric flow rate, ρ_m is the fluid mixture density, A_c and A_p are the areas of the ICV constriction and pipe, respectively, f is the Fanning friction factor, C_v is the dimensionless valve geometry coefficient, L is a characteristic length and D is the pipe diameter.

Based on these quadratic relationships between delta pressures and volumetric rate, the following features were built.

$$rdP_1P = \sqrt{|(P_1 - P)|} \tag{3-4}$$

$$rdP_2P = \sqrt{|(P_2 - P)|} \tag{3-5}$$

$$rdP_3P = \sqrt{|(P_3 - P)|} \tag{3-6}$$

$$rdP_1P_2 = \sqrt{|(P_1 - P_2)|} \tag{3-7}$$

$$rdP_1P_3 = \sqrt{|(P_1 - P_3)|} \tag{3-8}$$

$$rdP_2P_3 = \sqrt{|(P_2 - P_3)|} \tag{3-9}$$

where P is the pressure in the tubing, $P1$ is the pressure in the annulus of the upper zone, $P2$ is the pressure in the annulus of the intermediate zone, $P3$ is the pressure in the annulus of the lower zone.

All available combinations of ΔP were considered.

3.4.2 Flow Estimation Based on Pressure Convolution

Further features were constructed based on the work of (TIAN, 2018) on flow estimation based on pressure convolution. For each pressure, three features involving pressure and time were derived.

$$TC1_i = \sum_{j=1}^{i}(P_j - P_{j-1})log(t_i - t_{j-1}), i = 1, ..., n \tag{3-10}$$

$$TC2_i = \sum_{j=1}^{i}(P_j - P_{j-1})(t_i - t_{j-1}), i = 1, ..., n \tag{3-11}$$

where P_i and t_i are the pressure and time at the i-th data point, and n is the number of observations from the PDG.

(TIAN, 2018) states that these features deliver better results than features based on traditional pressure time formulation that involves division rather than multiplication. Two more features were built to evaluate this assumption in our data set.

$$TC3_i = \sum_{j=1}^{i}(P_j - P_{j-1})/log(t_i - t_{j-1}), i = 1, ..., n \tag{3-12}$$

$$TC4_i = \sum_{j=1}^{i}(P_j - P_{j-1})/(t_i - t_{j-1}), i = 1, ..., n \tag{3-13}$$

where P_i and t_i are the pressure and time at the i-th data point, and n is the number of observations from the PDG.

3.4.3 Interaction Between Features

The valve status variable can be used in combination with other features to make them more assertive.

The idea is that if the valve is closed, we have no flow coming from that zone, and therefore the parameters that are relevant for determining the partial flow of the zone can be set to zero. Since the status is a variable that has been

encoded as 1 if it is open and 0 if it is closed, a simple multiplication of this variable with the other features would already yield the desired result.

The multiplication of the features is the simplest way to evaluate the interaction between them, already bringing about possible nonlinearities for more simplified models.

A systematic way to perform feature multiplication is to generate a matrix of 2-by-2 combinations of column multiplications. This can be achieved through the PolynomialFeatures function of the scikit-learn library (PEDREGOSA et al., 2011), where all combinations are made and new columns are added to the original feature matrix.

3.4.4 Feature Selection

In the context of machine learning applications to oil and gas temperature sensor data, effective feature selection is essential to improve model interpretability, reduce overfitting, and minimize computational costs. According to (CHENG, 2024), feature selection techniques are generally grouped into filter, wrapper and embedded methods, each with distinct characteristics.

For this study, filter methods were adopted as the primary approach to feature selection. The process began with the construction of correlation maps to identify the features most strongly associated with the target variables. This approach, as described by (CHENG, 2024), evaluates the relevance of features based on their statistical relationship with the target variable, independently of any specific machine learning algorithm. Filter methods offer several advantages, particularly in scenarios involving large numbers of features, such as those generated by oil and gas temperature sensors. Their main benefits include computational efficiency, scalability, and the ability to rapidly eliminate redundant or irrelevant features before any modeling takes place.

Although more complex methods such as wrappers (e.g., forward selection) were considered to potentially identify optimal feature subsets, they were not explored in this work. Wrapper methods involve iterative model training and evaluation for each potential feature combination, which can quickly become computationally infeasible with large data sets or high-dimensional feature spaces. Given the extensive number of features and the exploratory objectives of this research, the computational burden of wrapper methods was deemed prohibitive for the scope of this thesis.

The exclusive use of filter methods is therefore justified by their simplicity, speed, and effectiveness in handling high-dimensional data, key require-

ments for the initial stages of modeling with sensor data sets. By enabling a systematic and computationally manageable evaluation of candidate features, filter methods supported the development of robust and scalable machine learning models, while also preserving the flexibility needed to iterate quickly during research.

3.5 Unsupervised Learning

In our study, the focus will be on unsupervised learning on dimensionality reduction and clustering, applying the following techniques to our data set.

3.5.1 Dimensionality Reduction

Dimensionality reduction is a critical technique in machine learning that involves reducing the number of input variables or features in a data set while preserving as much relevant information as possible. This process is essential for simplifying models, improving computational efficiency, and mitigating the curse of dimensionality, which can lead to overfitting and degraded performance in high-dimensional spaces.

Two widely used dimensionality reduction techniques are Principal Component Analysis (PCA) and t-Distributed Stochastic Neighbor Embedding (t-SNE). Both PCA and t-SNE serve different purposes in the context of dimensionality reduction, with PCA being more suitable for preprocessing and feature extraction in predictive modeling, while t-SNE excels in visualizing high-dimensional data in a comprehensible two- or three-dimensional space.

PCA (JOLLIFFE; CADIMA, 2016) is a linear technique that transforms the original features into a new set of uncorrelated variables called principal components, which are ordered by the amount of variance they capture from the data. By projecting the data onto a lower-dimensional space defined by the top principal components, PCA effectively retains the most significant variations in the data set while discarding less informative dimensions. This is particularly useful for tasks such as noise reduction, visualization, and data preparation for supervised learning.

t-SNE (MAATEN; HINTON, 2008) is a nonlinear dimensionality reduction technique primarily used for data visualization, particularly in high-dimensional data sets. Models the similarity between data points in the high-dimensional space and translates this similarity into a lower-dimensional rep-

resentation, emphasizing the preservation of local structures and relationships. Unlike PCA, which focuses on maximizing variance, t-SNE is adept at revealing clusters and patterns in the data, making it invaluable for exploratory data analysis and understanding complex data sets.

3.5.2 Clustering

Clustering is an unsupervised learning technique in machine learning that involves grouping a set of objects or data points into clusters based on their similarities, allowing for better understanding, organization, and analysis of the data. Various clustering algorithms exist, each with its own approach and assumptions. The most common algorithms include K-Means, Hierarchical Clustering and Gaussian Mixture Models (GMM).

3.5.2.1 Algorithms

K-means (BELYADI, 2021) is a partition clustering algorithm that aims to divide a data set into k distinct clusters by minimizing the within-cluster variance. The algorithm iteratively assigns data points to the nearest cluster centroid and updates the centroids based on the mean of the assigned points. The objective function to minimize is given by:

$$J = \sum_{i=1}^{k} \sum_{x_j \in C_i} ||x_j - \mu_i||^2 \tag{3-14}$$

where J is the total within-cluster variance, C_i represents the set of points in the i-th cluster, x_j is a data point, and μ_i is the centroid of the cluster i.

Hierarchical Clustering (BELYADI, 2021) builds a tree-like structure (dendrogram) to represent nested groupings of data points. It can be performed using either agglomerative (bottom-up) or divisive (top-down) approaches. In agglomerative clustering, the distance between clusters is typically measured using linkage criteria such as single-linkage (minimum distance), complete-linkage (maximum distance), or average-linkage (mean distance). The distance between two clusters C_i and C_j can be expressed as:

$$d(C_i, C_j) = \text{linkage}(C_i, C_j) \tag{3-15}$$

where the linkage function varies depending on the chosen strategy and the process continues until all points are merged into a single cluster.

GMM (Gaussian Mixture Models) (GéRON, 2023) are probabilistic models that assume data points are generated from a mixture of several Gaussian distributions, each representing a cluster. GMMs utilize the Expectation-Maximization (EM) algorithm to estimate the parameters of the Gaussian components. The probability density function of GMM can be expressed as:

$$p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k) \tag{3-16}$$

where $p(x)$ is the overall probability density function, K is the number of Gaussian components, π_k is the mixing coefficient for the k-th component (satisfying $\sum_{k=1}^{K} \pi_k = 1$), and $\mathcal{N}(x|\mu_k, \Sigma_k)$ is the Gaussian distribution with mean μ_k and covariance Σ_k.

Although the algorithms have very different clustering strategies, the key distinction among them lies in how they handle the determination of the number of clusters, with some necessitating prior knowledge and others providing more flexibility in exploring the underlying data structure.

K-Means clustering requires the number of clusters k to be specified beforehand, making it sensitive to the initial choice of k and potentially leading to suboptimal results if the true number of clusters is not known.

In contrast, Hierarchical Clustering does not require the number of clusters to be predetermined; instead, it builds a tree-like structure (dendrogram) that allows users to choose the number of clusters based on a desired level of granularity.

Gaussian Mixture Models, on the other hand, assume that the data is generated from a mixture of several Gaussian distributions and allow for soft clustering, where each data point can belong to multiple clusters with different probabilities, but still require the number of clusters to be specified in advance.

3.5.2.2 Cluster Analysis

An effective method for evaluating the quality of clustering results is the silhouette score (BELYADI, 2021). It evaluates how similar an object is to its own cluster compared to other clusters. Ranges from -1 to 1, a high value indicates that the object is well matched to its own cluster and poorly matched to neighboring clusters.

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \tag{3-17}$$

where $a(i)$ is the average distance between i and all other points in the same cluster, $b(i)$ is the minimum average distance between i and points in any other cluster, i.e., the nearest cluster that i is not a part of.

The overall silhouette score for the clustering is the mean of $s(i)$ for all data points.

By varying the number of clusters as an input parameter for algorithms like K-Means and analyzing the resulting silhouette scores, one can determine the optimal number of clusters that maximizes intra-cluster similarity while minimizing inter-cluster similarity. For example, as the number of clusters increases, the silhouette score may initially rise, indicating better-defined clusters, but could eventually decline if the clusters become too fragmented.

In contrast, for algorithms that do not require a predefined number of clusters, such as Hierarchical Clustering, other input parameters, like the linkage criteria for hierarchical methods, must be systematically varied to assess their impact on clustering performance.

By comparing silhouette scores across different clustering configurations, one can gain insight into the robustness and validity of the clustering results, ultimately guiding the selection of the most appropriate clustering method and parameters for the given data set. This comprehensive analysis helps ensure that the chosen clusters are meaningful and reflect the underlying data structure.

3.6 Supervised Learning

In our study, we will focus on supervised learning classification and regression problems, applying the following techniques to our data set.

3.6.1 Classification

The data set contained information on the opening status of the valves, so it is possible to explore classification methods based on the available pressure and temperature data.

The valve status classification problem can be described as a binary classification problem since there are only two possible statuses for this type of valve: open (coded 1) or closed (coded 0).

3.6.1.1 Algorithms

(GéRON, 2023) states that a a priori there is no model that is guaranteed to work better. So, since we do not have a specific guideline showing the best algorithms for the task at hand, we selected the most significant classifier algorithms described in the (JAMES et al., 2023) general section on classification.

Our first approach is to start as simply as possible with a linear classifier with the most relevant feature to establish a basis for further comparison. For this, we used the logistic regression linear classifier.

In the following, a general description of each classification algorithm is provided.

Logistic Regression (BELYADI, 2021) Used for binary classification. The decision boundary of a linear classifier:

$$f(x) = \beta_0 + \sum_{i=1}^{n} \beta_i x_i \tag{3-18}$$

where $f(x)$ is the output score, β_0 is the intercept (bias), β_i are the weights and n is the number of features.

The output is transformed using the sigmoid function to produce a probability.

$$P(y = 1|x) = \sigma(f(x)) = \frac{1}{1 + e^{-f(x)}} \tag{3-19}$$

where $P(y = 1|x)$ probability of positive class and σ sigmoid function.

The optimization process for linear classifiers typically involves minimizing a loss function, such as the binary cross-entropy loss.

Support Vector Classifier (SVC) (BELYADI, 2021) is a supervised learning model that finds the optimal hyperplane to separate classes in a high-dimensional space. The main hyperparameters include the regularization parameter C, the kernel type (e.g., linear, polynomial, RBF), and the kernel coefficient γ for nonlinear kernels. The decision function is represented as

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i K(x_i, x) + b \tag{3-20}$$

where α_i are the Lagrange multipliers, y_i are the class labels, $K(x_i, x)$ is the kernel function and b is the bias.

k-Nearest Neighbors (k-NN) (BELYADI, 2021) algorithm classifies a data point based on how its neighbors are classified. The main hyperparameters include the number of neighbors k and the distance metric (e.g., Euclidean or Manhattan). The classification decision is made by majority voting among the k nearest neighbors:

$$\hat{y} = \text{mode}(y_i) \quad \text{for } i \in \text{k nearest neighbors} \tag{3-21}$$

where $\hat{y}$ is the predicted class and y_i are the classes of the nearest neighbors.

Decision Tree Classifier (BELYADI, 2021) is a machine learning algorithm used for classification tasks, which divides data into subsets based on the values of input features to form a tree-like structure. The main hyperparameters that govern the behavior of a decision tree include the maximum depth of the tree max_depth, the minimum number of samples required to divide a node $min\_samples\_split$, and the minimum number of samples required to be in a leaf node $min\_samples\_leaf$. The decision tree algorithm selects the feature that maximizes the information gain or minimizes the impurity, typically using metrics such as Gini impurity or entropy for classification. The Gini impurity for a node can be expressed as

$$Gini(D) = 1 - \sum_{i=1}^{C} p_i^2 \tag{3-22}$$

where D is the data set at the node, C is the number of classes, and p_i is the proportion of instances belonging to class i at that node. The tree construction continues recursively, splitting the data at each node until one of the stopping criteria is met, such as reaching the maximum depth or having insufficient samples. The final output of the decision tree is determined by the majority class of instances in the leaf node, which makes it a straightforward but powerful technique for classification tasks.

Random Forest Classifier (BELYADI, 2021) is an ensemble learning method that constructs multiple decision trees during training and merges their predictions for better accuracy and robustness. The main hyperparameters include the number of trees $n_{\text{estimators}}$, the maximum depth of the trees, and the minimum samples required to split a node. The final prediction is made by majority voting:

$$\hat{y} = \frac{1}{n_{\text{estimators}}} \sum_{j=1}^{n_{\text{estimators}}} T_j(x) \tag{3-23}$$

where $\hat{y}$ is the predicted class and $T_j(x)$ is the prediction of the j-th tree.

3.6.1.2 Metrics

In supervised machine learning, the evaluation of classification models is based on a variety of performance metrics, each highlighting different aspects of predictive accuracy and reliability. Commonly used metrics include balanced accuracy, precision, recall, and the $F1$ score (BELYADI, 2021).

Balanced accuracy is particularly valuable when dealing with imbalanced data sets, as it averages the recall obtained on each class, ensuring that minority classes are fairly represented in the evaluation. Precision measures

the proportion of positive identifications that are actually correct, while recall assesses the proportion of actual positives that are successfully identified by the model. The $F1$ score, as the harmonic mean of precision and recall, provides a single measure that balances these two aspects, making it especially useful when it is necessary to account for false positives and false negatives.

Balanced Accuracy Balanced accuracy compensates for imbalanced data sets by averaging the recall obtained in each class. For binary classification, it is defined as:

$$\text{Balanced Accuracy} = \frac{1}{2}\left(\frac{TP}{TP+FN} + \frac{TN}{TN+FP}\right) \tag{3-24}$$

where TP means true positives, TN means true negatives, FP means false positives, and FN means false negatives.

- Range: 0 to 1
- 1 indicates perfect classification; 0.5 indicates random guessing for balanced binary classes.
- Good classification typically corresponds to values above 0.7–0.8, depending on the domain.

Precision Precision measures the proportion of correctly predicted positive observations to the total predicted positives.

$$\text{Precision} = \frac{TP}{TP+FP} \tag{3-25}$$

where TP means true positives and FP means false positives.

- Range: 0 to 1
- 1 means no false positives; 0 means all predicted positives are incorrect.
- Higher values are better; above 0.8 is strong, but the threshold depends on the application.

Recall (Sensitivity) Recall measures the proportion of actual positives correctly identified:

$$\text{Recall} = \frac{TP}{TP+FN} \tag{3-26}$$

where TP means true positives and FN means false negatives.

- Range: 0 to 1
- 1 means all actual positives are captured; 0 means none are.
- Higher values are better; above 0.8 is strong, especially in critical applications.

$F1$ **Score** $F1$ score is the harmonic mean of precision and recall, providing a balance between them:

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3-27}$$

- Range: 0 to 1
- 1 indicates perfect precision and recall; 0 means either precision or recall is zero.
- Higher values are better; values above 0.8 are generally considered strong.

Summary Table Together, these metrics offer a comprehensive framework for assessing the effectiveness of classification algorithms, advising the selection of models, and optimizing according to the specific requirements of the application domain.

What is considered a good value can vary significantly based on the problem context, class imbalance, and specific business requirements. In some cases, trade-offs between precision and recall are necessary, and metrics such as the $F1$ score help to summarize model performance in those situations.

Metric	**Range**	**Good Value**
Balanced Accuracy	0 to 1	> 0.7–0.8
Precision	0 to 1	Higher is better
Recall	0 to 1	Higher is better
F1 Score	0 to 1	Higher is better

Table 3.1: Summary of classification metrics, range, and good values.

3.6.2 Regression

The data set contained information on the total oil flow rate, so we could explore regression methods based on the available pressure and temperature data.

3.6.2.1 Algorithms

The same consideration applies to regression algorithms; since we do not have a specific guideline showing what the best algorithms are for the task at hand, we selected the most significant regression algorithms described in (BELYADI, 2021) and (JAMES et al., 2023).

Our first approach is to start as simply as possible with a linear regressor with the most relevant feature to establish a basis for further comparison. For this, we used linear regression.

In the following, a general description of each regression algorithm is provided.

Linear Regression (BELYADI, 2021) is a fundamental statistical method that is used to model the relationship between a dependent variable and one or more independent variables by fitting a linear equation to the observed data. The linear regression equation can be expressed as:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n + \epsilon \tag{3-28}$$

where y is the predicted value, β_0 is the intercept, β_i are the coefficients for each feature x_i, and ϵ represents the error term.

Ridge Regression (JAMES et al., 2023) is an extension of linear regression that introduces $L2$ regularization to prevent overfitting by penalizing large coefficients. The main hyperparameter is the regularization parameter α. The Ridge regression equation is given by minimizing the following cost function.

$$\text{Cost} = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2 + \alpha \sum_{j=1}^{n} \beta_j^2 \tag{3-29}$$

where m is the number of observations, y_i are the actual values, $\hat{y}_i$ are the predicted values, and β_j are the coefficients.

Lasso Regression (JAMES et al., 2023) is another form of linear regression that applies $L1$ regularization, which can reduce some coefficients to zero, leading to simpler models. The key hyperparameter is λ, which controls the strength of the regularization. The Lasso cost function is expressed as:

$$\text{Cost} = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{n} |\beta_j| \tag{3-30}$$

where m is the number of observations, y_i are the actual values, $\hat{y}_i$ are the predicted values, and β_j are the coefficients.

Elastic Net (JAMES et al., 2023) combines $L1$ and $L2$ regularization, making it particularly useful when there are correlated features. The main hyperparameters are λ_1 for $L1$ regularization and λ_2 for $L2$ regularization. The cost function for Elastic Net is:

$$\text{Cost} = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2 + \lambda_1 \sum_{j=1}^{n} |\beta_j| + \lambda_2 \sum_{j=1}^{n} \beta_j^2 \tag{3-31}$$

where m is the number of observations, y_i are the actual values, $\hat{y}_i$ are the predicted values, and β_j are the coefficients.

Support Vector Regression (SVR) (BELYADI, 2021) is an extension of SVM for regression tasks that aims to find a function that deviates from the actual target values by a value not greater than a specified margin ϵ. The main hyperparameters include the regularization parameter C and the kernel parameters (e.g., γ for the RBF kernel). The SVR optimization problem can be formulated as

$$\min_{w,b} \frac{1}{2}||w||^2 + C\sum_{i=1}^{m} \xi_i \tag{3-32}$$

subject to $y_i - (w^T x_i + b) \leq \epsilon + \xi_i$ and $(w^T x_i + b) - y_i \leq \epsilon + \xi_i$, where ξ_i are the slack variables.

Decision Tree Regression (BELYADI, 2021) is a non-parametric regression method that divides the data into subsets based on feature values. The main hyperparameters include the maximum depth of the tree, the minimum samples required to split a node, and the minimum samples required at a leaf node. The predicted value for a given input is the average of the target values in the leaf node:

$$\hat{y} = \frac{1}{N}\sum_{i=1}^{N} y_i \tag{3-33}$$

where N is the number of training samples in the leaf node.

3.6.2.2 Metrics

The quality of a regression model is commonly assessed using several statistical metrics, each providing unique information on different aspects of model performance (BELYADI, 2021).

In the following paragraphs, the equations, interpretation, and range of the metrics are described.

Coefficient of Determination (R^2) (BELYADI, 2021) measures the proportion of variance in the dependent variable that is predictable from the independent variables, with values ranging from $-\infty$ to 1. An R^2 value of 1 indicates a perfect fit, while values near zero or negative suggest that the model fails to capture the underlying data structure or performs worse than a simple mean-based prediction.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{3-34}$$

where y_i are the actual values, $\hat{y}_i$ are the predicted values and $\bar{y}$ is the mean of the actual values.

- Range: $-\infty$ to 1
- $R^2 = 1$ indicates a perfect fit.
- $R^2 = 0$ means the model not do better than predict the mean.
- $R^2 < 0$ means the model is worse than simply predicting the mean.
- Good regression typically corresponds to $R^2 > 0.7$–0.8, depending on the domain.

Mean Absolute Error (MAE) (BELYADI, 2021) quantifies the average magnitude of errors in predictions, disregarding their direction, and ranges from zero to infinity; lower MAE values denote more accurate models, although the interpretation of what constitutes a "low" value is context-dependent.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{3-35}$$

where y_i are the actual values, $\hat{y}_i$ are the predicted values.

- Range: 0 to ∞
- Lower values indicate better fit; 0 means perfect prediction.
- What is considered good depends on the context and scale of the data.

Mean Absolute Percentage Error (MAPE) (BELYADI, 2021) expresses prediction errors as a percentage of actual values, facilitating interpretability across different scales, with lower percentages indicating better performance; values below 10% are generally regarded excellent, while values above 50% are considered poor.

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{3-36}$$

where y_i are the actual values and $\hat{y}_i$ are the predicted values.

- Range: 0% to ∞%
- Lower percentages indicate better fit.
- MAPE $< 10\%$ is excellent, 10–20% is good, 20–50% is acceptable, $> 50\%$ is poor.
- May be misleading if actual values are close to zero.

Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) (BELYADI, 2021) measure the average squared differences between predicted and actual values, with RMSE being the square root of MSE to maintain the same units as the target variable. Both metrics are nonnegative and unbounded above, with lower values reflecting better model accuracy. However, because of their sensitivity to outliers, MSE and RMSE are often interpreted relative to the scale and variance of the data.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3-37}$$

where y_i are the actual values and $\hat{y}_i$ are the predicted values.

- Range: 0 to ∞
- Lower values indicate better fit; 0 means perfect prediction.
- Sensitive to outliers due to squaring.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3-38}$$

where y_i are the actual values and $\hat{y}_i$ are the predicted values.

- Range: 0 to ∞
- Lower values indicate better fit; 0 means perfect prediction.
- RMSE has the same units as the target variable.

Summary Table Collectively, these metrics provide a comprehensive framework for evaluating and comparing the predictive performance of regression models in machine learning applications.

Metric	**Range**	**Good Value**
R^2	$-\infty$ to 1	Closer to 1
MAE	0 to ∞	Lower values
MAPE	0 to ∞	$< 10\% - 20\%$
MSE	0 to ∞	Lower values
RMSE	0 to ∞	Lower values

Table 3.2: Summary of regression metrics, range, and good values.

3.6.3 Hyperparameter Optimization

In machine learning, parameters and hyperparameters play crucial but distinct roles in the training and performance of algorithms. Parameters are the internal configurations of a model that are learned from the training data during the training process, such as the coefficients in a linear regression model or the weights in a SVM. These parameters are adjusted through optimization techniques, such as gradient descent, on the basis of the error between the predicted outputs and the actual labels.

In contrast, hyperparameters are external configurations set before the training process begins and govern the overall behavior of the learning algorithm. Examples of hyperparameters include the learning rate in algorithms such as gradient descent for logistic regression, the maximum depth of a decision tree, and the number of neighbors k in the k-NN algorithm.

Unlike parameters, hyperparameters are not learned from the data, but are typically tuned through techniques such as grid search or random search to find the optimal settings that enhance model performance. The careful selection and tuning of both parameters and hyperparameters is essential for achieving high accuracy and generalization in traditional machine learning models, with hyperparameters often requiring more attention due to their significant impact on the training process and final model effectiveness.

As each algorithm has its own set of hyperparameters to be determined, a strategy was needed. For this task, the RandomizedSearchCV hyperparameter optimization technique was used together with TimeSeriesSplit to deal with the pressure and temperature time-series aspect of the data.

RandomizedSearchCV as introduced in (BERGSTRA; BENGIO, 2012), is a hyperparameter optimization technique that randomly samples a specified number of hyperparameter combinations from a defined search space, rather than exhaustively evaluating all possible combinations as in Grid Search. This approach offers several advantages, as it allows for a more efficient exploration of the hyperparameter space, particularly when the space is large and complex, as it can identify promising regions without the computational cost of evaluating every combination. Unlike Grid Search, which can become inefficient as the number of hyperparameters increases, since it evaluates every combination in a grid-like manner, Randomized Search provides a way to approximate optimal hyperparameters with a smaller number of evaluations. This makes it especially suitable for scenarios with high-dimensional parameter spaces, where some parameters may have little effect on the performance of the

model. By focusing on a randomized subset of the search space, Randomized Search can yield competitive results while significantly reducing computation time, making it a practical choice for hyperparameter tuning in machine learning workflows. But, as it performs a non-exhaustive search, it is important to verify the selected hyperparameter values obtained to check if they collapse in the extremes of the selected intervals; if so, a change in the interval of variation is necessary.

TimeSeriesSplit from Scikit-learn (PEDREGOSA et al., 2011) is a cross-validation method specifically designed for time-series data, where the temporal ordering of observations is crucial. Unlike traditional cross-validation techniques that randomly shuffle data, which can lead to data leakage and unrealistic training scenarios, TimeSeriesSplit respects the sequential nature of time series by ensuring that the training set consists of observations that precede the test set. This method progressively increases the size of the training set with each split, allowing models to be trained on all available past data while testing on future data. This is essential for time-series forecasting, because predictions are based on historical trends and patterns. Using conventional cross-validation could lead to overoptimistic performance estimates, as it may inadvertently include future information in the training phase, thus invalidating the model's ability to generalize to unseen data. By maintaining the temporal structure, TimeSeriesSplit provides a more realistic assessment of a model's predictive performance in time-dependent scenarios.

3.7 Workflow

The workflow for the classification and regression problems implemented is similar. An initial cleaning and preparation of the data is performed, followed by Min-Max scaling. The features are built and the selected algorithms with their hyperparameter range are set. The data set is divided into training and test. For each algorithm, RandomizedSearchCV within the TimeSeriesSplit is used to fine-tune the hyperparameters, and the training set is used to tune the parameters of the models using the appropriate metrics.

Algorithm 1 Workflow for Classification and Regression Models

1: **Input:** Raw dataset
2: Perform data cleaning and preparation
3: Construct features from the data
4: Normalize features
5: Define algorithms and their hyperparameter ranges
6: Split dataset into training and testing sets
7: **for** each algorithm **do**
8: Use RandomizedSearchCV with TimeSeriesSplit to tune hyperparameters
9: Train model using appropriate evaluation metrics
10: **end for**
11: **Output:** Trained models with tuned hyperparameters
12: Evaluate the trained models on the test data

Considering this workflow, in the end, we have the best version of each algorithm for the task at hand. To assess the generalization of the models, two experiments were conducted. The first uses the same well for training and testing, and the other uses different wells for each stage.

In Python version 3.11.0 [3], the workflow was developed using the Scikit-learn (PEDREGOSA et al., 2011), NumPy (HARRIS et al., 2020), SciPy (MCKINNEY, 2010), Matplotlib (HUNTER, 2007), and Seaborn (WASKOM, 2021) libraries.

This strategy aims to identify the algorithms that best adapt to the specific context and to fine-tune their parameters in an integrated manner.

[3]https://www.python.org/downloads/release/python-3110/

4
Results and Discussion

In this chapter, the results obtained from applying the developed methodology to the data set are presented and analyzed. The performance of clustering algorithms in identifying valve patterns, the effectiveness of classification models in predicting valve status, and the precision of regression models for oil rate estimation are discussed. The chapter also examines the impact of feature selection, data normalization, and temporal data handling on model performance and generalizability.

4.1
Exploratory Analysis

The variables present in the data set are pressure and temperature of the sensors in the annulus of each zone and their backups, the pressure and temperature of the sensor located inside the tubing, the oil and gas production rates, and the opening status of the valves for each zone.

A heat map showing the Pearson Correlation Coefficient (BELYADI, 2021) between the variables in the data set was built for Well A (see Figure 4.1) and Well B (see Figure 4.2), highlighting the most highly correlated variables and providing guidance in building the features to be used for model development.

A comprehensive exploratory analysis was conducted on the data set, which comprises pressure and temperature measurements recorded at one-minute intervals. This initial investigation yielded several critical insights into the operational performance and measurement characteristics of the wells.

For the first well, three distinct periods were identified in which both the main and backup sensors experienced data freezing. Notably, one of these periods persisted for approximately 11 hours, representing a significant data gap. Upon further inspection, it became evident that the backup sensor data was redundant, providing no additional unique information beyond that of the primary sensor. To maintain data integrity and avoid unnecessary duplication, the backup sensor data was excluded from subsequent analyses.

The analysis also revealed behavioral patterns in the valve operations. Specifically, Valve 1 was observed to open in pulses, while Valve 3 exhibited pulsed closures in both wells. These operational pulses introduced considerable noise into the pressure measurements, with temperature readings demonstrating a much smoother profile and less sensitivity to these transient events. This

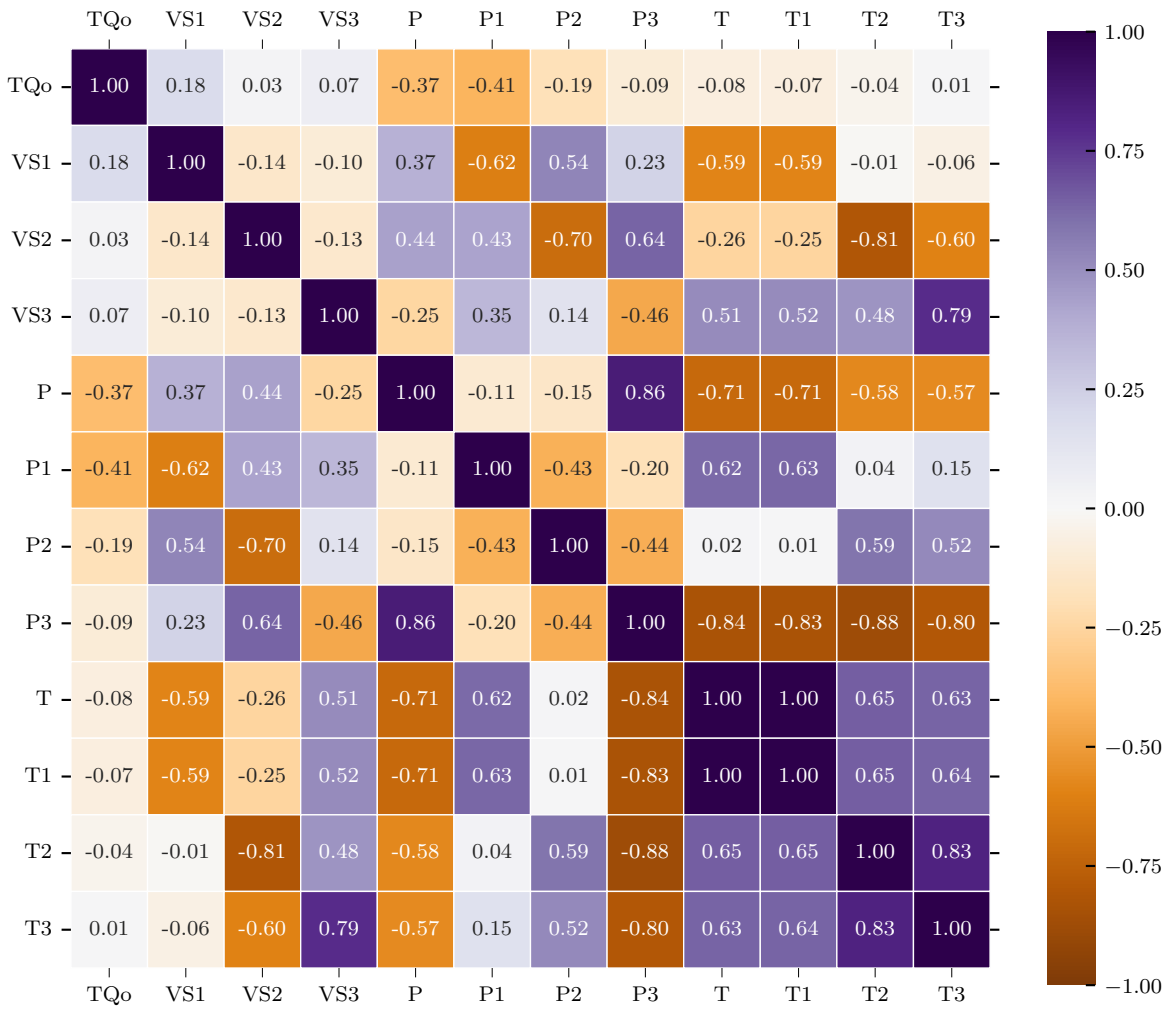

Figure 4.1: Correlation map between features and target variables for well A.

distinction highlights the importance of considering the nature of each physical quantity when engineering features and interpreting the data. Additionally, production data analysis demonstrated that the pressure consistently remained above saturation levels, ensuring single-phase flow conditions. A strong direct correlation was observed between gas and oil production rates, indicating redundancy in the gas rate data. Consequently, gas rate measurements were also removed to streamline the data set and focus on the most informative variables for subsequent modeling.

In summary, this exploratory analysis emphasizes the necessity of identifying measurement anomalies and understanding the operational characteristics of the wells and equipment. Such careful scrutiny is essential for ensuring data quality, guiding effective feature engineering, and ultimately supporting robust and accurate machine learning model development in oil and gas applications.

Analysis of pressure and temperature data reveals an uneven distribution

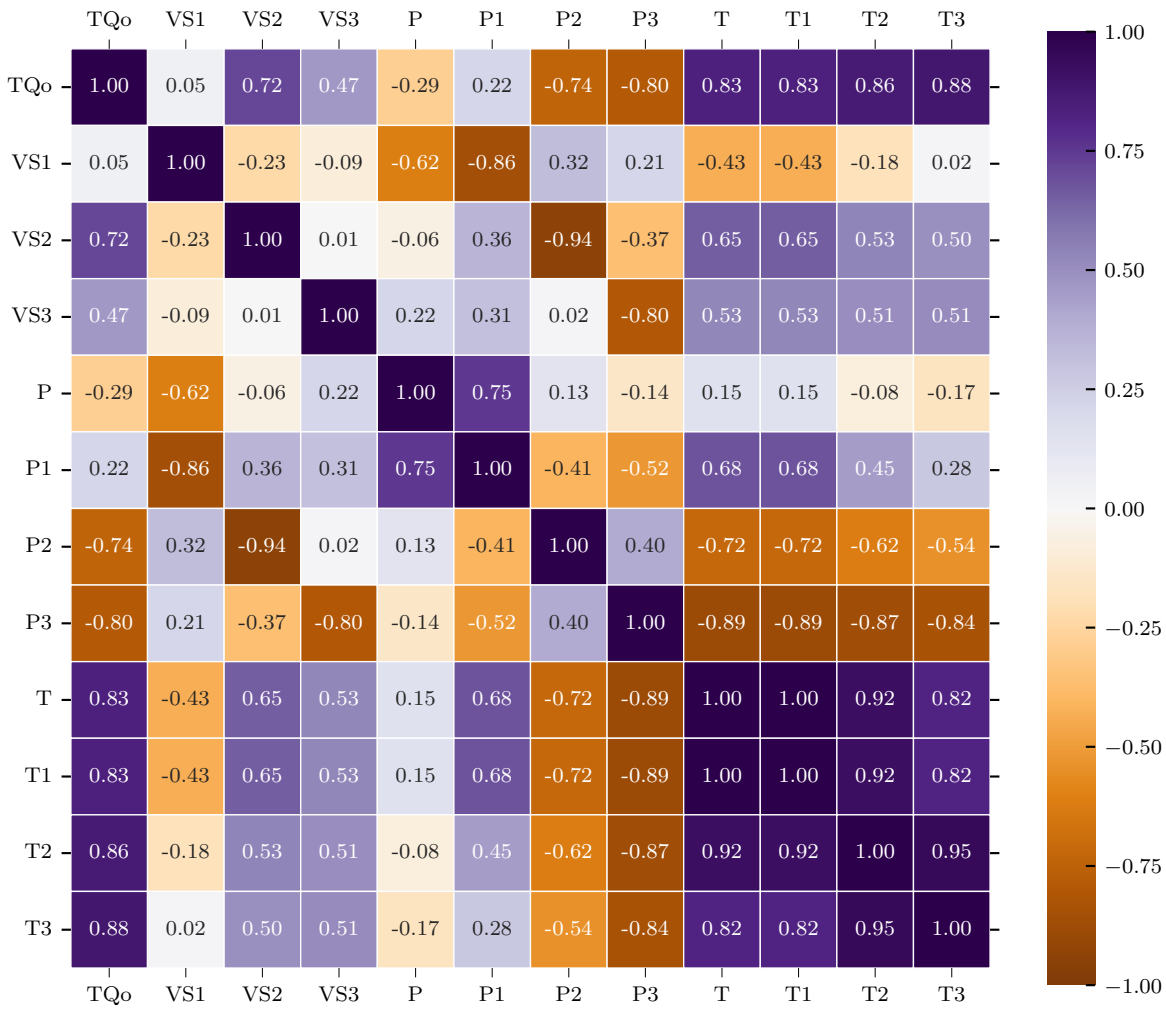

Figure 4.2: Correlation map between features and target variables for well B.

of values, as illustrated in Figures 4.3 and 4.4, which show the histograms for pressures (P, $P1$, $P2$, $P3$) and temperatures (T, $T1$, $T2$, $T3$). The histograms exhibit a notable concentration around some specific values, indicating that the measurements are not uniformly spread throughout the range but rather cluster around these points. This skewness in the distribution suggests that certain operational conditions or environmental factors may be influencing the readings, which warrants further investigation of the underlying causes of this concentration. Such insights are crucial to understanding the behavior of the system and to making informed decisions in operational management.

4.2 Feature Engineering

Feature engineering plays a pivotal role in the performance of machine learning models, particularly in complex domains such as oil and gas production optimization. In this study, we explored and compared different sets of

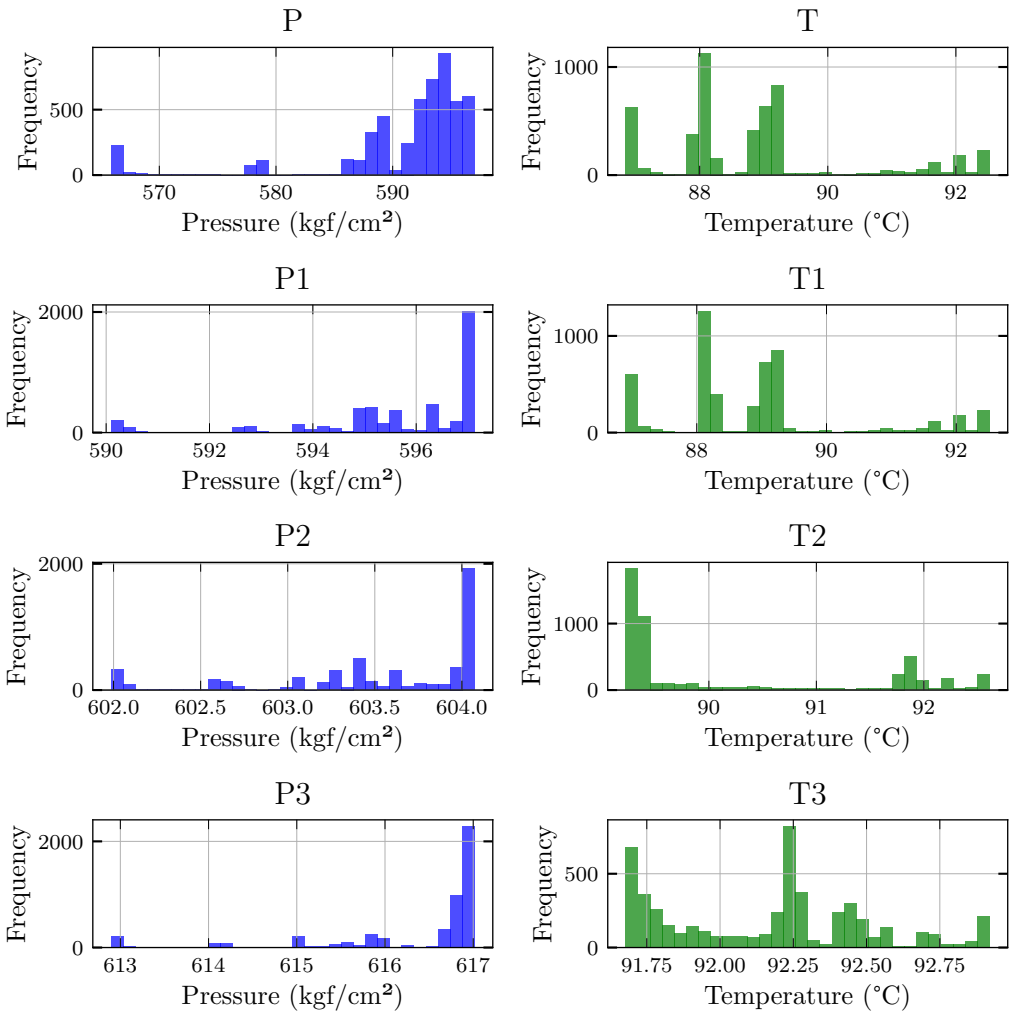

Figure 4.3: Pressure and temperature histogram for well A sensors.

features to characterize well behavior.

Pressure Among the various available signals, the pressure deltas, calculated from tubing and annular sensors, have shown distinct characteristics that can be used for downstream tasks such as the classification of valve status, as illustrated in Figure 4.5.

Analysis of the delta pressure profiles obtained from the tubing sensor reveals that while the overall shape of the signal remains consistent between different valve opening combinations, the relative values vary for each configuration. This suggests that tubing pressure deltas provide information about the system that is sensitive to the operational state, although in a manner that may require additional features or context to clearly distinguish between valve statuses.

In contrast, the pressure deltas of the annular sensors exhibit more pronounced and distinctive behaviors for each valve opening scenario. The annular

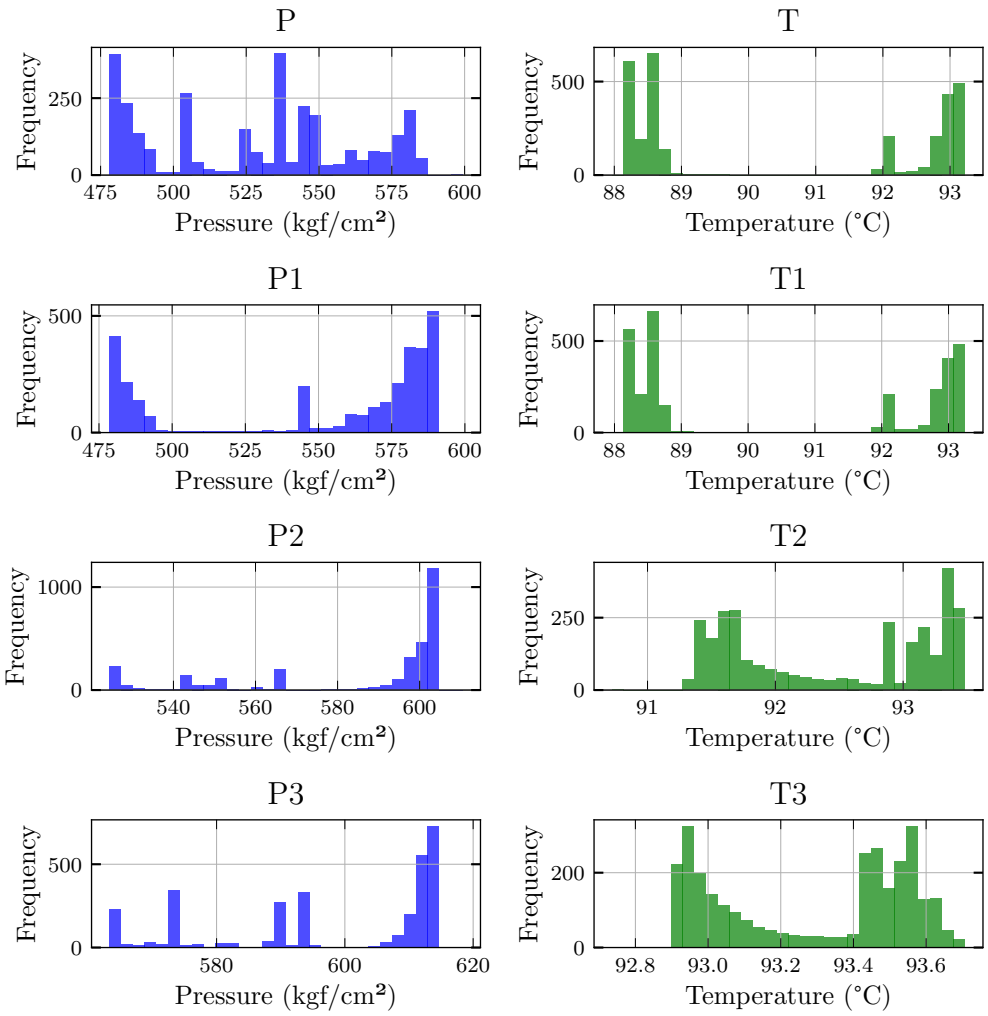

Figure 4.4: Pressure and temperature histogram for well B sensors.

pressure signals demonstrate clear variations in their profiles, corresponding uniquely to different operating states. This separation in signal characteristics indicates that annular pressure deltas can serve as a valuable source of information for accurately classifying valve status. Their ability to reflect changes in the system with greater clarity enhances the potential of machine learning models to learn reliable decision boundaries and improve classification accuracy.

In general, tubing pressure deltas offer consistency and sensitivity to operational changes, while annular pressure deltas provide more distinguishable patterns that are particularly useful for valve status classification. Incorporating both types of features, with consideration given to their individual strengths, can contribute to the modeling.

Temperature Temperature measurements constitute another important source of data for machine learning applications in oil and gas operations.

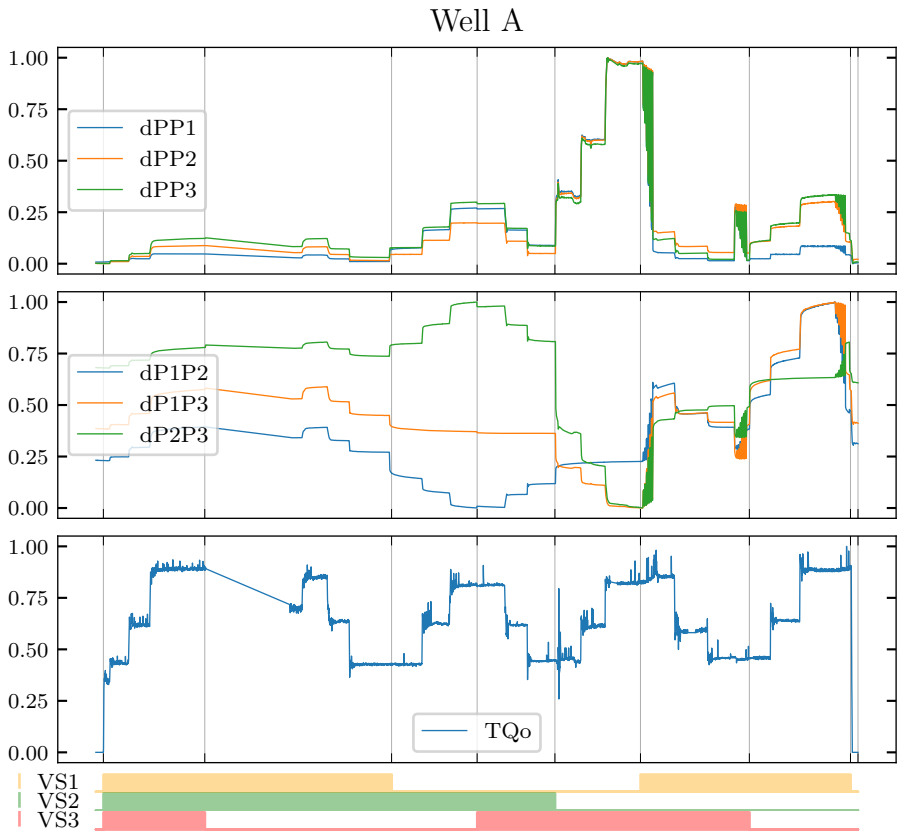

Figure 4.5: Features: Pressure deltas between sensors.

Compared to pressure signals, temperature readings are generally smoother, even during events such as valve closures, which can induce rapid fluctuations in pressure. This inherent smoothness can be advantageous for feature engineering, as it can reduce the impact of noise and outliers on model performance.

The spatial arrangement of sensors also plays an important role in the interpretation of temperature data. In particular, the tubing and annular sensors located near the superior zone are positioned close to each other. As a result, their measurements tend to be highly correlated, and the computation of temperature deltas between these sensors often serves only to amplify measurement noise rather than reveal new information; see Figure 4.6.

Another challenge arises from the influence of tubing flow on annular temperature readings. Since the temperature in the annulus can be affected by thermal exchange with the fluids moving inside the tubing, a direct interpretation of these measurements becomes less straightforward. Furthermore, because temperature naturally increases with depth, production from deeper zones can elevate the temperature observed at the sensors, further complicating the analysis.

To address these complexities, it is useful to consider the initial tempera-

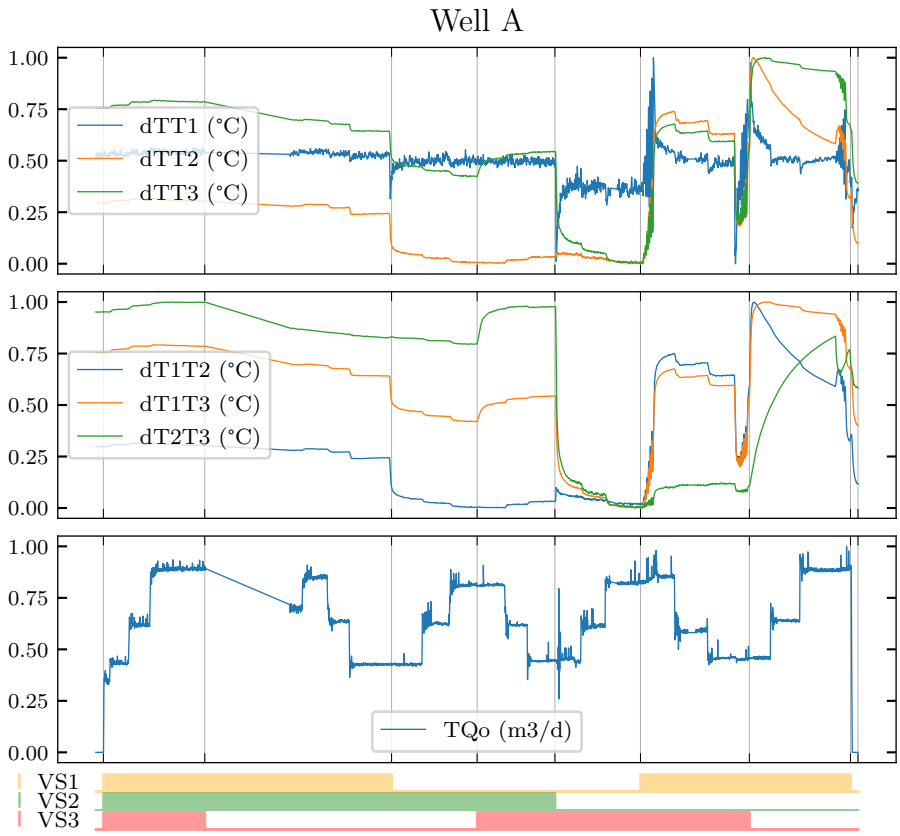

Figure 4.6: Features: Temperature deltas between sensors.

ture profile as a baseline, assuming the well is in a stable condition at the start of the observation. By calculating the temperature delta with respect to this initial state, we can derive features that may better capture meaningful deviations related to flow events or operational changes. These derived temperature features, which focus on changes in the initial condition rather than absolute values or simple sensor differences, can provide more informative signals for machine learning models, see Figure 4.7.

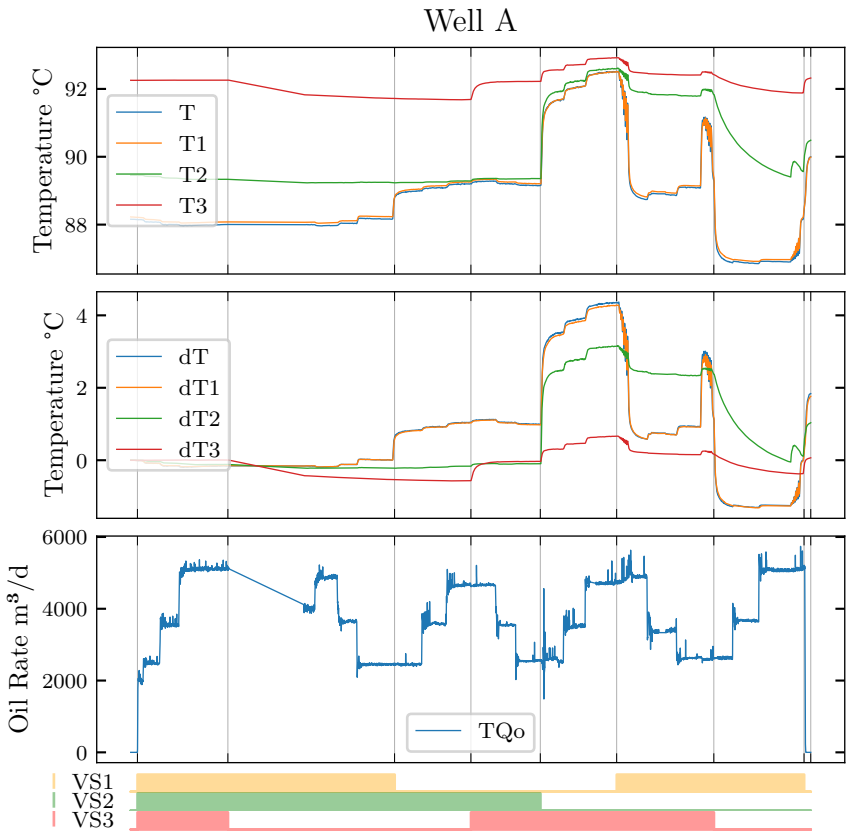

Figure 4.7: Features: Temperature deltas from the initial condition.

Tian Features We evaluated the Tian features ($TC1$ and $TC2$) alongside alternative features derived from traditional formulations ($TC3$ and $TC4$), as illustrated in Figures 4.8 and 4.9.

The first pair, $TC1$ and $TC2$, represents the features proposed by (TIAN, 2018), designed to capture essential aspects of pressure dynamics during well operations. Visual inspection reveals that $TC1$ closely follows the original pressure profile, effectively reflecting its fluctuations and transitions. This fidelity suggests that $TC1$ may be valuable for models that benefit from a direct correlation with pressure measurements. In contrast, $TC2$ exhibits a more smooth behavior, filtering out rapid oscillations and potentially reducing the impact of high-frequency noise. Such smoothing may be advantageous in scenarios where model robustness to measurement noise is desired.

The alternative features, $TC3$ and $TC4$, are based on conventional formulas traditionally utilized in well analysis. $TC3$ demonstrates a tendency to overshoot the original pressure signal, particularly during transition phases. Although this may highlight abrupt changes more prominently, it could also introduce artifacts that may mislead the model if not properly accounted for. $TC4$, on the other hand, appears to amplify noise, especially during

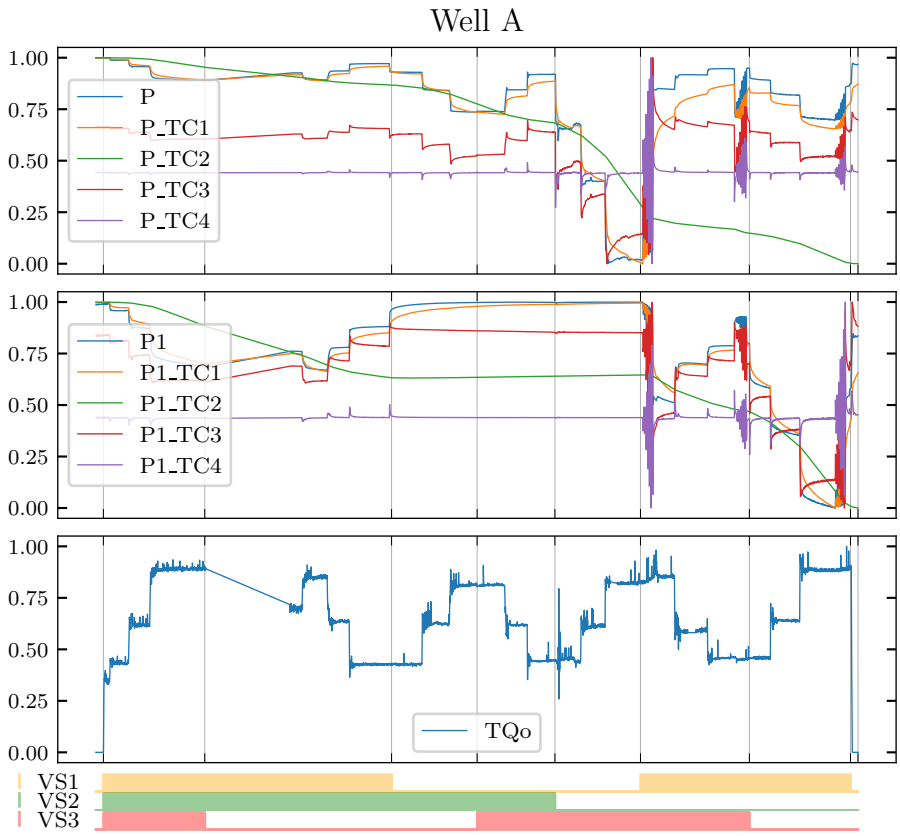

Figure 4.8: Features: Tian features and alternate Tian features for P and P1.

valve closures, which are typically associated with greater uncertainty of the measurement. This amplification suggests caution in its use, as it can degrade model performance unless further noise mitigation strategies are implemented.

(TIAN, 2018) argues that his proposed features outperform the traditional ones by offering a more faithful representation of pressure dynamics. Our findings corroborate these arguments: Tian features, particularly $TC2$, deliver smoother and more robust signals, whereas the traditional features tend to amplify noise or introduce undesirable overshooting. Therefore, we agree with Tian's conclusion that his feature formulations provide a superior basis for machine learning applications in oil and gas, as they improve the quality of the input data and, consequently, the predictive power and reliability of the models.

By systematically comparing these features, we aim to identify the most informative and robust representations for machine learning tasks. The insights obtained from this comparison guide the selection of features that best capture the underlying well dynamics, ultimately contributing to more accurate and reliable predictive models.

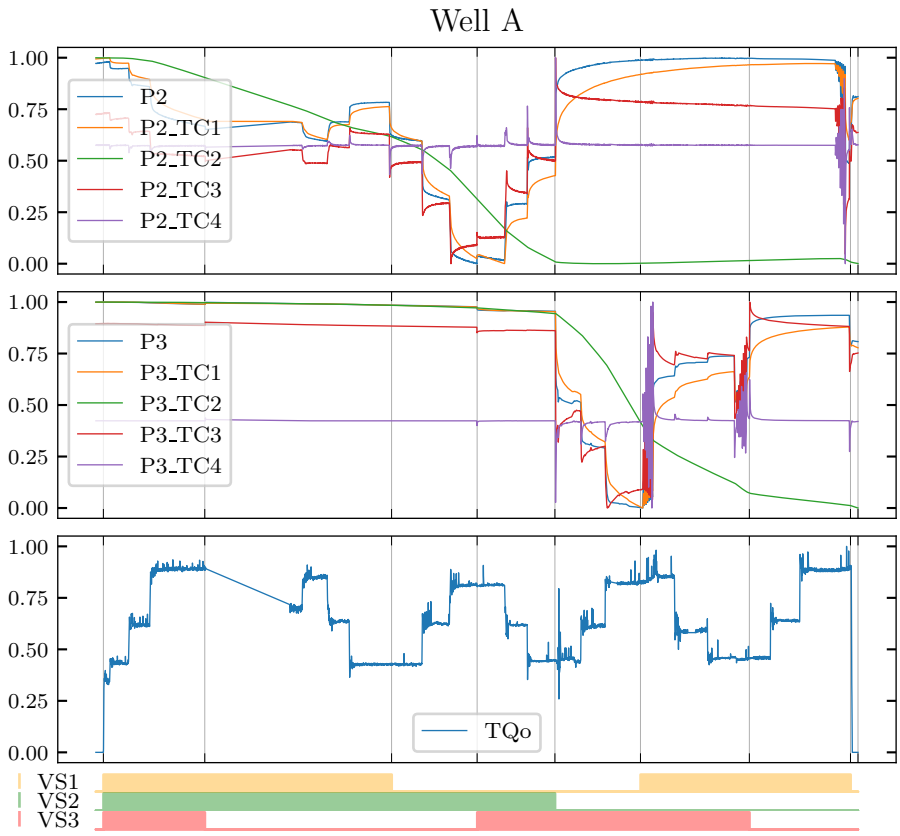

Figure 4.9: Features: Tian features and alternate Tian features for P2 and P3.

4.3 Normalization

In a machine learning context, when dealing with multiple features such as pressures, the choice of normalization method depends on the specific characteristics of the data and the model you are using.

Normalize All Together If the pressure features are on a similar scale and you want to maintain the relationships between them, you can normalize them together. This approach is common when using techniques like Min-Max scaling or Standardization (Z-score normalization) across all features.

Normalize Individually If the pressure features have different ranges or distributions, it might be beneficial to normalize each one individually. This ensures that each feature contributes equally to the model's performance and avoids any feature dominating the others due to differences in scale.

In general, here are some considerations.

– Model Sensitivity: Some models, such as k-NN or SVM, are sensitive to the scale of input features, so normalization is crucial.

- Feature Distribution: If features have different distributions, normalizing them individually can help improve model performance.
- Interpretability: Normalizing features together may make it easier to interpret their relationships, while individual normalization provides clearer insights into each feature's contribution.

Overall, it is often good practice to experiment with both approaches and evaluate model performance using cross-validation to determine which normalization method works best for your specific situation.

Normalizing data, especially in the context of pressure and temperature measurements, is a crucial step in data preprocessing that can significantly affect the analysis results. Although techniques such as min-max normalization can be useful for scaling data to a specific range (usually [0, 1]), it is essential to consider the physical meaning and relationships inherent in the original data. Maintaining the original relativity of pressure and temperature is important because these variables are often interconnected; changes in pressure can affect temperature and vice versa, especially in thermodynamic contexts. If Min-Max Normalization is applied indiscriminately, it may distort these relationships, leading to misleading interpretations or inaccurate modeling. Therefore, the choice of normalization method should be guided by the analysis goals, the nature of the data, and the physical principles governing the relationships between these variables. In scenarios where relative differences and relationships are critical, preserving the original scale and context of the data may be more important than scaling them for uniformity.

When pressures are normalized independently, the relative differences between various measurement stations, which are critical to estimating flow rates, can be obscured inadvertently. This loss of relational information can negatively impact the model's ability to capture the underlying dynamics governing production rates. To address this challenge, a more effective approach is to first compute pressure deltas between stations and then apply normalization to these derived features. By doing so, the essential information encoded in the pressure differences is retained while simultaneously ensuring that the data scale is suitable for machine learning algorithms. This strategy improves the interpretability of input features and helps the models establish a more direct correlation with the target variables, such as oil rate.

Figure 4.10 visually compares the raw pressures and the delta pressures, calculated from their initial values. It is evident that the delta pressure feature provides a clearer and more direct relationship with oil production rates. This approach aligns with (TIAN, 2018) work, where such derived features have demonstrated a particular value for regression tasks.

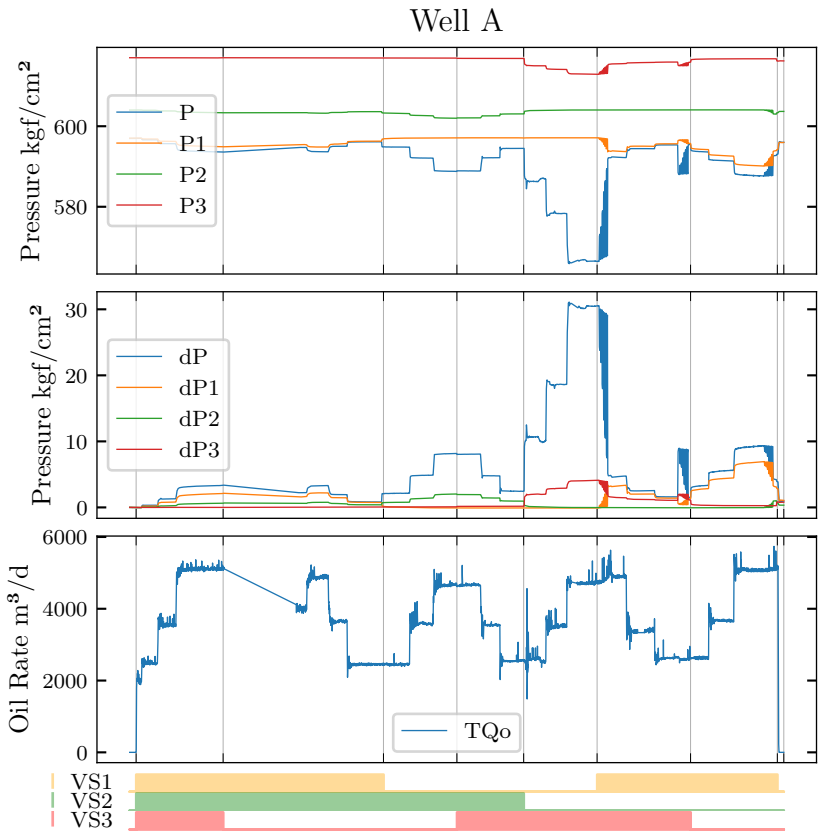

Figure 4.10: Features: Pressure deltas from the initial condition.

Furthermore, Figure 4.11 illustrates the normalized deltas, highlighting how this preprocessing step maintains the informativeness of the features while making them compatible with a variety of machine learning algorithms.

By carefully structuring the normalization process and prioritizing the derivation of physically meaningful features before scaling, this methodology ensures that critical operational information is preserved, ultimately supporting the development of more accurate and robust predictive models.

4.4 Unsupervised Learning

In this section, the unsupervised learning applications are discussed.

4.4.1 Dimensionality Reduction

First, we discuss the dimensionality reduction results. The analyses performed were PCA and t-SNE.

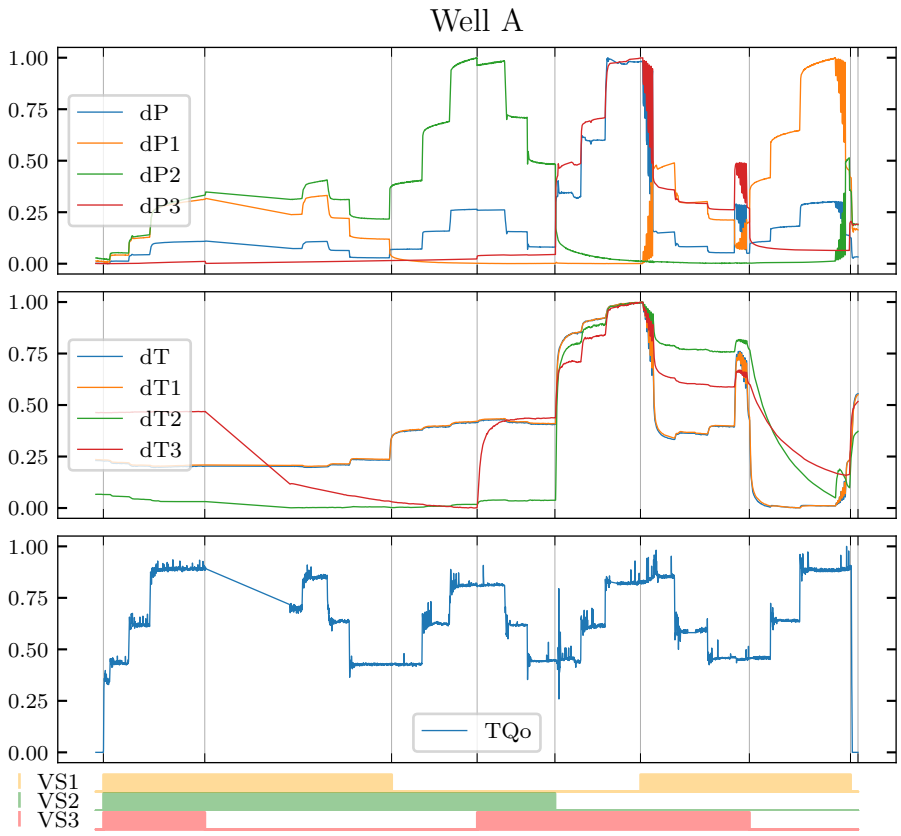

Figure 4.11: Features: Pressure deltas and temperature deltas scaled.

4.4.1.1 PCA

With the processed data set, a Principal Component Analysis (PCA) (JOLLIFFE; CADIMA, 2016) was performed, demonstrating that it is possible to transform the data set in a way that alleviates computational load by reducing at least three variables from the original dimension of the problem, Fig. 4.12.

Although PCA is effective for capturing variance and reducing dimensionality, it often sacrifices interpretability, as the principal components may not correspond directly to the original features or have meaningful interpretations in the context of the data. This loss of interpretability can be problematic, especially in fields such as oil and gas, where understanding the relationships between specific features is crucial for decision making. Consequently, when the goal is not just to improve model performance, but also to maintain or enhance the interpretability of the results, relying on PCA may not be the best choice. In such cases, it might be more beneficial to use feature selection methods or other dimensionality reduction techniques that retain the original feature space, allowing stakeholders to understand the implications of the

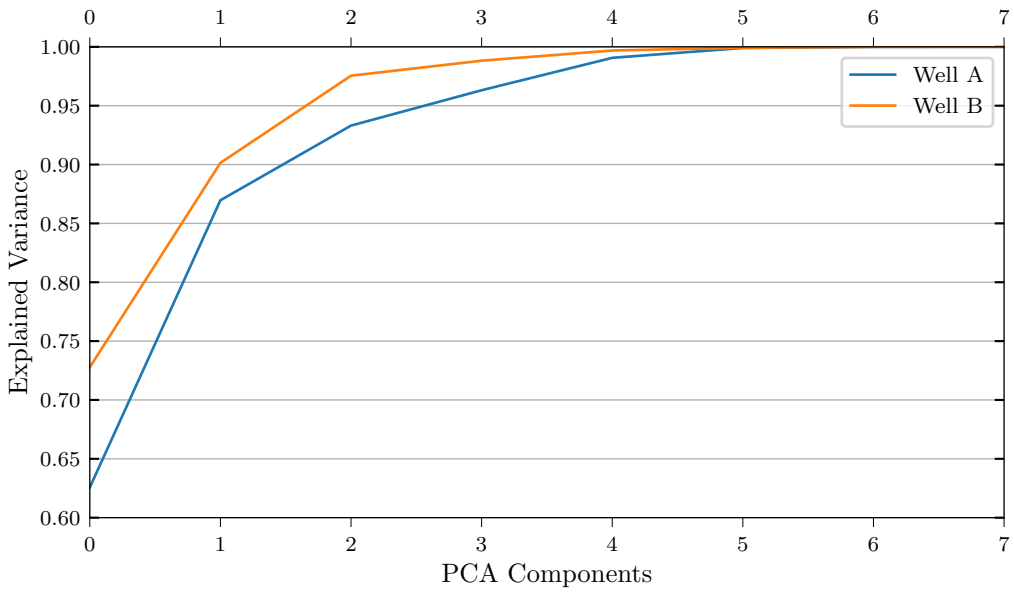

Figure 4.12: PCA Analysis for Well A and Well B.

model based on the actual features rather than abstract components. This need for interpretability often outweighs the computational efficiencies gained through PCA, making it essential to consider the context and objectives of the analysis when deciding whether to apply this technique.

4.4.1.2 t-SNE

A graphical representation in two dimensions of the problem was also performed using the t-SNE technique (MAATEN; HINTON, 2008). However, this representation did not show a clear separation between the open and closed valve classes in this two-dimensional space; see Fig. 4.15. However, for subsequent visualization purposes, the t-SNE technique was used.

Despite the different approaches of the techniques employed, both PCA and t-SNE revealed that the data set cannot be effectively reduced to a limited number of dimensions without incurring a significant loss of information. Consequently, given the small size of the data set, the decision was made to proceed with the analyses without implementing dimensionality reduction.

4.4.2 Clustering

Within the scope of the analysis, unsupervised learning, specifically clustering, will be used to identify hidden groups or patterns within the data.

In addition to simply identifying clusters, the approach enables the use of clusters as a new way to classify valve status in a multivariate context.

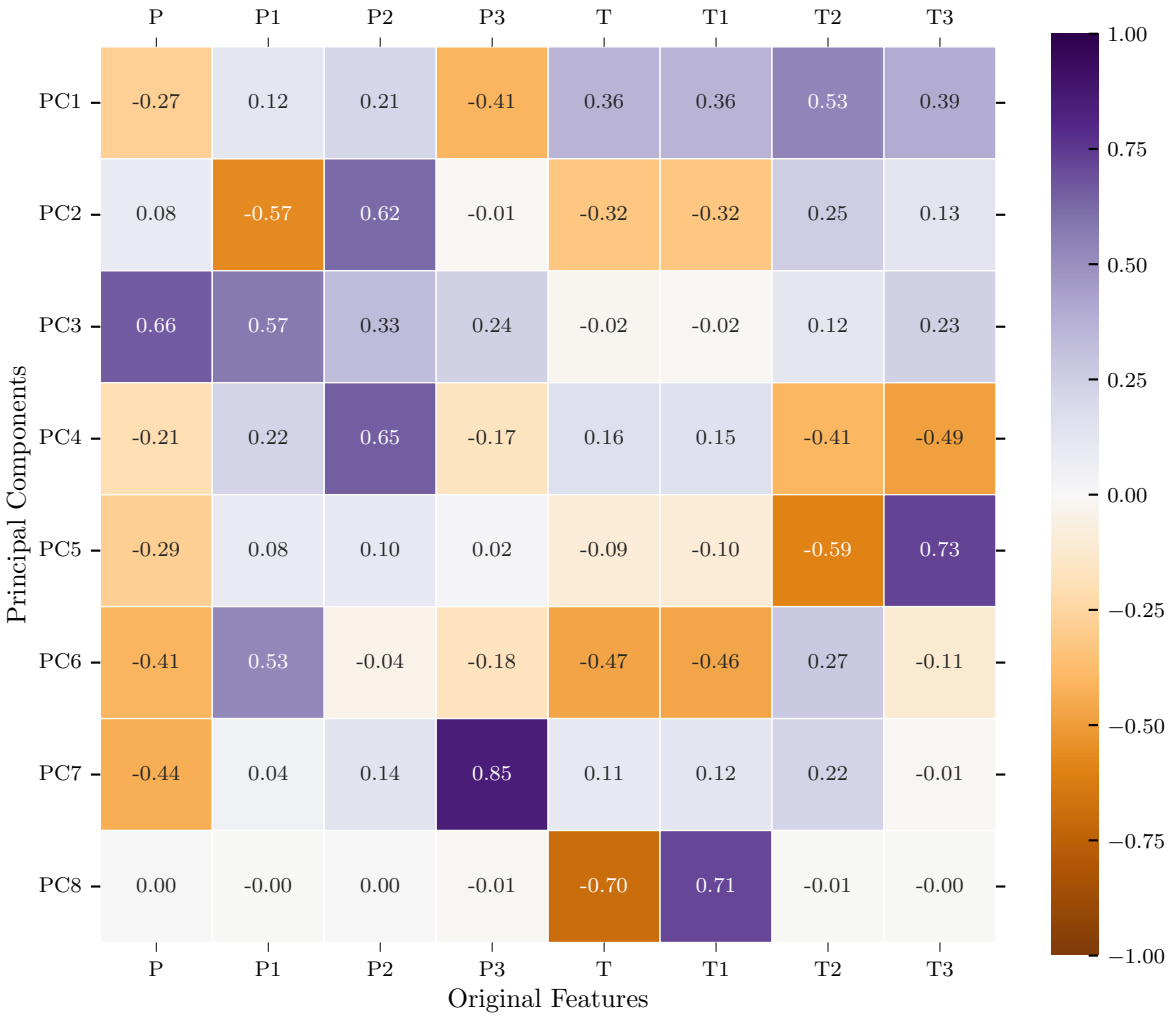

Figure 4.13: PCA Component loadings for Well A.

The relationship between assigned clusters and valve opening combinations is explored, developing a clustering model that can be applied to evaluate other data points as a classification model.

To visualize the clustering, we used the t-SNE representation in two dimensions, and the colors are the clusters. For each algorithm, the silhouette value is printed in the title to measure how well the clustering performed.

4.4.2.1 Cluster Analysis

Clustering is a fundamental technique in machine learning that facilitates the grouping of similar data points based on their inherent characteristics, enabling the extraction of meaningful insights from complex data sets. In this study, a thorough analysis of the histograms revealed a non-continuous distribution of the data, characterized by significant concentrations at specific values, which indicates the presence of distinct clusters within the data set,

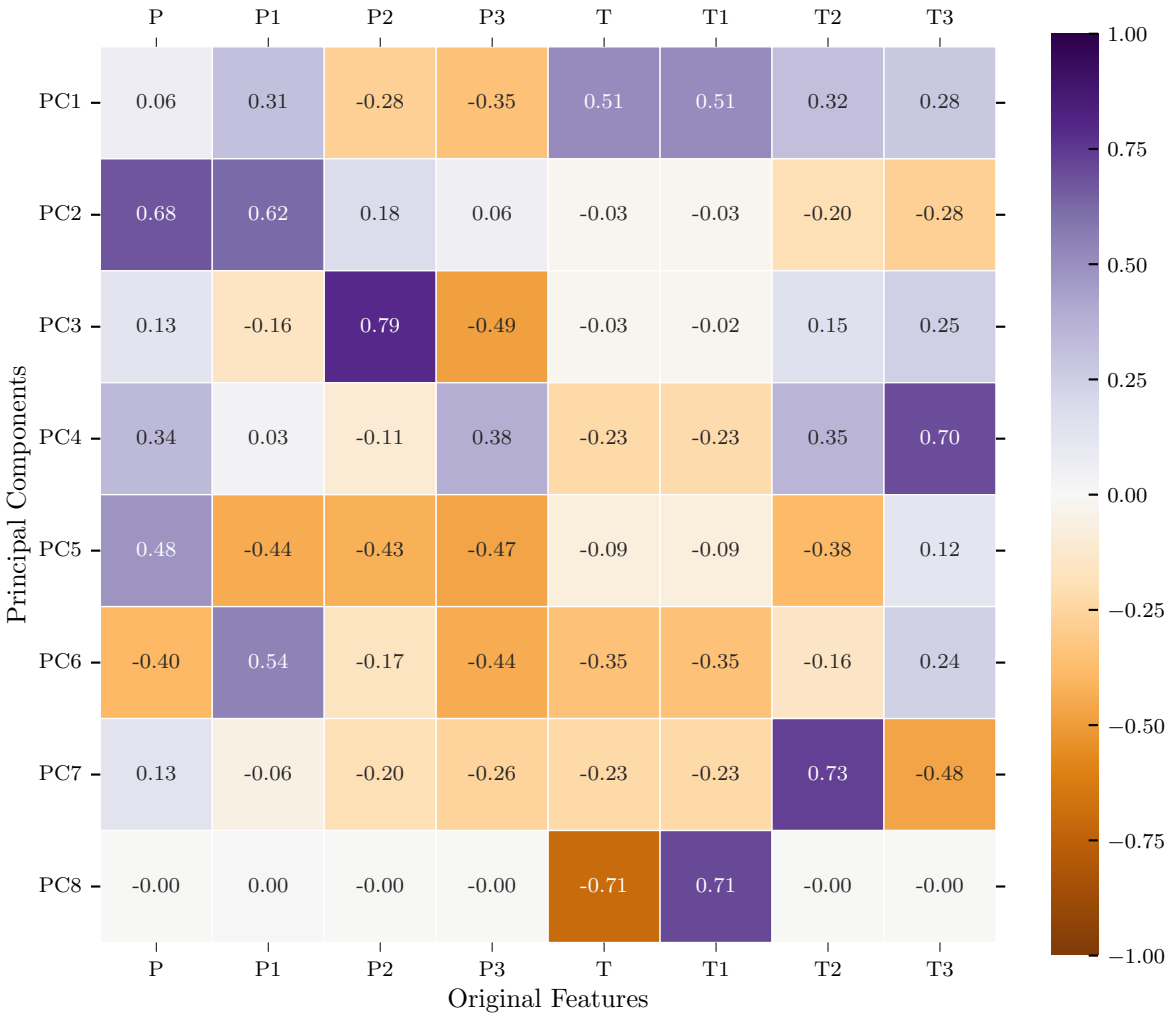

Figure 4.14: PCA Component loadings for Well B.

Figures 4.3 and 4.4.

Number of Clusters When cluster analysis is performed, one of the main hyperparameters in most algorithms is the number of clusters desired. This category of algorithms was used in our study to compare the performance of clustering on the same basis.

For each well, the ideal number of clusters was established first by analyzing the silhouette score as a function of the number of clusters for the K-Means, Hierarchical, and GMM algorithms. This method assesses the quality of clustering by considering the distance between points within a cluster and the distance to the nearest points in other clusters. A silhouette value close to 1 indicates good separation between clusters.

However, in this analysis, the silhouette score, which measures how similar an object is to its own cluster compared to other clusters, did not provide a clear indication of the optimal number of clusters, as it increased

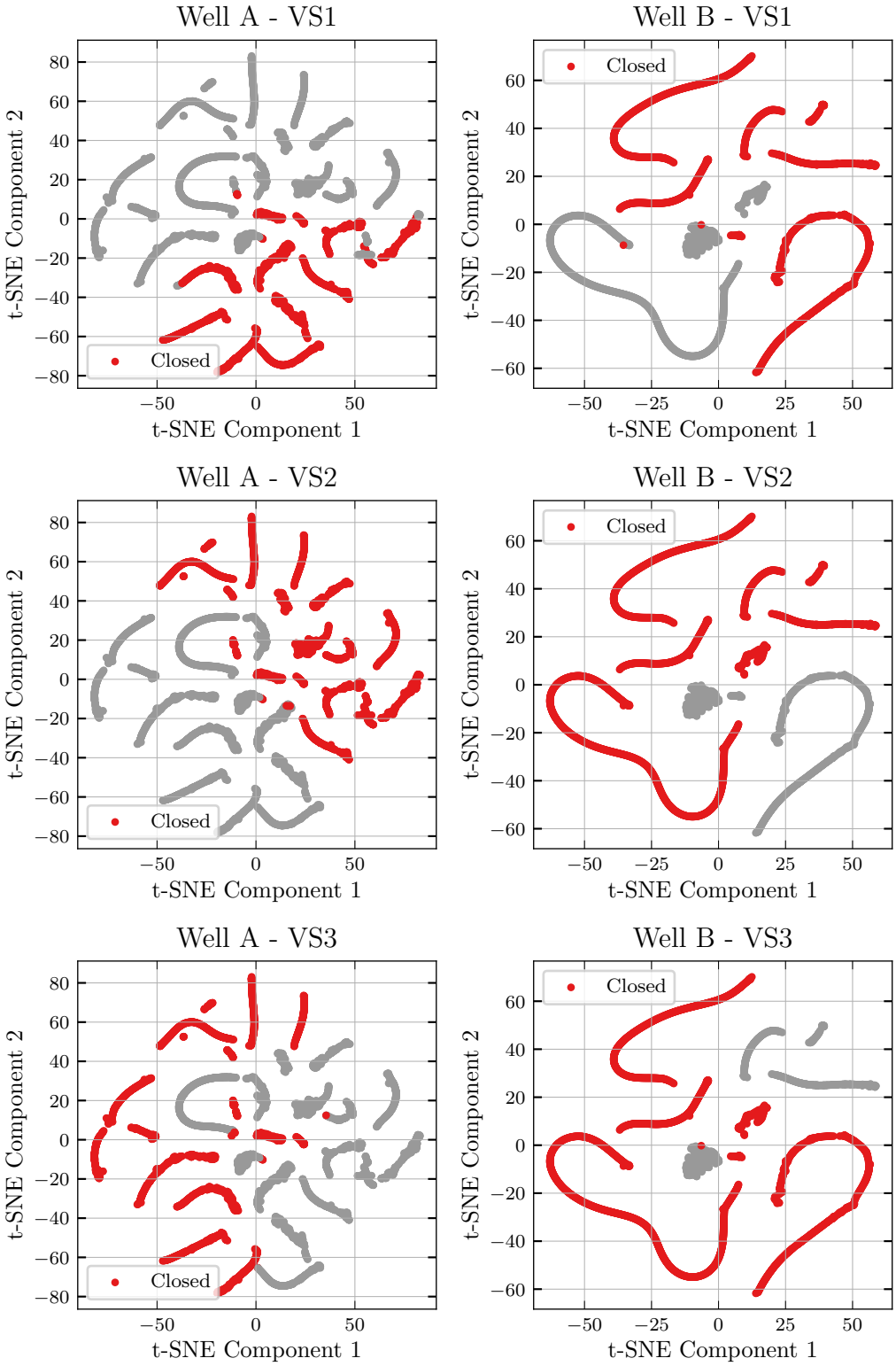

Figure 4.15: t-SNE Representation of Well A and Well B data sets.

consistently with the number of clusters for well A, Figure 4.16. This persistent increase suggests that more clusters may always provide better separation, but it fails to pinpoint a specific threshold beyond which additional clusters do not significantly enhance cluster quality, making it challenging to identify a definitive optimal cluster count based solely on the silhouette score. For well B, an optimal number of clusters could be determined, five clusters, Figure 4.17.

As the silhouette score was not definitive to establish the number of clusters for well A, another method was used, the Elbow Method. This method involves plotting the Within-Cluster Sum of Squares (WCSS) against the number of clusters and identifying the point where adding more clusters results in a marginal decrease in WCSS.

The elbow method is a widely used technique for determining the optimal number of clusters in a data set, particularly within K-means clustering. This method involves running the clustering algorithm across a range of possible cluster counts and calculating the WCSS for each value. By plotting the WCSS against the number of clusters, one can observe the trend and identify an "elbow" point where the rate of decrease in WCSS sharply changes, indicating the most appropriate number of clusters to use.

The application of the elbow method provided a graphical representation of the clustering performance, revealing a clear inflection point around 8 clusters for well A, Figure 4.18 and around 5 clusters for well B, Figure 4.19.

Using the same number of clusters, the algorithms have a similar performance as can be seen visually in Figures 4.20 and 4.21. By comparing the silhouette scores, we have a slightly better result for the k-means algorithm.

Mean Values for Each Cluster After defining the clusters, a detailed analysis of the mean pressure and temperature values is performed for each identified cluster. This allows for a better understanding of the distinctive characteristics of each group and how these features relate to the operating conditions of the wells.

The clustering was performed using the pressure and temperature data only, but the mean values for the valve status are plotted together to verify how the cluster relates to the combination of valve openings.

To be able to compare the clusters, we used a heat map with the resulting clusters on the y-axis and the features on the x-axis. The color is the mean value of the corresponding features in the cluster. White is used as a zero value to highlight when there is a high mean value for the cluster.

This comprehensive analysis culminated in the observation that the ex-

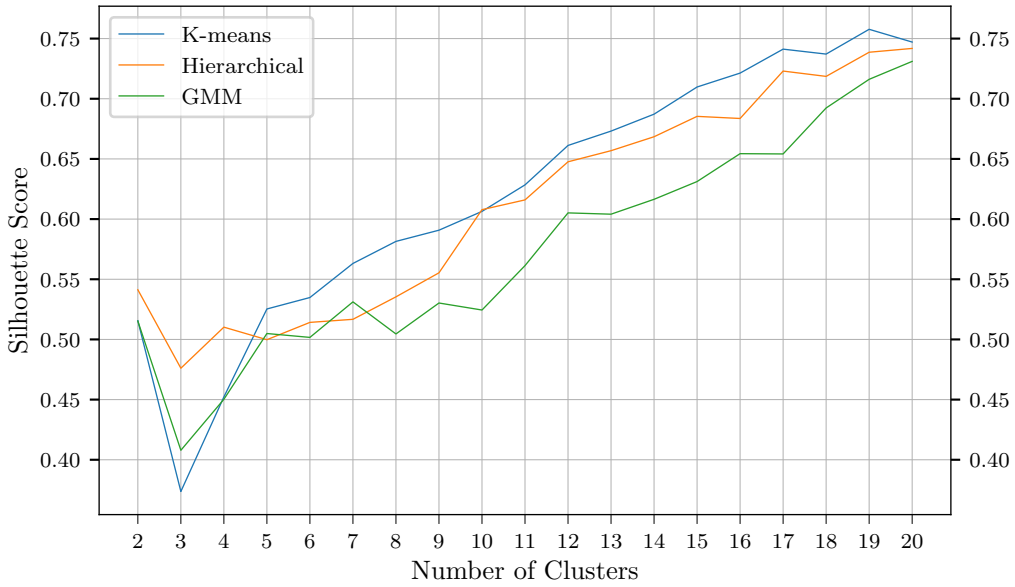

Figure 4.16: Silhouette score for increasing number of clusters for K-means, Hierarchical and GMM for well A.

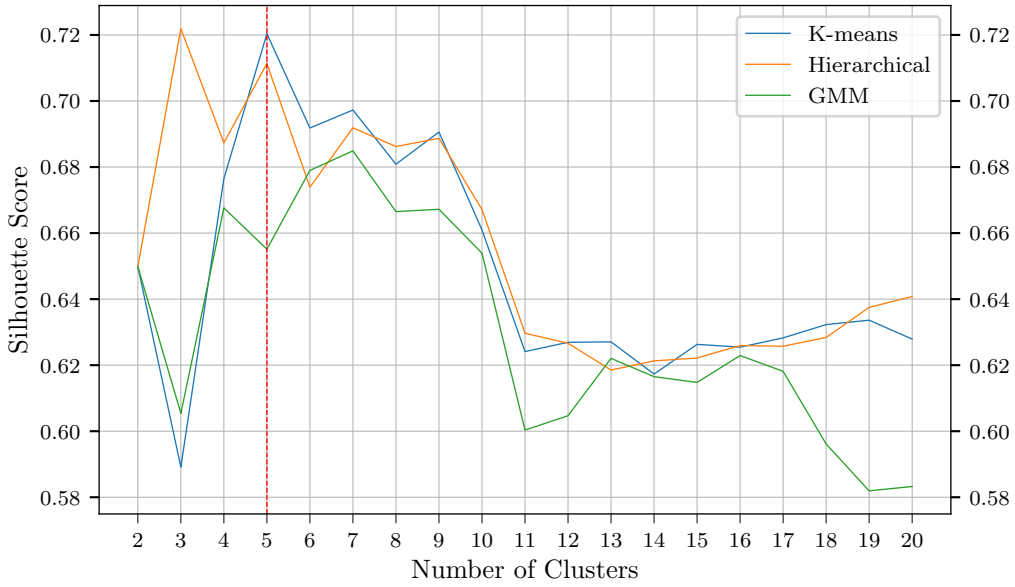

Figure 4.17: Silhouette score for increasing number of clusters for K-means, Hierarchical and GMM for well B.

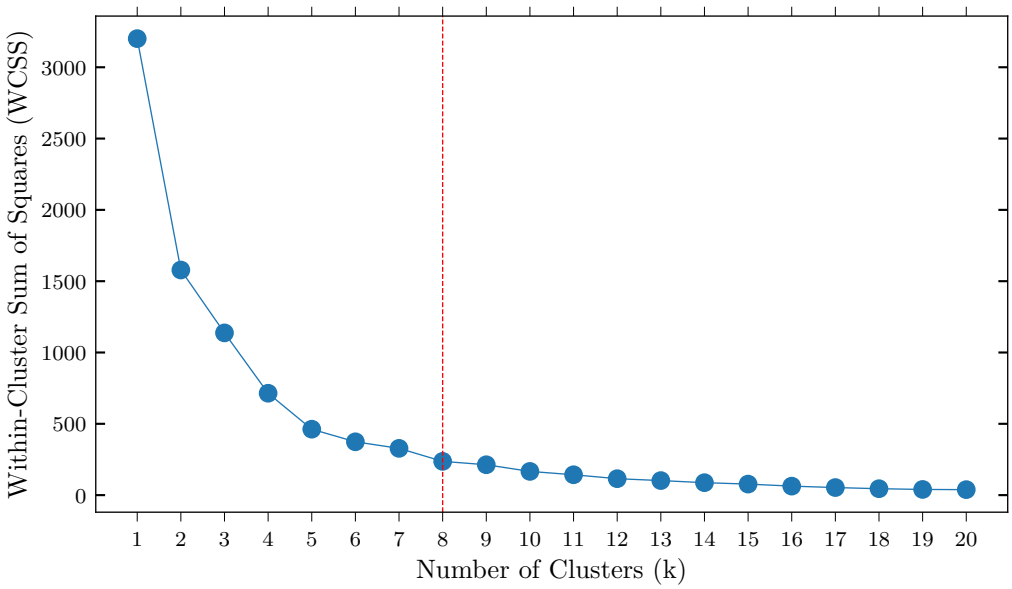

Figure 4.18: Elbow Method for Optimal k for well A.

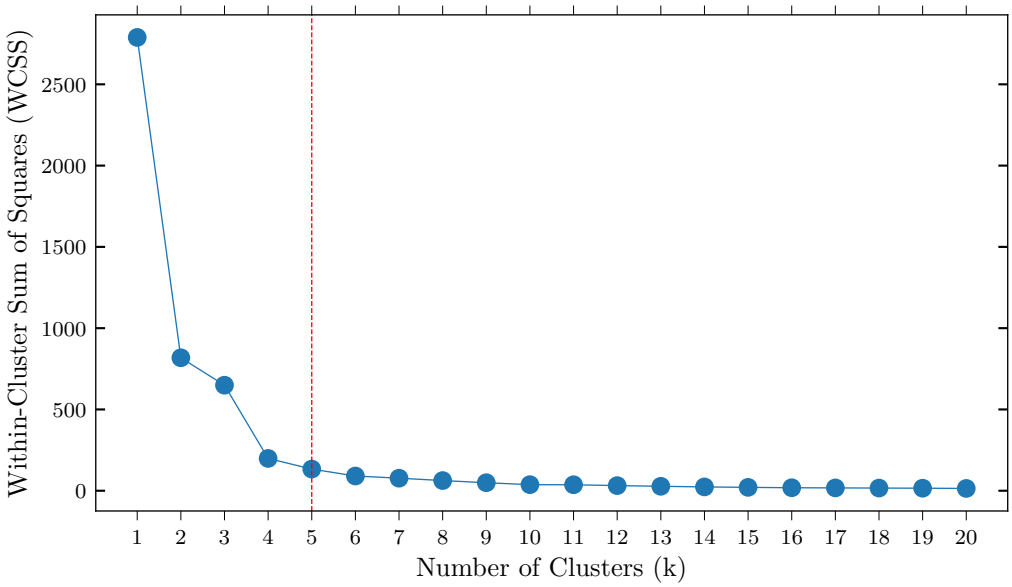

Figure 4.19: Elbow Method for Optimal k for well B.

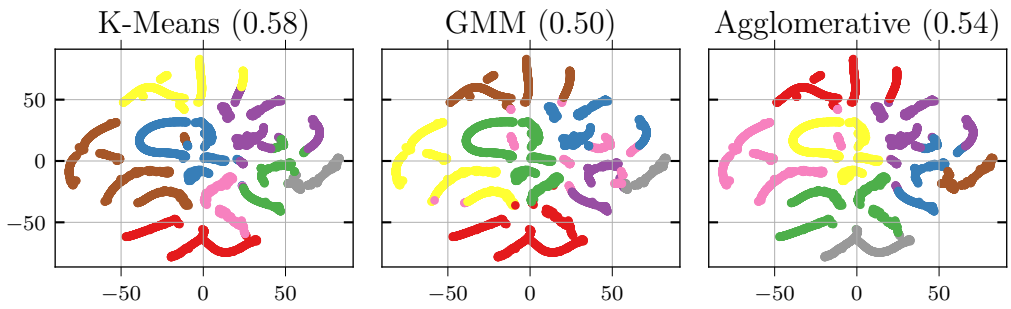

Figure 4.20: Clustering result for 8 clusters well A.

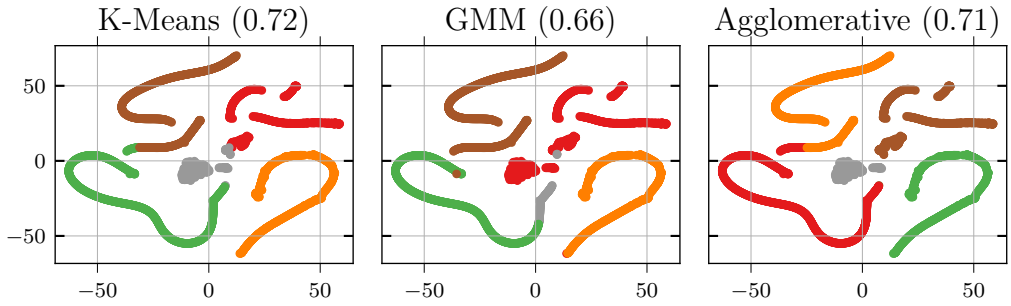

Figure 4.21: Clustering result for 5 clusters well B.

amination of the mean feature values within each cluster effectively illustrated the relationship between the clusters and the combinations of valve openings, highlighting the utility of clustering techniques to uncover intricate patterns within complex data, Figures 4.22 and 4.23.

There was a good match between clusters and combinations of valve openings, Figures 4.22 and 4.23. As the data set for well A had few points when all zones were closed, it was not possible for the clustering to retrieve this pattern well.

Clustering Valve Status Classification To build a classification model based on clustering, we can round the values of mean valve status for the train data set and use the model for further classification. This process takes Figures 4.22, 4.23 to Figures 4.24, 4.25. For a new point, the model assigns a cluster, then the classification is performed using the corresponding cluster values of the valve status.

To assess the quality of the proposed methodology, three evaluations were performed. In each figure, we have on top a confusion matrix showing the classification parameters and on the bottom a comparison between predicted and real values.

Train and test within the same well for well A, Figures A.1 and A.2.

Train and test within the same well for well B, Figures A.3 and A.4.

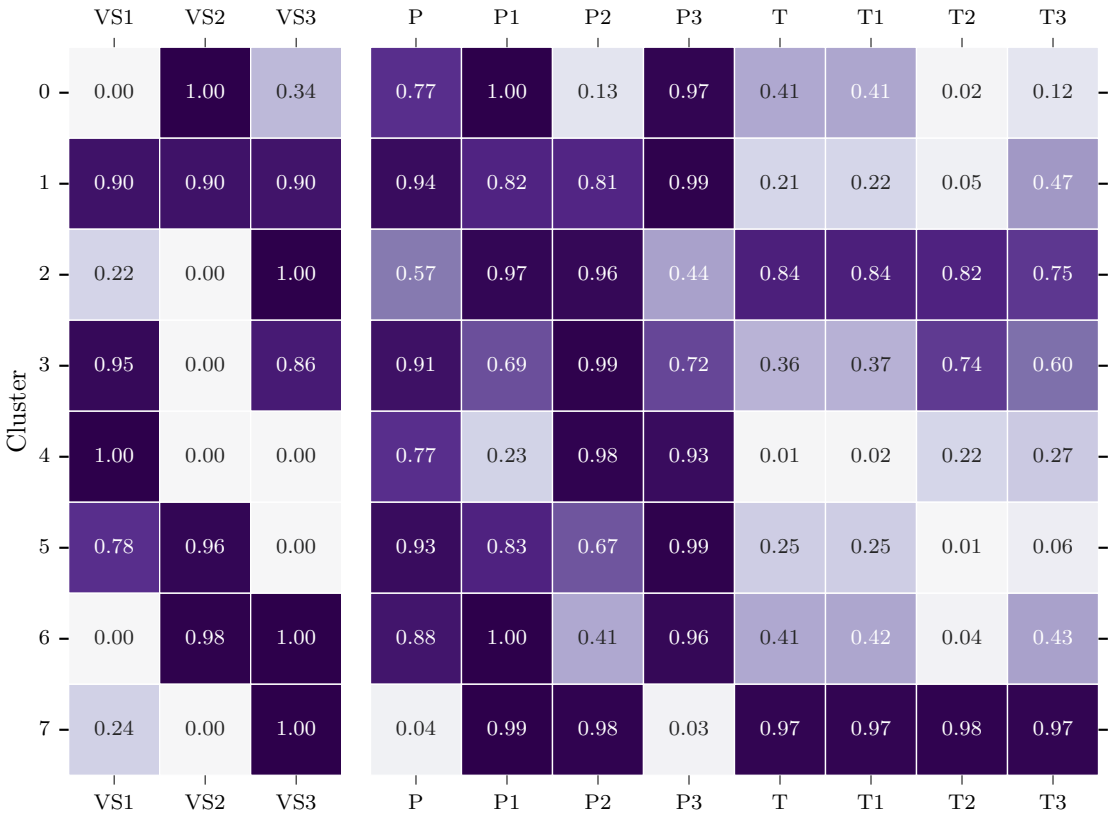

Figure 4.22: Mean feature values for each cluster well A.

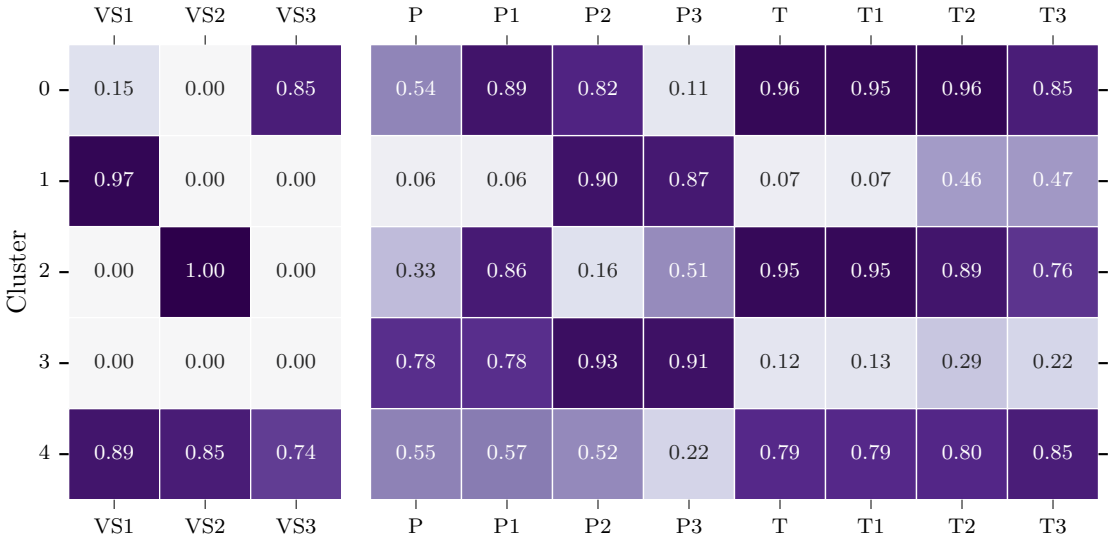

Figure 4.23: Mean feature values for each cluster well B.

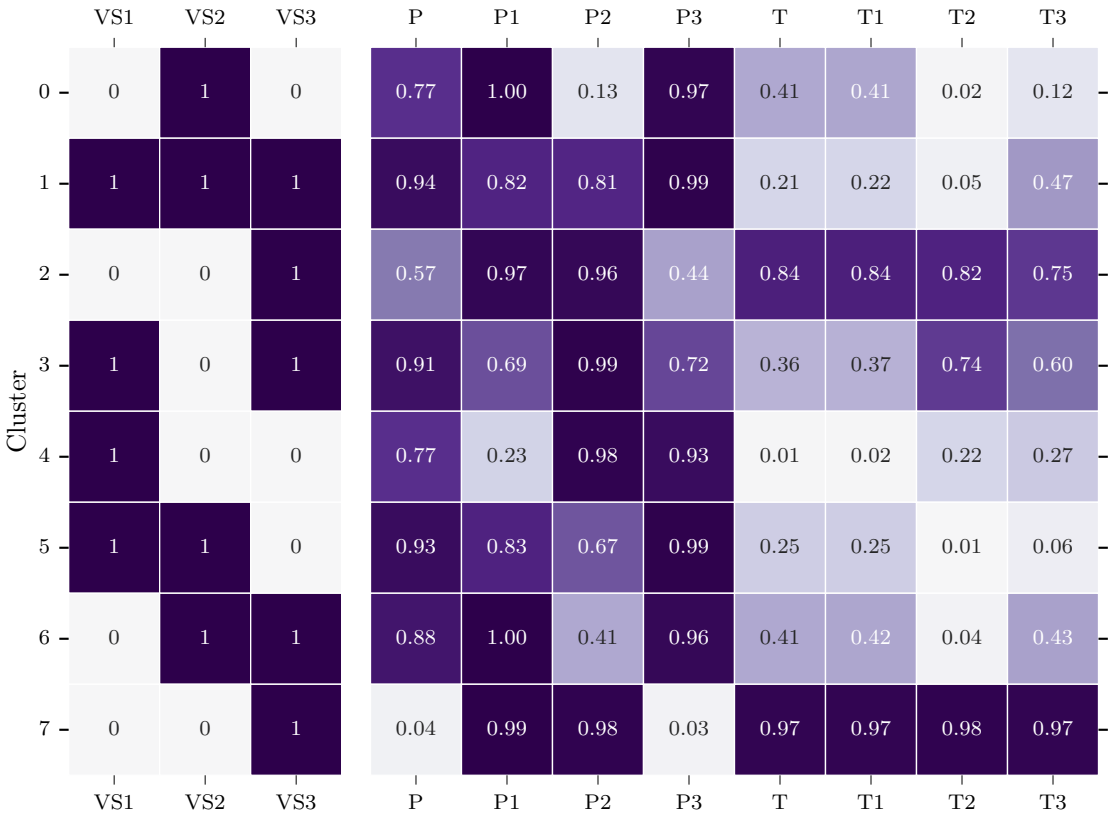

Figure 4.24: Rounded mean feature for valve status for each cluster well A.

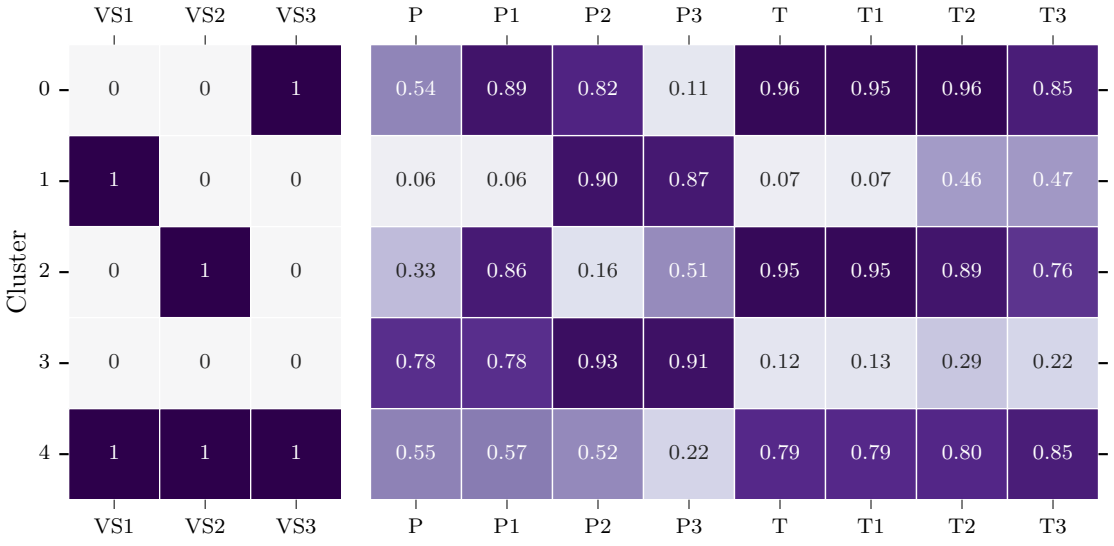

Figure 4.25: Rounded mean feature for valve status for each cluster well B.

Train	**Test**	**Valve**	**Accuracy**	**Recall**	**Precision**	$F1$
WA	WA	VS1	0.88	0.94	0.89	0.92
WA	WA	VS2	0.97	1.00	0.96	0.98
WA	WA	VS3	0.92	0.92	0.93	0.93
WB	WB	VS1	0.95	0.93	0.95	0.94
WB	WB	VS2	0.99	1.00	0.96	0.98
WB	WB	VS3	0.97	1.00	0.81	0.90
WA	WB	VS1	0.68	0.74	0.54	0.63
WA	WB	VS2	0.36	0.00	0.00	0.00
WA	WB	VS3	0.70	1.00	0.30	0.46

Table 4.1: Model Metrics for Clustering Classification.

Train well A and test well B, Figures A.5 and A.6.

In Table 4.1, we have a summary of the results with the calculated classification metrics for the proposed cluster classification.

Using this approach with raw pressure and temperature data, a good result is obtained by training and testing on the same well.

Transfer from one well to another using clustering of raw pressure and temperature data did not provide good results. By comparing the mean scaled pressure and temperature values for the same combination of valve openings for well A (Figure 4.24) and well B (Figure 4.25), it is evident that the clusters are very different. The relative productivity of each zone in each well is very different, making transfer between wells not possible.

4.5 Supervised Learning

In this section, the supervised learning applications are detailed. The first is the use of classification strategies for each valve status estimation, and the second is the regression of the oil rate based on pressure and temperature sensor data.

4.5.1 Well Status Identification

The described workflow was applied to the upper zone valve and then the model that performed best was adapted to predict the valve status of other zones. The initial model was built using the variables that were more correlated with the upper valve status, the tubing pressures, and the upper annulus; Figs. A.7 and A.8.

The first approach was to add the annulus pressures from other zones, a 10% improvement in the false-positive rate was achieved, Figs. A.9 and A.10.

Adding the temperatures did not improve the results; Figs. A.11 and A.12.

Upper Zone As the model still delivered a high rate of false positives (Figs. A.11 and A.12), features were constructed to improve the results. The first attempt was to create simple features using the temporal relationship between the variables based on the physics of the problem.

Combining these features with the initial pressures, a level of 10% of model error was obtained; Figs. A.13 and A.14.

Further features were constructed based on (TIAN, 2018) on flow estimation based on pressure convolution. For each pressure, four features involving pressure and time were derived.

With the selected features, a result that showed an error rate of approximately 5% was achieved, as illustrated in Figs. A.15 and A.16 and Table 4.2. However, it is evident that the transition from open state to closed state could be further optimized. Furthermore, given that the data are sourced from real-world conditions, there are peaks that require further investigation and resolution.

Intermediate Zone In the transfer to the intermediate zone, we achieved a significant result with an error below 0.5%. In this zone, there was only a valve closure, and the model was able to capture the transition well; Figs. A.17 and A.18 and Table 4.3.

Lower Zone For the lower zone, where the expectation was for the best result, as there is less interference with the other zones, the actual data exhibited some spurious behaviors that indicated transitions that did not actually occur, leading to a worse outcome than in the other zones; Figs. A.19 and A.20 and Table 4.4.

4.5.2 Oil Rate Prediction

To estimate the total flow rate, the initial approach was to use the features dP and P_{TC1} and P_{TC2} of the tubing sensor, as (TIAN; HORNE, 2019) have reported good results in the reconstruction of the flow rate using this set. However, significant challenges emerged. The algorithms were unable to capture flow behavior, failing even in the training set A.21. Linear regression, in particular, resulted in negative flow rate values, while other algorithms performed slightly better but did not adequately model the training data A.22.

Model	**Accuracy**	**Recall**	**Precision**	$F1$
Logistic Regression	0.91	1.00	0.74	0.85
Support Vector Classifier	0.95	0.98	0.86	0.91
k-Nearest Neighbors	0.87	0.95	0.70	0.81
Decision Tree Classifier	0.84	1.00	0.62	0.76
Random Forest Classifier	0.69	1.00	0.46	0.63

Table 4.2: Model Metrics for Upper Zone Classification.

Model	**Accuracy**	**Recall**	**Precision**	$F1$
Logistic Regression	0.99	1.00	0.95	0.97
Support Vector Classifier	1.00	1.00	0.97	0.99
k-Nearest Neighbors	0.97	1.00	0.84	0.91
Decision Tree Classifier	0.53	0.09	0.61	0.15
Random Forest Classifier	0.96	1.00	0.79	0.88

Table 4.3: Model Metrics for Intermediate Zone Classification.

Model	**Accuracy**	**Recall**	**Precision**	$F1$
Logistic Regression	0.77	1.00	0.28	0.44
Support Vector Classifier	0.80	1.00	0.31	0.48
k-Nearest Neighbors	0.57	1.00	0.18	0.30
Decision Tree Classifier	0.57	1.00	0.18	0.30
Random Forest Classifier	0.74	1.00	0.26	0.41

Table 4.4: Model Metrics for Lower Zone Classification.

The situation was further complicated by the test set, which consisted of valve combinations that were not present in the training set. Specifically, the situation where only reservoir A is producing.

In response to these limitations, the feature domain was expanded. By including Tian pressure features (TIAN, 2018) from additional sensors and incorporating pressure deltas between sensors, the models began to capture flow behavior more effectively in the training phase. With this enriched feature space, all algorithms, including linear regression, were able to fit the training data successfully; see Figure A.23 for reference. The construction of these nonlinear features allowed the models to represent complex relationships in the data.

A visual comparison of the results in the training set revealed similar behaviors between the models in the training set, but with very different predictions in the test set; see Figure A.24 for the evaluation of the DTR prediction.

Support Vector Regression (SVR) excelled in modeling all production plateaus and transitions, following pressure variations, and reducing output

noise; see Figure A.25.

Despite these improvements in the training phase, the results of the test set remained unsatisfactory due to the presence of previously unseen valve combinations. To address this, the data set was subdivided according to valve combinations and algorithms were trained specifically for each situation using the determined feature space. This targeted approach yielded good results for both training and test sets within each valve combination scenario; see Figure A.26, where the best model for each valve combination was chosen.

The same methodology was applied to Well B, demonstrating similar performance; see Figure A.27.

To evaluate the generalizability of the proposed method, the model trained in Well A was tested in Well B, see Figure A.28. As anticipated, the results were poor. Even among wells in the same reservoir, there were substantial differences in zone production, making direct model transfer impractical.

In conclusion, the results indicate that effective modeling requires segmentation by well and valve combination to ensure representative and accurate predictions for each scenario. Furthermore, the analysis demonstrated that it is possible to build a valve status classifier directly from pressure and temperature sensor data, which can be utilized to enhance flow rate estimation.

Train	**Test**	**R2**	**MAE**	**MSE**	**RMSE**	**MAPE**
WA	WA	0.8011	0.0224	0.0060	0.0772	2.63%
WB	WB	0.9797	0.0232	0.0019	0.0434	4.79%
WA	WB	0.6454	0.1588	0.0361	0.1900	inf%

Table 4.5: Model Metrics for Oil Rate Regression.

5
Conclusion

This thesis aimed to apply machine learning methodologies to a real-world oil and gas data set using purely data-driven models. Although the initial specific objective of production allocation among production zones was not fully achieved due to the lack of labeled datasets necessary for robust model training and validation, the work extensively applied a broad range of machine learning techniques to the available dataset. These included unsupervised methods such as dimensionality reduction and visualization techniques, as well as supervised methods including classification and regression. This comprehensive application allowed valuable analysis and insight into the pressure and temperature sensor data collected from oil and gas wells.

The experience gained throughout this research highlighted both the potential and current limitations of relying exclusively on data-driven approaches for complex production problems. In particular, the scarcity of labeled data emphasizes the need to integrate physical modeling with machine learning techniques to improve the accuracy and reliability of the model. Aligned with the thesis objectives of exploring machine learning capabilities in this context, future work should pursue hybrid modeling strategies that combine domain expertise and data-driven analytics. Such approaches are expected to significantly improve production allocation and optimization in complex multi-zone oil wells, advancing the state-of-the-art in the field.

5.1
Main Contributions

This thesis explored advanced machine learning applications for pressure and temperature sensor data in oil and gas production, focusing on the interpretation of Permanent Downhole Gauge (PDG) data, virtual flow metering (VFM), and valve status classification. Through a comprehensive literature review, the main challenges and opportunities in data-driven approaches for well monitoring and operation were identified.

Several machine learning techniques were implemented and evaluated using a complex multi-zone data set, which presented a more realistic and challenging scenario compared to the single-zone data sets commonly found in the literature.

Clustering methods, including Gaussian Mixture Models (GMM), hierarchical clustering, and k-means, were employed to uncover patterns associated

with various valve combinations. These patterns facilitated the development of a clustering-based classification approach to infer valve status.

Recognizing the valve status variable as a key operational factor, classification models were built to directly predict valve behavior from sensor data. The robustness and generalizability of these models were validated by training on data from one well and testing on data from another, demonstrating the importance of incorporating pressure-dependent and time-dependent features. Data normalization and the use of TimeSeriesSplit for hyperparameter optimization were also critical for handling the temporal nature of the data and ensuring model performance.

In addition, regression models were applied to estimate the total oil rate using features derived from (TIAN; HORNE, 2019) methodology. Segmenting the data set by zone combinations led to improved predictive accuracy, highlighting the importance of context-based modeling in multi-zone environments.

The results underscore the potential of machine learning techniques to enhance operational insight, automate well monitoring, and support decision making in oil and gas production. The approaches developed in this thesis provide a foundation for future research and practical applications, particularly in complex well configurations.

5.2 Future Work

Based on the findings and methodologies developed in this thesis, several promising avenues for future research are identified.

Extending to Other Well Completions and Sensor Placements The strategies and models presented here can be adapted and tested in a wider variety of well completions, including those with different sensor configurations and placements. This will enable validation and refinement of the approaches in diverse operational scenarios, enhancing their robustness and applicability.

Exploring Advanced Machine Learning Architectures Future work could investigate the use of deep learning models, such as Long Short-Term Memory (LSTM) networks and Transformer architectures, which are particularly effective in modeling sequential and time-series data. These architectures may capture more complex temporal dependencies in pressure and temperature data, potentially leading to improved predictive performance in valve status classification and flow metering tasks.

Feature Selection Strategies Implementing systematic feature selection methods, such as forward selection, can help identify the most informative features from sensor data, reduce model complexity, and improve generalization. This approach may also facilitate the interpretation of results and the discovery of new physical insights relevant to well operation.

Production Allocation per Reservoir Zone using Hybrid Approaches A crucial challenge for future research is the development of reliable production allocation models for each reservoir zone. Purely data-driven approaches often require extensive labeled data sets, which are not always available in real-world scenarios. Therefore, integrating physical models with machine learning can help generate synthetic data sets that reflect the underlying physical processes of multiphase flow and reservoir behavior. These models can serve as a basis for training and validating machine learning algorithms, bridging the gap between limited field measurements and the data requirements of advanced analytics. Hybrid approaches also improve the interpretability of the model and ensure that the predictions remain consistent with established physical principles.

6
Bibliography

AJAYI, A.; FASASI, T.; OKUNS, G. Real Time Flow Estimation Using Virtual Flow Measurement Techniques: A Field Application in Intelligent Well Completion. In: **Nigeria Annual International Conference and Exhibition**. Lagos, Nigeria: SPE, 2012. p. SPE–162948–MS. Disponível em: https://onepetro.org/SPENAIC/proceedings/12NAICE/12NAICE/SPE-162948-MS/159287.

ALJUBRAN, M. J.; HORNE, R. Surrogate-Based Prediction and Optimization of Multilateral Inflow Control Valve Flow Performance with Production Data. **SPE Production & Operations**, v. 36, n. 01, p. 224–233, fev. 2021. ISSN 1930-1855. Disponível em: https://doi.org/10.2118/200884-PA.

ALJUBRAN, M. J. J.; HORNE, R. Prediction of Multilateral Inflow Control Valve Flow Performance Using Machine Learning. **SPE Production & Operations**, v. 35, n. 03, p. 691–702, ago. 2020. ISSN 1930-1855, 1930-1863. Disponível em: https://onepetro.org/PO/article/35/03/691/450761/Prediction-of-Multilateral-Inflow-Control-Valve.

BALAJI, K. et al. Status of Data-Driven Methods and their Applications in Oil and Gas Industry. In: **SPE Europec featured at 80th EAGE Conference and Exhibition**. Copenhagen, Denmark: SPE, 2018. p. D031S005R007. Disponível em: https://onepetro.org/SPEEURO/proceedings/18EURO/18EURO/D031S005R007/216186.

BARRETT, E. et al. Improved Determination of Well Rate From Temperature and Pressure Distributions Along the Well. In: **North Africa Technical Conference and Exhibition**. Cairo, Egypt: SPE, 2012. p. SPE–150868–MS. Disponível em: https://onepetro.org/SPENATC/proceedings/12NATC/12NATC/SPE-150868-MS/159510.

BELYADI, H. **Machine learning guide for oil and gas using Python: a step-by-step breakdown with data, algorithms, codes, and applications**. Amsterdam: Gulf Professional Publishing, an imprint of Elsevier, 2021. ISBN 978-0-12-821929-4.

BERGSTRA, J.; BENGIO, Y. Random Search for Hyper-Parameter Optimization. **Journal of Machine Learning Research**, v. 13, n. 10, p. 281–305, 2012. ISSN 1533-7928. Disponível em: http://jmlr.org/papers/v13/bergstra12a.html.

BIKMUKHAMETOV, T.; JäSCHKE, J. Combining machine learning and process engineering physics towards enhanced accuracy and explainability of data-driven models. **Computers & Chemical Engineering**, v. 138, p. 106834, jul. 2020. ISSN 0098-1354. Disponível em: https://www.sciencedirect.com/science/article/pii/S0098135419313675.

BIKMUKHAMETOV, T.; JäSCHKE, J. First Principles and Machine Learning Virtual Flow Metering: A Literature Review. **Journal of Petroleum Science**

and Engineering, v. 184, p. 106487, jan. 2020. ISSN 0920-4105. Disponível em: https://www.sciencedirect.com/science/article/pii/S0920410519309088.

BRUNTON, S. L.; NOACK, B. R.; KOUMOUTSAKOS, P. Machine Learning for Fluid Mechanics. **Annual Review of Fluid Mechanics**, v. 52, n. Volume 52, 2020, p. 477–508, jan. 2020. ISSN 0066-4189, 1545-4479. Publisher: Annual Reviews. Disponível em: https://www.annualreviews.org/content/journals/10.1146/annurev-fluid-010719-060214.

CHENG, X. SSRN Scholarly Paper, **A Comprehensive Study of Feature Selection Techniques in Machine Learning Models**. Rochester, NY: Social Science Research Network, 2024. Disponível em: https://papers.ssrn.com/abstract=5154947.

DIASO, K. I. et al. Practical Hydrocarbon Allocation – A Machine Learning Approach. In: **SPE Nigeria Annual International Conference and Exhibition**. Lagos, Nigeria: SPE, 2023. p. D031S013R001. Disponível em: https://onepetro.org/SPENAIC/proceedings/23NAIC/23NAIC/D031S013R001/525957.

GRYZLOV, A. et al. Novel Methods for Production Data Forecast Utilizing Machine Learning and Dynamic Mode Decomposition. In: **Abu Dhabi International Petroleum Exhibition & Conference**. Abu Dhabi, UAE: SPE, 2020. p. D011S018R002. Disponível em: https://onepetro.org/SPEADIP/proceedings/20ADIP/20ADIP/D011S018R002/452566.

GRYZLOV, A.; SAFONOV, S.; ARSALAN, M. Intelligent Production Monitoring with Continuous Deep Learning Models. **SPE Journal**, v. 27, n. 02, p. 1304–1320, abr. 2022. ISSN 1086-055X. Disponível em: https://doi.org/10.2118/206525-PA.

GéRON, A. **Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: concepts, tools, and techniques to build intelligent systems**. Third edition. Beijing Boston Farnham Sebastopol Tokyo: O'Reilly, 2023. (Data science / machine learning). ISBN 978-1-0981-2246-1 978-1-0981-2597-4.

HARRIS, C. R. et al. Array Programming with Numpy. **Nature**, v. 585, n. 7825, p. 357–362, set. 2020. ISSN 1476-4687. Publisher: Nature Publishing Group. Disponível em: https://www.nature.com/articles/s41586-020-2649-2.

HUNTER, J. D. Matplotlib: A 2D Graphics Environment. **Computing in Science & Engineering**, v. 9, n. 03, p. 90–95, maio 2007. ISSN 1521-9615. Publisher: IEEE Computer Society. Disponível em: https://www.computer.org/csdl/magazine/cs/2007/03/c3090/13rRUwbJD0A.

JAMES, G. et al. **An Introduction to Statistical Learning: with Applications in Python**. Cham: Springer International Publishing, 2023. (Springer Texts in Statistics). ISBN 978-3-031-38746-3 978-3-031-38747-0. Disponível em: https://link.springer.com/10.1007/978-3-031-38747-0.

JOLLIFFE, I. T.; CADIMA, J. Principal Component Analysis: A Review and Recent Developments. **Philosophical transactions. Series A, Mathematical, physical, and engineering sciences**, v. 374, n. 2065, p. 20150202, abr. 2016. ISSN 1364-503X. Disponível em: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4792409/.

KADEM, M. A. et al. Real-Time Well Status Prediction Using Artificial Intelligence Techniques for Accurate Rate Allocation. In: **SPE Symposium and Exhibition - Production Enhancement and Cost Optimisation**. Kuala Lumpur, Malaysia: SPE, 2024. p. D021S006R001. Disponível em: https://onepetro.org/SPESM01/proceedings/25SM02/25SM02/D021S006R001/551858.

KARIMI, A. et al. Automated Well Status Identification in US Offshore Operations Using Machine Learning. In: **SPE Offshore Europe Conference & Exhibition**. Aberdeen, Scotland, UK: SPE, 2025. p. D021S009R007. Disponível em: https://onepetro.org/SPEOE/proceedings/25OE/25OE/D021S009R007/789359.

LIU, Y.; HORNE, R. N. Interpreting Pressure and Flow Rate Data from Permanent Downhole Gauges with Convolution-Kernel-Based Data Mining Approaches. In: **SPE Annual Technical Conference and Exhibition**. New Orleans, Louisiana, USA: SPE, 2013. p. D021S031R002. Disponível em: https://onepetro.org/SPEATCE/proceedings/13ATCE/13ATCE/D021S031R002/172765.

MAATEN, L. v. d.; HINTON, G. Visualizing data using T-sne. **Journal of Machine Learning Research**, v. 9, n. 86, p. 2579–2605, 2008. ISSN 1533-7928. Disponível em: http://jmlr.org/papers/v9/vandermaaten08a.html.

MCCRACKEN, M.; CHORNEYKO, D. Rate Allocation Using Permanent Downhole Pressures. In: **SPE Annual Technical Conference and Exhibition**. San Antonio, Texas, USA: SPE, 2006. p. SPE–103222–MS. Disponível em: https://onepetro.org/SPEATCE/proceedings/06ATCE/06ATCE/SPE-103222-MS/140266.

MCKINNEY, W. Data Structures for Statistical Computing in Python. **scipy**, maio 2010. Disponível em: https://proceedings.scipy.org/articles/Majora-92bf1922-00a.

NAGAO, M. et al. Reservoir Connectivity Identification and Robust Production Forecasting Using Physics Informed Machine Learning. In: . OnePetro, 2023. Disponível em: https://dx.doi.org/10.2118/212201-MS.

NEGASH, B. M.; HIM, P. C. Reconstruction of Missing Gas, Oil, and Water Flow-Rate Data: A Unified Physics and Data-Based Approach. **SPE Reservoir Evaluation & Engineering**, v. 23, n. 03, p. 1019–1030, ago. 2020. ISSN 1094-6470. Disponível em: https://doi.org/10.2118/199890-PA.

OLAMIGOKE, O.; ONYEALI, D. C. Machine learning prediction of bottomhole flowing pressure as a time series in the volve field. **International Journal of Frontiers in Engineering and Technology Research**, v. 2, n. 2, p. 020–039, jun. 2022. ISSN 27830497. Disponível em: https://frontiersrj.com/journals/ijfetr/content/machine-learning-prediction-bottomhole-flowing-pressure-time-series-volve-field.

PEDREGOSA, F. et al. Scikit-learn: Machine Learning in Python. **Journal of Machine Learning Research**, v. 12, n. 85, p. 2825–2830, 2011. ISSN 1533-7928. Disponível em: http://jmlr.org/papers/v12/pedregosa11a.html.

PóVOAS, M. D. S. et al. Artificial Intelligence in the Oil and Gas Industry: Applications, Challenges, and Future Directions. **Applied Sciences**, v. 15, n. 14,

p. 7918, jul. 2025. ISSN 2076-3417. Disponível em: https://www.mdpi.com/2076-3417/15/14/7918.

RAMCHARITAR, K.; RAMDHANIE, A. K. Using Machine Learning Methods to Identify Reservoir Compartmentalization in Mature Oilfields from Legacy Production Data. In: . OnePetro, 2021. Disponível em: https://dx.doi.org/10.2118/200979-MS.

SCHNITZLER, E. et al. Buzios Presalt Wells: Delivering Intelligent Completion In Ultra-Deepwater Carbonate Reservoirs. In: . OnePetro, 2021. Disponível em: https://dx.doi.org/10.4043/31116-MS.

SIRCAR, A. et al. Application of machine learning and artificial intelligence in oil and gas industry. **Petroleum Research**, v. 6, n. 4, p. 379–391, dez. 2021. ISSN 20962495. Disponível em: https://linkinghub.elsevier.com/retrieve/pii/S2096249521000429.

SORTICA, E. A. et al. Buzios: The Development of Well Construction in a Giant Pre-Salt Field. In: . OnePetro, 2023. Disponível em: https://dx.doi.org/10.4043/32246-MS.

TARIQ, Z. et al. A systematic review of data science and machine learning applications to the oil and gas industry. **Journal of Petroleum Exploration and Production Technology**, v. 11, n. 12, p. 4339–4374, dez. 2021. ISSN 2190-0558, 2190-0566. Disponível em: https://link.springer.com/10.1007/s13202-021-01302-2.

TIAN, C. **Machine learning approaches for permanent downhole gauge data interpretation**. Stanford University, 2018. Disponível em: https://search.proquest.com/openview/37732304bffc33f709898bd161da9bdf/1?pq-origsite=gscholar&cbl=18750&diss=y.

TIAN, C.; HORNE, R. N. Inferring Interwell Connectivity Using Production Data. In: **SPE Annual Technical Conference and Exhibition**. Dubai, UAE: SPE, 2016. p. D031S051R004. Disponível em: https://onepetro.org/SPEATCE/proceedings/16ATCE/16ATCE/D031S051R004/185026.

TIAN, C.; HORNE, R. N. Recurrent Neural Networks for Permanent Downhole Gauge Data Analysis. In: **SPE Annual Technical Conference and Exhibition**. San Antonio, Texas, USA: SPE, 2017. p. D011S008R007. Disponível em: https://onepetro.org/SPEATCE/proceedings/17ATCE/17ATCE/D011S008R007/193151.

TIAN, C.; HORNE, R. N. Applying Machine-Learning Techniques To Interpret Flow-Rate, Pressure, and Temperature Data From Permanent Downhole Gauges. **SPE Reservoir Evaluation & Engineering**, v. 22, n. 02, p. 386–401, maio 2019. ISSN 1094-6470, 1930-0212. Disponível em: https://onepetro.org/REE/article/22/02/386/207344/Applying-Machine-Learning-Techniques-To-Interpret.

WANG, F. et al. Field Application of Deep Learning for Flow Rate Prediction with Downhole Temperature and Pressure. In: **International Petroleum Technology Conference**. Virtual: IPTC, 2021. Disponível em: https://onepetro.org/IPTCONF/proceedings/21IPTC/21IPTC/D012S045R068/460627.

WASKOM, M. L. seaborn: statistical data visualization. **Journal of Open Source Software**, v. 6, n. 60, p. 3021, abr. 2021. ISSN 2475-9066. Disponível em: https://joss.theoj.org/papers/10.21105/joss.03021.

WU, X.; HUMPHREY, K.; LIAO, T. T. Enhancing Production Allocation in Intelligent Wells via Application of Models and Real-Time Surveillance Data. In: . OnePetro, 2012. Disponível em: https://dx.doi.org/10.2118/155031-MS.

A
Figures

In this appendix, the figures showing the evolution of each application are condensed.

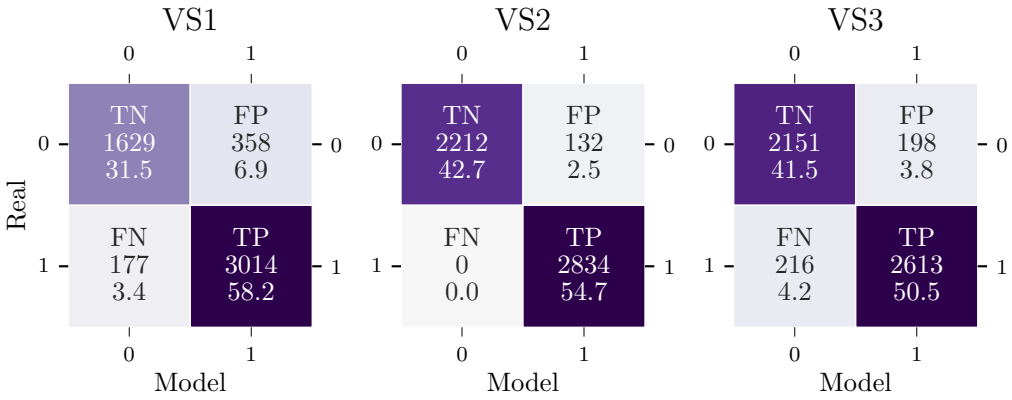

Figure A.1: Clustering Classification Confusion Matrix. Train: Well A, Test: Well A.

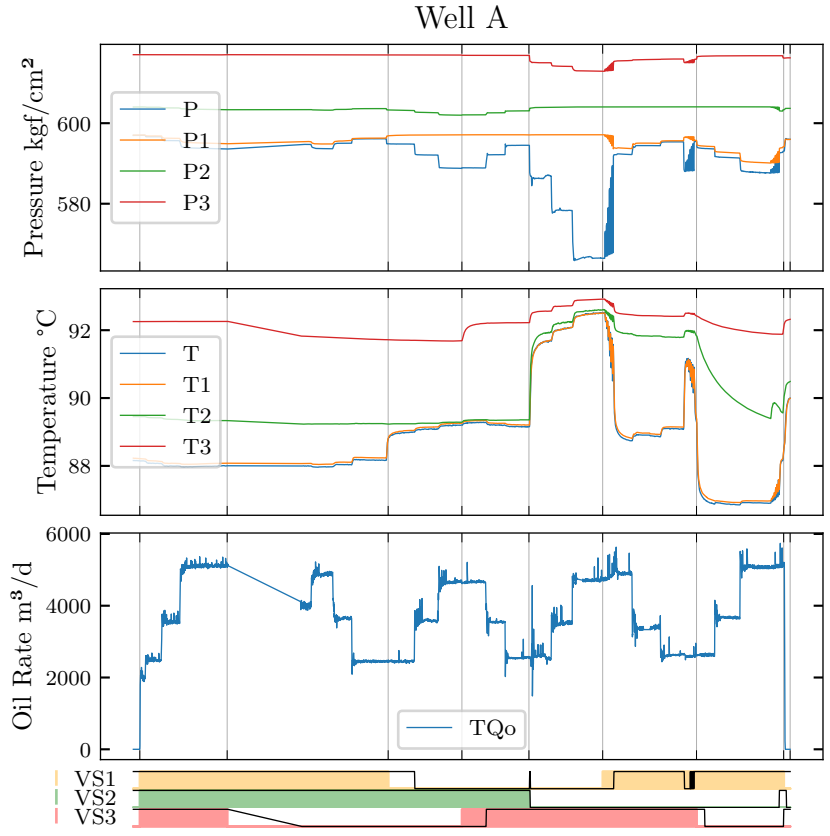

Figure A.2: Comparison Between Prediction and Actual Data for Clustering Classification. The bottom part of the figure shows the actual status of valve openings in each zone represented by colored areas, while the black lines represents the predicted values by the proposed model.

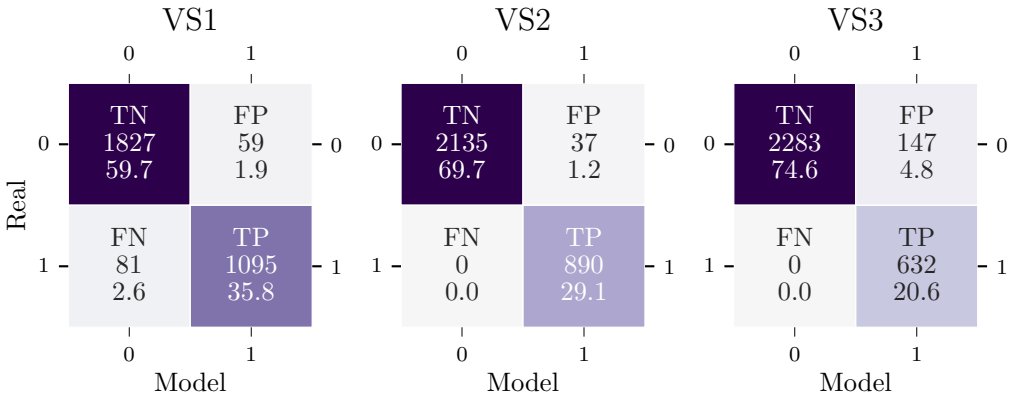

Figure A.3: Clustering Classification Confusion Matrix. Train: Well B, Test: Well B.

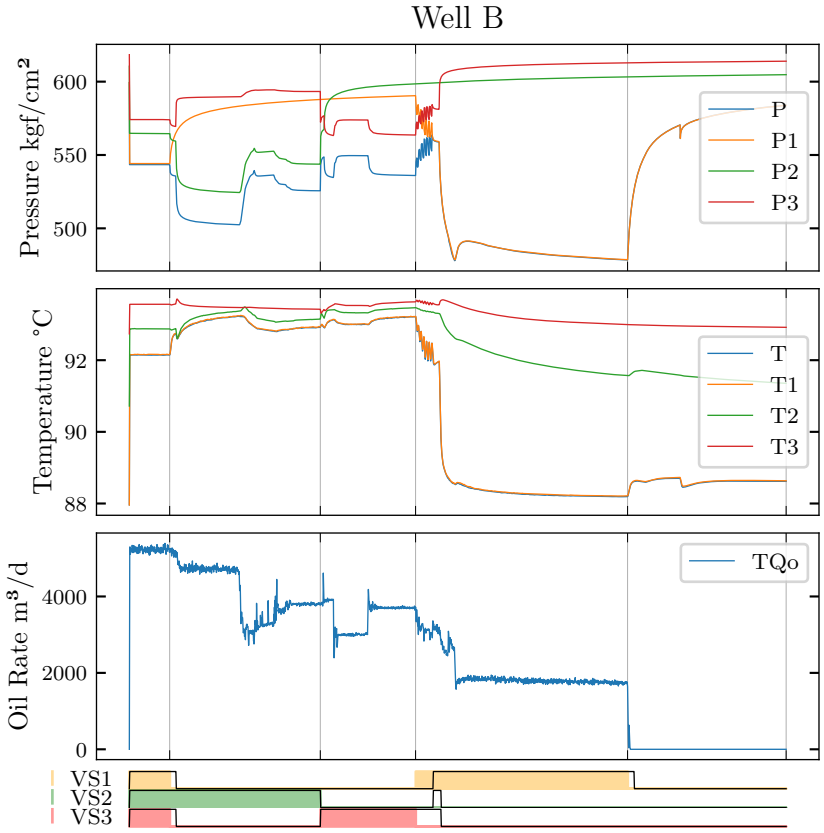

Figure A.4: Comparison Between Prediction and Actual Data for Clustering Classification. The bottom part of the figure shows the actual status of valve openings in each zone represented by colored areas, while the black lines represents the predicted values by the proposed model.

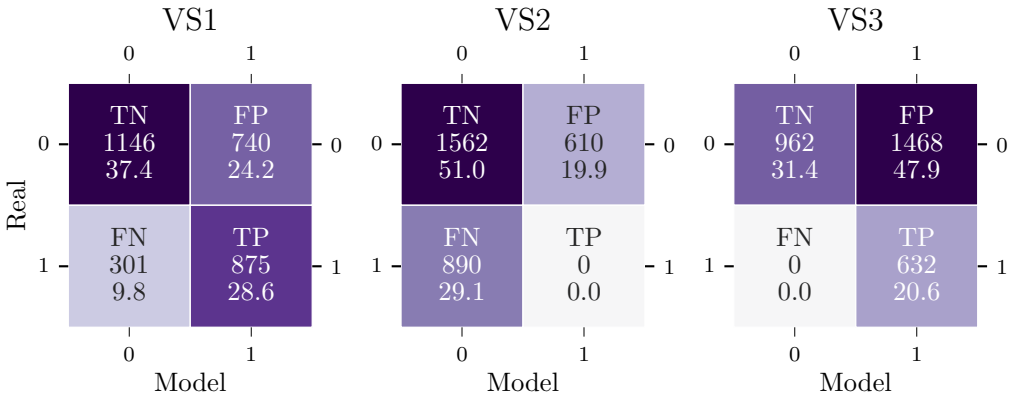

Figure A.5: Clustering Classification Confusion Matrix. Train: Well A, Test: Well B.

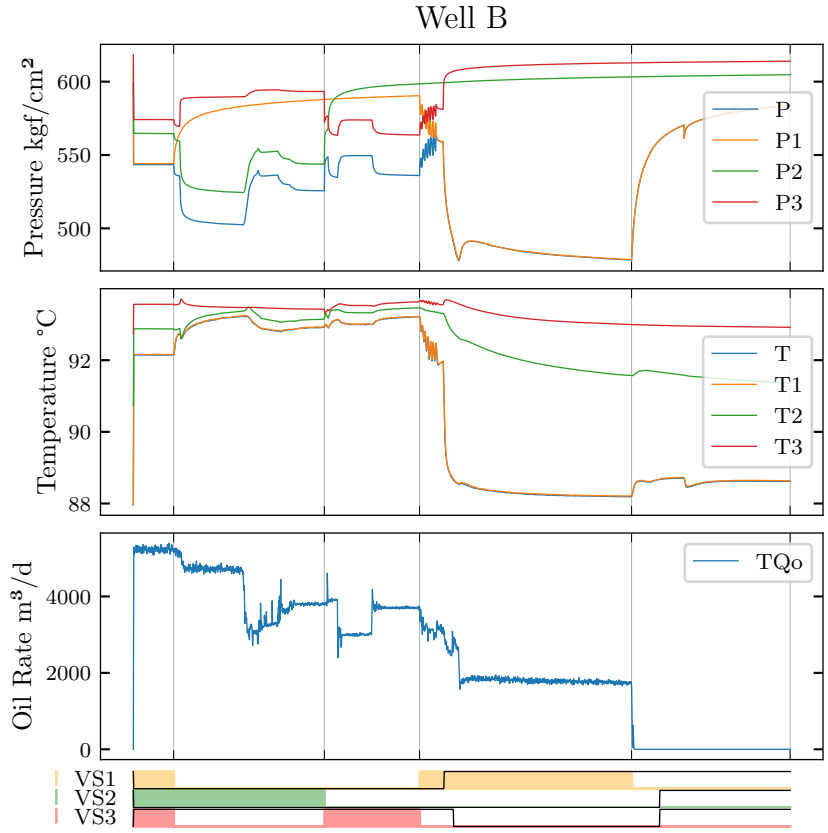

Figure A.6: Comparison Between Prediction and Actual Data for Clustering Classification. The bottom part of the figure shows the actual status of valve openings in each zone represented by colored areas, while the black lines represents the predicted values by the proposed model.

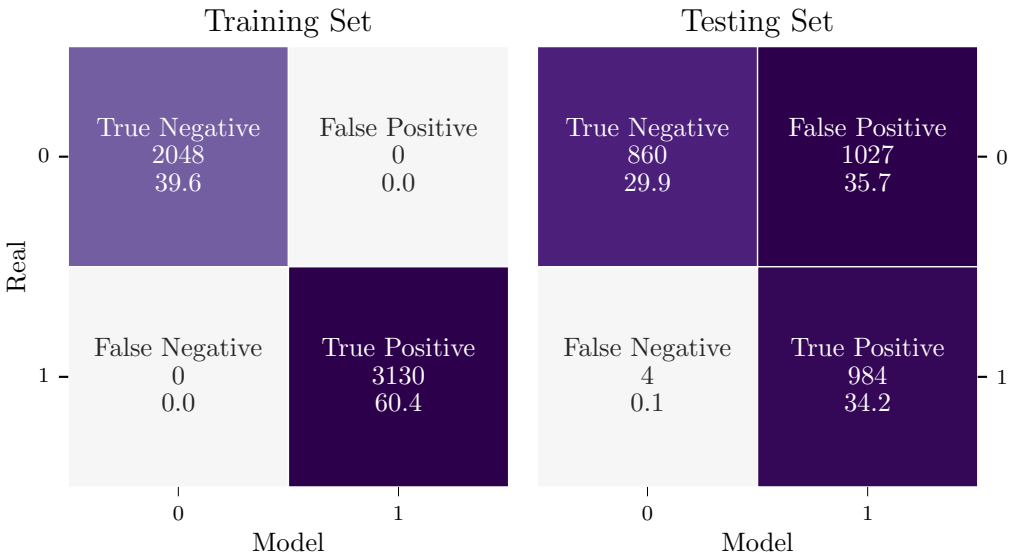

Figure A.7: Confusion matrix for Upper Valve. Features: P, $P1$.

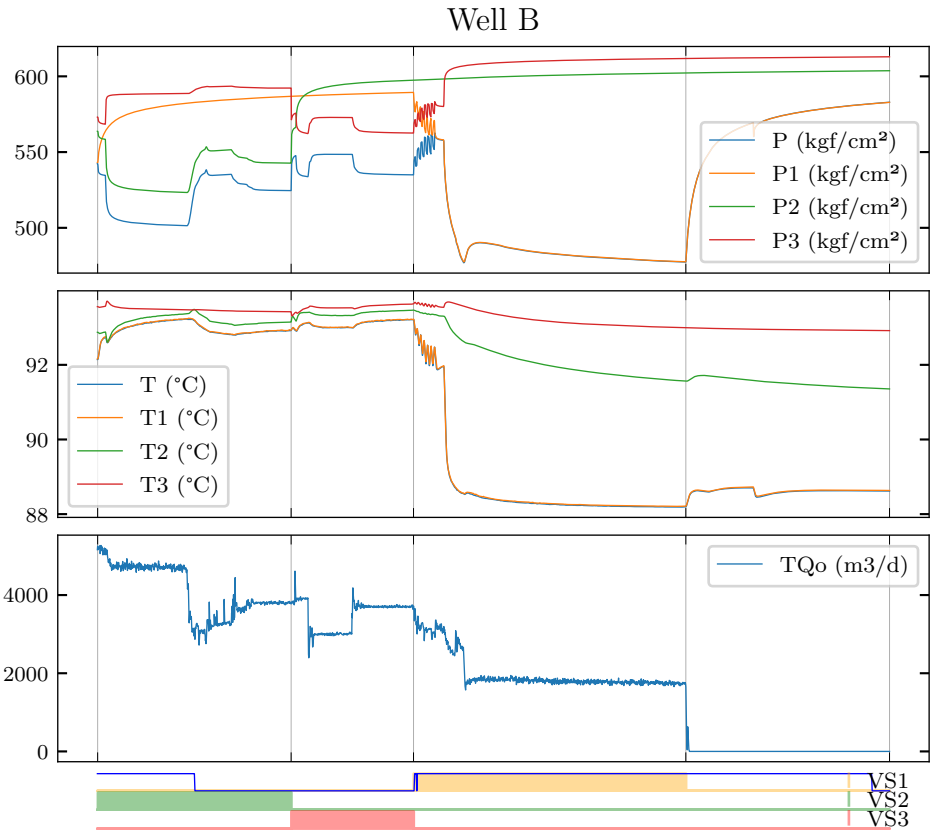

Figure A.8: Comparison Between Prediction and Actual Data for Upper Valve. Features: P, $P1$. The bottom part of the figure shows the actual status of valve openings in each zone represented by colored areas, while the blue line represents the predicted values by the proposed model.

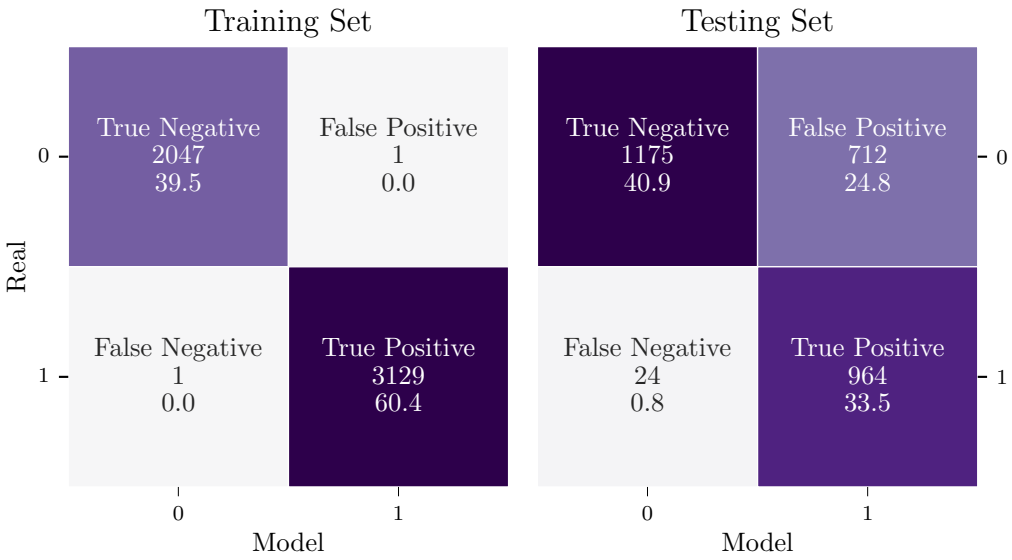

Figure A.9: Confusion matrix for Upper Valve. Features: P, $P1$, $P2$, $P3$.

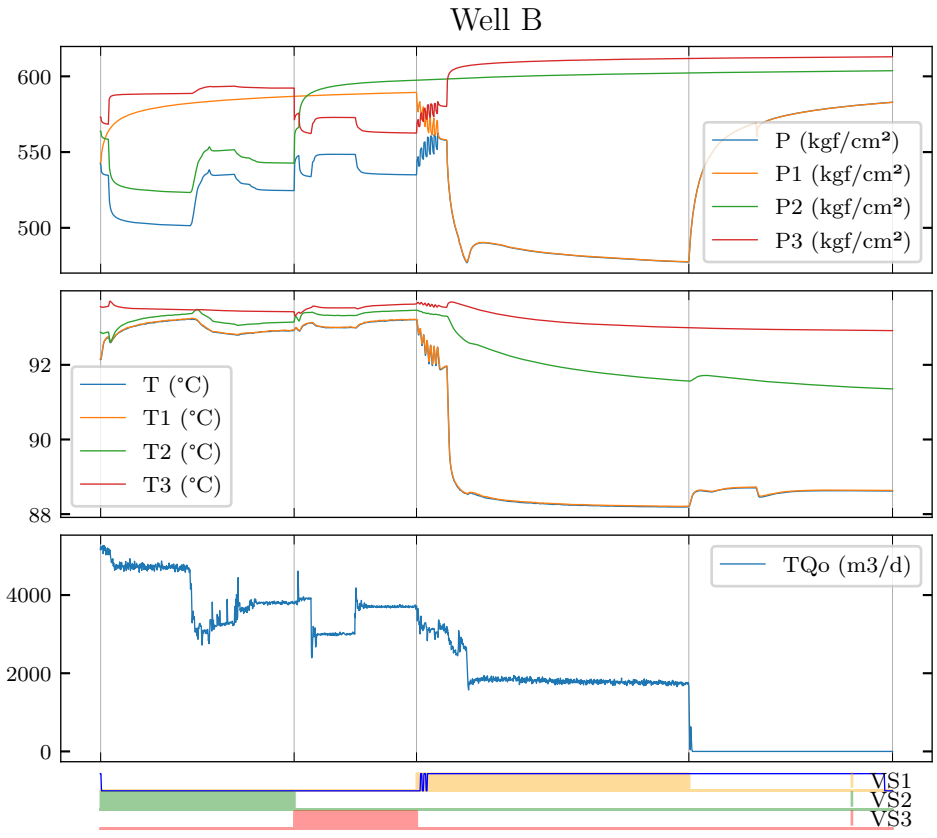

Figure A.10: Comparison Between Prediction and Actual Data for Upper Valve. Features: P, $P1$, $P2$, $P3$. The bottom part of the figure shows the actual status of valve openings in each zone represented by colored areas, while the blue line represents the predicted values by the proposed model.

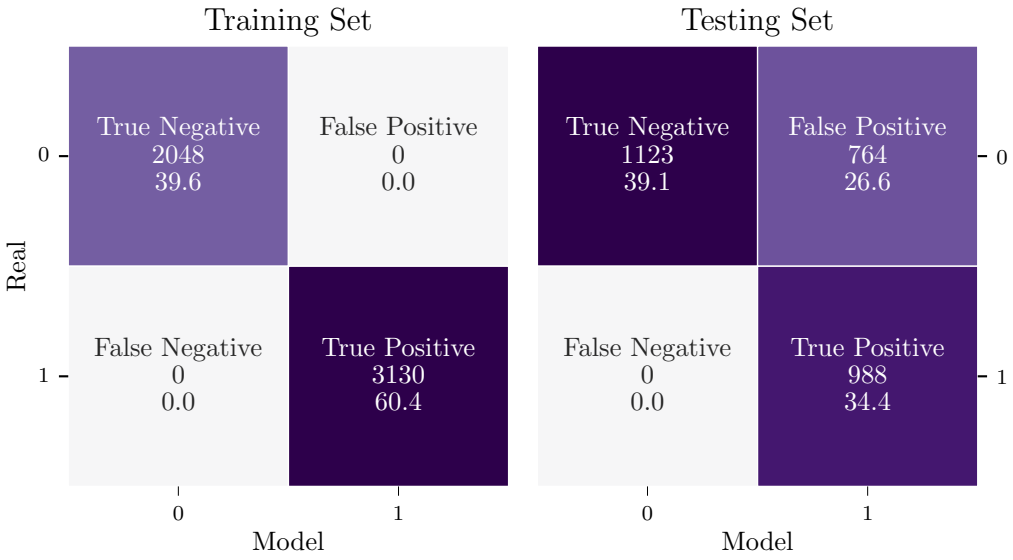

Figure A.11: Confusion matrix for Upper Valve. Features: P, $P1$, $P2$, $P3$, T, $T1$, $T2$, $T3$.

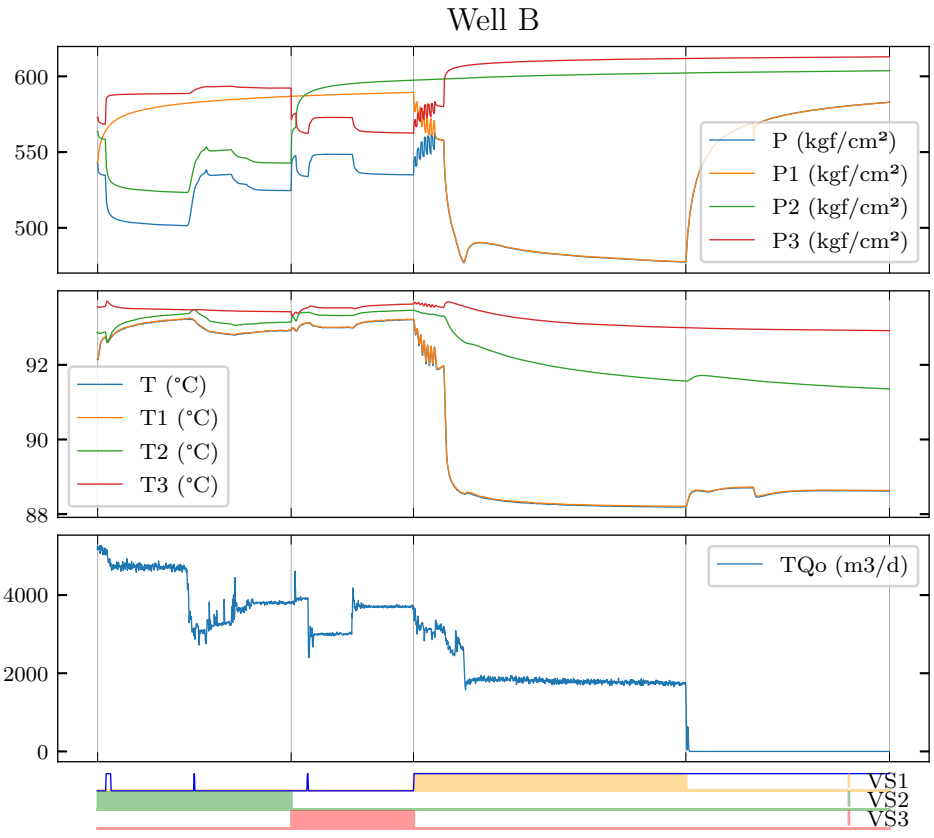

Figure A.12: Comparison Between Prediction and Actual Data for Upper Valve. Features: P, $P1$, $P2$, $P3$, T, $T1$, $T2$, $T3$. The bottom part of the figure shows the actual status of valve openings in each zone represented by colored areas, while the blue line represents the predicted values by the proposed model.

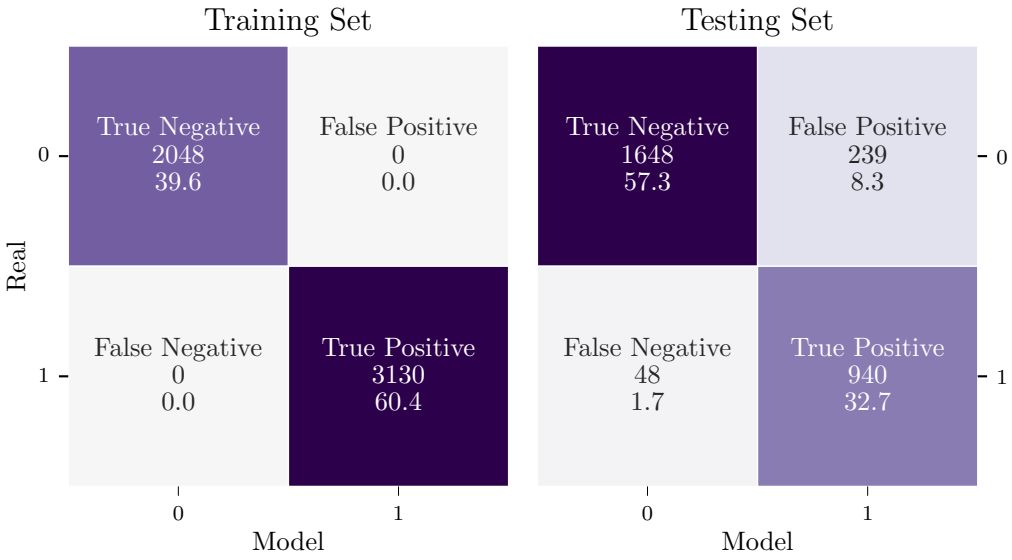

Figure A.13: Confusion matrix for Upper Valve. Features: P, $P1$, $dPP1$, $difP$, $difP1$.

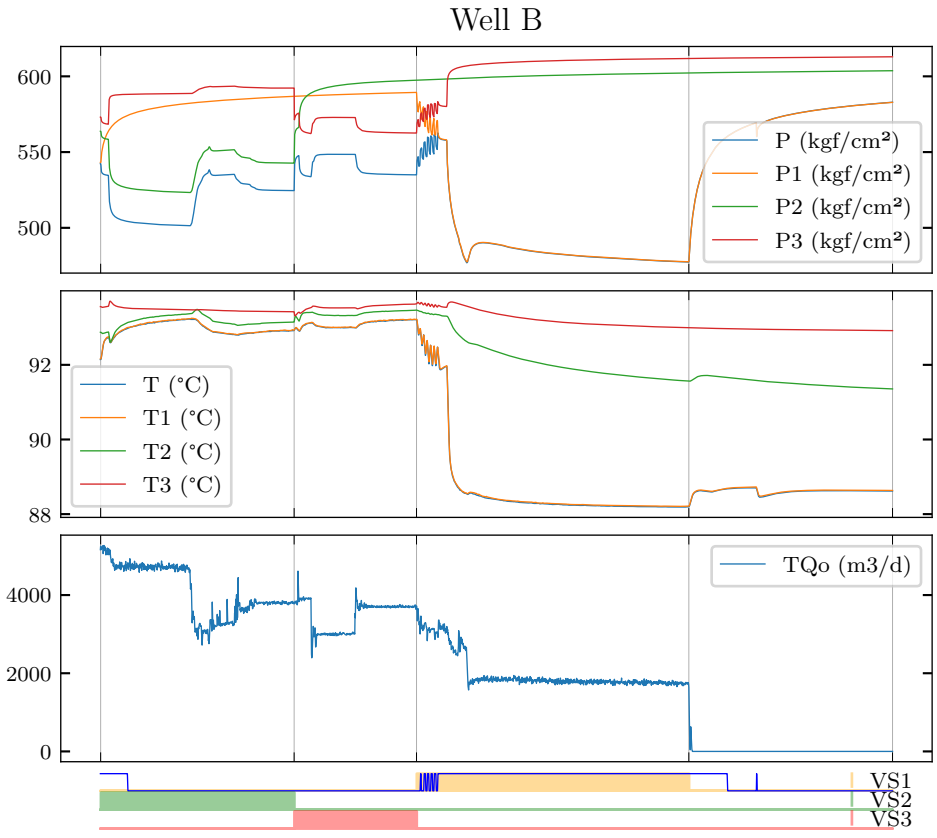

Figure A.14: Comparison Between Prediction and Actual Data for Upper Valve. Features: P, $P1$, $dPP1$, $difP$, $difP1$. The bottom part of the figure shows the actual status of valve openings in each zone represented by colored areas, while the blue line represents the predicted values by the proposed model.

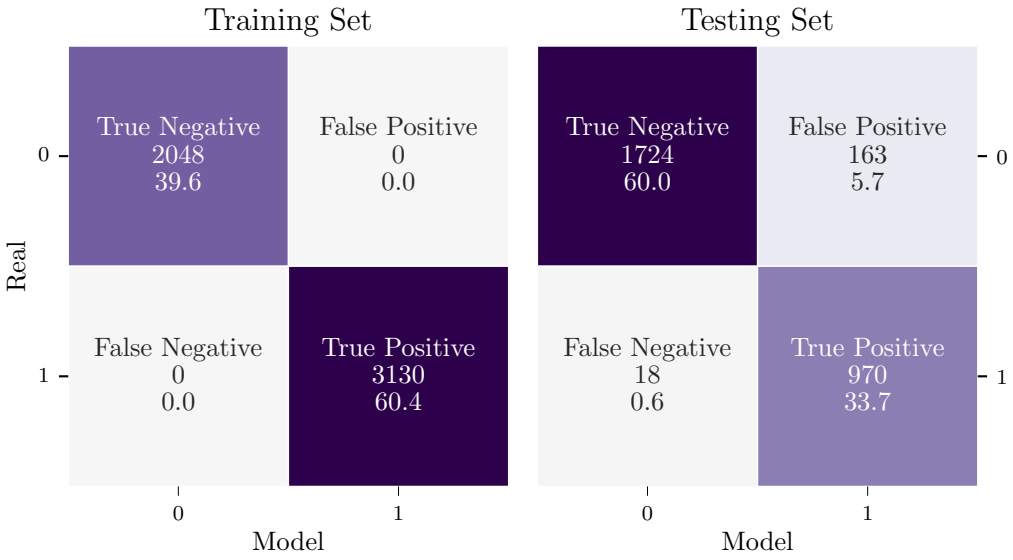

Figure A.15: Confusion matrix for Upper Valve. Features: P, $P1$, $dPP1$, $difP$, $difP1$, dP, $dP1$, P_{TC1}, P_{TC2}, $P1_{TC1}$, $P1_{TC2}$.

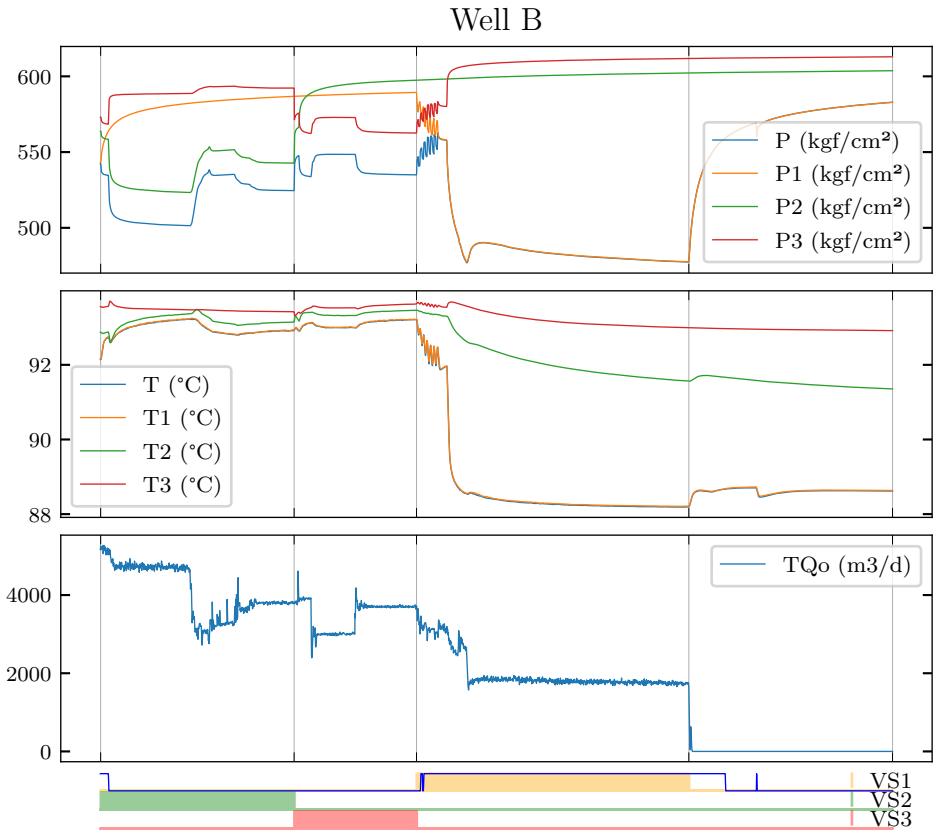

Figure A.16: Comparison Between Prediction and Actual Data for Upper Valve. Features: P, $P1$, $dPP1$, $difP$, $difP1$, dP, $dP1$, P_{TC1}, P_{TC2}, $P1_{TC1}$, $P1_{TC2}$. The bottom part of the figure shows the actual status of valve openings in each zone represented by colored areas, while the blue line represents the predicted values by the proposed model.

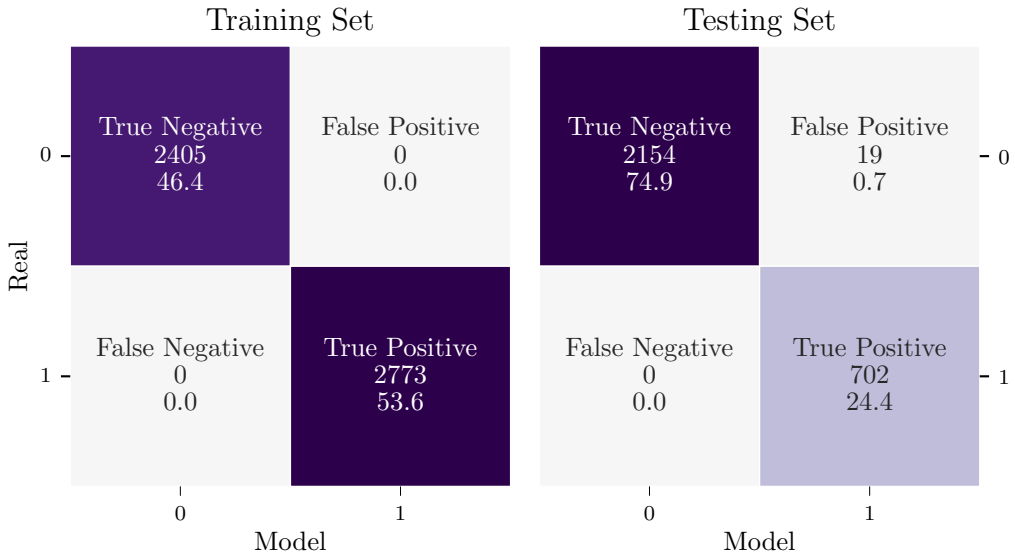

Figure A.17: Confusion matrix for Intermediate Valve. Features: P, $P2$, $dPP2$, $difP$, $difP2$, dP, $dP2$, P_{TC1}, P_{TC2}, $P2_{TC1}$, $P2_{TC2}$.

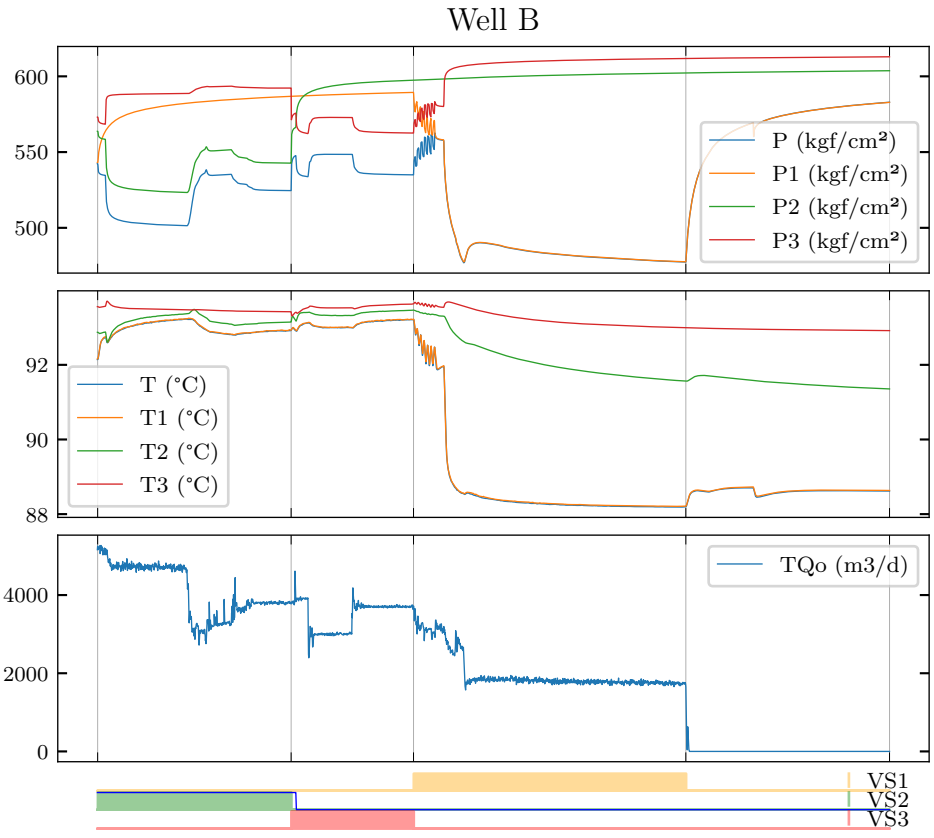

Figure A.18: Comparison Between Prediction and Actual Data for Intermediate Valve. Features: P, $P2$, $dPP2$, $difP$, $difP2$, dP, $dP2$, P_{TC1}, P_{TC2}, $P2_{TC1}$, $P2_{TC2}$. The bottom part of the figure shows the actual status of valve openings in each zone represented by colored areas, while the blue line represents the predicted values by the proposed model.

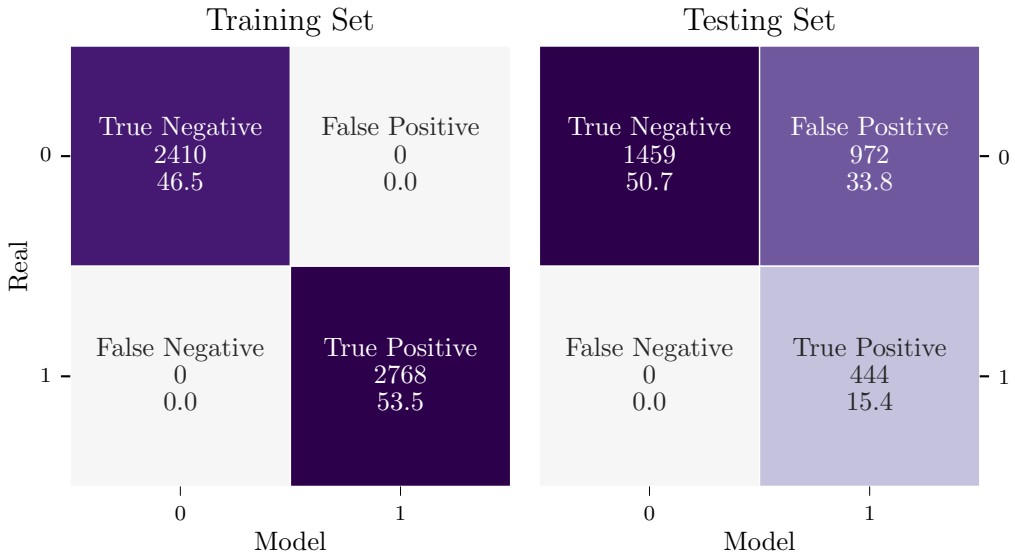

Figure A.19: Confusion matrix for Lower Valve. Features: P, $P3$, $dPP3$, $difP$, $difP3$, dP, $dP3$, P_{TC1}, P_{TC2}, $P3_{TC1}$, $P3_{TC2}$.

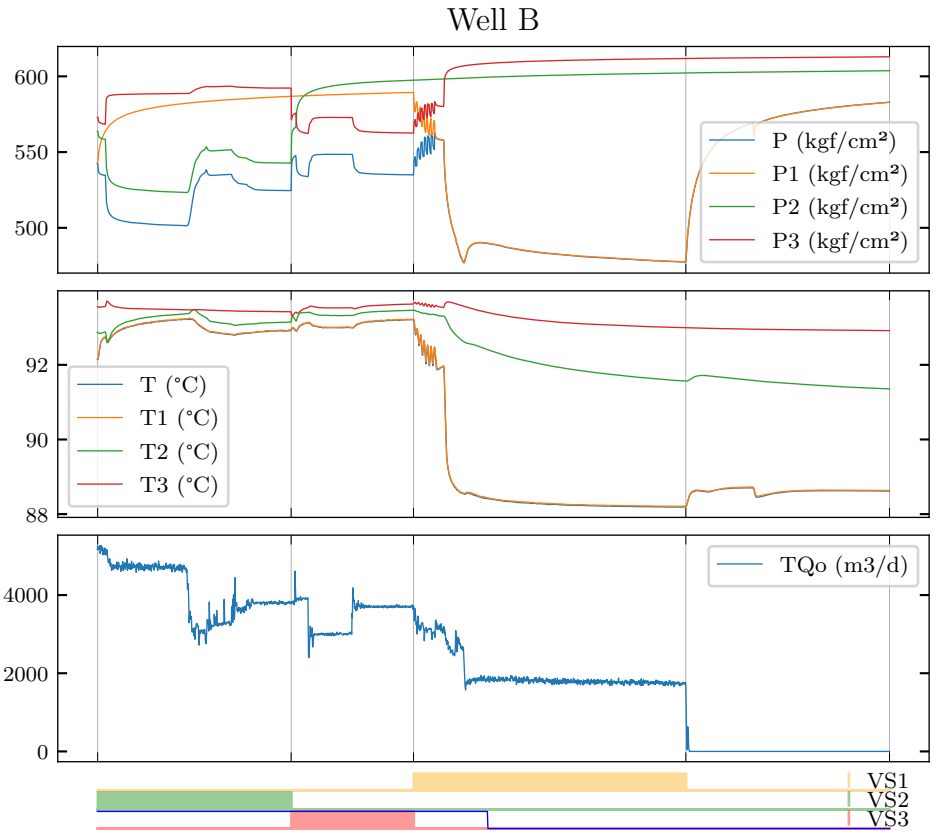

Figure A.20: Comparison Between Prediction and Actual Data for Lower Valve. Features: P, $P3$, $dPP3$, $difP$, $difP3$, dP, $dP3$, P_{TC1}, P_{TC2}, $P3_{TC1}$, $P3_{TC2}$. The bottom part of the figure shows the actual status of valve openings in each zone represented by colored areas, while the blue line represents the predicted values by the proposed model.

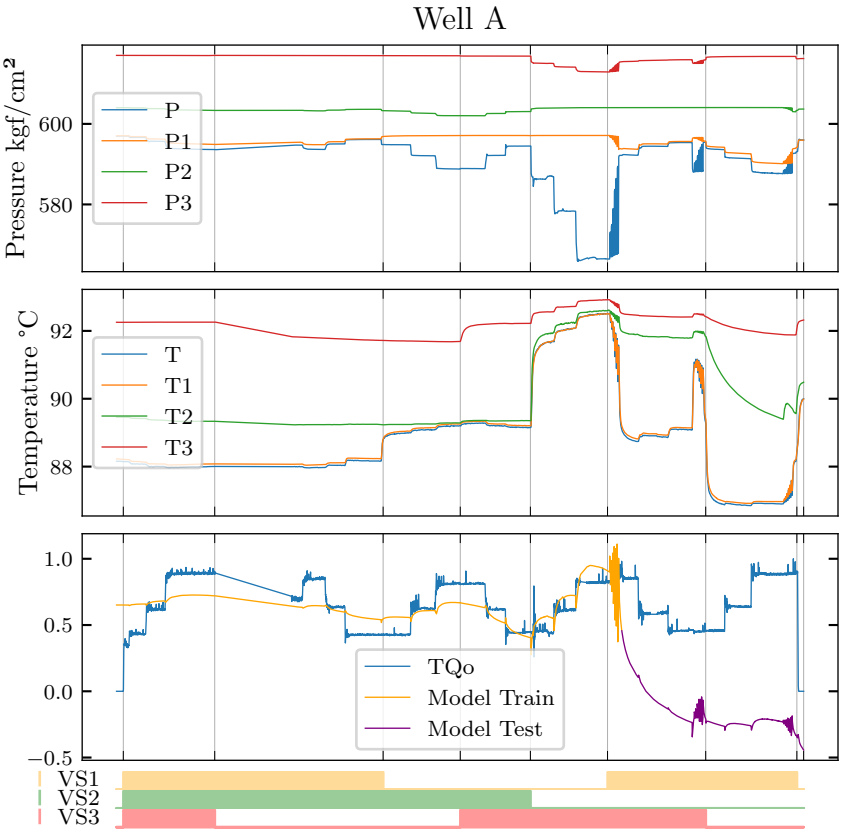

Figure A.21: Regression results Well A: Linear Regression, Features: dP, P_{TC1}, P_{TC2}

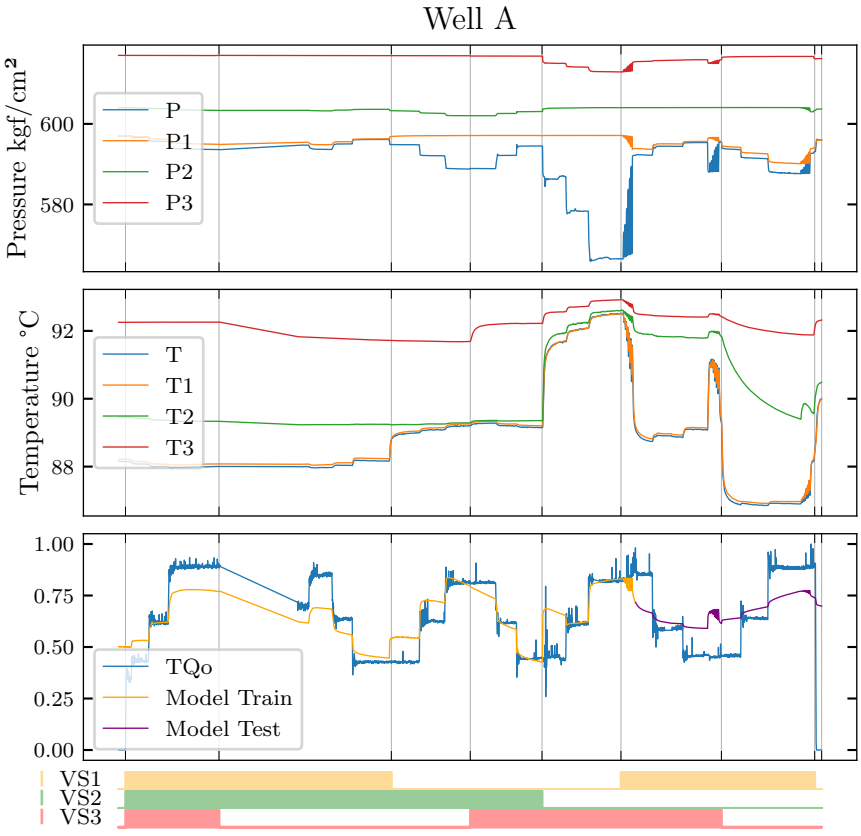

Figure A.22: Regression results Well A: Support Vector Regression, Features: dP, P_{TC1}, P_{TC2}

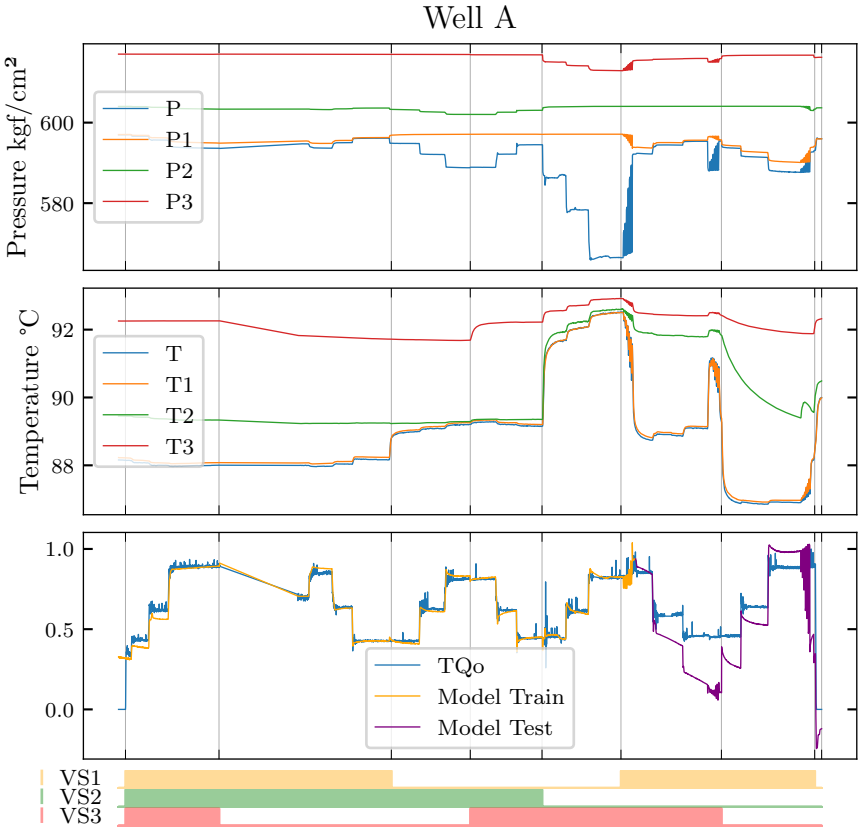

Figure A.23: Regression results Well A: Linear Regression, Features: dP, dPP_1, dPP_2, dPP_3, dP_1P_2, dP_1P_3, dP_2P_3, P_{TC1}, P_{TC2}, $P1_{TC1}$, $P1_{TC2}$, $P2_{TC1}$, $P2_{TC2}$, $P3_{TC1}$, $P3_{TC2}$

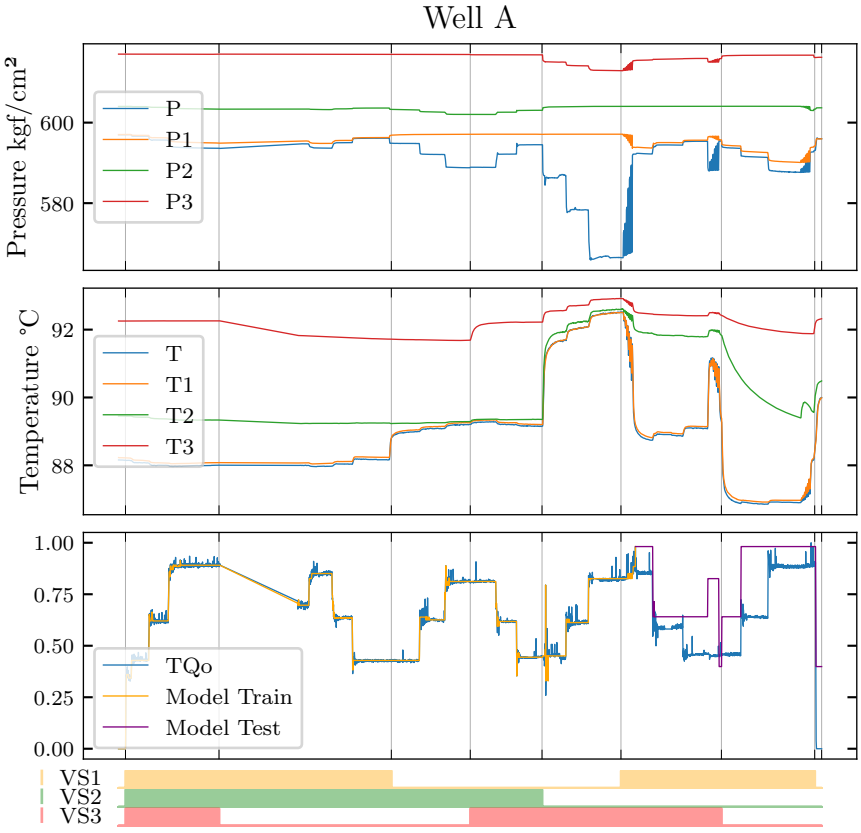

Figure A.24: Regression results Well A: Decision Tree Regression, Features: dP, dPP_1, dPP_2, dPP_3, dP_1P_2, dP_1P_3, dP_2P_3, P_{TC1}, P_{TC2}, $P1_{TC1}$, $P1_{TC2}$, $P2_{TC1}$, $P2_{TC2}$, $P3_{TC1}$, $P3_{TC2}$

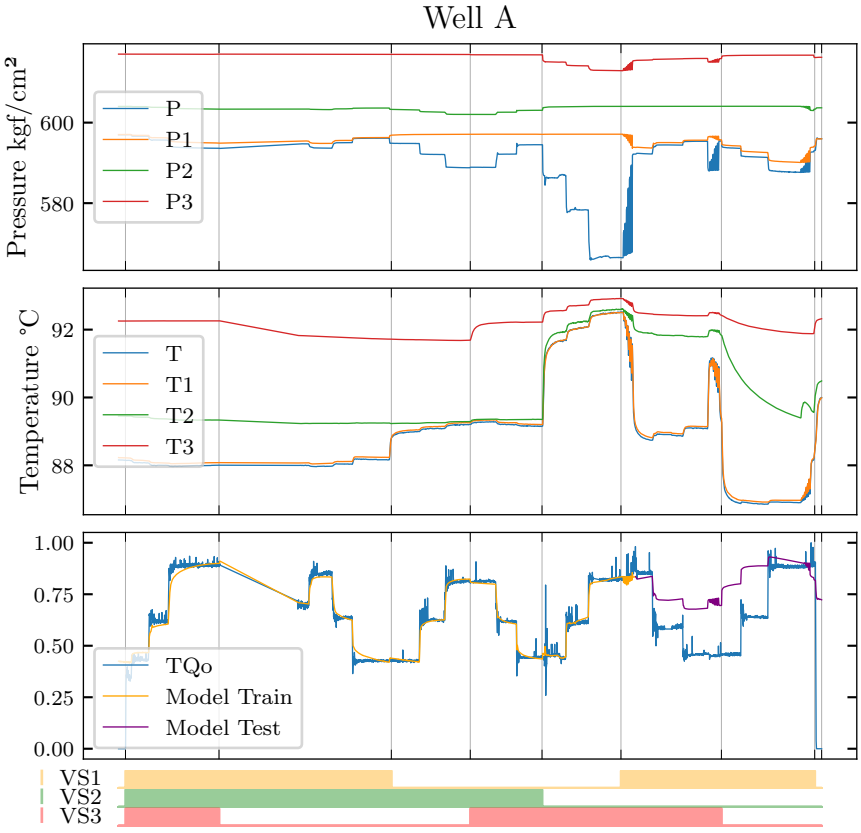

Figure A.25: Regression results Well A: Support Vector Regression, Features: dP, dPP_1, dPP_2, dPP_3, dP_1P_2, dP_1P_3, dP_2P_3, P_{TC1}, P_{TC2}, $P1_{TC1}$, $P1_{TC2}$, $P2_{TC1}$, $P2_{TC2}$, $P3_{TC1}$, $P3_{TC2}$

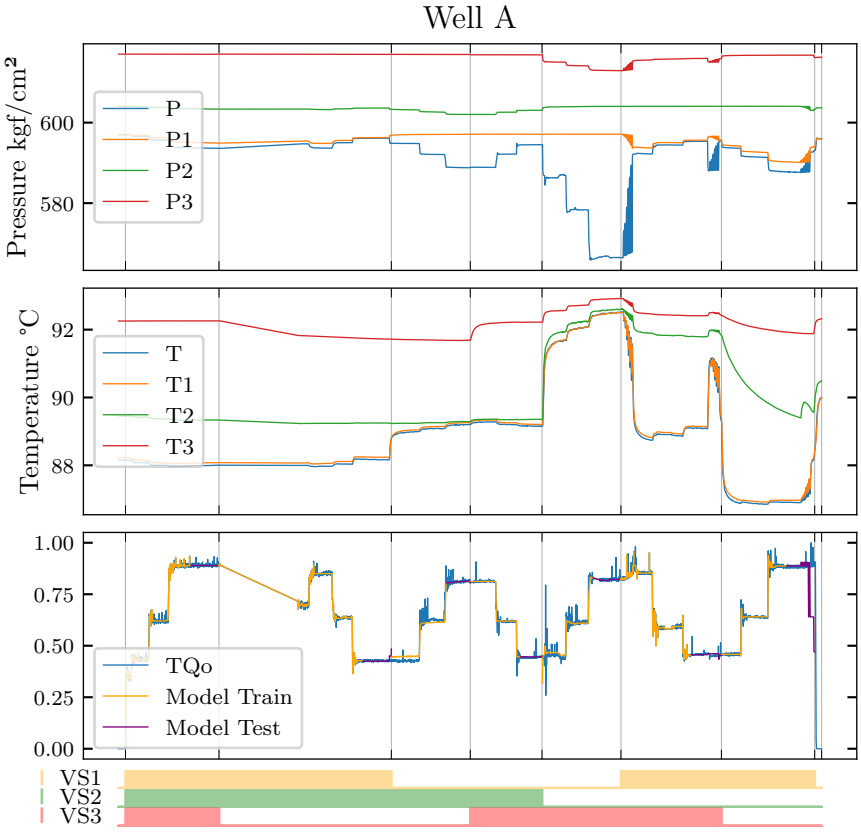

Figure A.26: Regression results Well A: Best algorithm each combination of valves, Features: dP, dPP_1, dPP_2, dPP_3, dP_1P_2, dP_1P_3, dP_2P_3, P_{TC1}, P_{TC2}, $P1_{TC1}$, $P1_{TC2}$, $P2_{TC1}$, $P2_{TC2}$, $P3_{TC1}$, $P3_{TC2}$

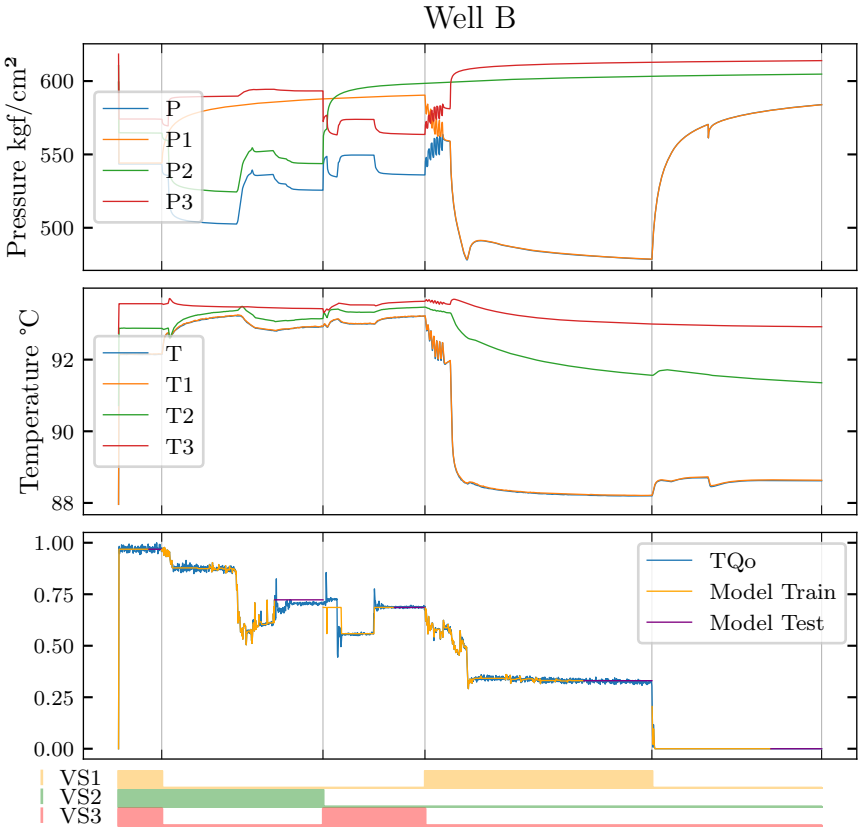

Figure A.27: Regression results Well B: Best algorithm each combination of valves, Features: dP, dPP_1, dPP_2, dPP_3, dP_1P_2, dP_1P_3, dP_2P_3, P_{TC1}, P_{TC2}, $P1_{TC1}$, $P1_{TC2}$, $P2_{TC1}$, $P2_{TC2}$, $P3_{TC1}$, $P3_{TC2}$

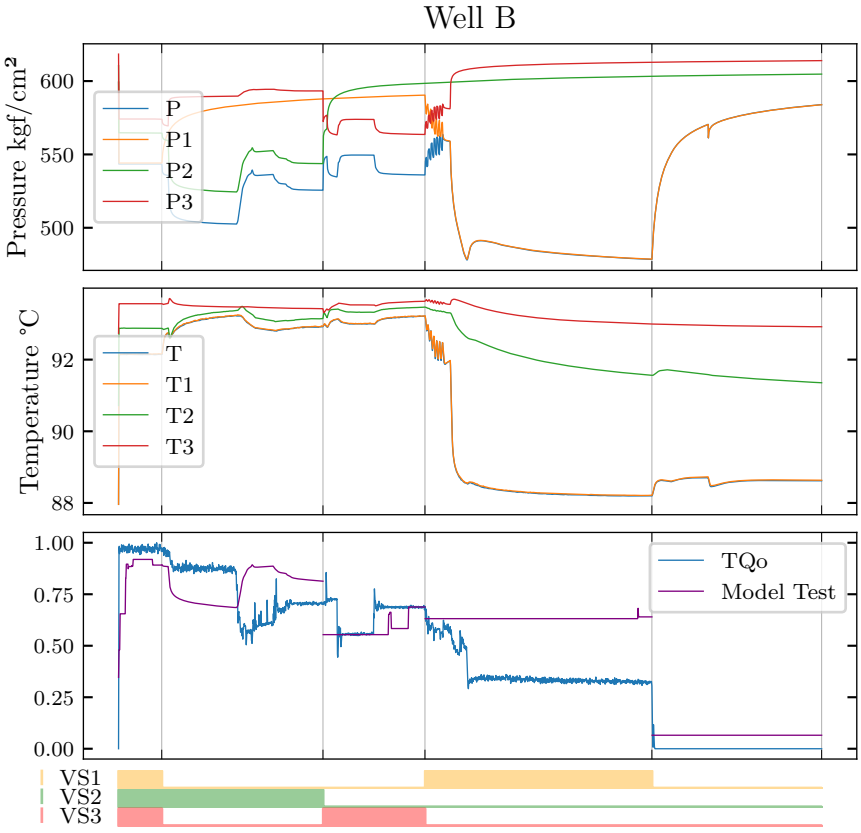

Figure A.28: Regression results Train Well A Test Well B: Best algorithm each combination of valves, Features: dP, dPP_1, dPP_2, dPP_3, dP_1P_2, dP_1P_3, dP_2P_3, P_{TC1}, P_{TC2}, $P1_{TC1}$, $P1_{TC2}$, $P2_{TC1}$, $P2_{TC2}$, $P3_{TC1}$, $P3_{TC2}$