

# Probabilistic Oil and Gas Production Forecasting using Machine Learning

by

Robert Andrais

BSc. Mechanical Engineering, University of Alberta (2013)

Submitted to the Department of Engineering and Management  
in partial fulfillment of the requirements for the degree of

Master of Science in Engineering and Management

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2021

© 2021 Robert Andrais. All rights reserved.

The author hereby grants to MIT permission to reproduce and to  
distribute publicly paper and electronic copies of this thesis document  
in whole or in part in any medium now known or hereafter created.

Author .....  
Department of Engineering and Management  
August 6, 2021

Certified by .....  
Aimé Fournier  
Research Scientist  
Thesis Supervisor

Certified by .....  
Bryan Moser  
Academic Director and Sr. Lecturer, System Design and Management  
Thesis Supervisor

Accepted by .....  
Joan Rubin  
Executive Director, System Design and Management Program



# **Probabilistic Oil and Gas Production Forecasting using Machine Learning**

by

Robert Andrais

Submitted to the Department of Engineering and Management  
on August 6, 2021, in partial fulfillment of the  
requirements for the degree of  
Master of Science in Engineering and Management

## **Abstract**

This thesis improves oil- and gas-well profitability by quantifying the uncertainty of the production-forecasting process, using probabilistic machine learning (ML) techniques. A Bayesian Neural Network successfully modelled a complex shale gas reservoir system (Eagle Ford), generating a production forecast with 5% mean absolute percent error. This result is 10%–35% more accurate than traditional decline curve analysis. These forecasts also quantified the epistemic and aleatory uncertainties, providing plausible probabilistic P10 and P90 values. This range provides analysts with the capability of making informed strategic decisions that consider risk. Next, the model was applied to predict reserves (estimated ultimate recovery) and the underlying reservoir quality. These predictions were combined with unsupervised learning techniques (Gaussian Mixture Modelling), creating gas and oil sweet-spot maps. Finally, this workflow's robustness was demonstrated by artificially reducing data by 93%; indeed, the algorithm could reproduce the full-dataset results with a 71%–91% Pearson correlation, despite this reduction. Supporting this workflow creation is an evaluation of relevant research, data processing, feature engineering, documentation of the probabilistic ML structure, and discussion of model performance using systems analysis.

Thesis Supervisor: Aimé Fournier  
Title: Research Scientist

Thesis Supervisor: Bryan Moser  
Title: Academic Director and Sr. Lecturer, System Design and Management



## Acknowledgments

I am grateful for my research supervisor, Aimé Fournier, who has provided critical feedback and guidance. This project has pushed my capabilities to the limit, investigating a cutting-edge problem without a defined roadmap for success. Without his support and probing questions, many key insights would be lost. I would also like to thank the 2020-2021 MIT and Chevron communities. I will always cherish the close friendships formed and hope that they continue to grow in the future. Lastly, I would like to thank my family, friends, and especially my fiancée, Kathryn. You have had to sacrifice so much to help me achieve this dream. 2020 has been a struggle with the pandemic and completing a 2-year program in 12 months. However, I can't imagine a better outcome for an otherwise terrible year. We've had to share this burden together, and I know it will make us stronger and better people for it. I will forever appreciate your support and being a constant source of light.

THIS PAGE INTENTIONALLY LEFT BLANK

# Contents

<b>1</b>	<b>Introduction</b>	<b>17</b>
1.1	Production Forecasting Overview . . . . .	18
1.1.1	Production Forecasting Issues . . . . .	19
1.1.2	Research Questions and Structure . . . . .	21
1.2	Literature Review . . . . .	23
1.2.1	Decline Curve Analysis (DCA) Overview . . . . .	23
1.2.2	Modified Hyperbolic Declines . . . . .	25
1.2.3	DCA Methodology Enhancements . . . . .	26
1.2.4	Type Well Generation Overview . . . . .	31
1.2.5	Summary of Lessons Learned . . . . .	38
1.3	Machine Learning Concept Generation . . . . .	41
1.3.1	Tree-Based Methods . . . . .	41
1.3.2	Neural Network . . . . .	43
1.4	Probabilistic Machine Learning Approaches . . . . .	46
1.4.1	Ensemble Based Method . . . . .	46
1.4.2	Bayesian Neural Networks . . . . .	48
1.4.3	Summary of Advantages and Disadvantages . . . . .	52
<b>2</b>	<b>Data Processing</b>	<b>55</b>
2.1	Data Processing . . . . .	56
2.1.1	Historical Production . . . . .	56
2.1.2	Well Metadata . . . . .	58
2.1.3	Data Cleaning and Validation . . . . .	62

2.2	Feature Engineering . . . . .	65
2.2.1	Unsupervised Learning of Static Parameters . . . . .	65
2.2.2	Dynamic Lag Features and Target Variable . . . . .	68
2.2.3	Summary of Input and Output Variables . . . . .	71
<b>3</b>	<b>Analysis Methodology</b>	<b>73</b>
3.1	Evaluation Criteria and Fitting Process . . . . .	74
3.2	Automated Decline Curve Analysis (DCA) . . . . .	76
3.3	Ensemble Models . . . . .	78
3.4	Bayesian Neural Network . . . . .	82
3.5	Application on Type Well Forecasts . . . . .	89
<b>4</b>	<b>Analysis Results</b>	<b>97</b>
4.1	Existing Well Forecasting Results . . . . .	97
4.1.1	Aggregate Comparison . . . . .	97
4.1.2	Indicative Individual Well Plots . . . . .	103
4.1.3	Generating EUR Output Variable . . . . .	107
4.1.4	Existing Well Forecasting Summary and Conclusions . . . . .	111
4.2	Type Well Results . . . . .	112
4.2.1	Restricted Dataset Forecasting . . . . .	115
4.2.2	Type Well Forecasting Summary and Conclusions . . . . .	124
<b>5</b>	<b>Discussion and Conclusions</b>	<b>125</b>
5.1	Existing Well Forecasting Discussion . . . . .	125
5.1.1	Ensemble Method Discussion . . . . .	126
5.1.2	Bayesian Neural Network (BNN) Method Discussion . . . . .	129
5.2	Type Well Forecasting Discussion . . . . .	136
5.2.1	Restricted Dataset Discussion . . . . .	141
5.3	Conclusions . . . . .	144
<b>6</b>	<b>Future Analysis</b>	<b>149</b>
6.1	Summary of Improvements to Current Workflow . . . . .	149

6.2 Future Applications . . . . .	150
<b>A Tables</b>	<b>155</b>
A.1 Cross-Validation . . . . .	155
A.2 Oil Existing Well Forecast Results . . . . .	157
<b>B Figures</b>	<b>159</b>
B.1 Well Plots - Existing Well Forecasting . . . . .	162
B.2 Well Plots - EUR Forecast . . . . .	172
B.3 Oil Existing Well Forecasts . . . . .	178
B.4 Sweet Spot Mapping Full Dataset . . . . .	183
B.5 Restricted Dataset Forecasts . . . . .	187

THIS PAGE INTENTIONALLY LEFT BLANK

# List of Figures

1-1	Well count and engineering design changes over time . . . . .	20
1-2	Thesis layout of workflow methodology . . . . .	22
1-3	Traditional DCA Example . . . . .	24
1-4	Example of Unconventional Reservoir following Lognormal Distribution	34
1-5	Example of Data Handling Strategies . . . . .	36
1-6	Tree Based ML Algorithm Fundamentals . . . . .	42
1-7	Neural Network Algorithm Fundamentals . . . . .	44
1-8	Plot of Activation Functions . . . . .	45
1-9	Aleatory and Epistemic Error Summary . . . . .	47
1-10	Example of Heteroscedastic Dataset . . . . .	49
2-1	Eagle Ford Well Plot . . . . .	56
2-2	Nondimensionalizing Cumulative Volume using a 12 month datum and converting to log scale . . . . .	59
2-3	Initial High Collinearity within Dataset . . . . .	61
2-4	Outlier Cleanup Process . . . . .	63
2-5	Final Collinearity after feature refinement . . . . .	64
2-6	Scree plot for $k$ -means analysis of optimal cluster number . . . . .	66
2-7	CGR Trends across the Eagle Ford . . . . .	67
2-8	BIC Plot for GMM Analysis of optimal cluster number . . . . .	68
2-9	GMM vs $k$ -means cluster comparison . . . . .	69
2-10	Backcasting methodology to validate model performance . . . . .	70
2-11	Test Train Split Methodology . . . . .	72

3-1	Full Analysis Workflow Overview . . . . .	73
3-2	Visualization of the cross-validation process . . . . .	75
3-3	Automated DCA with Shift Factor . . . . .	76
3-4	Workflow diagram of Automated DCA . . . . .	77
3-5	Ensemble Method Overview . . . . .	78
3-6	Random perturbation overview for the Ensemble . . . . .	80
3-7	Workflow diagram of Ensemble Models . . . . .	81
3-8	BNN Methodology Overview . . . . .	84
3-9	Model Structure for the BNN . . . . .	85
3-10	Example of random sampling for each weight and output node . . . . .	86
3-11	Depiction of the Monte Carlo Forecasting Process . . . . .	87
3-12	Workflow diagram of the BNN methodology . . . . .	88
3-13	Validation Plot of Predicted Geological Features . . . . .	90
3-14	Plot of the Eagle Ford divided into grid cells . . . . .	91
3-15	Monte Carlo sampling method for Ensemble training . . . . .	92
3-16	Type Well EUR Workflow for Ensemble Method . . . . .	94
3-17	Type Well Workflow for BNN Method . . . . .	95
4-1	Validation Plot of Existing Well Performance for Gas . . . . .	99
4-2	Plot of MAPE Performance on gas . . . . .	100
4-3	Percentages of Wells within Probabilistic Range . . . . .	101
4-4	Residual Analysis for Gas Forecast . . . . .	103
4-5	Existing Well Forecast Example 1 . . . . .	104
4-6	Existing Well Forecast Example 2 . . . . .	105
4-7	Existing Well Forecast Example 3 . . . . .	106
4-8	Type Well Forecast Example 1 . . . . .	108
4-9	Type Well Forecast Example 1 for BNN Method . . . . .	109
4-10	Correlation between EUR and Normalized Month on Production . . . . .	110
4-11	Validation Plot of BNN Gas Performance on Type Well Prediction . . . . .	112
4-12	Validation Plot of BNN Oil Performance on Type Well Prediction . . . . .	113

4-13	Gas Hotspot Map with Full Dataset . . . . .	114
4-14	Oil Hotspot Map with Full Dataset . . . . .	115
4-15	GMM Cluster Results with Full Dataset . . . . .	116
4-16	Well Count Differences between Full and Restricted Datasets . . . . .	117
4-17	Full and Restricted Dataset Correlation . . . . .	119
4-18	Comparison of Gas Hotspot map between Full and Restricted Datasets	121
4-19	Comparison of Oil Hotspot map between Full and Restricted Datasets	122
4-20	Comparison of GMM map between Full and Restricted Datasets . . .	123
5-1	Plot of accuracy, precision, and bias tendencies . . . . .	127
5-2	Calculations Performed for Ensemble Models as a function of $\tau$ . . . . .	128
5-3	Residual comparison of Month 40 to 50 . . . . .	131
5-4	Relative Molecule and Pore Throat Sizes . . . . .	133
5-5	Phase envelope for different fluid types . . . . .	134
5-6	Example of Kalman Filter . . . . .	135
5-7	Expected OGR behaviour versus Reservoir Pressure . . . . .	135
5-8	Oil Shrinkage to Standard Conditions . . . . .	137
5-9	GMM Map for Full Dataset and Tier 1 Areas Overlaid . . . . .	139
5-10	MAPE Results Plotted Spatially . . . . .	143
5-11	GMM Map for Restricted Dataset and Tier 1 Areas Overlaid . . . . .	144
6-1	Illustration of a Potential Full Type Well Forecast . . . . .	151
6-2	Example of a Tradespace Analysis . . . . .	152
6-3	Example of Local Type Well Forecasting . . . . .	153
B-1	Plot of the Eagle Ford divided into grid cells without wells . . . . .	159
B-2	Main code structure of the BNN . . . . .	160
B-3	Validation of lognormal ensemble prediction distributions . . . . .	161
B-4	Existing Well Forecast Example 4 . . . . .	162
B-5	Existing Well Forecast Example 5 . . . . .	163
B-6	Existing Well Forecast Example 6 . . . . .	164

B-7 Existing Well Forecast Example 7 . . . . .	165
B-8 Existing Well Forecast Example 8 . . . . .	166
B-9 Existing Well Forecast Example 9 . . . . .	167
B-10 Existing Well Forecast Example 10 . . . . .	168
B-11 Existing Well Forecast Example 11 . . . . .	169
B-12 Existing Well Forecast Example 12 . . . . .	170
B-13 Barplot analysis of wells exhibiting divergence . . . . .	171
B-14 Type Well Forecast Example 2 . . . . .	172
B-15 Type Well Forecast Example 2 for BNN Method . . . . .	173
B-16 Type Well Forecast Example 3 . . . . .	174
B-17 Type Well Forecast Example 3 for BNN Method . . . . .	175
B-18 Type Well Forecast Example 4 . . . . .	176
B-19 Type Well Forecast Example 4 for BNN Method . . . . .	177
B-20 Validation Plot of Existing Well Performance for Gas and Oil . . . . .	178
B-21 Plot of MAPE Performance on gas and oil . . . . .	179
B-22 Percentages of Wells within Probabilistic Range for Gas and Oil . . . . .	180
B-23 Percentages of Wells within the P10–P90 vs P50–P90 . . . . .	181
B-24 Residual Analysis for Oil Forecast . . . . .	182
B-25 Gas Hotspot Map with Full Dataset - P10 Forecast . . . . .	183
B-26 Gas Hotspot Map with Full Dataset - P90 Forecast . . . . .	184
B-27 Oil Hotspot Map with Full Dataset - P10 Forecast . . . . .	185
B-28 Oil Hotspot Map with Full Dataset - P90 Forecast . . . . .	186
B-29 Validation Plot of BNN Performance on Existing Well Prediction - Restricted Dataset . . . . .	187
B-30 Validation Plot of BNN Performance on Type Well Prediction - Restricted Dataset . . . . .	188

# List of Tables

2.1	Normalizing Cumulative Volumes to Producing Month . . . . .	57
2.2	Summary of Process Treatment for each Input Feature Type . . . . .	60
2.3	Example of one-hot encoding to make the values machine-interpretable.	62
2.4	CGR as a proxy for fluid type . . . . .	67
2.5	Example of Generating Lag and Target Production Features based on months before forecast start ( $t_{\text{forecast}}$ ) . . . . .	70
2.6	Summary of the treatment methods for each variable class after processing and feature engineering. . . . .	72
3.1	Summary of Adjustments made to predict EUR for Type Well Process	89
4.1	Aggregate Statistical Summary . . . . .	102
4.2	Angular Similarity Results for Probabilistic Curves . . . . .	107
4.3	Summary of Model Performance on Restricted Dataset . . . . .	118
4.4	Comparison of GMM Probabilistic Results between Full and Restricted Datasets . . . . .	120
5.1	Recap of Research Questions . . . . .	125
A.1	XGBoost Hyperparameter Cross-Validation . . . . .	155
A.2	Random Forest Hyperparameter Cross-Validation . . . . .	155
A.3	LightGBM Hyperparameter Cross-Validation . . . . .	156
A.4	Feedforward Neural Network Hyperparameter Cross-Validation . . . .	156
A.5	BNN Hyperparameter Cross-Validation . . . . .	156
A.6	Aggregate Statistical Summary - Gas and Oil . . . . .	157

# List of Summaries

1.1.1 Primary Research Questions . . . . .	21
1.2.1 DCA Assumptions violated by Shale Gas Wells . . . . .	24
1.2.2 Weaknesses Associated with Modified Hyperbolic DCA . . . . .	25
1.2.3 RTA modifications for Shale Reservoirs . . . . .	28
1.2.4 Lessons Learned from Existing Well Forecasts . . . . .	38
1.2.5 Lessons Learned from Type Well Forecast . . . . .	39
1.4.1 Ensemble Method Advantages and Disadvantages . . . . .	52
1.4.2 BNN Method Advantages and Disadvantages . . . . .	53
4.1.1 Existing Well Forecast Results . . . . .	111
4.2.1 Type Well Forecast Results . . . . .	124
5.3.1 Existing Well Forecast Conclusions to Research Questions . . . . .	145
5.3.2 Type Well Forecast Conclusions to Research Questions . . . . .	146
6.1.1 BNN Workflow Improvements . . . . .	149

# Chapter 1

## Introduction

Machine Learning is on the verge of disrupting the oil and gas industry. With improvements in computational power, available sensor data, and open-source algorithms come powerful tools that can impact all engineering disciplines. However, unlike other industries, some of the inherent qualities of oil and gas make it a larger beneficiary. Unlike a vehicle or device designed and built to precise specifications, a reservoir cannot be manufactured and will always carry substantial uncertainties. Petroleum engineers are challenged with delineating the natural properties of a complex system by interpreting indirect and potentially erroneous signals kilometers away from the source. Historically, this process created analysis subjectivity with numerous physical models capable of explaining the same phenomena. As a result, predictions exhibited tremendous variance between practitioners and have generally comprised deterministic values without any risk estimation. Without including probability, the true unpredictability of the well forecasts has been obscured, resulting in unfavourable commercial agreements and sub-optimal facility designs. Incorrect decisions can lead to disastrous economic consequences, considering the substantial capital required to implement an oil and gas project.

Statistical methods provide one mechanism to combat this problem, but rely on large datasets to inform the most likely outcomes. Consequently, this requirement poses an issue for fields with relatively low well counts compared to more established plays. This thesis seeks to develop a rigorous methodology that applies machine

learning (ML) principles to generate probabilistic production forecasts. These forecasts are then used to predict the reservoir’s expected ultimate recovery (EUR) and identify the best-performing regions with limited data. However, probabilistic ML is still an emerging field with no consensus on the preferred approach; to provide a solution requires researching cutting-edge ML developments, which build upon the best practices of modern production-forecasting techniques.

## 1.1 Production Forecasting Overview

Production forecasting is one of the most critical tasks for a petroleum engineer. Broadly, this process is used for two primary purposes: Analyzing future production on existing wells, and expected performance on new producers. Predicting performance on existing wells is critical across many domains. From a production engineering standpoint, forecasting when a well is likely to stop flowing is crucial to avoid unplanned downtime. As gas rates drop, the liquid starts to accumulate in the wellbore until there is insufficient pressure to deliver volumes to the surface; this process is known as liquid loading. The minimum rate needed to provide enough drag force and overcome the hydrostatic and frictional forces is called critical velocity [54]. Therefore, intersecting the production forecast range with the critical velocity provides the latest and earliest dates that liquid loading can be encountered; this information allows the production engineer to execute artificial lift projects before the earliest date, preventing liquid loading.

Additionally, production forecasting carries a significant monetary impact from a marketing and commercial perspective. Requirements vary regionally for each commodity, but typically require a forecast, also referred to as a nomination, of expected production days [38] to months beforehand [20]. Failure to deliver within a certain tolerance of the nominated volume can result in production restrictions, penalties, or the reduction of realized prices.

These forecasts also play a critical role when predicting performance in undeveloped areas. *Type Wells* are composed of analogous wells in the region and used to

generate curves for gas and oil that represent the most likely production rates over time. These Type Well forecasts are necessary to generate valid economic decisions for regional development, and size infrastructure appropriately. Pipelines, compressors, gas plants, and oil batteries are all multi-year projects that are planned based on the outputs from these forecasts. Incorrect assumptions could lead to disastrous consequences, with take or pay obligations set beyond the deliverability of the wells or facility bottlenecks that limit throughput.

### 1.1.1 Production Forecasting Issues

With the tremendous monetary impacts discussed, it is unsurprising that a significant amount of research has been conducted into both Existing and Type Well forecasts. Existing Well forecasting has progressed from traditional decline curve analysis [8] with the inclusion of physics-based approaches [13, 29] and improved empirical methods [24, 76]. Similarly, best practices for generating type wells have been led by the Society of Petroleum Evaluation Engineers (SPEE) publishing Monograph 3 for predicting expected ultimate recovery (EUR) in undeveloped reservoirs [12]. Despite this research, there are still many critical limitations (a thorough evaluation of these techniques is discussed in [Section 1.2](#)). First, though accuracy has improved for producing wells, the forecasts are inherently deterministic and do not represent a range of possible outcomes. The inclusion of probability would drastically improve the quality of analysis for production engineers and commercial negotiators, as risk can be incorporated into the evaluation. Pragmatically, these use cases do not seek perfect accuracy; instead, by providing a plausible low-end prediction, adequate mitigations can be put in place, such as installing artificial lift systems before the possible onset of liquid loading, or providing nominations at reasonable levels.

Second, the best practices for generating Type Wells rely on statistical methodologies which are sensitive to the number of data points available and data stationarity. This impact is compounded by the natural evolution of drilling and completion designs such as proppant injected and lateral length, as shown in [Figure 1-1](#). These changes effectively reduce analogous data points because if future well designs

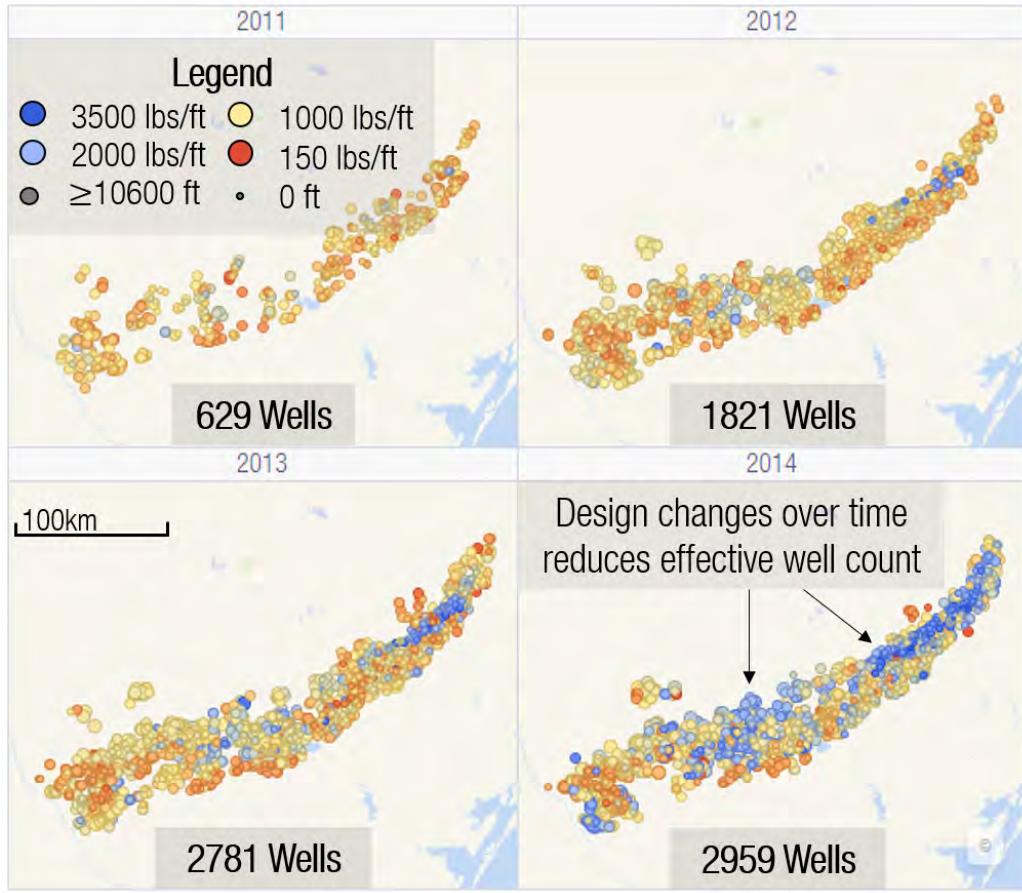


Figure 1-1: Plot of producing wells added to the Eagle Ford Reservoir over time, with markers colored by proppant per foot and sized by lateral length. Though the number of wells added per year increases, the design drastically changes, as evidenced by the shift in the color spectrum. Plot generated using Spotfire [2].

strongly deviate, disproportionate weight will be placed on unrepresentative designs. This limitation provides value in normalizing design variations and has been explored through machine learning techniques [3, 14, 36, 52]; however, these conversions are also deterministic. An improvement would be normalizing for drilling and completion parameters and quantifying the uncertainty of this process.

Third, though the Existing Well forecasts generate the EUR used for the Type Well process, these two methods are independent. There is the potential to utilize a single pipeline to propagate errors and create a repeatable workflow, providing up-to-date estimates of the best reservoir areas.

### 1.1.2 Research Questions and Structure

The product's intended goal to deliver value to stakeholders (system problem statement) can be summarized as follows: to improve well profitability by quantifying the underlying uncertainties in the forecasting process using probabilistic machine learning techniques. The associated research questions are shown in [Summary 1.1.1](#), with an outline of the analysis process in [Figure 1-2](#).

#### Summary 1.1.1: Primary Research Questions

1. Do probabilistic machine learning techniques improve well forecasting compared to existing methodologies in terms of accuracy, ease-of-use, and risk quantification?
2. Can the most productive regions of the reservoir be identified with limited data, defined by less than a thousand wells?
3. Are computational power and practitioner demands sufficiently accessible to create a sustainable and repeatable workflow?

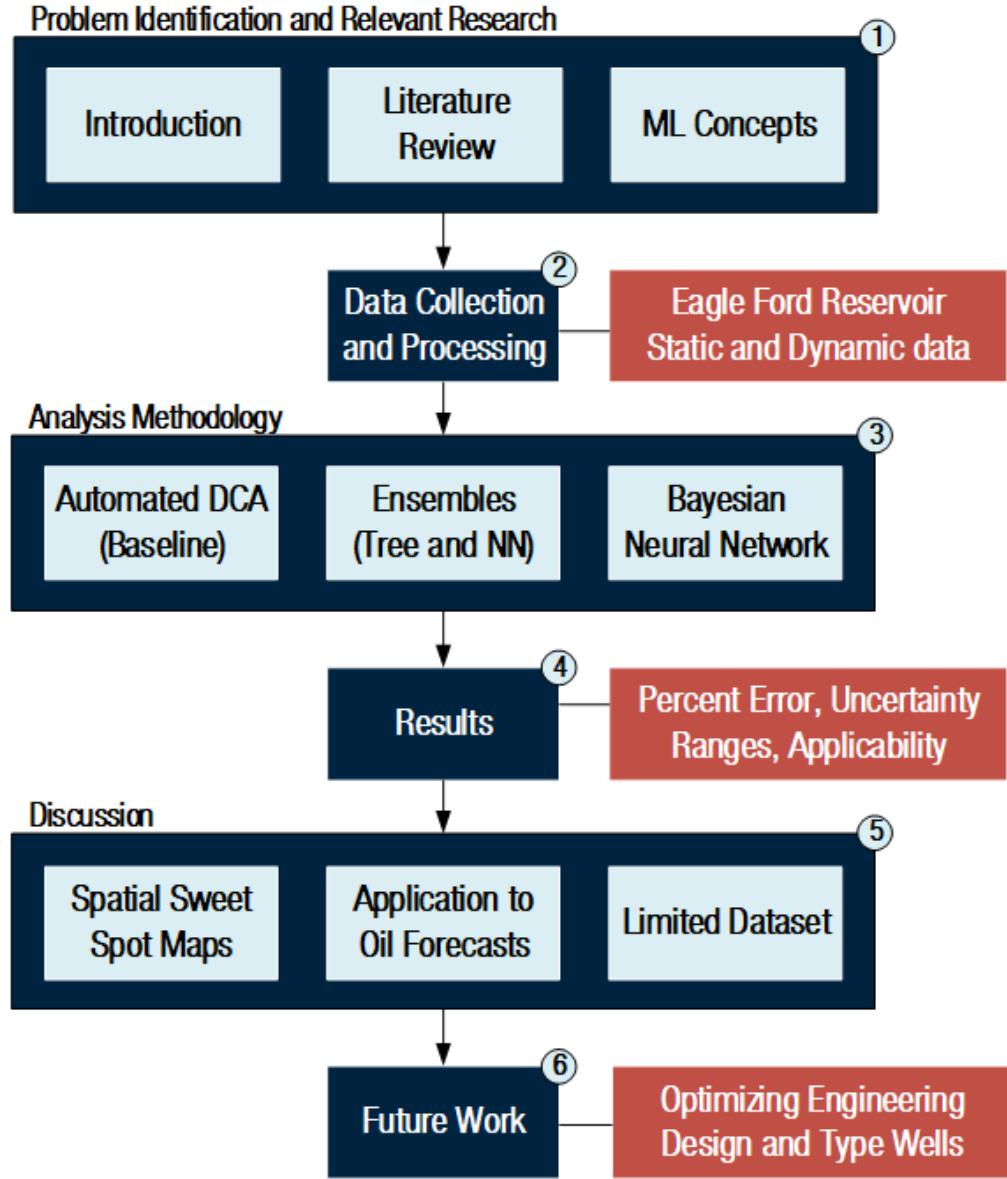


Figure 1-2: Depiction of the thesis workflow, with the numbers indicating the corresponding chapters. First, a thorough evaluation of relevant research and theory will be performed, forming the knowledge foundation for the rest of the thesis. Next, the data-processing decisions will be outlined, creating a common set of features for modelling. The data will then be used for two probabilistic forecasting methodologies and contrasted against traditional Decline Curve Analysis (DCA). How the algorithms perform in different domains and use cases will be discussed for accuracy and robustness. Finally, future work that can build off of these techniques will be described.

## 1.2 Literature Review

This section analyzes the evolution of published literature for Existing and Type Well forecasting. Understanding these methods requires discussion of the context in which they were generated. Throughout the 20th century, production consisted of conventional wells sourced by shale reservoirs. Shale itself was deemed of little interest to develop due to its ultra-low permeabilities generating uneconomic production rates. However, the advent of multiple-fracture stimulation technology unlocked the ability to target the source rock directly, and spurred the development of unconventional wells in the 2010s. As traditional methods for forecasting were applied, engineers quickly discovered issues. These methods used simplifying assumptions for conventional wells and were incompatible with shale; a new set of techniques was required to capture the nuances of unconventional wells.

### 1.2.1 Decline Curve Analysis (DCA) Overview

Decline Curve Analysis is the most common form of production forecasting due to its high degree of flexibility and relative simplicity. This method was developed by Arps in 1945 using an empirical approach to predict future production [8]. Typically, this method relies on the judgement of practitioners to fit data plotted on a rate-time or rate-cumulative basis. An example of this process is shown in [Figure 1-3](#). Equations [1.1](#) to [1.2](#) are used to generate the forecasts [40]:

$$q = \frac{q_i}{(1 + ba_i\Delta t)^{\frac{1}{b}}}; \quad (1.1)$$

$$a_i = \frac{1}{b}[(1 - d_i)^{-b} - 1]. \quad (1.2)$$

These equations use  $b$  a dimensionless value normally between 0 to 1 that affects the degree of curvature,  $q_i$  the initial production rate,  $\Delta t$  the producing time in years, and  $d_i$  the initial decline rate per year. By convention,  $b = 0$  is known as exponential,  $b = 1$  harmonic, and all other  $b$  values hyperbolic. This method of forecasting well

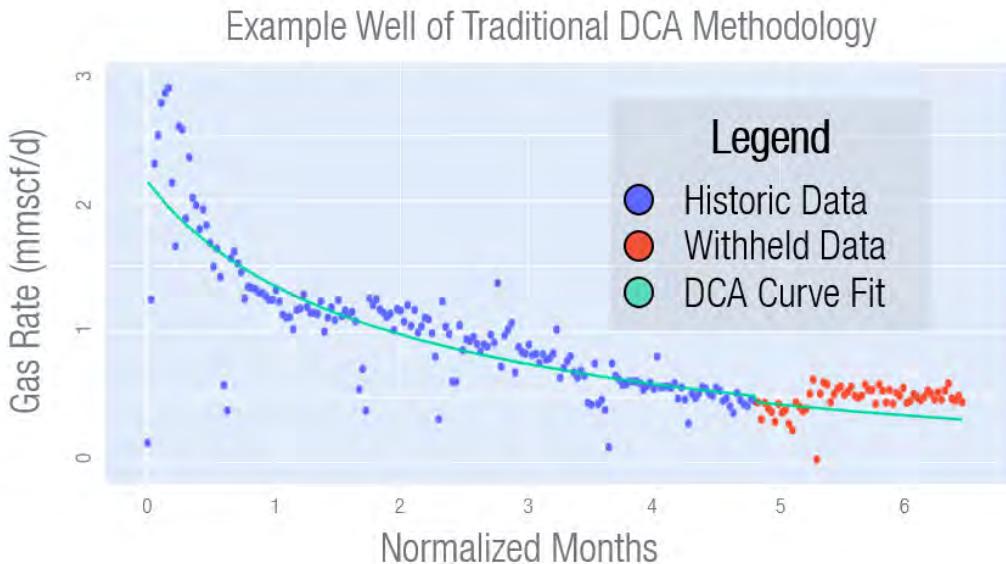


Figure 1-3: An example of a traditional decline curve analysis fit through historical data to predict withheld future production, shown in red. This method is prone to many sources of error. Plot generated from Plotly [42].

production became an industry-standard practice; however, Shale Reservoirs violate some of the fundamental assumptions necessary to have an accurate DCA forecast, as shown in [Summary 1.2.1](#) [19].

#### Summary 1.2.1: DCA Assumptions violated by Shale Gas Wells

1. Well must have fixed drainage radius (steady-state boundary dominated flow (BDF) regime).
  - Shale reservoirs have ultra-low permeabilities causing the wells to produce in transient flow for years before entering BDF. Transient flow creates a continually shifting drainage radius.
2. Constant well flowing (bottomhole) pressure.
  - This assumption is unlikely to occur due to consistent operational setpoint changes for choke sizes, pipeline network effects, and interference from nearby producers. Pressure buildup and depletion are highly nonlinear processes, which exist at far longer timescales than for conventional wells.

3. Well has a consistent productivity index and static reservoir properties.
- Numerous factors can cause productivity loss over time, including propellant crushing, parent-child well frac hits, and pressure-dependent permeability. Productivity from Shale wells is dominated by the fracture network, which is more susceptible to changes than the rock matrix.

### 1.2.2 Modified Hyperbolic Declines

Due to these violations in unconventional, a  $b$  value greater than 1 will occur, creating a horizontal asymptote for production rate and causing reserves to tend to infinity. To compensate, Robertson published a modified Arps equation in 1988 [70], transitioning the  $b$  value to 0 once the decline reaches a minimum percent change per year ( $d_{\text{lim}}$ ). Once  $d_{\text{lim}}$  is encountered, the curve will follow an exponential decline utilizing [Equation 1.3](#) to [1.5](#) from [40]:

$$q = q_{\text{lim}} e^{a_{\text{lim}}(t-t_{\text{lim}})}; \quad (1.3)$$

$$q_{\text{lim}} = q_i \left( \frac{a_{\text{lim}}}{a_i} \right)^{\frac{1}{b}}; \quad (1.4)$$

$$t_{\text{lim}} = \frac{\left( \frac{q_i}{q_{\text{lim}}} \right)^b - 1}{ba_i}. \quad (1.5)$$

Typical values for  $d_{\text{lim}}$  range between 5–10% and are estimated using analog wells with sufficiently long production histories. Despite this modification, numerous weaknesses affect both the accuracy and utility of the model. The main sources of criticism is given by [Summary 1.2.2](#) [35, 50, 55]:

#### Summary 1.2.2: Weaknesses Associated with Modified Hyperbolic DCA

- The DCA equation does not consider reservoir fluids, geological parameters, and well designs. Formulas are empirically derived with no direct connection to physics-based equations. Applying a terminal exponential decline is a

- heuristic to avoid infinite reserves and can be prone to error.
2. The process does not directly apply lessons learned from analogous wells with similar properties. Transferring this knowledge is the practitioner's responsibility outside of the algorithm and is prone to omission.
  3. DCA is inherently subjective, with only a single deterministic curve generated. It is difficult to measure whether the fit is accurate from performance metrics alone, as each analyst must apply judgement to emphasize different portions of the historical period considered representative.
  4. Generating forecasts is a time-consuming process generally conducted manually. With potentially thousands of wells requiring forecasts, either a single deterministic curve is created, or an entire field is aggregated to create a system-level prediction, which is highly inaccurate. Single predictions obscure the uncertainty associated with the prediction process.

### 1.2.3 DCA Methodology Enhancements

Several different approaches have been explored over time to improve the forecasting process and compensate for these weaknesses. This thesis will discuss three strategies, including rate transient analysis, improved empirical methods, and machine learning.

#### Rate Transient Analysis

The first significant modification was the incorporation of a more rigorous physics-based methodology to the Arps equations. Broadly, these approaches are known as rate-transient analysis (RTA), a form of pressure-transient analysis (PTA). RTA incorporates both rate and pressure, accounting for dynamics in operating conditions and fluid properties. Fetkovich was one of the first to incorporate analytical solutions of the diffusion equation [29]. This method focused on a single flow regime (constant-

pressure radial flow) and integrated results with DCA for boundary-dominated flow.

Fetkovich also utilized a series of assumptions to make the process easier to apply. First, **dimensionless rate-time variables** were used to compress the solutions into characteristic curves, simplifying the fitting process. Additionally, pseudovariables were introduced to account for fluid behaviour differences in gas and oil. Specifically, gas properties, including viscosity ( $\mu_g$ ), compressibility factor ( $Z$ ), and compressibility ( $c_t$ ) change with pressure. Utilizing pseudopressure ( $p_{\text{pseudo}}$ ) and pseudotime ( $t_{\text{pseudo}}$ ) in Equations 1.6 and 1.7 accounts for these changes, allowing gas to be treated like a liquid with constant properties [28]. The results of this process showed that the  $b$  value corresponds with specific reservoirs and flow regimes; these changes are a significant improvement over DCA, as rather than utilizing *any*  $b$  value, the range is limited to *physically permissible bounds*:

$$p_{\text{pseudo}} = 2 \int_{p_0}^p \frac{p}{\mu_g Z} dp; \quad (1.6)$$

$$t_{\text{pseudo}} = \mu_g c_t \int_0^t \frac{1}{\mu_g c_t} dt. \quad (1.7)$$

One weakness in the Fetkovich method is that it does not consider transient flow and can never achieve a  $b > 1$ . Blasingame, in 1989, advanced this RTA technique by allowing for variability in rate and pressure [13]. Generally, these RTA methods require simple boundary conditions, with either constant flow rate or pressure to create tractable solutions; however, it would be impossible for a well to produce at a fixed rate or pressure for its entire lifecycle. Typically, the rate starts high and diminishes over time as energy is depleted. Blasingame's critical insight was the concept of material balance time ( $t_{\text{mb}}$ ) represented in Equation 1.8:

$$t_{\text{mb}} = \frac{Q}{q}. \quad (1.8)$$

Rather than analyzing the entire production history at once, the curve is divided into individual components. Each data point is equivalent to a constant rate ( $q$ ), producing for the period ( $t_{\text{mb}}$ ), to generate the cumulative volume ( $Q$ ). Thus, by superimposing

all of the simple components together, complex curvatures can be modelled. This approach successfully maintains the constant-rate assumption required to solve the flow equations, while providing the capability of determining transient and boundary-dominated flow regimes.

The next major evolution in RTA occurred with Shale reservoirs. In 2013, Clarkson studied this phenomenon in-depth, summarizing the RTA research to date and providing commentary on the adjustments necessary for unconventional [19]. The key differences with conventional reservoirs are highlighted in [Summary 1.2.3](#).

#### Summary 1.2.3: RTA modifications for Shale Reservoirs

1. Natural fractures in the matrix and high initial pressures generate complex, tortuous completion geometries; these fractures significantly deviate from the bi-wing planar shapes observed in conventional reservoirs.
  - The sequence and observed flow regimes are different for unconventional. Radial and elliptical transitional flows are not observed in shale reservoirs. Linear flow signatures tend to dominate.
2. Pressure-dependant permeability and gas adsorption from organic material in the matrix alters reservoir properties over time.
  - Pseudotime and pseudopressure calculations need to compensate for these effects and utilize modified equations.
3. Three-phase (gas, condensate, and water) and non-Darcy (e.g., turbulent) flow alters the fluid behaviour and requires modifications.
  - Incorporate relative permeability changes as a function of fluid saturation. This adjustment is a complex procedure based on estimating multi-phase flow potential.

In conclusion, RTA techniques provide rigorous physics-based solutions. However, these methods require the quantification of numerous parameters, notably pressure, fluid properties, and fracture geometry. These values are not typically available on

competitors, limiting analysis scope when evaluating new areas. Additionally, as shown in [Summary 1.2.3](#), the reservoir system is highly complex, making solutions with **first principles challenging to derive**. Many authors have offered simplifications to make this process easier; regardless, these solutions are still inherently subjective, with numerous potential hypotheses capable of fitting the problem. In favourable conditions, these methods are significantly more accurate than DCA but require both sufficient data and technical expertise to reach this accuracy. Ultimately, the ease of DCA makes it more ubiquitous compared to RTA techniques.

## Improved Empirical Curve Fitting Methods

Another approach to overcome the limitations of DCA is through improved empirical curve fitting methods. In 2010, Valko published the concept of a stretched exponential production decline model (SEPD) [\[76\]](#). Like DCA, this forecasting method uses only production volumes, avoiding the data burden associated with RTA. SEPD includes a sum of exponentially decaying functions, adding  $n$  and  $\tau$  parameters to the DCA equation shown in [Equation 1.9](#). The parameter  $n$  dictates the size of the distribution tails, while  $\tau$  is the median value of the time constants. The main improvement of this technique is that it can generate a finite estimated ultimate recovery (EUR) without requiring a  $d_{\text{lim}}$  parameter since the function can naturally decay into an exponential function later in life due to the variable tail lengths [\[46\]](#):

$$q = q_i e^{1 - \left(\frac{t}{\tau}\right)^n}. \quad (1.9)$$

Duong in 2011 also published an improved empirical method [\[24\]](#). This approach combines elements from RTA and aligns with observations from Clarkson. The most critical assumption is that production maintains a linear flow regime, as it is dominated by fracture flow and has negligible contributions from the matrix. [Equation 1.10](#) governs Duong's model, where  $m$  is slope and  $a$  is intercept of the log-log plot for  $q/G_p$  vs  $t$ :

$$\frac{q}{q_i} = t^{-m} e^{\frac{a}{1-m}(t^{1-m}-1)}. \quad (1.10)$$

Though this method improves the underlying physics compared to DCA, the constant linear flow assumption is inappropriate after the transition to boundary-dominated flow. This effect demonstrates the fundamental weakness of empirical techniques: liberties must be taken with the simplifying assumptions to reduce the reservoir complexity to a manageable level. It is improbable to have a simple empirical equation that can fully capture all conditions and flow regimes. Also similar to the RTA techniques, Duong demonstrated how utilizing dimensionless values provides several simplifications and improves the model fitting process. This effect is accomplished by dividing rates by the maximum value ( $q_{\max}$ ) and time by when  $q_{\max}$  was encountered ( $t_{\max}$ ). The overall impact of these changes was removing the  $a$  factor, as shown in Equations 1.11 and 1.12:

$$t_m = \frac{t}{t_{\max}}; \quad (1.11)$$

$$\frac{q}{q_{\max}} = t_m^{-m} e^{\frac{m}{1-m}(t_m^{1-m}-1)}. \quad (1.12)$$

These advanced empirical models showed improved performance compared to Arps' DCA, but failed to match the accuracy of detailed RTA models. The main benefit of this approach was reduced complexity, which allows them to be used on publicly available datasets that would be incompatible with RTA due to missing inputs. However, these techniques are not commonly included in many commercial software packages or require a complex parameter fitting process, offsetting the main benefit. DCA offers a more straightforward implementation, which is a critical need for analysts, who are often required to manually forecast hundreds or thousands of wells. These practitioners do not have the capacity to implement a more demanding process.

## Machine Learning Methods

With the advancements in computational power, the next major trend was incorporating Machine Learning (ML) into the forecasting process. These analyses serve as a bridge between empirical and physics-based methodologies. ML provides the flexibility of training the algorithms on whatever input features are available without explicitly deriving the relationships based on first principles; thus, numerous data

types can be incorporated, such as geospatial, completion, geological, and production data. Note that the theory behind these techniques is explored in greater detail in [Section 1.3](#).

In 2013, Bansal was one of the first to attempt production forecasting utilizing an artificial neural network (ANN) [10]. This model consisted of a feedforward neural network of well log, seismic, and completion data. The output contained predictions of cumulative oil production at 3-month timeslices between 3 and 24 months. The main advantage of this approach was that it simplifies the high degree of reservoir complexity, described in [Section 1.2.3](#). Utilizing a large amount of input data allowed the model to capture complex effects caused by nearby producers, completion designs, and geological factors without a derivation of the underlying physics. In 2019, Han built on this ANN structure with the addition of unsupervised learning techniques ( $k$ -means) to cluster wells based on similar properties [36]. Finally, Li in 2020 utilized dynamic production resampling (DPR) to convert cumulative volumes into a dimensionless value similar to the RTA and the Duong techniques [55]. DPR demonstrated a significant improvement in forecasting accuracy. This benefit is likely attributed to compressing the solution space into characteristic shapes, allowing the algorithm to learn from similar trends and improving the translation between wells.

#### 1.2.4 Type Well Generation Overview

The previous section provided a thorough description of different Existing Well forecasting techniques; however, this process raises the question of how new wells with insufficient data or no production history can be forecasted. This category of analysis is referred to as *Type Well*, as introduced by Lee [48]. Typically, these forecasts rely on statistical methods of aggregating analogous producers with similar characteristics. Incorporating statistics is necessary due to the nature of shale reservoirs. As described in [Summary 1.2.3](#), unconventionalals are distinct from conventionals due to complex fracture geometries and long horizontal laterals, leading to localized pressure-depleted zones. These effects are compounded by random variations in the executed versus planned treatment, pressure loss injecting through the lateral, and uncertainties in

controlling sand (proppant) placement in each frac stage [64]. As a result, shale reservoirs are highly heterogeneous, and have been referred to as statistical plays since performance can significantly vary within a relatively small region [9]. These effects place greater reliance on Type Wells, as if the area's aggregate production is accurate, system-level strategic decisions are valid even if individual predictions are wrong.

A standard industry practice utilizes the nomenclature of P10, P50, and P90 to correspond to the 10th, 50th, and 90th percentiles of a cumulative probability curve. P10 would indicate a 10% probability of having volumes **below this value** or a 90% chance of being above. Note that this nomenclature is not always consistent as some companies misinterpret percentiles as being greater than this value; a P10 in this context would indicate a 10% chance of being above this value. This thesis will always utilize the **less than** convention. Additionally, these values align with important metrics for reserve bookings, which are referred to as *proved*, *probable*, and *possible*. The Securities Exchange Commission (SEC) requires that every company report the total *proved* reserves each year, which have a 90% chance of being produced and corresponds with the P10 volume [67]. The *probable* volume corresponds with the P50, while *possible* is P90. These two categories are fundamental internal metrics to manage shareholder expectations, but are not required to be reported to the SEC. Many investors look at the reserves-replacement ratio (RRR) represented in [Equation 1.13](#). If this ratio is less than 100%, the company shows signs that current operations are not sustainable, negatively impacting the stock price. Therefore, it is critical to track *probable* and *possible* reserves, as they can be moved to *proved* once derisked, improving the RRR:

$$\text{Reserves Replacement Ratio} = \frac{\text{Proved Reserves Added}}{\text{Volumes Produced}}. \quad (1.13)$$

Due to the potentially high business impact, the Society of Petroleum Evaluation Engineers (SPEE) published Monograph 3 in 2010, guiding the determination of probabilistic ranges for undeveloped reserves [12]. This document was created by a

committee of ten industry experts, who introduced statistical concepts that formed the foundation of all future Type Well work. One key statistical assumption was that expected ultimate recovery (EUR) always follows a **lognormal probability distribution**. Lognormal is a type of continuous distribution in which the natural logarithm of the random variable is normally distributed [26]. The cause of this behaviour is due to a corollary of the central limit theorem that states that the **sum** of independent random variables tends to follow a normal distribution. That is, Gibrat's law states that the **product** of independent random variables tends to follow a lognormal distribution [22]. When applied to the context of reserves, the Volumetric Equation (1.14) for oil in place ( $N$ ) demonstrates Gibrat's law. The equation consists of the **product** of reservoir area ( $A$ ), reservoir thickness ( $h$ ), porosity ( $\phi$ ), irreducible water saturation ( $S_w$ ), and oil formation volume factor ( $B_{oi}$ ) [39]:

$$N = 7758 \frac{Ah\phi(1 - S_w)}{B_{oi}}. \quad (1.14)$$

These variables will tend to be lognormally distributed; thus, the product  $N$  of the multiplicative equation should also follow a lognormal distribution. This tendency has been proven accurate across numerous unconventional reservoirs and provides a critical tool in generating type wells [30, 59]. When properly grouped into analogous regions, plotting the log of EUR and percentiles results in a straight line consistent with a lognormal distribution for EUR near its median value (see Figure 1-4). This process allows the analyst to fit a slope representing the most likely production distribution, which is a recommended best practice from the Unconventional Assessment and Valuation course by Rose and Associates [9]. The expected value (EV) of the reserves is equal to the mean of the distribution, which can be calculated with Equations 1.15 to 1.17, where  $e^\mu$  is the median,  $\sqrt{e^{\sigma^2} - 1}e^{\mu+\sigma^2/2}$  is the standard deviation, and EV is the expected value or mean. Using these equations along with Figure 1-4 allows the *proved*, *probable*, and *possible* reserves to be determined:

$$\mu = \frac{\ln(P10) + \ln(P90)}{2}; \quad (1.15)$$

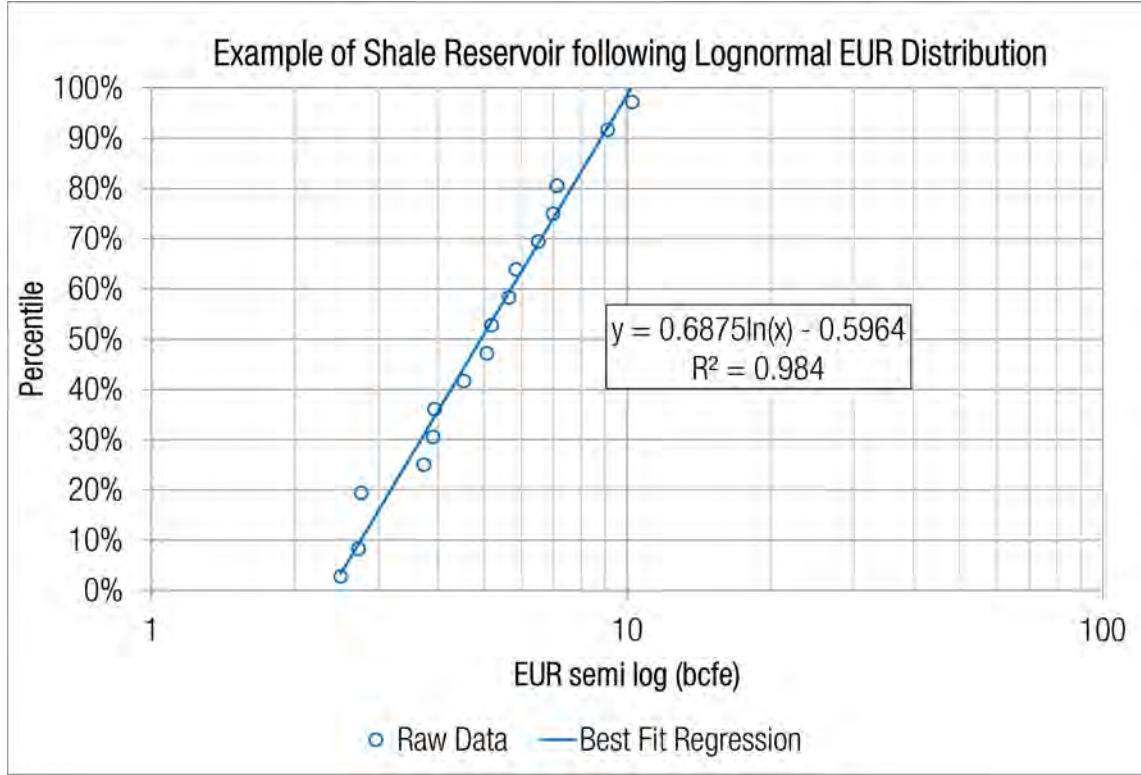


Figure 1-4: An example of a semi-log plot to determine uncertainty range for reserves in an unconventional shale gas field. The straight-line fit is evidence that a lognormal distribution exists, providing a tool to generate an uncertainty range.

$$\sigma = \frac{\ln(P90) - \ln(P10)}{2\sqrt{2} \operatorname{erfc}^{-1} \frac{1}{5}}, \quad (1.16)$$

where  $\sqrt{2} \operatorname{erfc}^{-1} \frac{1}{5} \approx 1.282$ ;

$$EV = e^{\mu + \sigma^2/2}. \quad (1.17)$$

### Weaknesses of SPEE Monograph 3

Although this method provided advancements in determining a probabilistic range, there are several limitations. Lee demonstrated that to be an effective technique, a minimum of 50 to 100 wells are necessary for each region; however, in practice, there are often insufficient wells to satisfy this requirement [48]. Additionally, SPEE Monograph 3 is susceptible to numerous types of bias, as shown by Freeborn and Russell [31, 63, 64]. First, though numerous commercial packages exist, the industry generally performs forecasting manually. This process introduces subjectivity, as an analyst

must distinguish signal from noise, creating tendencies to underestimate or overestimate production. Also, due to the procedure being applied to potentially thousands of wells, an engineer may not have enough time to forecast each well individually. Instead, a common pitfall is to include only historical production into the type curve, introducing **survivorship bias** by excluding newer producers with less data [64].

Understanding survivorship bias requires the context in which an oil and gas reservoir is typically developed. First, a field will evolve by drilling wells to explore different regions and appraise the potential. Experiments are also conducted with drilling and completion parameters to improve production. The last phase is a development program, which adds substantially more wells to the best areas of the field. If there is any improvement in the well designs, the new producers should outperform the old ones in the same acreage. Second, even if the engineering design has no impact, developing the best portions of the reservoir first leads to changing reservoir quality over time. Shale reservoirs are highly susceptible to this effect, as old wells could create low-pressure zones or new wells could damage the existing wells during completion (frac-hits); overall, these effects reduce future well productivity [59]. A dilemma emerges, as generating an accurate Type Well requires sufficient data points; however, due to changing effects over time, though the total number of wells increases, the number of **representative wells** stays low. This effect violates the **principle of stationarity** that is usually assumed for achieving a valid statistical result.

If production from new producers can be imputed accurately, they can be incorporated into the type curve process; effectively, these wells will increase the number of representative designs, reducing the effect of non-stationarity. An example of this strategy is provided in [Figure 1-5](#). However, this approach creates a tradeoff as the imputation will generate some degree of error. Therefore, the increase in knowledge provided must outweigh the potential error introduced. This principle is discussed further in the context of uncertainty types in [Section 1.4.1](#). Other methods to mitigate a non-stationary problem vary based on engineering and geological parameter types. Engineering parameters consist of design variables that can be directly controlled. Examples include lateral length, proppant tonnage, frac fluid volume, tubular

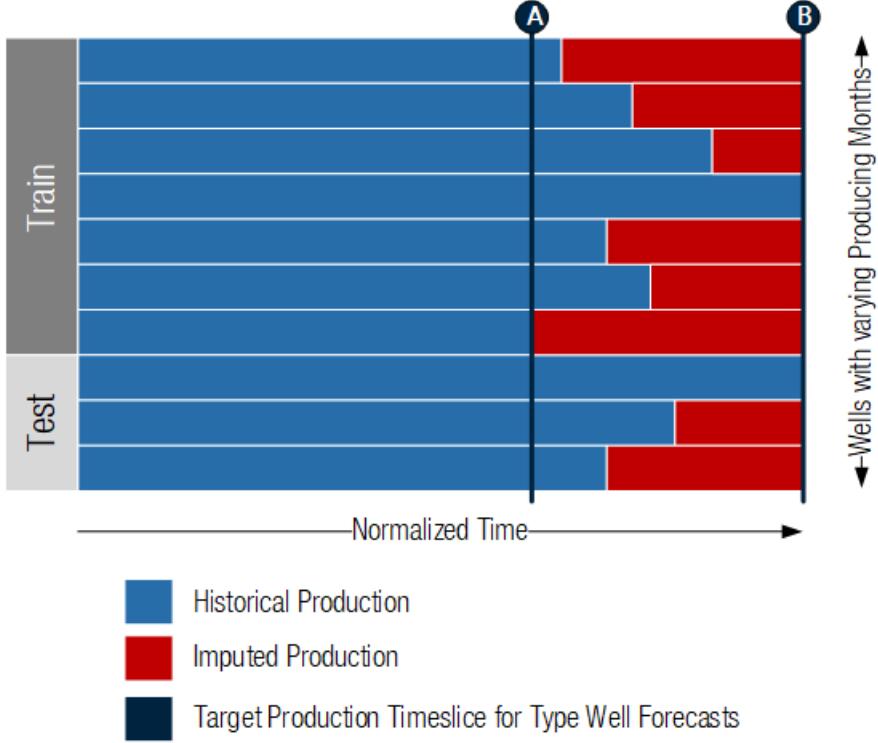


Figure 1-5: An illustration of two approaches to generate the Type Well target variable. The bar length corresponds with the number of months on production. Timeslice *A* truncates data based on the fewest producing months, ensuring only measured data are utilized. However, since Type Well forecasts aim to predict late-time production, without imputation only older producers, with sufficiently long histories can be incorporated, resulting in survivorship bias due to misrepresentative designs. Timeslice *B* demonstrates another approach, where data are imputed to predict future production. This strategy increases the number of representative wells and decreases the effect of survivorship bias. The downside created is the potential error generated in the imputation process; if this error is limited, approach *B* should theoretically improve forecast accuracy compared to *A* [64].

diameters, and choke sizes. Conversely, geological parameters are not under the direct control of an engineer, such as fluid density, viscosity, pressure, permeability, porosity, and natural fracture density. Since they are naturally occurring, they can only be enhanced or exploited indirectly by engineering variables. Several heuristics have been proposed to compensate for changes in engineering parameters, by Freeborn [31] and Miller [59]. These methods apply normalization factors to adjust production results using simple ratios of key parameters. An example is presented in Equation 1.18, where  $l$  is the fracture lateral length,  $p$  is the proppant tonnage, and  $\Delta q$  is the change

in production rate after year 1.

$$\text{Scaling Factor} = \left( \frac{l_{\text{well}}}{l_{\text{analog}}} \right) \left( \frac{p_{\text{well}}}{p_{\text{analog}}} \right) \left( \frac{\Delta q_{\text{well}}}{\Delta q_{\text{analog}}} \right) \quad (1.18)$$

A similar principle can be applied to geological parameters. Yip offered a new approach to exploring the variation of unseen geological values [83]. His work demonstrated that as additional information is gained, the sizes of the regional areas diminish continually, as some reservoir properties are highly discretized; therefore, incorporating sparse geological data can create the false conclusion that the reservoir is relatively homogeneous. Techniques such as Monte Carlo simulation are required to represent the true uncertainty in undeveloped regions.

## Machine Learning Enhancements

Normalization of the production data to engineering and geological features is an improvement that reduces data non-stationarity. However, applying simple ratios assumes a purely linear transformation, which is a poor approximation of reality. A simple example occurs with lateral length. As horizontal length increases, so does the contacted reservoir area; though, at *large deviations*, the stimulated rock volume (SRV) near the end of the well (toe) may not match the SRV at the beginning (heel). Since a pressure differential enables fluid flow, the toe will always experience a greater pressure loss than the heel, reducing fracture quality during stimulation and the effective drawdown when operating [31, 48]. To generate a nonlinear correlation, in 2016, Alaboodi applied an artificial neural network to the problem [3]. The goal of the analysis was to determine the degree that engineering and geological variables impact EUR. These volumes were estimated manually through Arp's and SEPD analysis. A plot was then generated, demonstrating the effect that each feature had on EUR in isolation. Results showed that of the 34 tested parameters, **none** of the features followed a linear trend. The results also supported that both engineering and geological parameters had a substantial impact on EUR.

Bowie, in 2017, also performed a similar analysis on 200 wells in the Duvernay

[14]. That paper directly compares multivariate linear regression and a neural network model. Based on the test results, the neural network significantly outperformed multivariate linear regression based on accuracy metrics. Bowie also described how the model could tailor engineering designs, with a potential improvement of 20–100% if more optimal selections were made. Lastly, Kocoglu in 2020 compared a neural network model to other advanced geostatistical techniques such as Kriging [52]. Kriging is a sophisticated process commonly used for geological mapping, which considers the autocorrelation between nearby points [27]. Covariance equations and variograms are used to calculate the spatial correlation as a function of distance. The variograms consist of different patterns, including circular, spherical, exponential, Gaussian, and linear; based on the analyst’s interpretation of the expected trend, the most appropriate variogram is selected for the application. Kocoglu performed DCA manually to estimate the EUR and then applied ML and geostatistical techniques. Results showed that the neural network outperformed the geostatistical methods based on accuracy and generalizability metrics.

### 1.2.5 Summary of Lessons Learned

With the theory behind forecasting techniques established, the key lessons learned are provided by [Summary 1.2.4](#) and [1.2.5](#):

#### Summary 1.2.4: Lessons Learned from Existing Well Forecasts

1. The assumptions for Arps Decline Curve Analysis (DCA) with a modified hyperbolic equation violate fundamental assumptions for unconventional shale reservoirs. This method should be used with caution.
  - Lessons Learned: An ideal methodology needs to be relatively simple to implement or will never be used at scale, even if more accurate. Automation is critical to generate a sustainable workflow.
2. Rate transient analysis provides a significant improvement in accuracy over DCA with the inclusion of physics-based principles. However, it requires sig-

nificantly more inputs that may not be available, and is inherently subjective due to numerous complex mechanisms.

- Lessons Learned: Incorporating geological, engineering, and spatial data is essential to capture the underlying physics and transfer knowledge between wells. Each data type plays an important role.
3. Enhanced empirical equations incorporate some principles of rate-transient analysis (RTA) but are not an ideal solution as they are specialized for specific conditions.
- Lessons Learned: Fit must apply to the entire production history. Using dimensionless variables improves model performance, as curves are compressed into characteristic solutions that are more generalizable.
4. No method provides a non-heuristic-based solution to generating an uncertainty range.
- Lessons Learned: An ideal forecast removes analyst subjectivity while directly quantifying the probabilistic uncertainty range.

#### Summary 1.2.5: Lessons Learned from Type Well Forecast

1. Generating probabilistic estimates for P10, P50, and P90 are critical business processes but rely on large well counts, which may not be available.
  - Lessons Learned: The ideal methodology needs to provide a solution for limited data points. EUR is expected to follow a lognormal distribution, supposing it is the product of numerous multiplicatively generated properties.
2. Due to changes in engineering and geological parameters over time, this category of problem is non-stationary, leading to errors in applying statistical methods.

- Lessons Learned: Must incorporate normalization techniques to try to provide a stationary problem. Adjustment factors are unlikely to follow linear trends. An ideal method can interpret well results and create local geological clusters of equivalent quality.
3. Numerous discrete and manual processes exist, resulting in a complex and slow workflow.
- Lessons Learned: Process must be relatively fast to implement and include newer wells when capable of accurately imputing future production; failure to do so leads to survivorship bias, generating results that are unrepresentative of current designs on limited datasets.
4. Methodology for forecasting existing wells is largely held separate from type well generation. If any bias occurs during the existing well forecasting, it will translate directly into the type well.
- Lessons Learned: Opportunity to utilize a pipeline that can propagate errors from extrapolation, normalization, and spatial analysis into the uncertainty range.

## 1.3 Machine Learning Concept Generation

Now that the needs of Existing and Type Well forecasts have been established, this thesis will transition to applying Machine Learning (ML) techniques to generate a probabilistic forecast to mitigate known limitations. First, ML and Artificial Intelligence (AI) require definition, as they are often used interchangeably. AI is a broader term, defined as a computer-based technique that mimics natural human intelligence. ML is a specialization within AI for applications that create functions without being explicitly programmed [4]; the chosen functions depend on the nature of the problem but fundamentally attempt to maximize *learning* i.e., correctly assigning classes or computing characteristics of patterns within the data. In *supervised learning*, these patterns can be used to solve either *classification* or *regression* problems. Classification involves assigning results into various categories, while regression creates models to predict a numerical output from an input, to adequately fit a given input-output data set. This thesis requires a regression-based solution, which will be investigated using two main algorithms: tree-based and neural networks.

### 1.3.1 Tree-Based Methods

Decision trees are a popular class of machine-learning algorithms. They consist of recursive partitioning of features to classify points either for regression or classification [11]. Figure 1-6 depicts this process visually. A question is asked at each node resulting in either a true or a false response. These questions are flexible and could consist of whether a feature quantity is above or below a certain value, or equal to 0 or 1 for Boolean variables [75]. Unlike for other methods, data normalization is not strictly required since the cutoffs will adjust to the range of individual features. Generally, each tree-based method sorts the sequence of decisions based on minimizing impurity (Gini coefficient) [11]. The intuitive interpretation behind this concept is to organize the tree structure so that decisions that **increase predictability** the most are asked **first**. Decisions deeper into the tree are refinements and have a reduced impact. For regression problems, the output value ( $\hat{y}$ ) is the mean of the data points

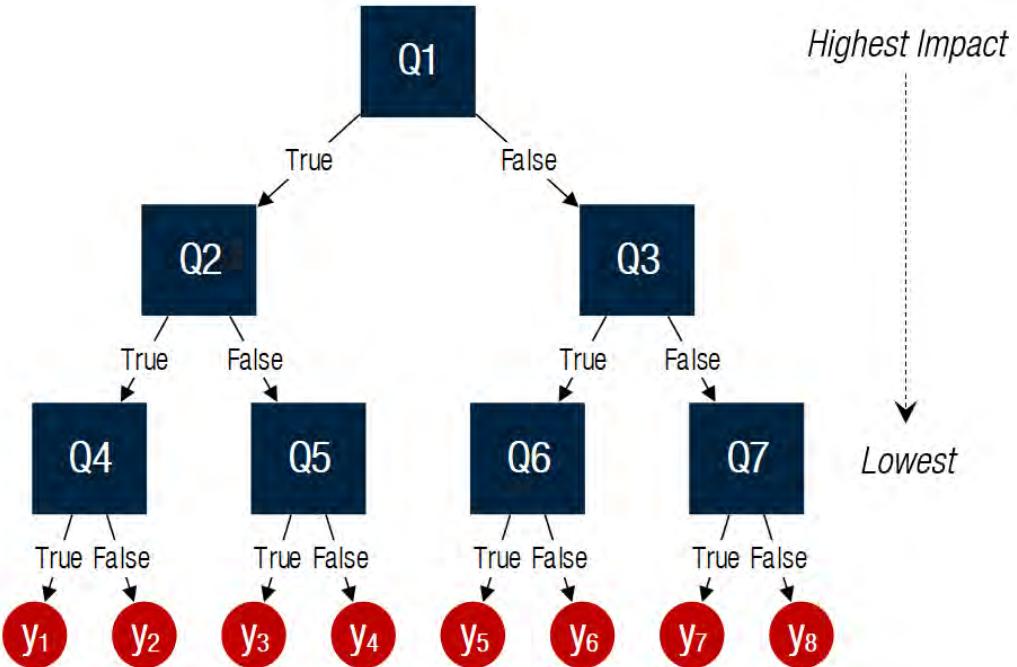


Figure 1-6: Example of a Tree Based ML structure. Conceptually a series of true or false questions are asked to minimize impurity and identify what decisions are most predictive of the target value. Each algorithm has different settings that control where branches are added and how deep or complex they can become.

which satisfy the same decision criteria.

Tree-based methods are practical algorithms, as they apply across a wide range of problems, model nonlinear behaviour, and function effectively with limited datasets [75]. Two modern applications are Random Forest [62] and XGBoost [17]. Though each follows a similar structure, there are differences in regard to where branches are located, tree depth, and the variables included. Random Forest (RF) is a type of ensemble-based decision tree. Rather than forming a single tree with all variables, it generates a series of weak predictors by randomly selecting a subset of the available input features [62]. The predicted value is then the arithmetic average from the predictions of all weak trees. This method has been proven effective, as it avoids overfitting data by only utilizing a subset of available inputs for each tree and considers the effect of less critical variables. Overall the random forest algorithm results in a series of deep trees that control variance by averaging numerous predictors to generate a consensus.

XGBoost (extreme gradient boosting) is another form of decision tree analysis. It differs from RF by taking into consideration where the error is occurring in the model. Rather than solving for a series of independent trees, it calculates where the maximum error is occurring and sequentially adds targeted splits to compensate. In the  $n^{\text{th}}$  calculation round, another tree is added to fit the residual error. Lastly, XGBoost incorporates regularization, which penalizes model complexity [17]. The net effect of these assumptions is a shallower tree compared to Random Forest but with stronger learners that adopt to error trends. Both approaches have shown to be effective in ML competitions, with the preferred method depending on the dataset.

### 1.3.2 Neural Network

Neural Networks (NN) are a class of ML algorithms based loosely on the brain's biological structure. The NN functions like a neuron passing information between synapses; these values are then transformed by linear and nonlinear functions, resulting in an output value. [Figure 1-7](#) depicts this process visually. First, the features of the model ( $x_1, x_2, \dots, x_n$ ) are passed into an input layer and then to a *hidden layer* consisting of numerous *nodes* or *units*. The analyst can freely specify the number of nodes and layers desired to fit the problem. A key characteristic is that edges connect each node to some or all nodes from a previous layer. Each edge ( $i$ ) has an associated weight ( $W_i$ ), which multiplies the input features or output of a node [68]. Additionally, a trainable bias term ( $B_i$ ) is added, to shift node values. Finally, the  $i$ -sum of  $W_i x_i + B_i$  is calculated at each node to give a value ( $z$ ). This sum, using the generic [Equation 1.19](#), is passed through an activation function  $f(z)$  that can take a variety of forms shown by Equations [1.20](#) to [1.25](#) [56]:

$$z = \sum_{i=1}^n (W_i x_i + B_i). \quad (1.19)$$

[Figure 1-8](#) demonstrates examples of the plotted activation functions. The result of this process is an output layer, which generates a value ( $\hat{y}$ ). Though this structure seems complicated, in its simplest form, it behaves identically to linear regression.

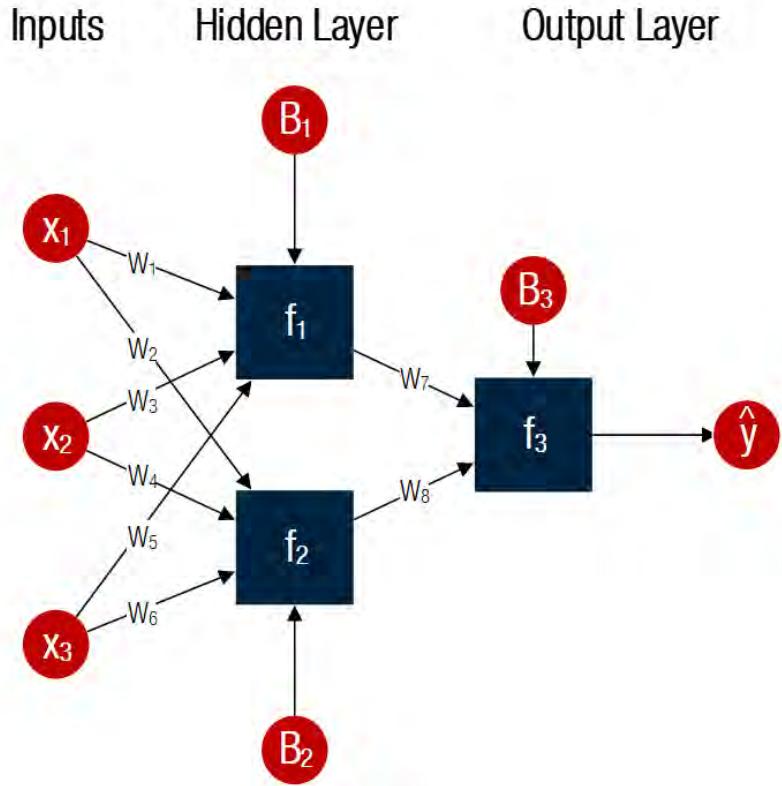


Figure 1-7: An example of a Neural Network structure. Key features include nodes ( $f$ ) directly connected by edges (represented by arrows) to all nodes in the previous layer. These connections pass inputs ( $x$ ), multiplied by weights ( $W$ ), and shifted by a bias term ( $B$ ). The sum of this process ( $z$ ) is transformed either linearly or nonlinearly through an activation function  $f(z)$ . Repeating this process through all nodes and layers generates an output ( $\hat{y}$ ).

Suppose there was only **one input** variable and **one output** node with a **linear activation function**. The form would consist of  $\hat{y} = f(W \cdot x + B)$  or  $\hat{y} = m \cdot x + b$  substituting [Equation 1.25](#). The NN can model more complex behaviours by including more inputs, nodes, layers, and nonlinear activation functions.

$$\text{sigmoid}(z) = \frac{1}{1 + e^{-z}}; \quad (1.20)$$

$$\text{relu}(z) = \max(0, z); \quad (1.21)$$

$$\text{swish}(z) = \frac{z}{1 + e^{-\beta z}} \xrightarrow{\beta \rightarrow \infty} \text{relu}(z); \quad (1.22)$$

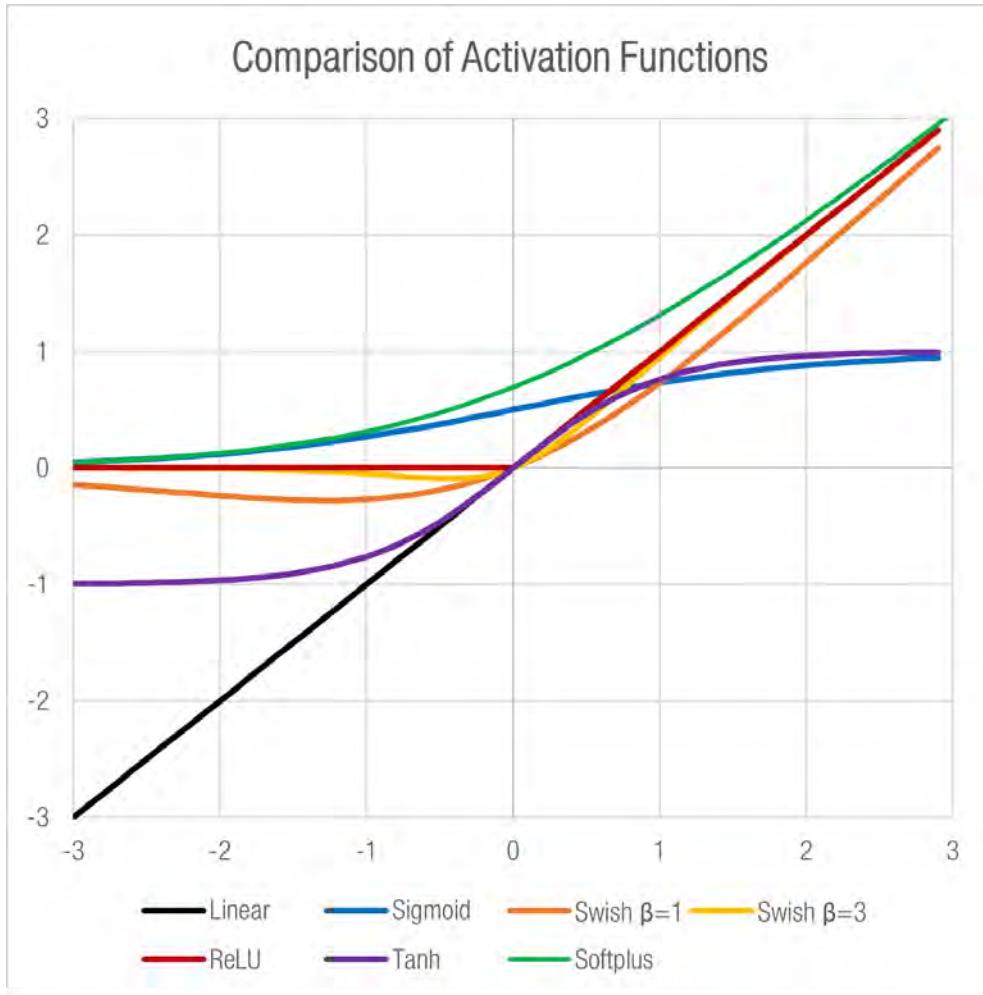


Figure 1-8: Common activation functions are plotted for a neural network on dimensionless axes. The choice of activation function is critical to introduce nonlinearity and can strongly dictate how a model performs.

$$\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} = 2 \operatorname{sigmoid}(2z) - 1; \quad (1.23)$$

$$\operatorname{softplus}(z) = \ln(1 + e^z) = \int_{-\infty}^z \operatorname{sigmoid}(w) dw; \quad (1.24)$$

$$\operatorname{linear}(z) = z. \quad (1.25)$$

The concept for NNs has been around since the 1950s but has only seen a modern resurgence due to increases in computational power, graphical processing units (GPUs), increased dataset size, and improved training algorithms. One of the most significant barriers for NNs was determining how to fit weights and biases at each

layer. This problem was solved by a concept called *gradient descent*. First, the *loss function*, a measure of how closely the model predictions match the target variables is computed [6]; this performance can be calculated through numerous forms and is discussed further in [Section 1.4.2](#). Next, the chain rule is applied to calculate the partial derivatives of the loss function with respect to the weights and biases [75]. The process of calculating the gradients for each layer is called *back-propagation*. Employing this technique allows quick computation of thousands of trainable parameters to mimic patterns observed in the data.

## 1.4 Probabilistic Machine Learning Approaches

The machine learning concepts from [Section 1.3](#) provide the engine behind a probabilistic forecasting tool. However, equally important is the frame and structure needed to harness their abilities. Since this probabilistic ML is still an active research area, two fundamentally different approaches will be tested: Ensemble-based and Bayesian Neural Networks. To identify which approach is preferred, they will be evaluated on gas production forecasts, which will be easier to predict than oil due in part to decreased product volatility and more accurate well measurements. Once the preferred method is selected, it will then be applied to forecast oil and tested on a reduced dataset to determine the methodology's robustness.

### 1.4.1 Ensemble Based Method

The first category of probabilistic forecasts is using an ensemble of predictors. This technique has been researched extensively in Meteorological Sciences, providing numerous best practices. Similar to the Oil and Gas Industry, probability is tremendously important when predicting the weather. For example, customers want to know the likelihood of rain, high and low temperatures, and maximum wind speed. Schaeÿbroeck and Vannitsem, in 2016, published a summary of probabilistic forecasting research using model ensembles [78]. The concept behind this approach is similar to Random Forest from [Section 1.3.1](#). A spread in predictions is generated by utilizing

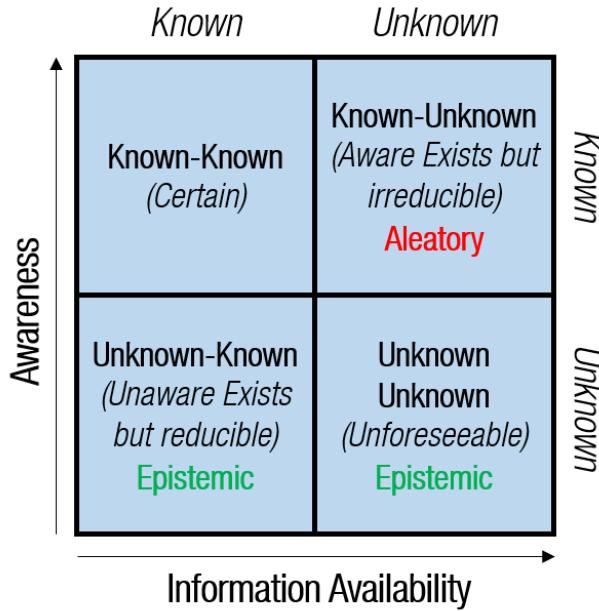


Figure 1-9: Uncertainties Categorized based on awareness and information availability. The ML frameworks must be able to consider **both** Epistemic and Aleatory Uncertainties to generate a representative probabilistic forecast.

a series of models; however, unlike random forest, the spread itself can be analyzed to inform the underlying forecast uncertainty.

There are two primary forms of uncertainty possible, namely *aleatory* and *epistemic*, shown in Figure 1-9. *Aleatory uncertainties* are *known-unknowns*; the analyst knows that they exist, but exact measurements are unknown to the model. Common sources of aleatory uncertainty are when there is inherent randomness in a variable or measurement limitations [25]. A probability distribution can be quantified for the mean and standard deviation of aleatory uncertainties; however, the error cannot be reduced to zero, as there is a fixed standard deviation regardless of the number of measurements. Conversely, *epistemic uncertainties* are unaware to the analyst and are either *unknown-knowns* or *unknown-unknowns*. This type of uncertainty is typically viewed as the error in the model approximating natural phenomena. As more data is acquired, the *unknown-knowns* can be observed and accounted for, reducing the total error [7]. Generally, epistemic uncertainty is more complicated to determine as the model needs to understand what it does not know - a difficult task for humans

and machines alike.

For ensemble-based models, aleatory uncertainties can be considered by providing small perturbations in the initial conditions or parameters, accounting for measurement errors in the input data [78]. The epistemic uncertainty is approximated by having numerous models predict the output, providing several viewpoints and possibilities *unknown* to an *individual* model.

### Challenges with Ensemble Models

Both petroleum and weather forecasting involve making predictions on data that are heavily *heteroscedastic*. Heteroscedasticity occurs when the magnitude of one variable impacts the variance of another [1]. An example of this effect is given in Figure 1-10. This tendency invalidates many statistical tests which assume constant variance. Also, it poses a significant problem in the context of forecasts, as heteroscedasticity, by definition, will likely have a greater spread outside the range used for training.

Additionally, ensemble models are inefficient as a significant number are required to generate a range of predictions. The sheer amount of models poses a data management challenge and could make training infeasible without parallelization. Schaeybroeck demonstrated that a *spread-based* method could mitigate this limitation. Instead of fitting several models, a probability distribution is fit over a batch of outputs to approximate the effect of a larger ensemble. This work suggested that a lognormal distribution is a reasonable assumption for this application [78].

#### 1.4.2 Bayesian Neural Networks

The other category of probabilistic forecasts is Bayesian Neural Networks (BNN). The principle behind this concept is to utilize Bayes' Theorem to fit probabilistic weights in a NN model. Bayes' Theorem is a method of statistical inference to update prior probabilities based on the addition of new evidence [18, 75]. The intuitive benefit of this approach is incorporating prior knowledge into the calculation; this

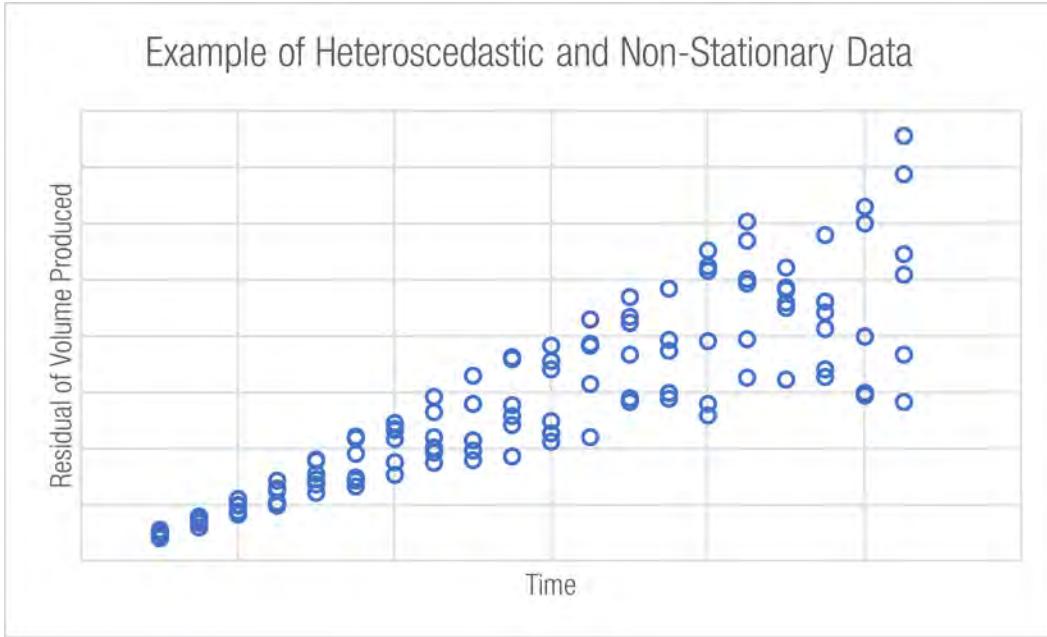


Figure 1-10: Production forecasting is expected to be strongly heteroscedastic due to the cumulative impact of variables interacting over time. This effect creates a significant problem for extrapolation, as the future spread will always be wider than the data used for training.

knowledge avoids overfitting new information and generating overconfident predictions. The BNN method employed in this thesis is based on the *Bayes by Backprop* paper published by Blundell et al. from *Google Deepmind* in 2015 [16]. The application of this technique has been simplified using the *Tensorflow Probability* Python library in 2018, which builds on the *Tensorflow* Neural Network API [51, 56, 73, 77]. For simplicity, only a high-level overview of the principles will be provided, but further details can be obtained from these sources [16, 18].

[Equation 1.26](#) is the adaptation of Bayes' Theorem to predict the model weights ( $w$ ) based on data ( $D$ ), with  $P$  denoting probability [51]. The **prior** term is  $P(w)$ . The **likelihood** is  $P(D|w)$  which is the conditional probability of obtaining the data given the weights. Finally, the **posterior** is  $P(w|D)$  or the conditional probability of the weights after utilizing the training data. The posterior is determined by assuming a distribution for  $P(w)$ , calculating the likelihood  $P(w|D)$ , and applying

[Equation 1.26](#) to get the posterior  $P(w|D)$ :

$$P(w|D) = \frac{P(w) P(D|w)}{P(D)}. \quad (1.26)$$

However, in practice, this calculation is intractable and requires an approximation of the posterior. The method proposed by Blundel is to utilize another function to perform variational learning, and simulate this term [\[16\]](#). A trainable parameter  $\theta$  is added to create a variational posterior  $q(w|\theta)$ . The value of  $\theta$  is solved by minimizing the difference between  $q(w|\theta)$  and  $P(w|D)$  by calculating the Kullback-Leibler Divergence ( $D_{\text{KL}}$ ) using [Equation 1.27](#):

$$D_{\text{KL}}(q||P) = \sum_x q(x) \log \frac{q(x)}{P(x)}. \quad (1.27)$$

A  $D_{\text{KL}}$  value near zero would indicate that the distributions are in close agreement. The last component required is the *loss function*. For regression problems, conventionally, a mean squared error (MSE) equation is utilized. However, the choice of MSE is made to **maximize the likelihood** assuming that the residuals follow a normal distribution [\[51\]](#). This effect can be observed by inspecting the log-likelihood of a normal distribution, shown in [Equation 1.28](#) [\[34\]](#):

$$\sum_{i=1}^N \log p(y_i|x_i; w) = -N \log(\sigma) - \frac{N}{2} \log(2\pi) - \frac{N}{2\sigma^2} \text{MSE}. \quad (1.28)$$

where  $x_i$  are the input,  $w$  is the model parameter,  $\sigma$  the standard deviation,  $-N \log(\sigma) - \frac{N}{2} \log(2\pi)$  is constant, and

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i(x_i; w))^2. \quad (1.29)$$

By utilizing MSE ([Equation 1.29](#)), it is equivalent to solving for the value of  $w$  that provides the highest average likelihood for  $N$  samples [\[34, 47\]](#). This result implies that MSE assumes a normal distribution with a constant  $\sigma$ . Note that [Section 1.4.1](#)

showed a constant  $\sigma$  is a flawed assumption for this problem type due to heteroscedasticity. As a result, when training the BNN, the more general form can be utilized by maximizing the log-likelihood or following convention, minimizing the negative log-likelihood (NLL) of [Equation 1.30](#). Finally, the total loss function is the sum of these two potential errors: the KL Divergence and the NLL. [Equation 1.31](#) is often referred to as the ELBO or Evidence Lower Bound [16]:

$$\text{NLL} = -\frac{1}{N} \sum_{i=1}^N \log L(y_i|\theta); \quad (1.30)$$

$$\text{Loss Function} = D_{\text{KL}} + \text{NLL} = -\text{ELBO}. \quad (1.31)$$

Aleatory and epistemic uncertainties are directly quantified by applying probabilistic layers. A thorough explanation is provided by Sountsov from Tensorflow [69]. In short, the final output node is a probabilistic layer with a trainable mean and standard deviation. These values create a distribution that matches the underlying aleatory uncertainty. For the epistemic uncertainty, a trainable posterior function is used within a Dense Variational hidden layer. This layer generates a probabilistic weight rather than a deterministic value. Random weights are sampled during each run, based on the trained distribution, which mimics the effect of an ensemble with only one model. The advantage of this approach is that it can directly solve the underlying uncertainties present in the data instead of relying on heuristics. This structure is discussed further in [Section 3.4](#) and [Appendix B-2](#).

## Challenges with Bayesian Neural Network Models

The main challenge with this methodology is complexity. The statistical principles used are considerably more specialized than what is typically exposed to an engineer. Without this knowledge, there is the risk of misapplication, which is amplified by introducing roughly double the number of trainable parameters of a standard NN. Tensorflow Probability provides a relatively easy-to-use interface to offset this problem but can be misapplied without experience. Additionally, since the posterior term is an approximation, if there is a significant deviation from the actual value, the error

associated with this technique could be considerable.

### 1.4.3 Summary of Advantages and Disadvantages

The principles learned from the probabilistic machine learning frameworks are given by [Summary 1.4.1](#) and [1.4.2](#):

#### Summary 1.4.1: Ensemble Method Advantages and Disadvantages

##### 1. Advantages:

- Flexible to use in combination with tree based or neural network algorithms. These techniques are much simpler than the Bayesian Neural Network (BNN) at a component level.
- The effect of a small model ensemble can be mitigated through probabilistic fitting; this assumption provides a more approachable workflow.
- Can be understood without specialized statistical knowledge.

##### 2. Disadvantages:

- The probabilistic range is heuristic-based and may not give a true reflection of the aleatory and epistemic uncertainties.
- Vulnerable to heteroscedasticity, as extrapolating will yield variances greater than those observed in training. It's recommended to limit the forecast period based on the longest producing well in the dataset to mitigate this problem.
- This method is inefficient, requiring numerous models, resulting in slow training, forecasting, and complex data handling. Though simple on a component level, managing these interfaces is a complicated process.

## Summary 1.4.2: BNN Method Advantages and Disadvantages

### 1. Advantages:

- Can directly quantify the epistemic and aleatory uncertainties present in the data without relying on heuristics.
- Efficient, as only one model can replace the numerous models required in the ensemble approach.
- Training time is longer than a standard neural network (approximately double) but should be approachable without parallelization.
- Ability to utilize the Tensorflow API provides freedom to specify desired distributions and apply constraints to match expected behaviour.

### 2. Disadvantages:

- Concept utilizes more advanced statistical principles and requires experience to avoid misapplication.
- Requires a neural network with a complex loss function and trainable posterior. More difficult to achieve than the simple ensemble-based algorithms.
- Heteroscedasticity is still a concern with this method and one should avoid imputing volumes beyond the longest producing well in the dataset.

THIS PAGE INTENTIONALLY LEFT BLANK

# Chapter 2

## Data Processing

Datasets were assembled from the Eagle Ford reservoir located in southeast Texas, to test these probabilistic ML theories. This reservoir has numerous features that make it suitable to analyze. First, it was one of the earliest unconventional fields developed in 2010, which provides a substantial quantity of data to analyze as shown in [Figure 2-1](#). With over 22,000 wells available, the effect of data size can be investigated by restricting the number of wells; if predictions from a smaller dataset can approximate the full size, it would indicate the algorithm's robustness. Additionally, the Eagle Ford reservoir is inclined, resulting in variable fluid properties from dry gas to black oil. This feature provides the unique capability of testing both gas and oil predictions, which can inform the method's applicability to other reservoirs.

The dataset utilized was from IHS Markit, an analytics consultant and data provider [\[41\]](#). IHS aggregates data from the Texas Railroad Commission (RRC), the regulatory body for the Eagle Ford. The RRC sets requirements for each operator to report, including production volumes, drilling and completion design, geological data, and permitting. Due to the large quantity of data, this information cannot be easily retrieved in bulk, so IHS provides a more accessible database to purchase. Since a third party maintains these data, the overall quality was higher than unprocessed information, but still requires extensive cleaning before it can be analyzed. This chapter describes the processing and feature engineering performed. All models will utilize the same train and test datasets to eliminate cleaning as a source of variance.

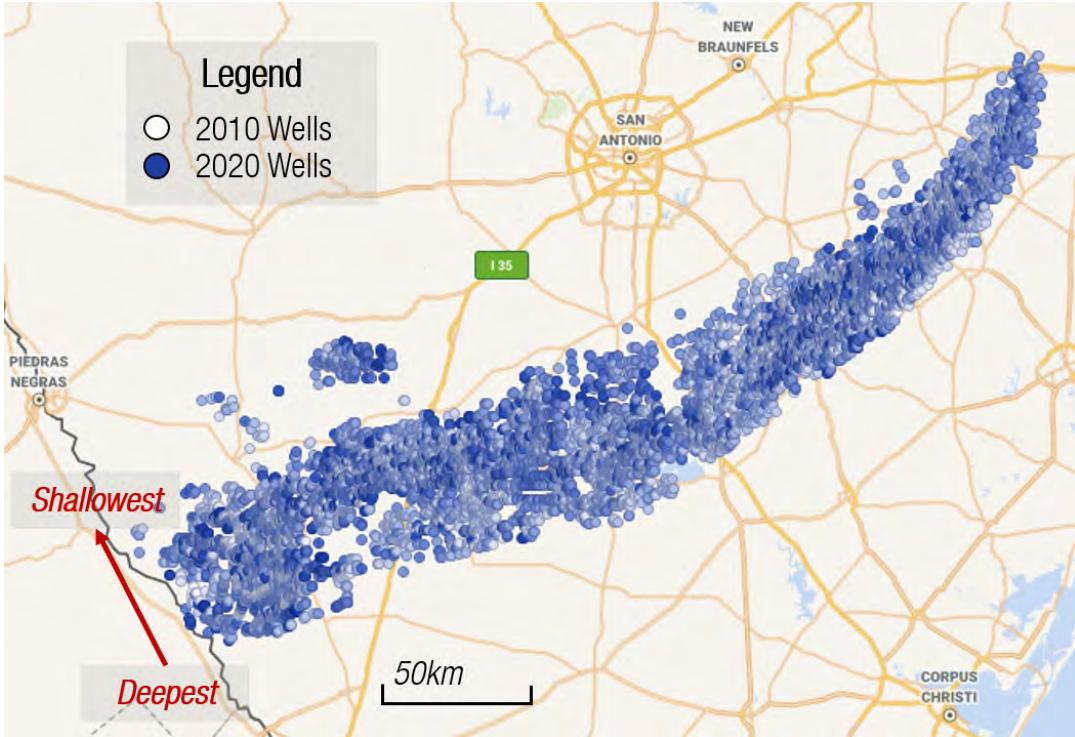


Figure 2-1: The Eagle Ford Reservoir location in South East Texas. Currently there are over 22,000 producing wells drilled from 2010 to 2020 (with drilling year indicated as a color gradient from white to blue), necessitating extensive data processing.

## 2.1 Data Processing

The Eagle Ford dataset contains over 1200 unique features and up to 10 years of production history, creating a total size of approximately 22 million data points. The data consist of two categories: historical production and static well metadata.

### 2.1.1 Historical Production

Historical production contains monthly oil and gas volumes, along with the corresponding production hours. However, this information cannot be used without processing, due to the variation of reliability. The duration when a well produces fluctuates considerably each month, as there are numerous sources of possible downtime, including plant outages, pipeline constraints, maintenance, and equipment failures. Without accounting for reliability differences, there would be significant noise in the data. A simple example is if two wells produced exactly 1000 bbls in the same month,

Table 2.1: Converting calendar volumes into normalized monthly timesteps using interpolation. Highlighted section provides an example for month 4.

The diagram illustrates the conversion process from raw production data to normalized summary data. It consists of two tables. The top table, titled "prod\_data\_processed.head(10)", shows raw production data for 10 wells. The bottom table, titled "prod\_data\_summary.head(6)", shows the same data after conversion, with the third row highlighted. A green arrow points from the top table to the bottom table, labeled "After Conversion".

prod_data_processed.head(10)						
	uwi	Cum_Hours	Cum_Gas_MMSCF	Cum_Oil_MBBL	Gas_MSCFD	Oil_BBLSD
0	42123337170000	46	9.4	0.3	4696.5	168.5
1	42123337170000	312	55.1	1.7	4156.0	124.1
2	42123337170000	1056	161.9	8.5	3444.6	218.5
3	42123337170000	1776	234.5	13.5	2419.5	167.1
4	42123337170000	2520	321.7	18.3	2814.1	153.7
5	42123337170000	3264	401.5	22.6	2573.1	141.6
6	42123337170000	3984	484.5	26.7	2766.3	135.7
7	42123337170000	4728	561.8	31.4	3139.2	151.5
8	42123337170000	5448	657.1	35.4	2510.4	134.2
9	42123337170000	6192	725.8	38.5	2215.1	100.2

prod_data_summary.head(6)					
	uwi	Normalized_Month	Cum_Hours	Cum_Gas_MMSCF	Cum_Oil_MBBL
0	42123337170000	1.0	720.0	113.7	5.4
1	42123337170000	2.0	1440.0	200.6	11.1
2	42123337170000	3.0	2160.0	279.5	15.9
3	42123337170000	4.0	2880.0	360.3	20.4
4	42123337170000	5.0	3600.0	440.2	24.5
5	42123337170000	6.0	4320.0	528.4	28.8

the algorithm would conclude that both wells are equivalent. However, suppose one well flows for only half the month. To match cumulative production, the productivity of that well must be double. Therefore, without separating the effects of downtime from productivity, the algorithm will not generate conclusions correctly. Standardization was performed by computing a running total of cumulative volumes and hours produced by well [72]. These values were interpolated to correspond with precisely 1 month (720 hr) timesteps, as shown in [Table 2.1](#). Additionally, rather than utilizing calendar dates, a **normalized month** variable was introduced, counting the number of months from when the well started producing; this process ensures that each timeslice corresponds with the **same hours on production**.

As discussed in [Section 1.2.3](#), forecasts can be improved by non-dimensionalizing the production volumes, which from [Section 1.2.4](#) are expected to follow a lognormal

distribution. [Equation 2.1](#) combines these effects to generate dimensionless cumulative production  $\hat{Q}_i$  at month  $i$ :

$$\hat{Q}_i = \ln\left(\frac{Q_i}{Q_{12}}\right). \quad (2.1)$$

This equation uses 12-month cumulative volume ( $Q_{12}$ ) as a datum to divide cumulative volume ( $Q_i$ ). The choice of 12 months corresponds with the minimum amount of time necessary to construct a representative decline curve; wells with less than this amount cannot be used for training or testing and were removed. A plot of this process is given in [Figure 2-2](#).

### 2.1.2 Well Metadata

The other primary dataset contains well metadata that comprise four categories: administrative, spatial, geological, and engineering design variables. [Table 2.2](#) summarizes the treatment methods with examples of each category. The biggest challenge was filtering down over 1200 unique variables to the most impactful ones. **Administrative variables** contain legal permit numbers and well statuses. Though important in other contexts, these variables have no bearing on production and can be removed except for the unique well identifier (UWI). The UWI is a unique key given to each well and is utilized for joining the production data with the well metadata.

**Spatial data** consist of the geographical location of the wells and are critical values for mapping. Additionally, these data have increased importance in the context of the **limited geological variables available**. Since the data originate from reported values to the RRC, the amount of data disclosed will **vary by operator**. An operator could have a different legal interpretation of what variables are required for reporting or could be in non-compliance. As a result, each well will have **different levels of available features**. Notably, geological data are considered highly confidential and are not typically disclosed publicly. [Section 1.2.4](#) demonstrated how geological variables have a considerable impact on production, putting pressure to include these data; however, algorithm performance is also heavily dependent on sample size. For

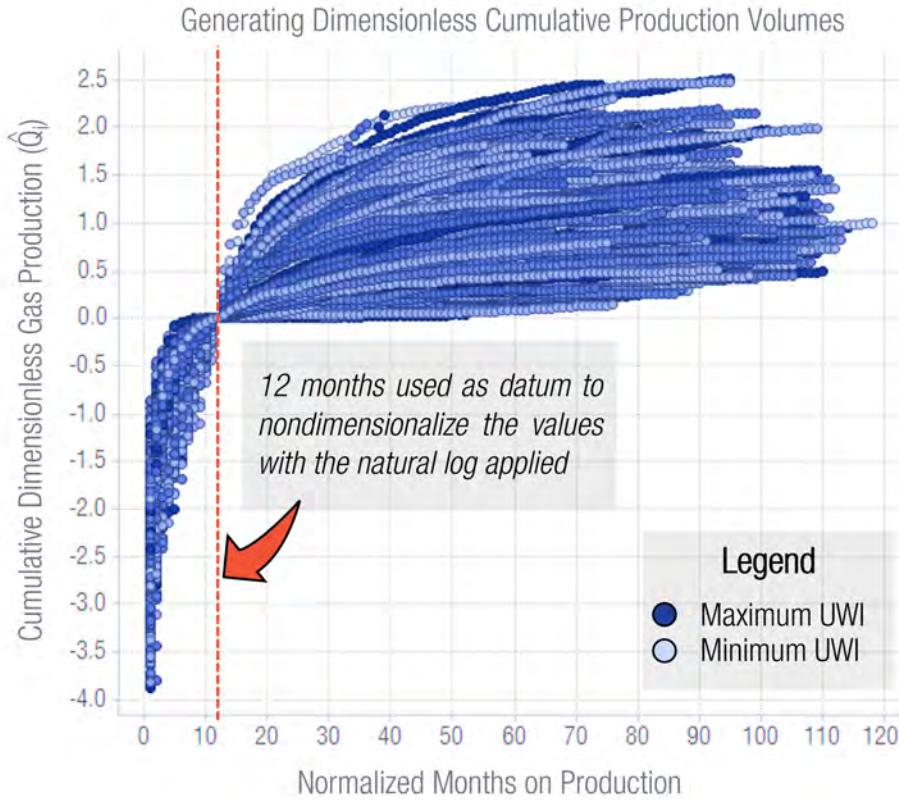


Figure 2-2: Cumulative dimensionless volumes ( $\hat{Q}_i$ ) are plotted by the unique well identifier (UWI) as a color gradient. [Equation 2.1](#) normalizes production using a datum of 12 months ( $Q_{12}$ ) in million standard cubic feet (mmscf). This approach improves predictive power, as the algorithm can identify analogous wells based on their characteristic shape, reducing the system complexity associated with absolute values. Additionally, this process makes the algorithm more generalizable to other regions or reservoirs.

example, of the potential 22,000 wells in the data set, only 5,000 had the variable *clay percentage* populated. Geological features generally cannot be imputed as they are heterogeneous across the reservoir, and standard methods of filling missing values, such as using an average, are too inaccurate. Therefore, if *clay percentage* were included as an input variable, the dataset would be reduced to only the 5,000 wells with this field available. Variables that reduce data set size must be used sparingly, even if expected to be highly impactful. For this analysis, a variable must be populated on 85% of wells or be imputable to be considered for inclusion. This value may seem high but consider that missing features are unlikely to align perfectly; well count will

Table 2.2: Four categories of variables exist for Well Metadata and require the specific treatments shown in red boxes. Examples are provided of available static features.

Administrative	Spatial	Geological	Engineering Design
<ul style="list-style-type: none"> <li>• Unique Well Identifier (UWI, API)</li> <li>• Data Source</li> <li>• Well Status</li> <li>• Status Code</li> <li>• Milestone Dates</li> <li>• Permit Number</li> </ul> <div style="background-color: #e0f2e0; padding: 5px; margin-top: 10px;"> <i>Preserve only UWI</i> </div>	<ul style="list-style-type: none"> <li>• Surface Latitude and Longitude</li> <li>• Bottomhole Latitude and Longitude</li> <li>• Total Vertical Depth</li> </ul> <div style="background-color: #e0f2e0; padding: 5px; margin-top: 10px;"> <i>Use Unsupervised Learning to Cluster wells and blend spatial and geological features to enhance forecasting</i> </div>	<ul style="list-style-type: none"> <li>• Oil Density</li> <li>• Well Log Data (Gamma Ray, nPhi, dPhi)</li> <li>• Target Zone</li> </ul>	<ul style="list-style-type: none"> <li>• Drilling (measured depth, tortuosity, dogleg severity, kickoff point)</li> <li>• Completion (fluid type, proppant tonnage, fluid volume, pump rates/pressures)</li> <li>• Production (choke size, casing size)</li> <li>• Spacing (distance to nearest neighbour, parent/child well)</li> </ul> <div style="background-color: #e0f2e0; padding: 5px; margin-top: 10px;"> <i>Standard Scale and Remove Collinearity</i> </div>

quickly drop, as each one must pass a series of sequential filters.

The last category of well metadata contains the **engineering design** parameters, which are under the direct control of an engineer. Commonly these parameters constitute a basis of design (BOD), which evolves over time to optimize performance or reduce cost. For example, an engineer decides on the lateral length to drill, the amount of proppant to inject, and the perpendicular distance of an infill (child) well to an existing (parent) well. Engineering variables are also critical in dictating the performance of a well and are more **readily reported** than geological values. The two primary data considerations are standard scaling and removing collinearity. **Standard scaling** is a process that fits a Gaussian distribution, forcing the scaled data  $\hat{X}$  to have a mean ( $\mu$ ) of 0 and standard deviation ( $\sigma$ ) of 1 through [Equation 2.2](#) [62, 75]:

$$\hat{X} = \frac{X - \mu}{\sigma}. \quad (2.2)$$

This method is recommended as each feature ( $X$ ) spans a different **range of values**

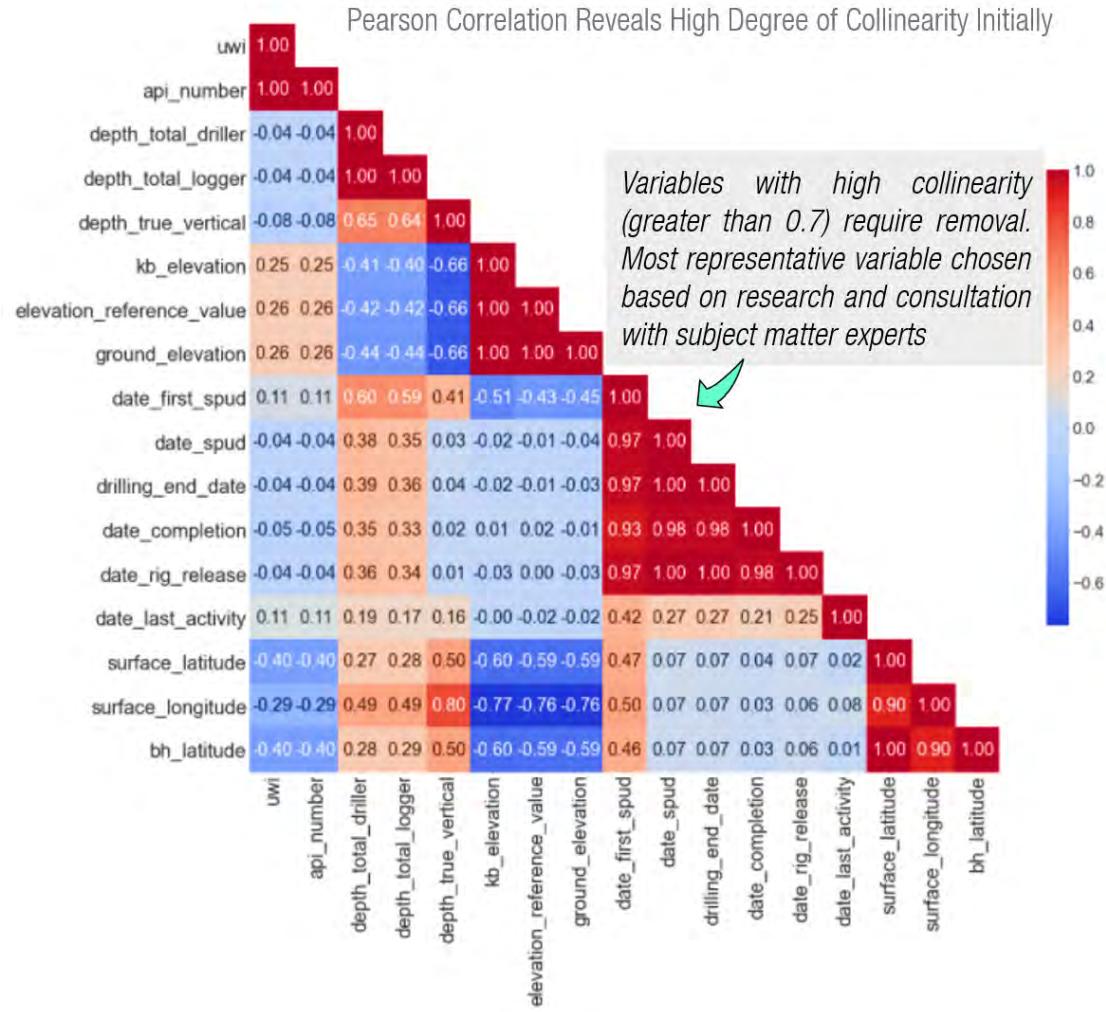


Figure 2-3: Example of initial collinearity before removing variables. High collinearity must be avoided to avoid confounding which variable is impacting production [81]. Suppose one variable had a substantial impact on production, and a second variable was correlated to the first but has limited impact on productivity. The model will decrease the weighting of Variable 1 and spread the impact to Variable 2. When applying this model, the analyst may underestimate the effect Variable 1 plays. Additionally, when forecasting on test data, the model accuracy would be diluted by the increased weighting on Variable 2, which in reality has little impact.

based on the units utilized. For example, proppant injected could be 2000 lbs/ft, while choke size is 18/64 in. With a  $10^3$  magnitude difference, the algorithm may incorrectly weigh the variable importance; this effect was mitigated by standardizing to a common mean and standard deviation.

**Collinearity** is also undesirable, as high amounts (greater than 0.7) can confound

Table 2.3: Example of one-hot encoding to make the values machine-interpretable.

The diagram illustrates the process of one-hot encoding. On the left, a table titled "Categorical Variable" lists four categories: A, B, C, and D. An arrow points from this table to the right, where another table shows the resulting one-hot encoding. This second table has columns labeled "Var\_A", "Var\_B", and "Var\_C". The rows correspond to the categories A, B, C, and D. For category A, the "Var\_A" column has a value of 1 and the others are 0. For category B, "Var\_B" is 1 and the others are 0. For category C, "Var\_C" is 1 and the others are 0. For category D, all three columns are 0.

Categorical Variable	Var_A	Var_B	Var_C
A	1	0	0
B	0	1	0
C	0	0	1
D	0	0	0

which variable is impacting production [11]. For example, if two variables are strongly correlated, then the model will reduce the importance of both, losing the interpretability of the dominant factors. This effect can also cause decreased performance in the test dataset if one variable is more predictive than the other. [Figure 2-3](#) demonstrates the high degree of collinearity present in the data initially using the Seaborn library [81]. Subject matter experts were consulted to select the variable expected to have a **more significant impact** on production. Once chosen, the collinear variables were removed manually from the dataset.

Finally, categorical variables require conversion to be interpreted by the neural network models. One-hot encoding was performed, converting each category into a series of dummy variables with either a 0 or 1. A simple example is presented in [Table 2.3](#). A categorical variable with  $n$  unique elements is transformed into  $n - 1$  variables, yielding three additional columns. When a categorical variable equals  $A$ , it will have a 1 present in the Var\_A column and 0 in all others. This process repeats for the other categories except for the  $n^{\text{th}}$  variable. Adding an  $n^{\text{th}}$  variable is informationally redundant, since a value of  $D$  is equivalent to having all zeroes and does not explicitly require a Var\_D column.

### 2.1.3 Data Cleaning and Validation

The last step of the data processing process was cleaning and validating the results. After applying the methods summarized in [Table 2.2](#), the number of variables dropped from 1200 to 23. Next, manual cleaning was performed to remove outliers from the dataset. Each variable was plotted against cumulative production to identify outliers based on their *distribution* and *plausible variable range*, as shown in [Figure 2-4](#).

The principle of removing the *minimum amount of data possible* was followed to avoid imposing biases. Values that seem off-trend in isolation could be legitimate based on their unique combination of variables; therefore, key insights could be lost by removing these points. However, not removing clear outliers may also lead to error. Since standardization (Equation 2.2) utilizes the sample mean and standard deviation, these values could be skewed by outliers resulting in an unequally distributed feature and introduce error.

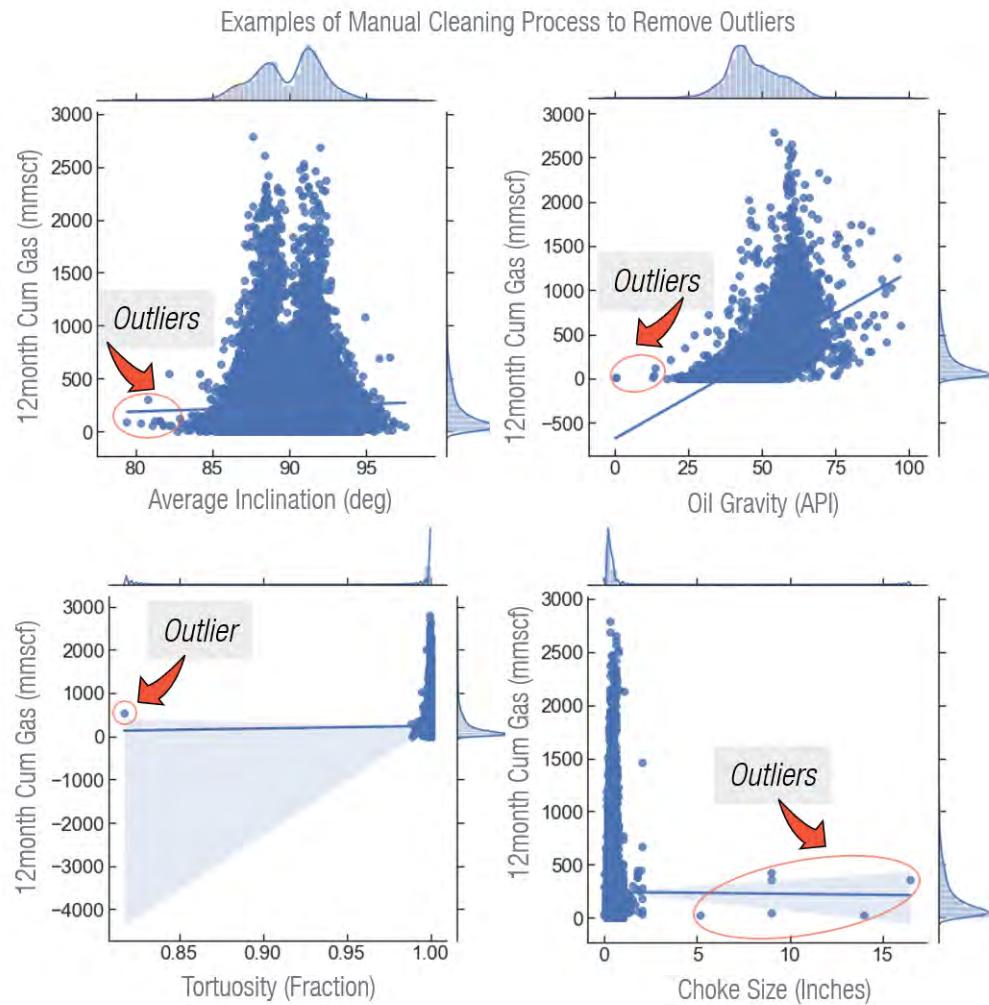


Figure 2-4: Example of the outlier cleaning process of selected input features versus cumulative gas production at 12 months. Obvious outliers must be removed, as they have the potential to skew the dataset when standard scaling. Additionally, the plausible range of variables was considered. For example, oil gravities less than 20 API (bitumen) are unexpected for the Eagle Ford and indicate erroneous data [33].

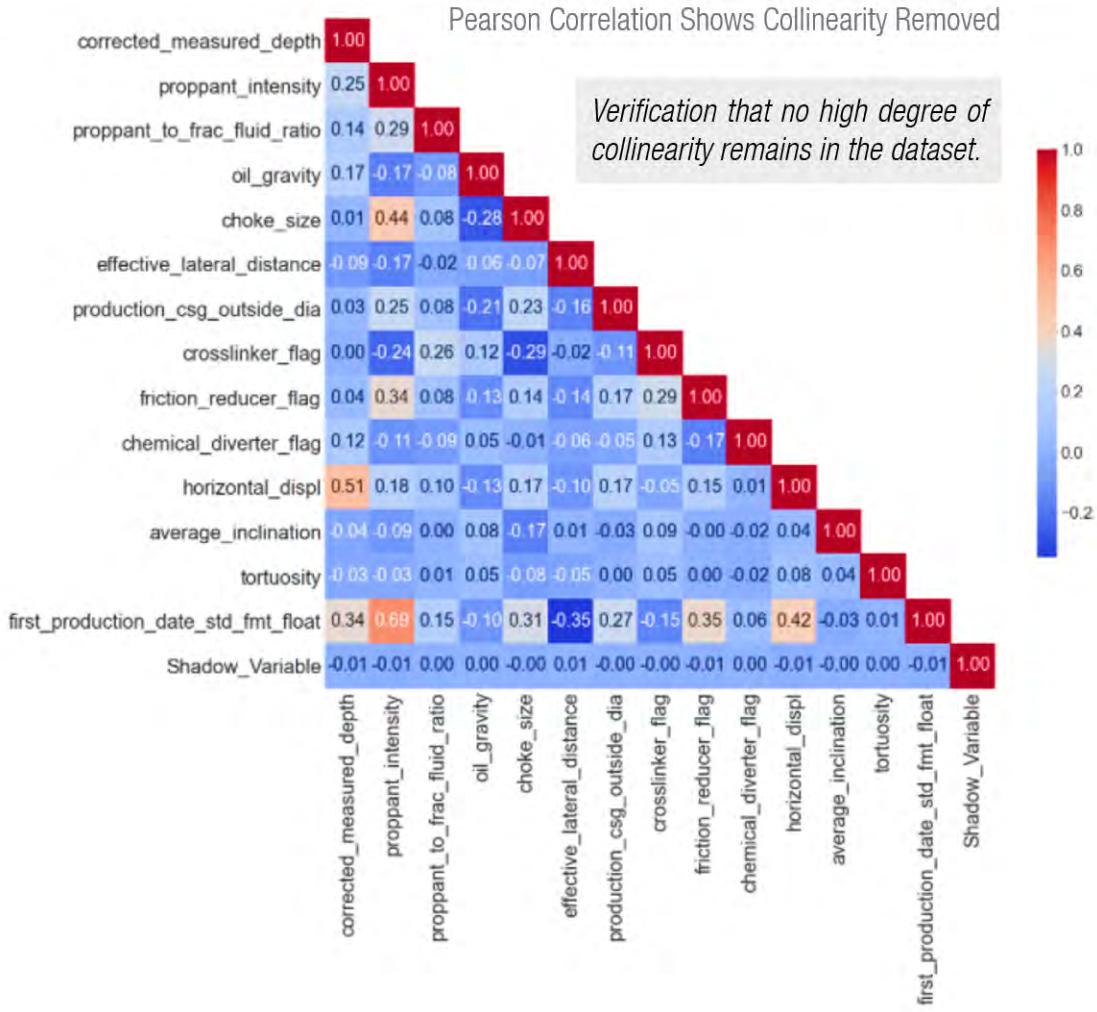


Figure 2-5: Final collinearity after refining variables. None remain with a correlation greater than 0.7, indicating that the variables are sufficiently independent.

The remaining variables were then validated to all fall below 0.7 mutual collinearity. The results of that process are shown in [Figure 2-5](#). These variables have been referred to as **static features** since they are based on initial conditions and are **unchanging within the model**. This terminology does not necessarily reflect the conditions within the reservoir. Variables such as the contributing lateral length or proppant volume can change over time; however, these changes are not directly measured with the data available and were treated as static within the algorithms.

## 2.2 Feature Engineering

This section builds on the static features by generating additional geological and dynamic production lag features. Once generated, all data sources were merged to combine input features and designate a target output variable. These values were then split into train and test datasets to fit the ML models and test performance.

### 2.2.1 Unsupervised Learning of Static Parameters

All previous ML techniques discussed were versions of **supervised learning**, where an algorithm is trained to predict a known target variable. Conversely, **unsupervised learning** is a separate branch of ML that attempts to group data that is **not labelled**. The main goal is to detect similarities between entities and group them into distinct clusters. This technique can be leveraged to mitigate the effects of limited geological data and changing fluid properties across the reservoir. This concept has been shown by Han [36] to improve forecast performance.

Since the reservoir is slanted and immobile in the source rock, depth directly ties to the fluid type. Deeper portions of the reservoir are under higher pressure and temperature, resulting in increased thermal cracking of the organic kerogen and eventually yielding dry gas [71]. Shallower depths are less mature and generate black oil. Since the reservoir's depth gradually changes, there will be a progression of fluid properties from dry gas to wet gas, gas condensate, volatile oil, and black oil. Additionally, flow mechanisms are considerably different between gas and oil, so separating wells based on fluid type is critical to improving model performance (see [Section 1.2.3](#)). Two forms of unsupervised learning were tested: *k*-means and Gaussian Mixture Modelling (GMM).

One of the simplest forms of unsupervised learning is *k*-means clustering. This technique seeks to minimize the Euclidian distance from the center (centroid) of a cluster to the surrounding points [62]. First,  $n$  centroids are assumed by the model. These centroids are randomly dispersed across the dataset and then, through iteration, shifted to minimize the sum squared distance. When no observations are

Scree Plot used to Determine Optimal Cluster Number for *k*-means

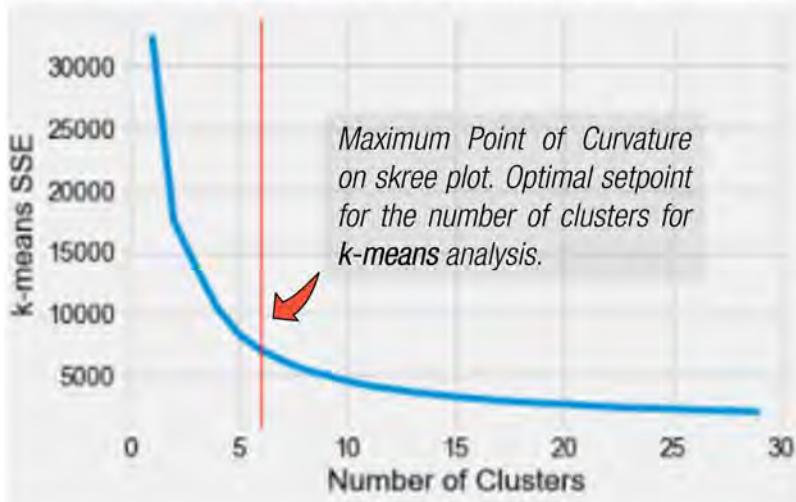


Figure 2-6: Scree plot for *k*-means analysis. Maximum point of curvature occurs with 6 clusters. Graphed using Matplotlib [37].

reassigned to another cluster, the algorithm stops [11]. Since the number of clusters was initially guessed, a best practice is to vary  $n$  and evaluate how an additional cluster improves performance. This process is called a **scree plot analysis** and seeks to find the maximum point of curvature or the **knee of the curve**. *k*-means was applied across multiple dimensions for latitude, longitude, true vertical depth, and oil gravity. The intention was to represent areas that are **spatially contiguous** with **similar fluid properties**. Figure 2-6 was generated from this technique, suggesting 6 clusters are optimal. For **circular-shaped clusters** with equal size and variance, *k*-means is ideal; however, this assumption is poor for the Eagle Ford based on the fluid properties. Condensate gas ratio (CGR)<sup>1</sup> is often used as a proxy for fluid type, with thresholds marking transitions between fluids shown in Table 2.2.1. Figure 2-7 demonstrates the CGR trends spatially. The critical observation is that CGR exhibits parallel bands spanning long distances; therefore, this phenomenon is **better represented by elliptical clusters**.

<sup>1</sup>The terms condensate and oil are used interchangeably in this analysis to refer to C<sub>5</sub><sup>+</sup> liquid hydrocarbon chains. In actuality, they have different definitions based on density and the location where they are formed (oil in the reservoir and condensate in the wellbore).

Table 2.4: Summary of utilizing CGR as a proxy for fluid type adopted from [71]

CGR Cutoff (bbls/mmscf)	Dry Gas	Wet Gas	Gas Condensate	Volatile Oil	Black Oil
Lower Bound	-	10	67	333	400
Upper Bound	10	67	333	400	10,000

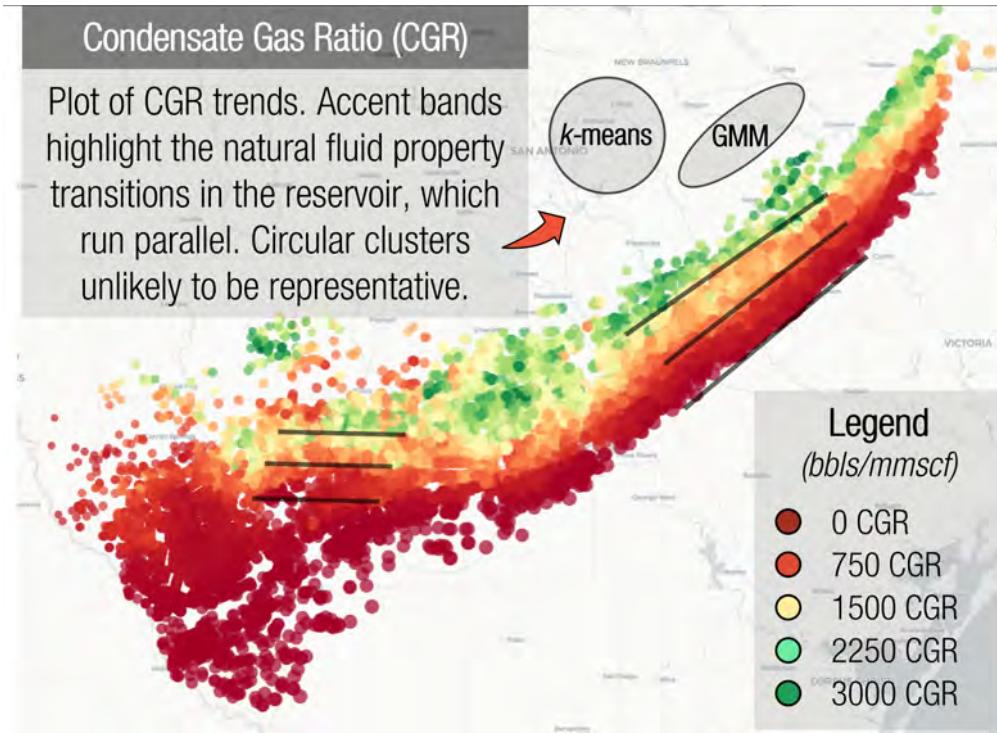


Figure 2-7: Plot of Year 1 Average CGR. Distinct bands occur over large distances, which make clusters more likely to be elliptically shaped than circular. GMM should theoretically have greater performance on this dataset based on cluster shape.

**Gaussian Mixture Modelling (GMM)** is far better suited to handle this type of condition by assuming data points are represented by a series of  $n$  Gaussian distributions [62]. Similar to Section 1.4.2, the maximum likelihood was solved by fitting the distribution shape with trainable covariance matrices, allowing for the formation of ellipses.  $k$ -means is actually a simplified form of GMM, where the covariance matrix is restricted to be diagonal and identical [11]. Finally, to solve for the optimal number of clusters, the Bayesian Information Criterion (BIC) was calculated [79]. BIC predicts the probability of explaining the underlying distribution while penalizing for model complexity. A plot of the results for GMM is shown in Figure 2-8.

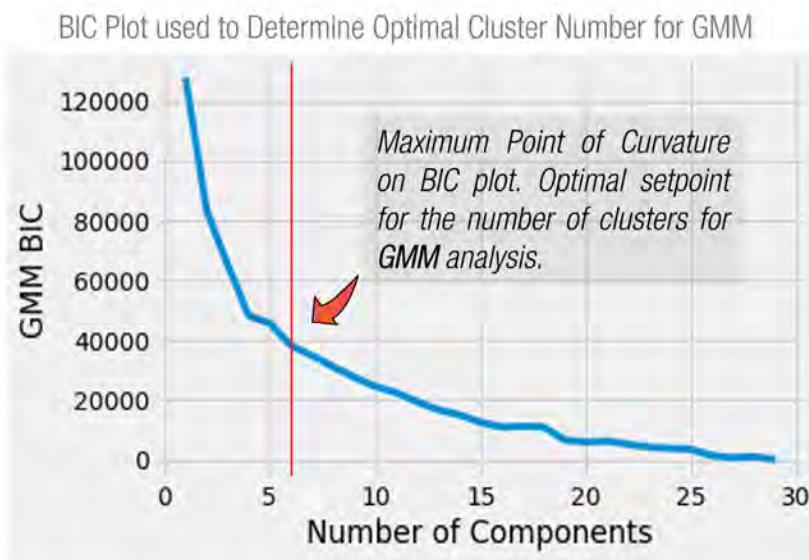


Figure 2-8: BIC Plot for GMM analysis. Maximum point of curvature occurs with 6 clusters.

After clustering wells using  $k$ -means and GMM, the results were inspected to determine the preferred methodology. Figure 2-9 investigates the cluster boundaries in the same geographical region. As expected from theory, the  $k$ -means clustering is **less successful** at capturing the unsymmetrical distribution of fluid properties; this effect was evident when evaluating the cluster assignments near boundaries, as there was significant overlap. The ideal technique should have **minimal overlap** as the clusters are intended to represent distinct sections of similar fluid types and geology. The geological similarity was assumed based on proximity, but the quality of analysis would be enhanced with more geological data or geoscientist interpretations. As a result, **GMM is the recommended clustering method** and was added as a feature to the dataset. The input variables used to derive the clusters were dropped to avoid collinearity.

### 2.2.2 Dynamic Lag Features and Target Variable

The last components needed before performing the analysis were the dynamic production lags and the target output variable. All of these parameters originate

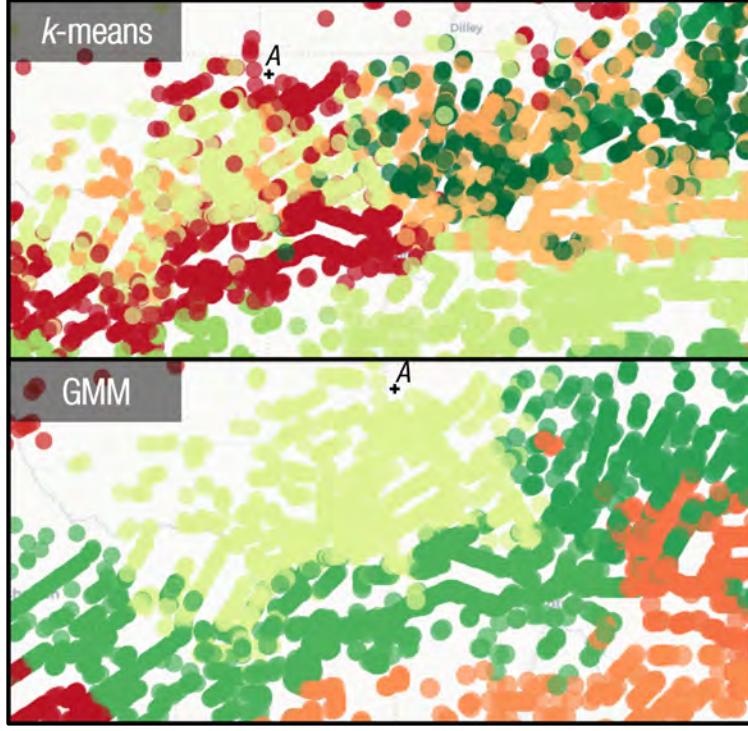


Figure 2-9: Comparison of  $k$ -means and GMM results with each color corresponding to an individual cluster. Looking the same geographical region (A) illustrates indistinct cluster boundaries for  $k$ -means; this effect is non-ideal and supports the implementation of GMM.

from the historical production data discussed in [Section 2.1.1](#). Since the analysis aims to predict future production, careful attention must be made to avoid revealing information that the algorithm should not access (data leakage). Backcasting is a common process for time series problems to avoid this effect and has been advocated by Freeborn and Russell as being a best practice [64]. Rather than splitting data randomly, the start of the forecasting period is defined, which separates data between values known and unknown to the model. [Figure 2-10](#) demonstrates this concept visually. The historical production before the start of the forecast can be added as **dynamic lag variables** into the input features (see [Table 2.5](#)). Conversely, the production after the forecast start constitutes the **target output variables**. The lag variables are referred to as **dynamic** as they must be updated throughout the forecasting process based on the desired prediction month; this process is further discussed in Chapter 3.

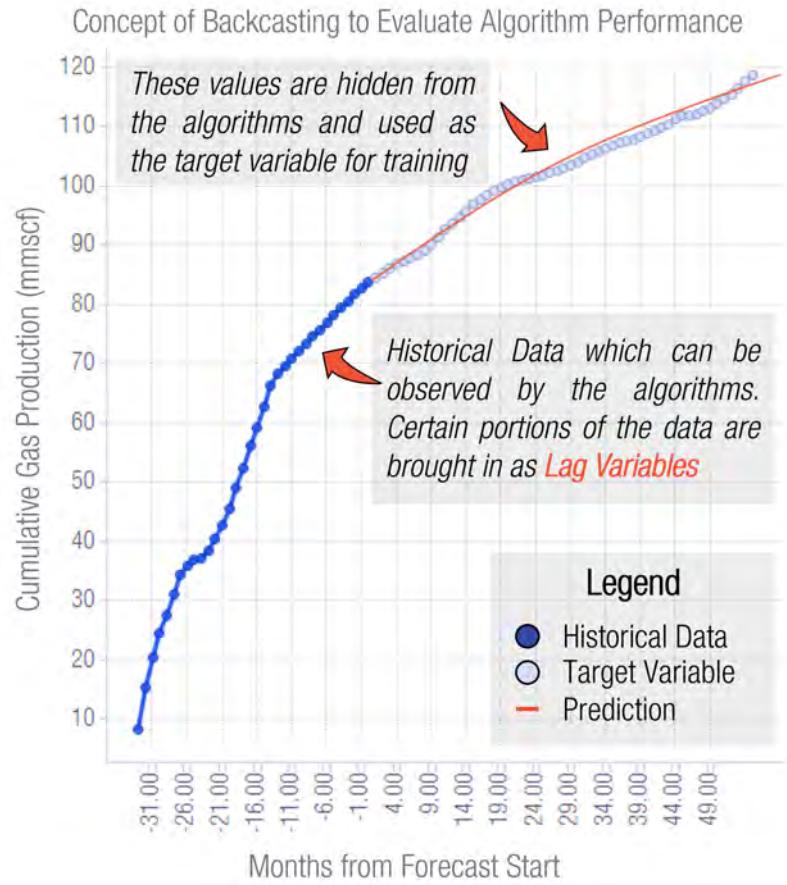


Figure 2-10: Concept behind the backcasting methodology. Algorithms will be trained based on historic data to predict future volumes (target variable). The months prior to the forecast start are used to populate dynamic lag variables.

Table 2.5: Example of Generating Lag and Target Production Features based on months before forecast start ( $t_{\text{forecast}}$ ).

uwi	Normalized_Month	$t_{\text{forecast}}$	Norm_Cum_Gas	Target_Variable
42123337170000	33	-8	0.483	
42123337170000	34	-7	0.491	
42123337170000	35	-6	0.498	
42123337170000	36	-5	0.506	
42123337170000	37	-4	0.512	
42123337170000	38	-3	0.525	
42123337170000	39	-2	0.533	0.552
42123337170000	40	-1	0.543	
42123337170000	41	0	0.552	

Lag_Month_6	Lag_Month_5	Lag_Month_4	Lag_Month_3	Lag_Month_2	Lag_Month_1
0.498	0.506	0.512	0.525	0.533	0.543

The choice of the start of the forecasting period has a significant impact on model performance. Wells with more production history should be easier to predict due to additional data points and a lower degree of change in cumulative production. The forecast starting point was randomized for each well with [Equation 2.3](#), ensuring that the algorithms are applicable for varying periods:

$$\text{Forecast Start} = \text{Ceiling}((t_{\max} - 12) \cdot X) + 12. \quad (2.3)$$

A random value ( $X$ ) was generated for each well following a uniform distribution between 0.1 and 0.9. This value multiplies the last production month,  $t_{\max}$ , to get the months from year 1. Since at least 12 months of historical production was required, this equation ensures the earliest the forecast can start was on month 13 and the maximum being  $t_{\max} - 1$ . The lag inputs were then merged with the static data using UWI as the identifier.

Finally, the data was split into **train and test wells**. 70% of wells were randomly assigned into the training dataset and the remaining into the test. These designations were held constant for the remainder of the analysis to guarantee that no data was leaked from a previous step. [Figure 2-11](#) provides an example of this process, with each color representing which data segment was used for inputs, fitting, and testing model performance. All dataframes are stored in Github Large File Storage (LFS) as pickle files to be retrievable without recalculation [[77](#)]. Pickle files are immutable objects loaded from storage, ensuring that all algorithms consistently reference the same train/test wells and input data.

### 2.2.3 Summary of Input and Output Variables

[Table 2.6](#) provides a summary of the processing and feature engineering performed in this chapter, along with the inputs ( $X$ ) and outputs ( $Y$ ) used for analysis in the remainder of this thesis. The result of this process was a decrease from 22,000 to 13,048 wells and 12,000 to 23 static features.

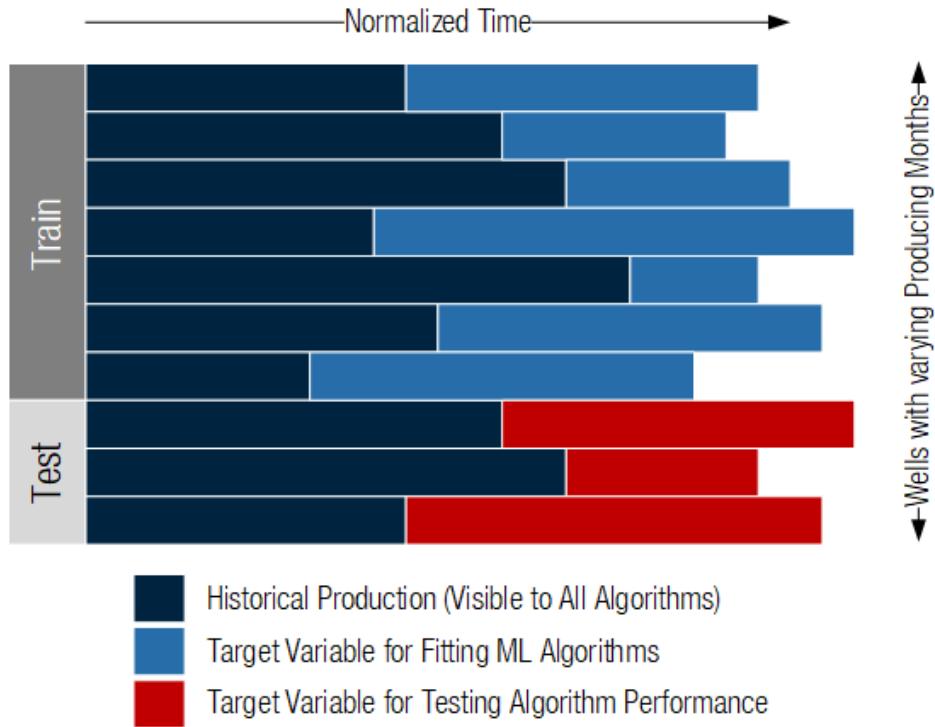


Figure 2-11: 70/30 split was used to designate wells into train and test bins. Bar length represents the number of months. Note how each well has varying historical and forecast months based on [Equation 2.3](#), ensuring applicability in different conditions.

Table 2.6: Summary of the treatment methods for each variable class after processing and feature engineering.

Static Data			Dynamic Data		Target Variable
Admin	Engineering	Geospatial	Gas Lags	Oil Lags	Forecast Start
Keep UWI	Remove Collinearity	GMM Unsupervised Clustering	Calculate Normalized Gas and Oil Values based on 12m cumulative volume		Randomly Generate the Forecast Start Month using a uniform distribution
Merge gas and oil lags	Standard Scale and One-Hot Encode	Drop Input Variables	Generate gas and oil lag variables n months prior to Forecast Start		Timing be between Month13 and $t_{max} - 1$

← Input Features ( $X$ ) → ← Output ( $Y$ ) →

70/30 Test Train Split made by well. Designations are preserved for the entire process to prevent data leakage

# Chapter 3

## Analysis Methodology

A probabilistic forecast methodology can now be built from the theory established in Chapter 1 and the processed data from Chapter 2. A summary of the entire analysis pipeline is demonstrated in Figure 3-1. The Existing and Type Well forecasting methods will be discussed, highlighting the analysis approach, key assumptions, and adjustments to established theory.

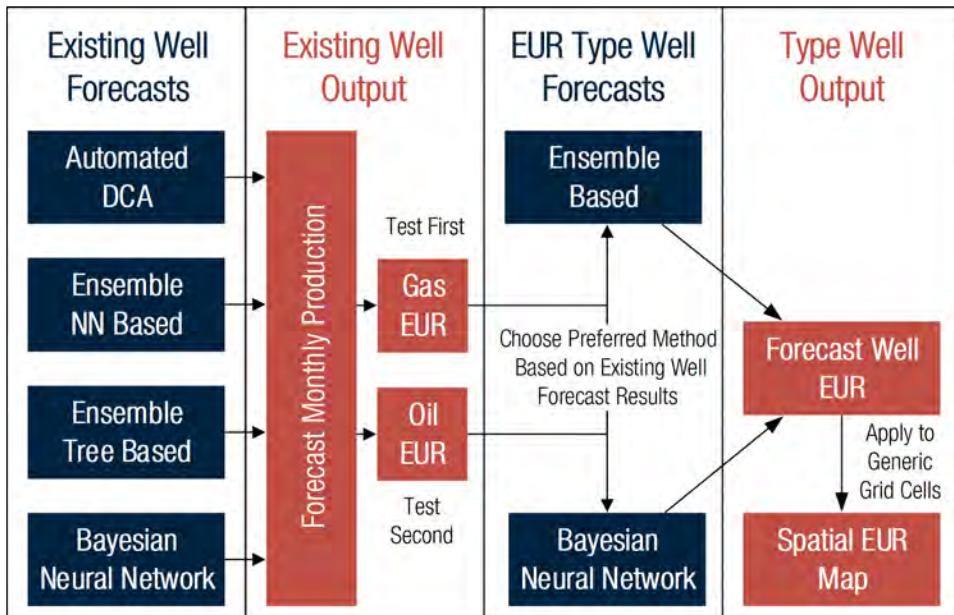


Figure 3-1: Diagram of the full analysis workflow. First, four Existing Well Forecasting algorithms will be tested. The output of that process constitutes the target variable (EUR) for Type Well Forecasting.

### 3.1 Evaluation Criteria and Fitting Process

First, a standard set of metrics were required to evaluate model performance. Equations 3.1 to 3.2 summarize commonly used methods, where the actual value is  $y_i$ , predicted is  $\hat{y}_i$ , and the number of data points is  $n$ :

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n}}; \quad (3.1)$$

$$\text{Loss Function} = D_{\text{KL}} + \text{NLL} = -\text{ELBO}. \quad (3.2)$$

As discussed in Section 1.4.2, the loss function selected should maximize the likelihood of the underlying distribution. RMSE will be utilized for the Automated DCA and Ensemble Methods for simplicity, while the Bayesian Neural Network will minimize the  $-\text{ELBO}$  [51].

Additionally, the coefficient of determination ( $R^2$ ) and out-of-sample  $R^2$  ( $\text{OSR}^2$ ) provide convenient methods to evaluate model performance, given by Equations 3.3 and 3.4 [11]:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}; \quad (3.3)$$

$$\text{OSR}^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y}_{\text{train}})^2}. \quad (3.4)$$

The choice of applying  $R^2$  and  $\text{OSR}^2$  is based on the dataset type.  $R^2$  is utilized for training data, while  $\text{OSR}^2$  is performed on unseen values in the test. Note that these formulas have the same form, except for  $\text{OSR}^2$  using the mean value of the training dataset ( $\bar{y}_{\text{train}}$ ), rather than the mean of the test. Intuitively, this metric determines how predictive the model is compared to a naive approach of predicting values based on the average of the data available [11]; since the test dataset is obscured from view, this metric must utilize the average from the training data.  $\text{OSR}^2$  is regarded as the more crucial metric since it removes the effect of overtraining or the risk that an algorithm memorizes the data rather than finding generalizable patterns. These equations provide a score typically between 0 and 1, though a negative score is possible

Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5

Used for Training the Model
Validation – Compute Errors

Figure 3-2: Visualization of the cross-validation process. Application prevents over-training, overusage of test data, and improved selection of hyperparameters.

if the prediction is worse than using the training dataset mean ( $\bar{y}_{\text{train}}$ ). Since the score is dimensionless, it provides a simple basis of comparison, with a model closer to 1 being desirable. However, this value is often misleading, varying based on the input range of the data points, is not applicable after data transformations, and is not an accurate reflection of goodness of fit [21]. For these reasons, OSR<sup>2</sup> is included only as a secondary metric.

Finally,  $k$ -folds cross-validation was performed to optimize hyperparameters and improve model accuracy. Cross-validation splits the dataset into  $k$  folds or partitions. For every iteration, one fold was held as a validation set, where the RMSE or  $-\text{ELBO}$  was calculated. This process was repeated  $k$  times until all folds have been used for validation, as depicted by Figure 3-2. Performing cross-validation avoids overtraining, as performance can be gauged on the validation set without continually referencing the test data [75]. Over-reliance on test performance is not recommended, as it can lead to these data being effectively part of the training dataset and are not valid indicators of unseen values. Cross-validation is also commonly used to find the best combination of hyperparameters or adjustable settings within an algorithm [60]. Note that a 5-fold cross-validation process was performed for each algorithm. A summary of the parameters tested and selected is given in Appendix A.1.

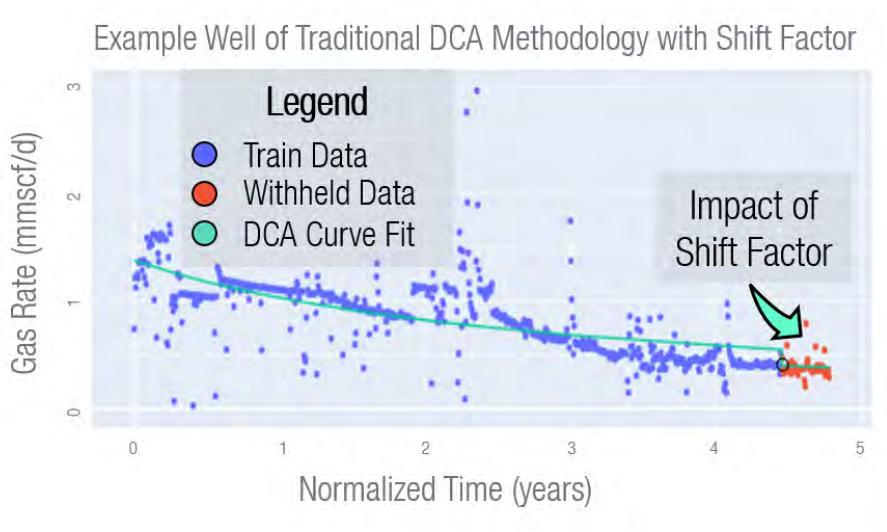


Figure 3-3: DCA automated with Scipy Curve Fit Optimizer [80]. The  $q_{adj}$  factor allows the forecast to respond to near-term changes. Production step changes can occur due to operational setpoint adjustments, facility constraints, surface choke alterations, or pressure effects from pipeline hydraulics. An analyst will commonly fit a curve through historical data and then shift the trend to align with the new setpoints.

## 3.2 Automated Decline Curve Analysis (DCA)

Before evaluating the performance of the probabilistic machine learning methodologies, a baseline was necessary to gauge success. From Section 1.2, the most common industry practice is to utilize DCA with a modified hyperbolic equation. Therefore, DCA was selected as the benchmark and was automated to apply at scale to 13,048 wells. To solve for the  $q_i$ ,  $b$ , and  $d_i$  parameters, the **scipy curve fit optimizer library** was utilized [80]. This optimizer minimizes RMSE by finding the best combination of parameters for Equations 1.1 and 1.2. Acceptable parameter bounds were imposed from recommended values in SPEE Monograph 3, speeding convergence [12]. Note that each section will store the results as a pickle file or Tensorflow HDF5 object [77], allowing trained models to be easily retrieved and used for forecasting.

The DCA methodology was largely left unchanged, providing a fair comparison against standard processes. However, one modification was employed to ensure that the forecast aligned with the most recent production data. An adjustment factor

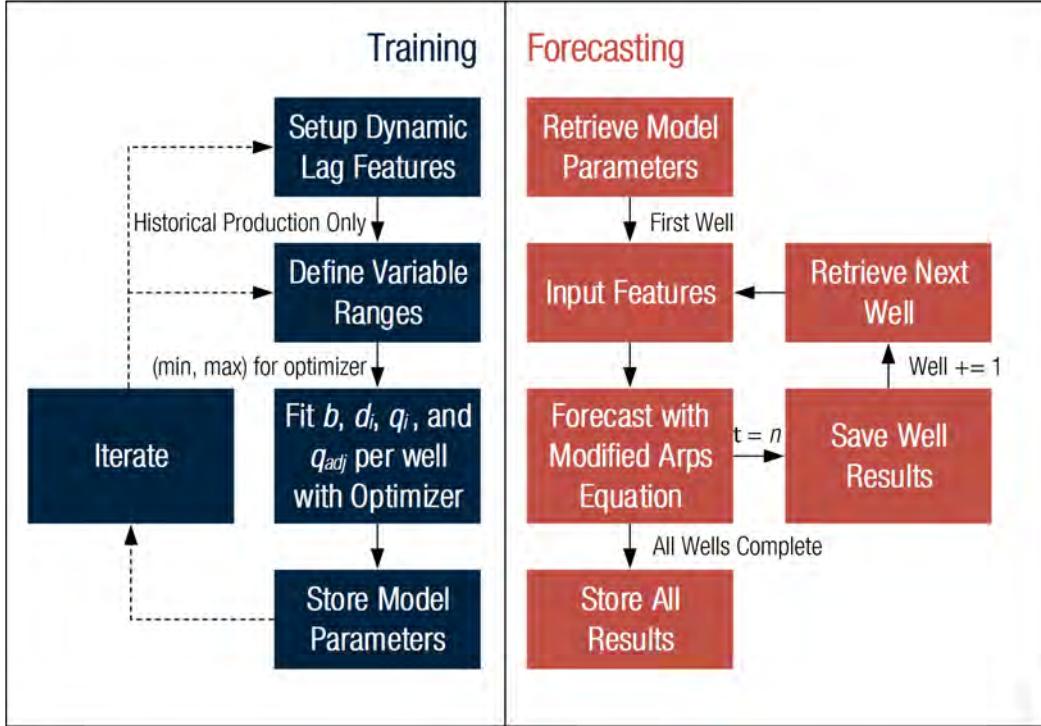


Figure 3-4: Workflow diagram of Automated DCA. First, training occurs by fitting the best combination of parameters. These parameters were then retrieved and used to forecast the test dataset, using backfitting to gauge performance.

$(q_{adj})$  was calculated using [Equation 3.5](#) and demonstrated in [Figure 3-3](#):

$$q_{adj} = \frac{\hat{q}_n}{q_n} q_i. \quad (3.5)$$

After fitting the optimal parameters, the algorithm calculates the ratio of average predicted rate ( $\hat{q}_n$ ) to actual production ( $q_n$ ) over the last two months before the forecast start. This method is an industry best practice to align the forecast with current trends. The  $q_{adj}$  value replaces the  $q_i$  term in the Arps' equations.

[Figure 3-4](#) describes the automated DCA algorithm in more detail. Notably, unlike the machine learning algorithms, there is no difference between train and test datasets because there is no explicit mechanism to transfer knowledge between wells (see [Summary 1.2.2](#)). The curve fitting process was completed on **historical production only** to provide the parameters for  $b$ ,  $d_i$ ,  $q_i$ , and  $q_{adj}$ .

### Workflow to Generate Probabilistic Forecast with Ensembles

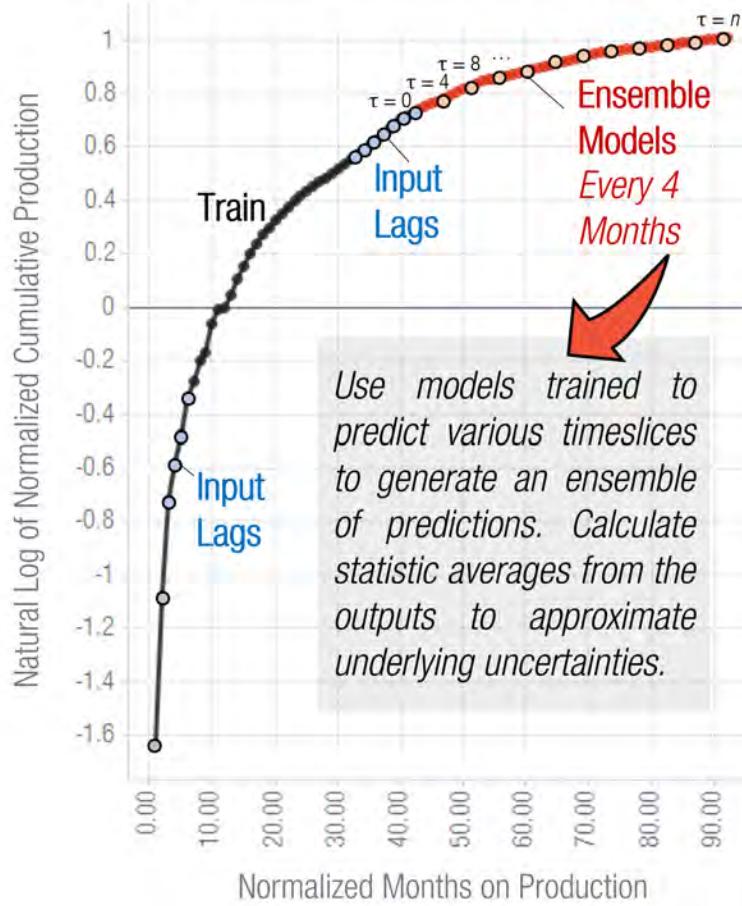


Figure 3-5: Ensemble Method fits models at  $\tau = 4X$  increments until  $n = 60$ , where  $X \in \mathbb{N}$ . Input lags were created from the first and last 6 months. Timeslices were necessary to increase variability. If only one  $\tau$  is forecasted, the spread in the predictions would be unrealistically low and generate overconfident uncertainties.

### 3.3 Ensemble Models

The first class of probabilistic machine learning is ensemble-based models. The concept behind this approach is to fit numerous models to predict a **specific forecast month** ( $\tau$ ), as illustrated in [Figure 3-5](#). In this example, a model was trained to predict 1 month ahead ( $\tau = 0$ ), with a separate model predicting month 5 ( $\tau = 4$ ); this pattern continues until  $\tau = 60$ , creating a total of 15 models. Additionally, the **algorithm type** was also varied for each timeslice. Ensemble-based models are flexible in terms of the algorithm chosen and gain their strength by increasing the

diversity of viewpoints. As a result, **three tree-based methods**, XGBoost [17], Random Forest [62], and LightGBM [49], were fit for each  $\tau$  period, increasing the total number of models to 45. Similarly, an ensemble of Neural Networks was created using Tensorflow [56]. This structure consisted of **three models for each period**, which vary based on the number of nodes and hidden layers. The inclusion of neural networks within the ensemble provides a common basis to compare against the BNN.

**Twelve dynamic production lags** were chosen for input data from the *first* and *last* six months of production before the forecasting period. This number is the maximum allowable since the earliest forecast starting point is month 13, and all features must be populated. The choice of splitting lags between the beginning and end allowed for the effects of early operating strategies to be considered. As described by Clarkson [19] and [Summary 1.2.3](#), operational decisions such as the differential pressure between the wellbore and reservoir (known as drawdown) impacts pressure-dependent properties of the reservoir and flow regime transitions. Consequently, including these starting lags can manifest impacts on long-term predictions.

The next step was generating the forecast. The trained models were retrieved from storage and the input features configured. A **for-loop** creates a **batch of predictions** for each  $\tau$ . A *spread-based ensembles* approach was followed, fitting a distribution on the outputs of the prediction month ( $t$ ); recall that Schaeybroeck described this approach as a best practice to simulate the effect of a larger ensemble [78]. Based on the literature review, cumulative volumes follow a lognormal distribution (see [Summary 1.2.5](#)); however, since the natural logarithm was applied during the standardization process, a normal distribution was selected as it creates an equivalent lognormal distribution in the original units. This assumption was independently validated in [Appendix B-3](#). Next, the production lags were dynamically adjusted for each forecast iteration. The normal distribution fit at time  $t$  was used to *commit* the predicted volume for the most recent lag, overwriting this value and shifting all others by one. This process sets up the next forecasting round, targeting  $t + 1$ . Predictions outside of  $t$  were incorporated in the distribution fitting process once  $t$  was increased to that specific month. This process was repeated to forecast the desired range.

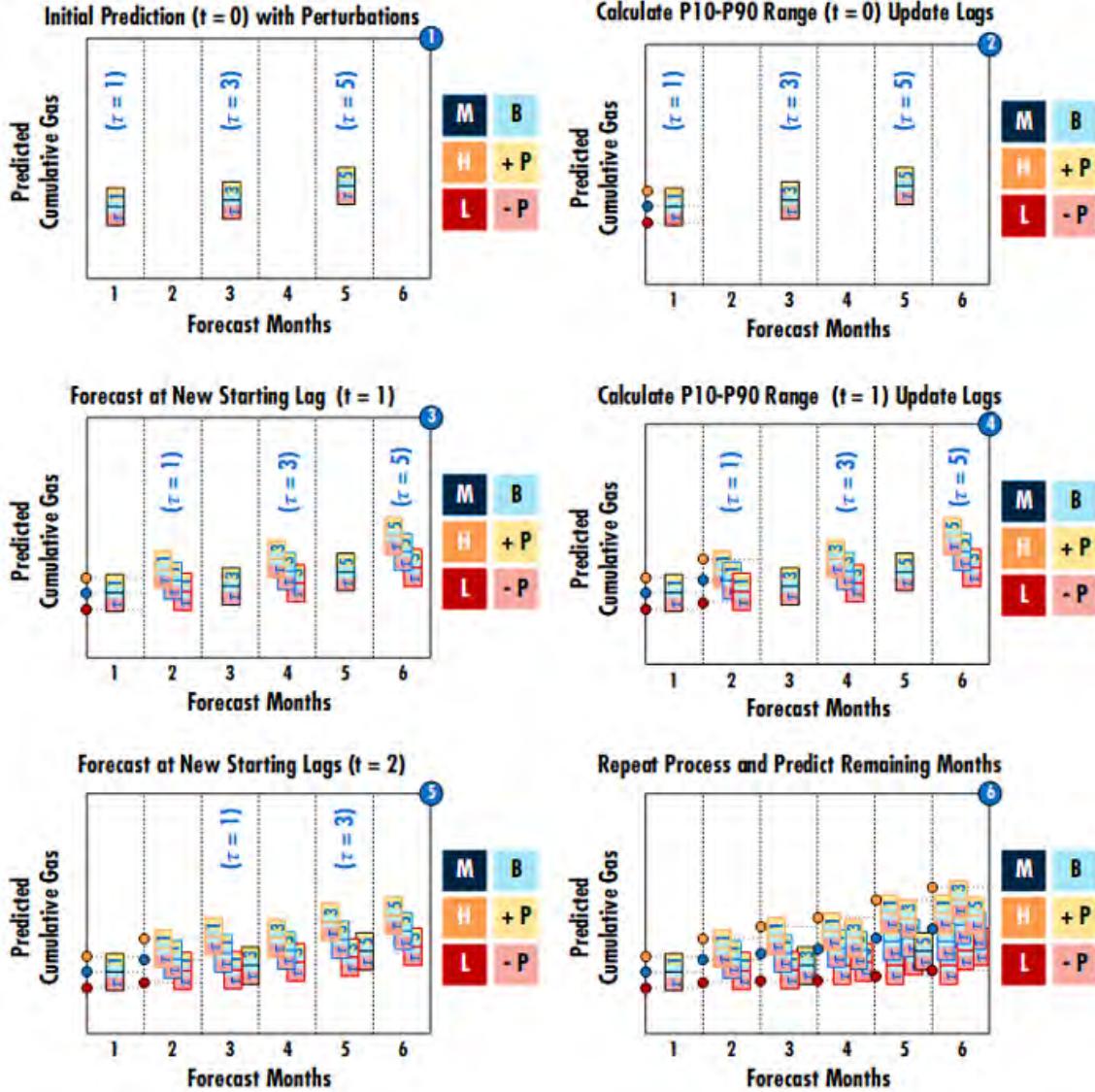


Figure 3-6: Random perturbations were applied to simulate uncertainty on the input parameters. Each run alternates between base static inputs ( $B$ ), and a random set of  $\pm 5\%$  perturbations ( $+P$ ), followed by the inverse perturbation ( $-P$ ). For simplicity,  $+P$  in the diagram is depicted as having an arbitrary positive effect, but the magnitude and directionality will vary in practice depending on the combination. After a round of iterations, a normal distribution was fit, committing the values at time  $t$  and generating mid ( $M$ ), high ( $H$ ), and low ( $L$ ) base volumes for the next iteration. The **boxes** represent the **three predictions** ( $B$ ,  $+P$ ,  $-P$ ) made by each algorithm for every  $\tau$  and base volume. Note spread increases over time, matching the expected non-stationary behaviour.

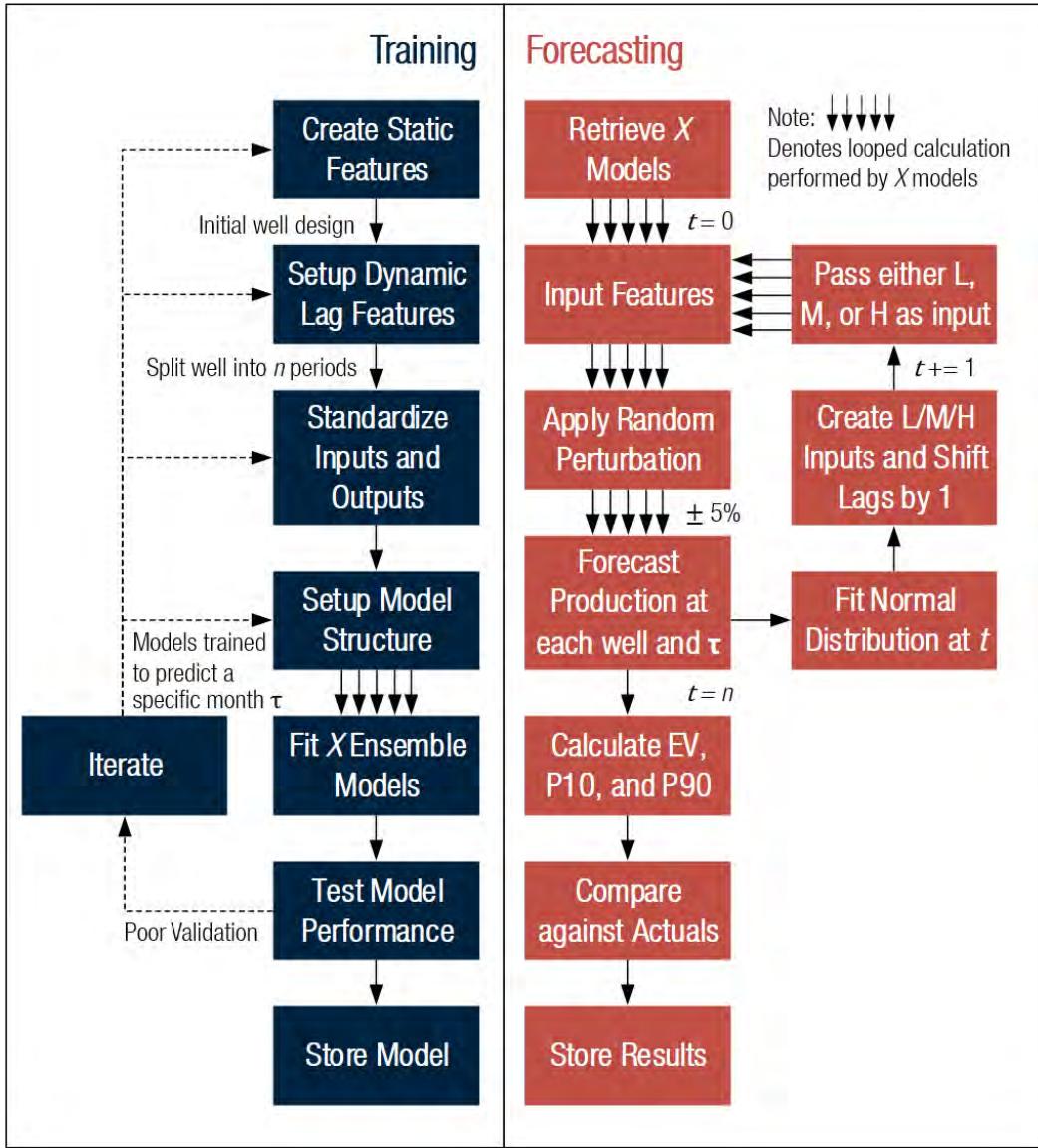


Figure 3-7: Workflow diagram of Ensemble Models. Compared to other methods, the strength of the workflow lies in the relative simplicity of training tree-based and feed-forward neural network models. However, this simplicity is lost when forecasting the probabilistic range. The method relies on trained models at various  $\tau$ , adding random perturbations to the input data, and three base volumes to create sufficient variation to approximate uncertainty. This approach introduces the risk that the uncertainty range does not appropriately reflect reality, by varying the level of noise.

Finally, to increase the spread in potential outcomes, two additional modifications were made, as shown in [Figure 3-6](#). First, the distribution at time  $t$  was used to calculate mid (P50), low (P5), and high (P95) estimates; these values constitute new base starting volumes for each round. A random perturbation was also applied to the static well inputs shifting values by  $\pm 5\%$ ; this multiplier emulates the effects of aleatory uncertainty by generating measurement variances for the input values. A summary of the full training and forecasting process is given by [Figure 3-7](#).

## 3.4 Bayesian Neural Network

Though the ensemble-based method is simple on a component level, this strategy has clear weaknesses. First, there is an inherent tension between forecast accuracy and generating a probabilistic range. Increasing the size of  $\tau$  is necessary to create a spread sufficient to quantify the uncertainties; however, if  $\tau$  is too large, these models can cause divergence and steer the forecast into unrepresentative low or high values. The fundamental problem is that the uncertainty range is heuristically derived, and as shown in [Figure 3-6](#) there are several assumptions that need to be tuned to have a meaningful output. Adjusting the percentile cutoffs for the base volumes and the degree of perturbation randomness will impact both forecast uncertainty and accuracy. These assumptions may also not translate appropriately to other reservoirs or smaller datasets.

The Bayesian Neural Network (BNN) class of probabilistic forecasts potentially addresses these weaknesses. Though more theoretically complex, there are many advantages at a systems level that simplify the application. The first simplification is that the forecast can be trained to **only predict 1 month ahead of time** ( $\tau = 0$ ). Rather than generating a batch of forecasts, each well was **split into  $n$  forecast periods** starting from the forecast start date and extending to the last month of production. This training was possible as the BNN does not rely on multiple  $\tau$  predictions to generate a probabilistic range; instead, the uncertainties were quantified within the NN structure. Simply stated the BNN **predicts a distribution rather**

**than a single value.** Figure 3-8 demonstrates how this concept alters the dynamic lags by creating  $n$  copies of each well in the **training dataset**; note that **only the input lags change** and the **static values were constant** by well. This method of dividing wells into components allows the algorithm to learn the **entire production profile**, which should theoretically improve late-life predictions. A different strategy for the dynamic lags was also employed to improve performance for this concept. Rather than pulling values from the beginning and end of the production history, the **last 12 months before the forecast start** were used exclusively. Since the algorithm predicts one month into the future, the most recent data were viewed as the best indicator of the following month.

The BNN also allows for some adjustments in the target output to improve the prediction. Inspecting Figure 3-8 illustrates how volumes plateau, with minimal changes later in time. This observation can be leveraged to reduce the likelihood of extrapolation errors and prevent the model from overestimating uncertainty. First, the change in cumulative volume ( $\Delta N$ ) was calculated with Equation 3.6:

$$\Delta N = N_n - N_{n-1}. \quad (3.6)$$

This assumption provides a relatively narrow range of possible predictions, as  $\Delta N \geq 0$  and likely  $\Delta N < \epsilon$ , where  $\epsilon$  is a small value. These properties can be incorporated into the BNN structure to improve model convergence, as seen in Figure 3-9. A Softplus activation function was applied to the mean and standard deviation in the Dense Variational layer (Equation 1.24). Since this activation creates a gentle curve with values slightly above zero, the output of the BNN is a better reflection of the expected well behaviour (see Figure 1-8 for a plot of the activation functions).

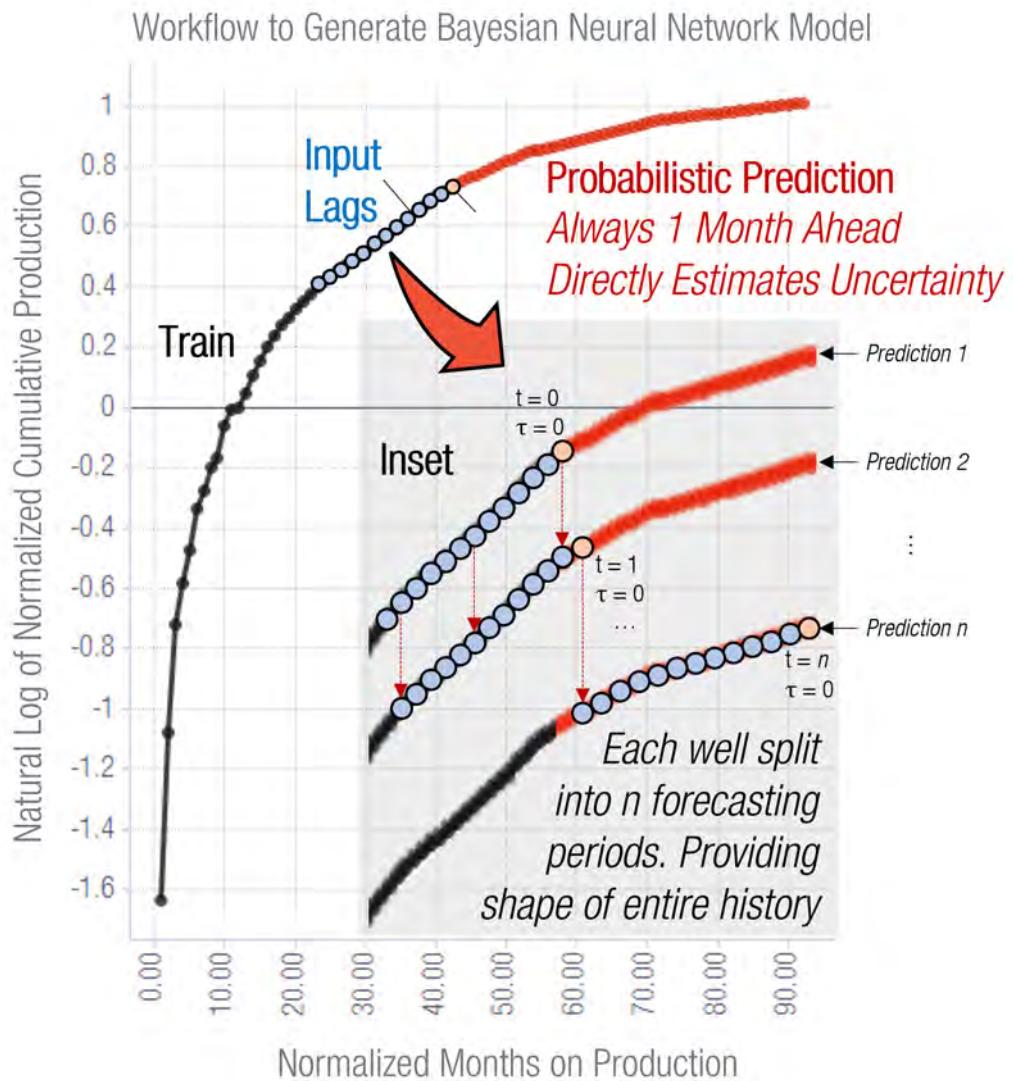


Figure 3-8: The BNN Method fits models at fixed  $\tau = 0$  increments on the training dataset, with a shifting timeslice  $t$  creating  $n$  versions of each well. Uncertainty was directly quantified by the probabilistic layers, allowing the entire production profile to be matched.

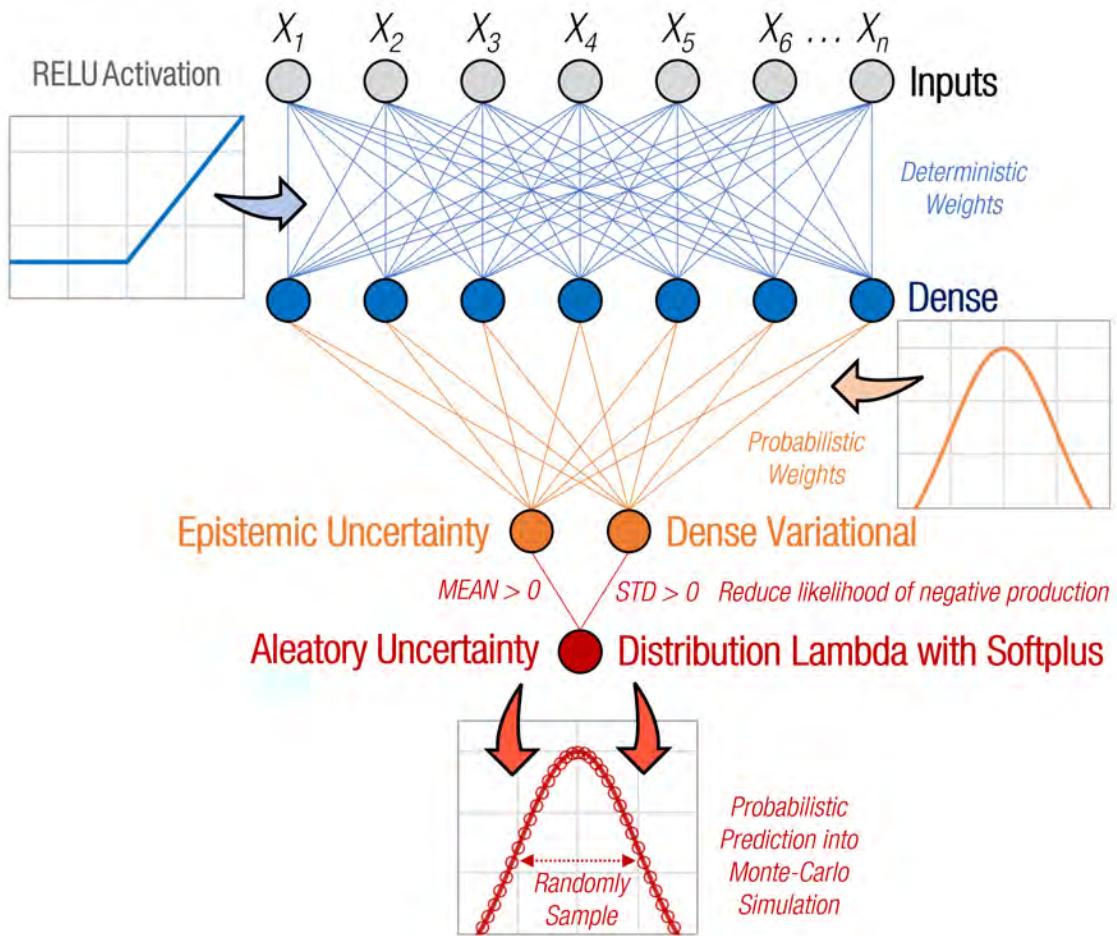


Figure 3-9: Model Structure for the BNN. Inputs were passed to a Dense layer of  $n$  nodes with relu activation functions. Next, Bayesian Inference was performed within the Dense Variational layer. As discussed in [Section 1.4.2](#), a variational posterior term was employed to approximate the true posterior (see [Appendix B-2](#)). Probabilistic weights were fit to account for epistemic uncertainty. A softplus activation function was applied, passing a mean and positive standard deviation to the output node. The final output was a normal distribution trained to match the aleatory uncertainty. When forecasting, both the weights and output distributions were randomly sampled; these values were used in a Monte Carlo simulation to create a probabilistic range.

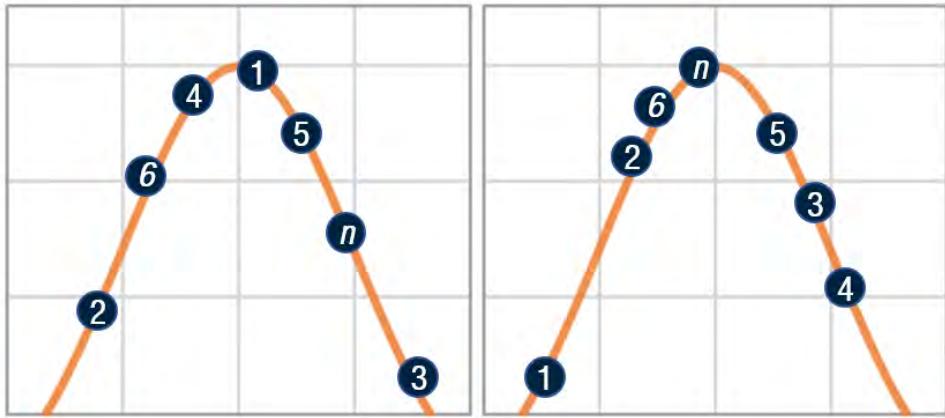


Figure 3-10: An example of random sampling occurring at each of the  $n$  dense variational nodes and 1 output layer. The numbers correspond with  $n$  hypothetical calculations. A random value was sampled at every iteration following each trained distribution; with a large enough sample size, the distributions are perfectly recreated.

Lastly, this structure allows for the aleatory and epistemic uncertainties to be calculated. From [Section 1.4.2](#), the functions were fit based on minimizing the negative log-likelihood and KL Divergence score using the Tensorflow Probability Library [\[56\]](#). For the epistemic uncertainty, rather than a deterministic weight, a Gaussian distribution was trained. Values were then **randomly sampled from these distributions** applying a different weight to the input values for each calculation. This process can be visualized in [Figure 3-10](#). The emergent effect is that each time the model runs, it will have a **unique combination of weights applied**, resulting in a single model providing different viewpoints. A similar process was done for the output node. Since the prediction attempts to match the target variable by training an output distribution, the mean and standard deviation solved represent the aleatory uncertainty [\[16\]](#). A Monte Carlo simulation was performed to generate a forecast for unseen data. For each well, 1000 runs were conducted, which in effect simulates possible production forecasts. A detailed visualization is provided in [Figure 3-11](#). Probabilistic curves were then fit through each timestep. A summary of the entire training and forecasting process is given by [Figure 3-12](#).

## Using a BNN to Generate MC Production Forecasts

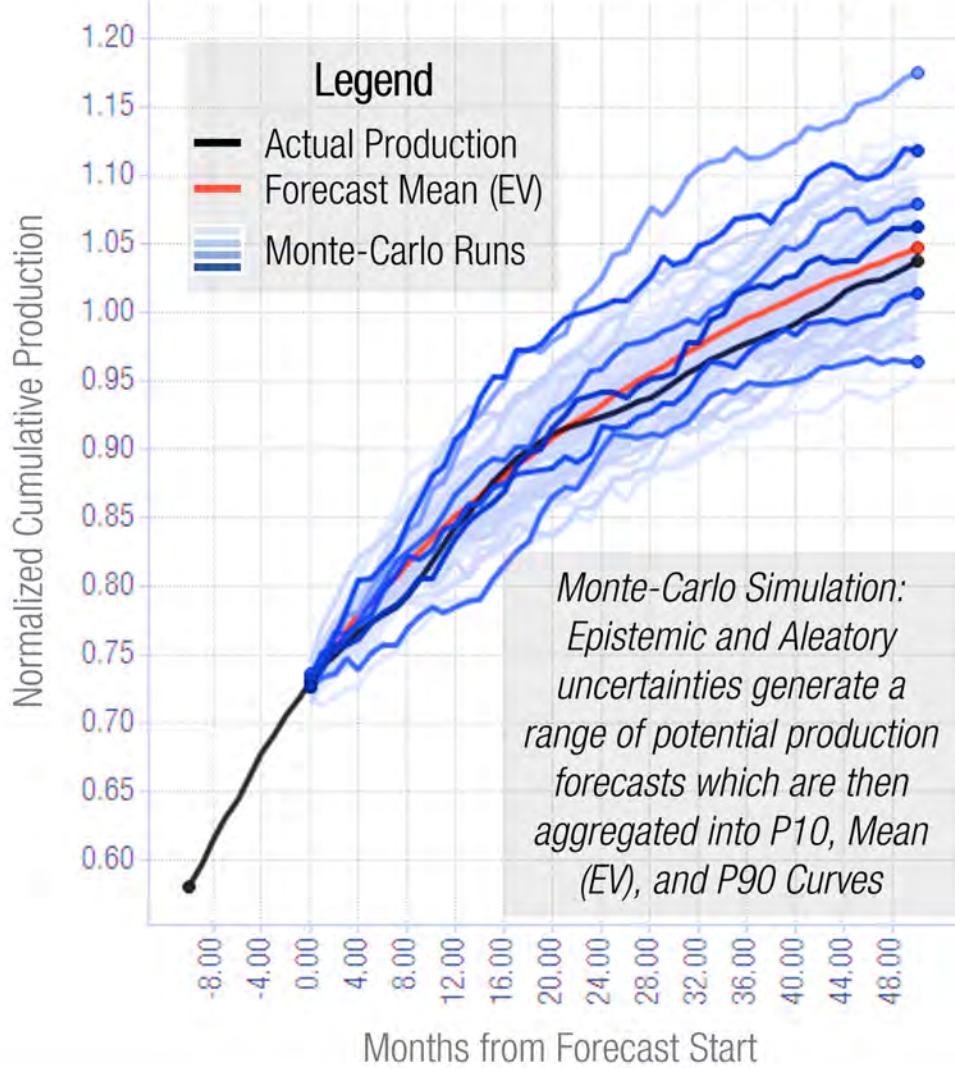


Figure 3-11: Depiction of the Monte Carlo forecasting process applied after training the BNN to the test dataset. Only one month ahead was forecasted ( $\tau = 0$ ), using random sampling of the model's distributions to predict the incremental volume. This value then updated the dynamic lags, replacing the most recent input and shifting all others by 1. The process was repeated, simulating 1000 runs to create a series of possible forecasts by well; these runs are plotted as a blue color gradient to distinguish arbitrary curves. Afterwards, statistical summaries were taken at each timestep to construct probabilistic curves for P10, Mean, and P90. Cumulative production occasionally dropped if the distribution of the well has a small mean and large standard deviation. However, this effect was removed with percentile curves.

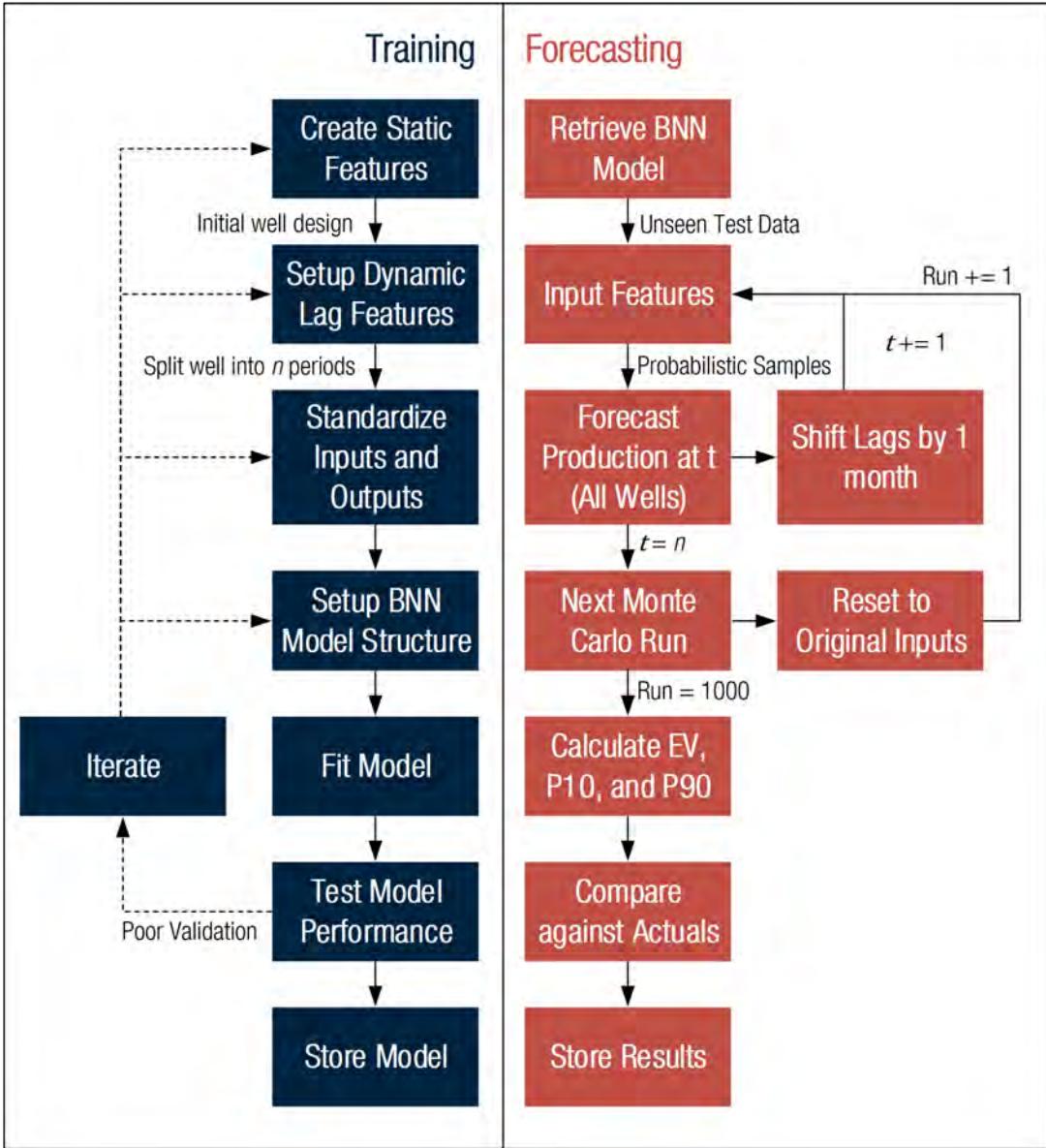


Figure 3-12: Workflow diagram of the BNN. Compared to the ensemble method, the individual components are more complex theoretically but provide a streamlined workflow without heuristics. Rather than training numerous models, a single model was fit on training data to quantify the uncertainties. When forecasting, a Monte Carlo simulation was applied to provide a range of possible production forecasts. Finally, probabilistic curves were formed generating P10, P50, Mean (EV), and P90 predictions.

Table 3.1: Summary of the adjustments to the processed data when applied to predict EUR. Major changes include removing the dynamic lags and predicting geospatial data for total vertical depth and oil gravity using latitude and longitude coordinates. These adjustments allow the EUR prediction to be translated from models trained to predict well production to generic grid cells.

Static Data			Dynamic Data		Target Variable
Admin	Engineering	Geospatial	Gas Lags	Oil Lags	Forecast Start
Keep UWI	Remove Collinearity	<i>Establish Model to predict oil density as function of depth and location</i>	<i>Calculate Normalized Gas and Oil Values based on 12m cumulative volume</i>	<i>Generate gas and oil lag variables n months prior to Forecast Start</i>	<i>Predict the EUR only based on number of months to have a representative EUR</i>
Merge gas and oil lags	Standard Scale and One-Hot Encode				

← Input Features ( $X$ ) → ← Output ( $Y$ ) →

70/30 Test Train Split made by well. Used to test model performance at predicting EUR.

After training model apply on grid cells predicting geospatial features and assuming average engineering parameters

### 3.5 Application on Type Well Forecasts

The previous workflows describe how probabilistic forecasting can apply to existing wells. However, with slight modifications, the output of these processes can also be utilized to form Type Wells. This concept aims to predict the expected ultimate recovery (EUR) for an **individual section** in the Eagle Ford assuming a **standardized engineering design**. The existing wells form the basis of training and testing performance before applying to generic sections to identify gas and oil *Sweet-Spots*. As a result, the input data requires modification to address this use-case; a summary of the changes is provided in [Table 3.1](#). The primary change for the input data was the **removal of the dynamic lags**. Since Type Wells are for new areas without production data, the algorithms must predict EUR based on the **static features alone**. This workflow will only discuss predicting the EUR for simplicity, but the process can be repeated for any month.

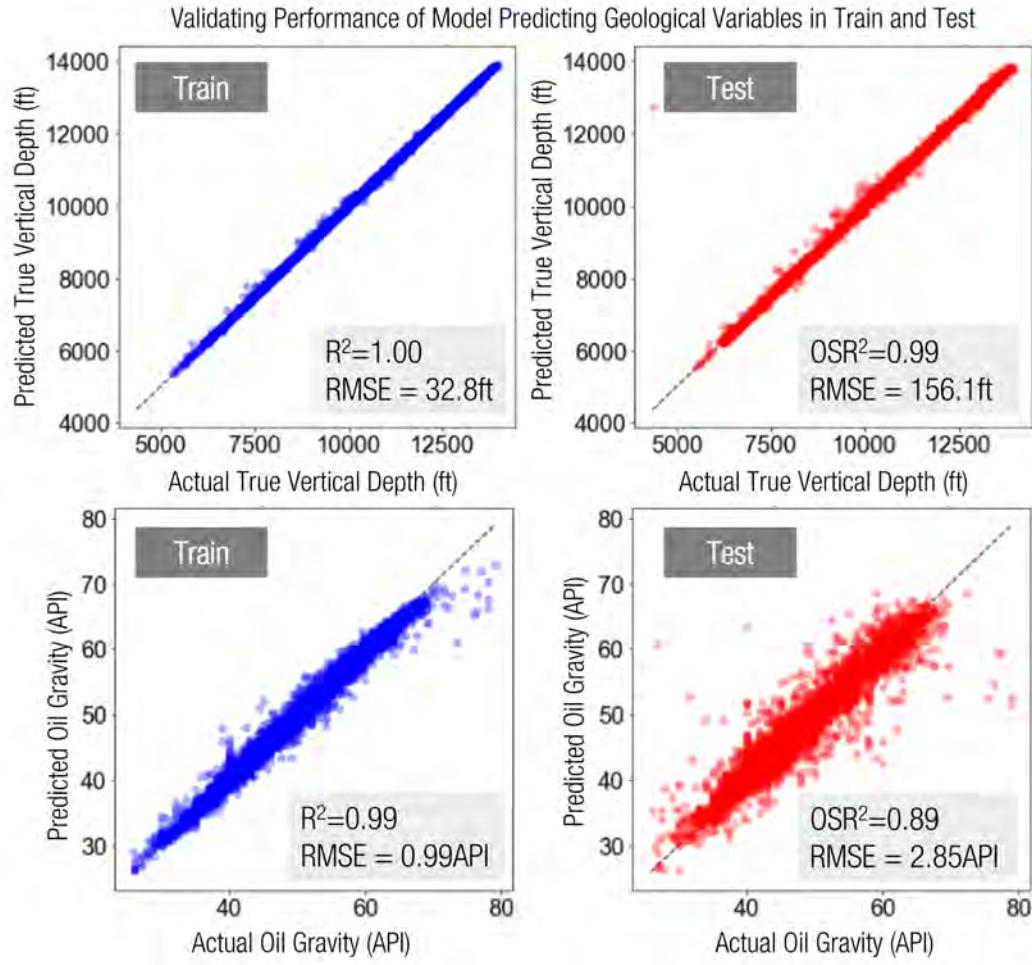


Figure 3-13: Validation plot comparing actuels ( $x$ -axis) compared to predicted values ( $y$ -axis). A perfect prediction would follow the dashed diagonal line. Training results are shown in blue and test in red. The high OSR $^2$  results support that this methodology is valid to generate simple geological parameters. Plot format is based on the literature [75].

As discussed in [Section 2.1.2](#), geological variables are expected to be high impact but are sparsely populated within this dataset. Additionally, most of these variables cannot be forward-filled, as they are too heterogeneous. However, oil density and total vertical depth (TVD) can be imputed due to a predictable reservoir slope and a large pool of wells. If a model can be trained to predict these features on existing producers, the model can then be applied to interpolate areas without well control. A Random Forest algorithm [\[62\]](#) was trained to predict TVD and oil gravity, using latitude and longitude. The performance results are shown in [Figure 3-13](#) showing

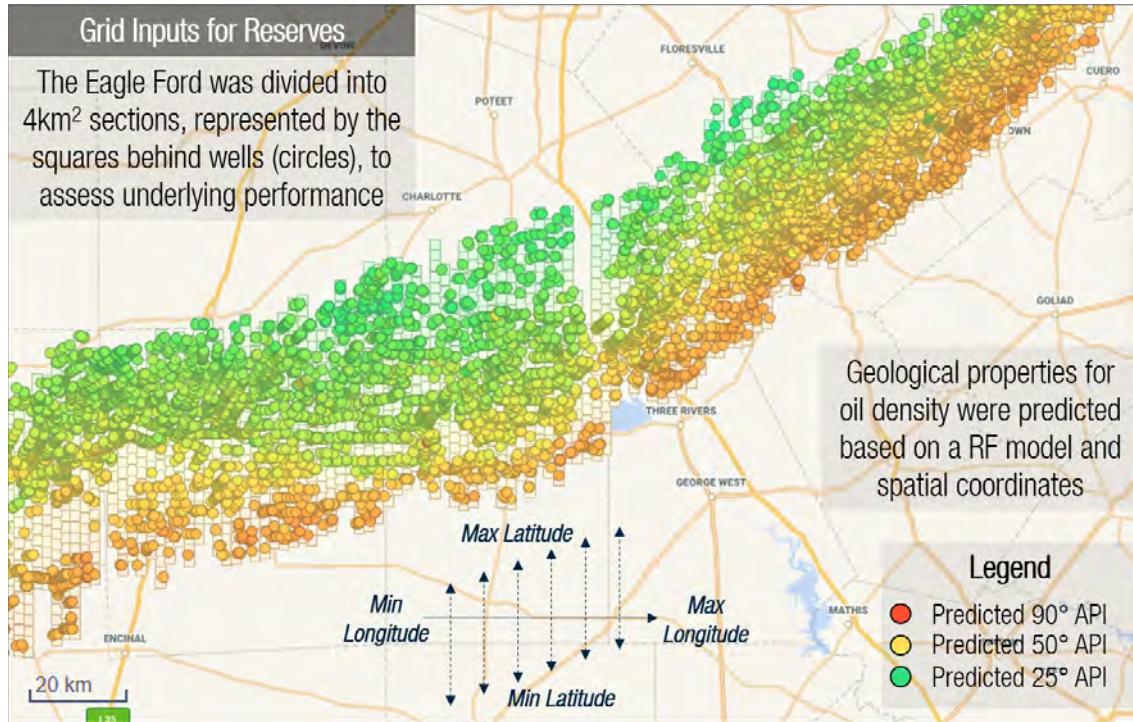


Figure 3-14: Plot of the Eagle Ford split into 4-km<sup>2</sup> sections. The circles represent known wells, while the squares are the grid cells approximating possible development areas; this grid was formed from the maximum and minimum longitude and latitude of nearby wells. A trained RF model was then applied to each grid cell to predict TVD and oil gravity. A graph with only grid cells is displayed in [Appendix B-1](#).

strong predictability with an OSR<sup>2</sup> of 0.99 for TVD and 0.89 for oil gravity.

Next, an additional input dataset was constructed by splitting the Eagle Ford reservoir into grid cells. First, the algorithm is trained to predict EUR on existing wells, then applied to forecast the underlying performance of the reservoir assuming a standardized design. The maximum and minimum longitude was identified, establishing the grid bounds. An array was formed by splitting this range into 2-km increments horizontally. Then, for each horizontal increment, the maximum and minimum latitude were calculated based on wells within the vicinity. Similarly, this vertical range was divided into 2-km increments, creating 4-km<sup>2</sup> grid cells. Finally, the trained random forest model was used to predict each cell's TVD and oil gravity. The methodology and example results are plotted in [Figure 3-14](#) and [Appendix B-1](#).

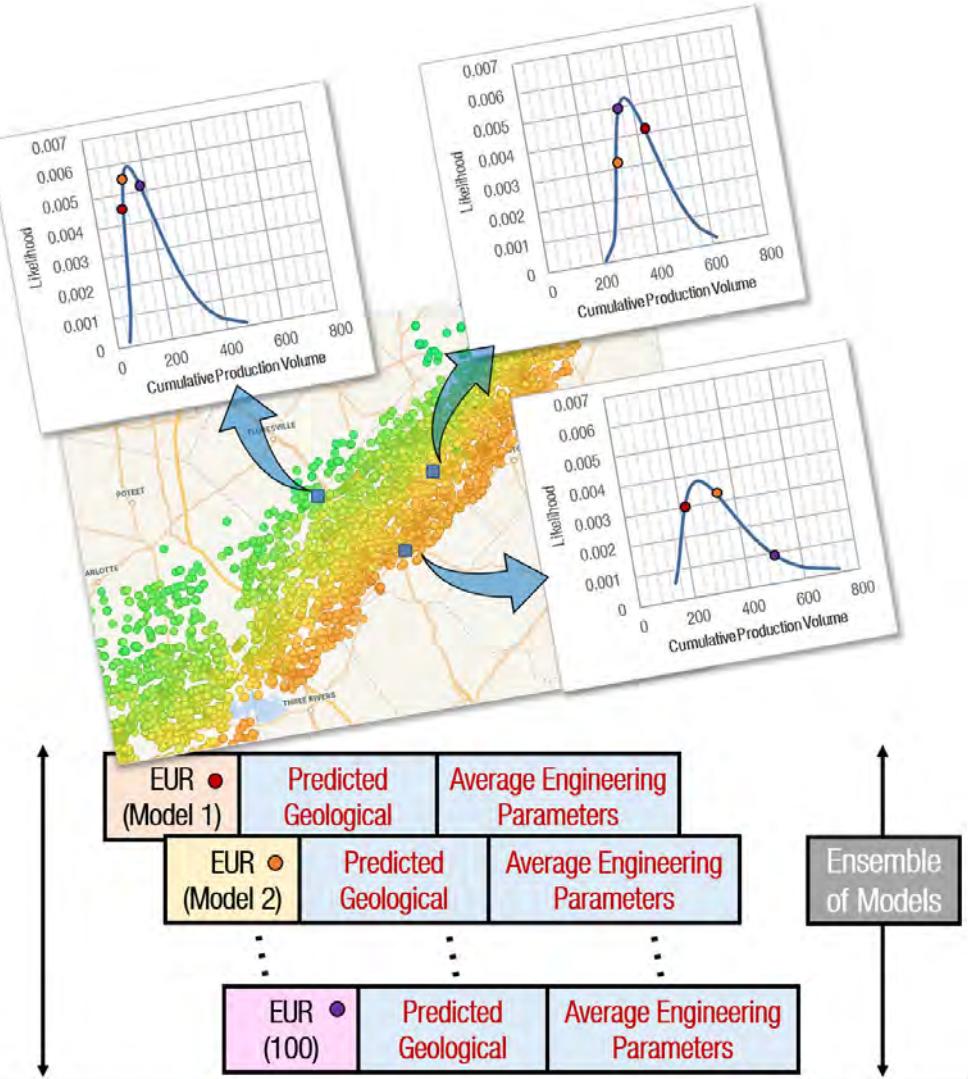


Figure 3-15: Illustration of the Monte Carlo sampling method for ensemble training, using the uncertainty distributions developed from the existing well forecasts. The EUR distribution was randomly sampled at each run and used as the target variable to fit an XGBoost model. This process was repeated 100 times, creating an ensemble.

With the input values modified, the probabilistic forecasting methods can be applied to predict EUR. These methods build off the existing well forecast results, leveraging the cumulative volume uncertainty distributions. For the ensemble-based method, a Monte Carlo process was used to train a set of 100 models. However, unlike traditional Monte Carlo simulations, the input values were constant, and the target output was randomly sampled off the EUR uncertainty distributions. The process was shown in greater detail in [Figure 3-15](#).

The full ensemble-based workflow is demonstrated by [Figure 3-16](#). When forecasting on the grid cells, a standardized well design was defined as the **average of the training dataset**. This assumption is a simplification for modelling purposes. If this workflow is deployed, it is recommended that the operator modify this assumption to match their standardized well design. Since this thesis intends only to evaluate the concept merit and not prescribe a specific design, the exact parameters chosen are not critical. Additional commentary on generating a prescriptive design is provided in [Section 6.2](#).

The BNN process applied to predict EUR is given by [Figure 3-17](#). Like the ensemble-based approach, the output of the existing well forecast was used to train a BNN. Note that GMM clustering was applied to the grid cell predictions to identify *Sweet Spot* areas for both methods and is discussed further in Chapter 4.

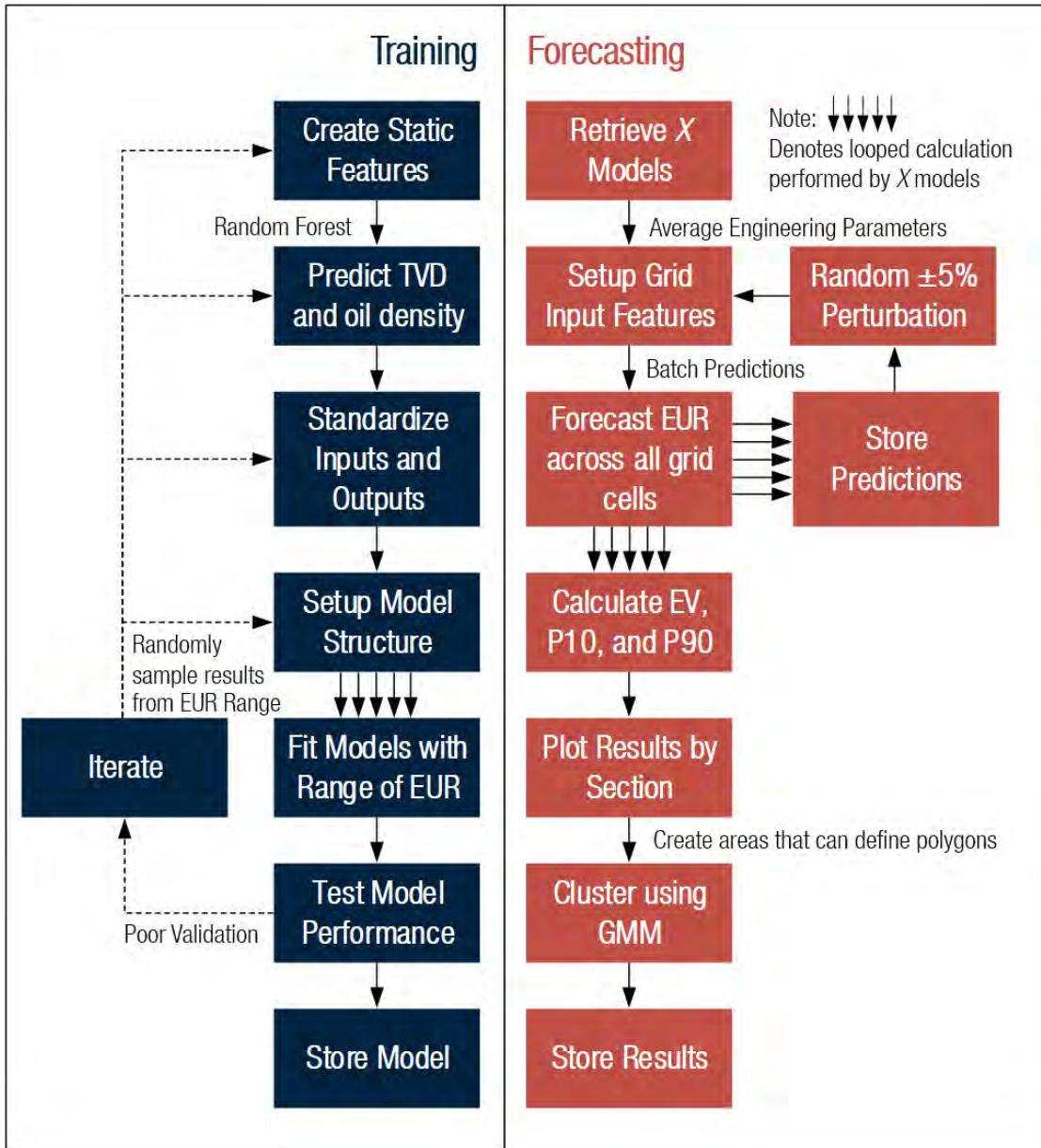


Figure 3-16: This workflow leverages the results of the uncertainty range calculated from the existing well forecasts. A Monte Carlo sampling process of EUR was performed **varying the target output** and used to train 100 models. These models were then applied to forecast EUR on each grid cell populated with average engineering parameters with random  $\pm 5\%$  perturbations. These results were converted into percentile ranges and clustered using GMM.

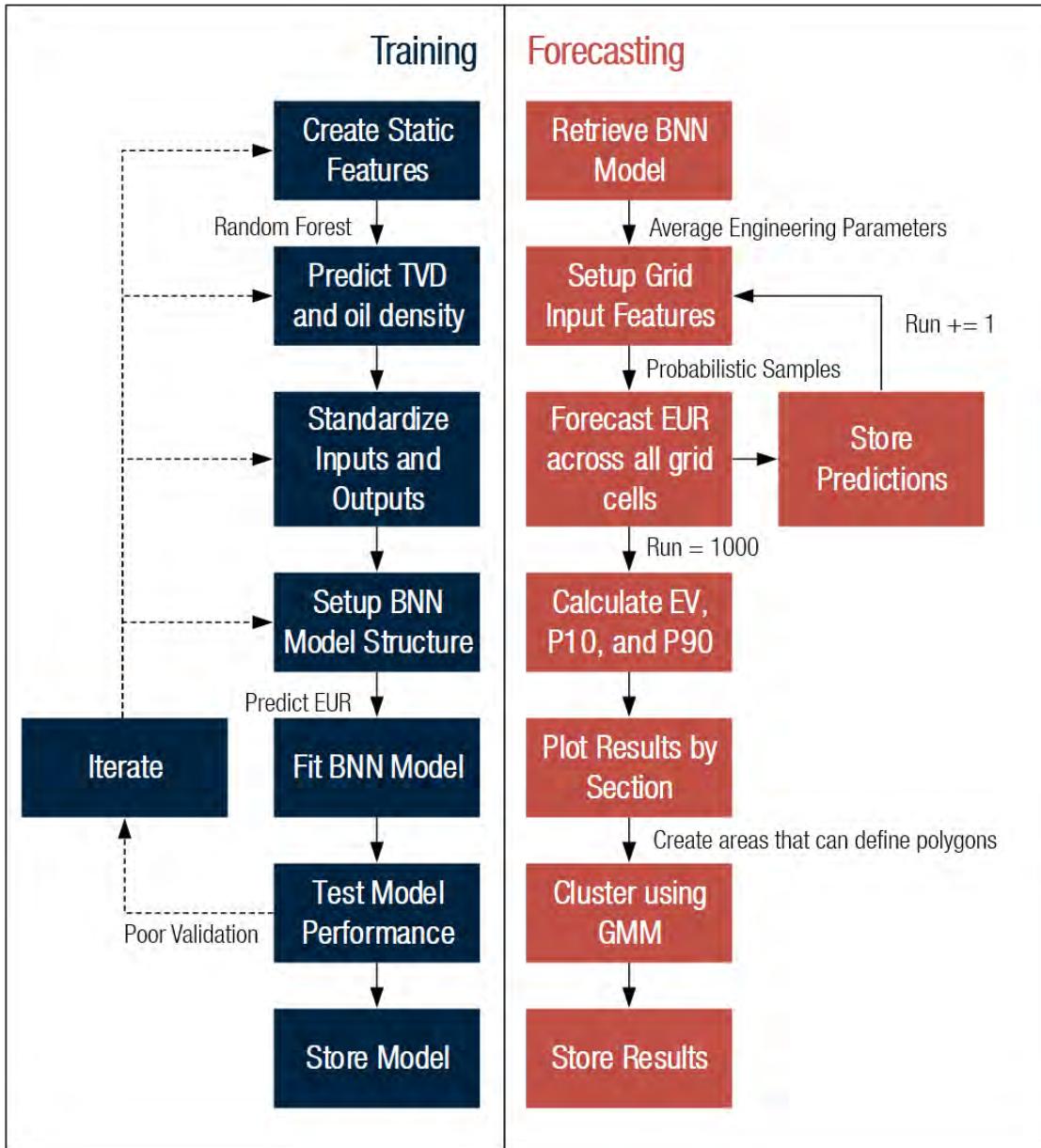


Figure 3-17: This workflow builds off the results of the BNN method for forecasting existing well production. The same layer structure shown in [Figure 3-9](#) was applied to predict cumulative production, as this structure can directly quantify aleatory and epistemic uncertainty. The model was then applied using a Monte Carlo simulation to forecast EUR distributions on each grid cell. Finally, results were converted into percentile ranges and clustered using GMM to generate sweet spot mapping.

THIS PAGE INTENTIONALLY LEFT BLANK

# Chapter 4

## Analysis Results

### 4.1 Existing Well Forecasting Results

This chapter provides the results of the established probabilistic forecast methodologies for Existing and Type Well workflows. First, the findings of the Existing Well forecasts will be analyzed in aggregate to identify emergent effects. Critical observations will be identified and summarized for each algorithm. Examples will then be shown for indicative wells, illustrating typical outputs and generating hypotheses of larger-scale trends. Chapter 5 will then build off these findings, interpret root causes, and discuss implications on the research questions.

#### 4.1.1 Aggregate Comparison

From [Section 3.1](#), evaluation criteria were established for training each model class. However, when evaluating overall model performance, the mean absolute percent error (MAPE) will be used as the primary criterion; these values are more straightforward to interpret than other metrics, since they are relative to the target variable. [Equation 4.1](#) is used to calculate this value, where  $y_i$  is the actual,  $\hat{y}_i$  the predicted, and  $n$  the number of samples. The remaining metrics are presented as supporting values,

with their performance varying based on the units and variable magnitude:

$$\text{MAPE} = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right|. \quad (4.1)$$

After training the forecast algorithms, these models were applied to the test dataset. Only the **initial static and dynamic features** from before the forecasting period were provided. The algorithms then update the dynamic lags purely with internally predicted values, and no reconciliation occurs with actual volumes. Therefore, it is expected that errors will propagate over time, as variances can accumulate and cause the model to drift. A validation plot of the mean results is provided by [Figure 4-1](#). The actual volume is plotted on the  $x$ -axis and the predicted value on the  $y$ -axis. Note that a perfect prediction will follow the diagonal line. All methods showed reasonably strong performance with an OSR<sup>2</sup> between 87-89%. However, the **BNN had the best results consistently** with the lowest MAPE and RMSE scores. The **Ensemble - Tree Based** method had the **second-highest performance**, followed by the Automated DCA and Ensemble - NN methods. The DCA and NN approaches exhibited more spiked regions from the diagonal line, indicating a propensity to deviate from measured values. This effect can be more clearly seen by plotting MAPE results from the start of the forecast ( $t$ ), as shown in [Figure 4-2](#).

[Table 4.1](#) provides a summary of these results binned in increments of 10 months for MAPE, OSR<sup>2</sup>, and RMSE. **BNN demonstrated a 10 to 35% performance improvement** over other methods after 25 months, which increases in magnitude over time. Since the Existing Well model is used to estimate the ultimate recovery of the well, having unreliable late time forecasts is a concern for the Type Well process, which utilizes these predictions as the target variable. Next, the probabilistic component of the production forecast can be contrasted. [Figure 4-3](#) calculates the percentage of wells that fall within the P10-P90 range.

[Figure 4-3](#) demonstrates mixed results for the BNN, with the probability range adequately bracketing actual production for one year capturing between 60 to 94% of wells. Afterwards, the fraction of wells within the P10-P90 range dropped to

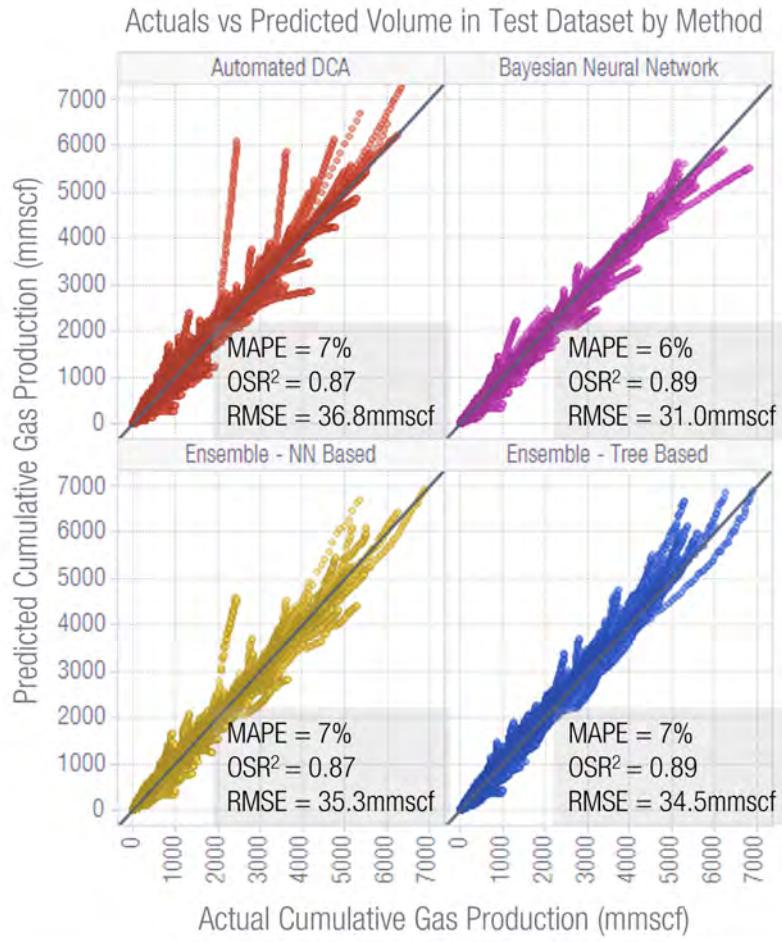


Figure 4-1: Validation plot demonstrating model performance on unseen test data. Overall the BNN had the best results for all three metrics. Notably, RMSE was 10% better than the next closest algorithm.

40%, indicating that the range predicted was too narrow. Conversely, the ensemble methods started overly tight capturing only 40% of wells; however, after one year 80% of the wells fell within the P10-P90 range, providing a more desirable result. However, [Figure 4-3B](#) demonstrates that the ensemble methods were skewed with a high percentage of values (70-80%) falling within the P10-P50 range; conversely, the BNN exhibited balanced results, with 50% above and below. Investigating the residual distributions provides additional clarity on these findings, as shown in [Figure 4-4](#).

The Automated DCA method was aligned on average with actual production, with approximately equal residuals; this result supports that the shift factor  $q_{\text{adj}}$  is working as anticipated. DCA is the benchmark for the mean results, as there is no defined

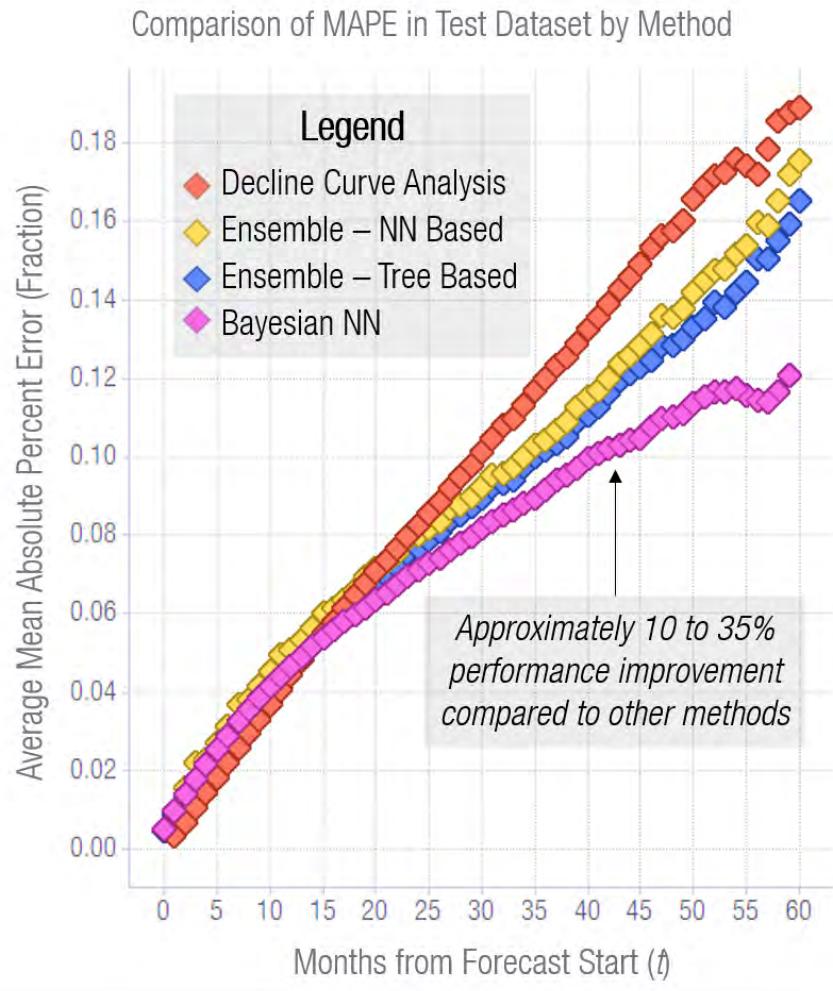


Figure 4-2: Plot of MAPE versus the start of forecast ( $t$ ). All Probabilistic ML algorithms demonstrate improvements over DCA after 25 months. The BNN has the strongest results, which increase in significance with  $t$ . The best performing Ensemble Method was the Tree-based.

process to generate the P10 and P90 forecasts. Overall, the **BNN** had the **lowest residual range**, defined as the difference between the P90 and P10 residuals, which aligned with the lowest RMSE results. Despite having a significantly more narrow range than the ensemble methods, the negative residuals matched strongly with the P10 forecast; this result indicates a good representation of the low-end forecast range. However, the residuals were highly biased to be larger than the P90, indicating the algorithm was **underestimating the upside potential**.

Both Ensemble approaches exhibit similar trends. The Tree-based method gener-

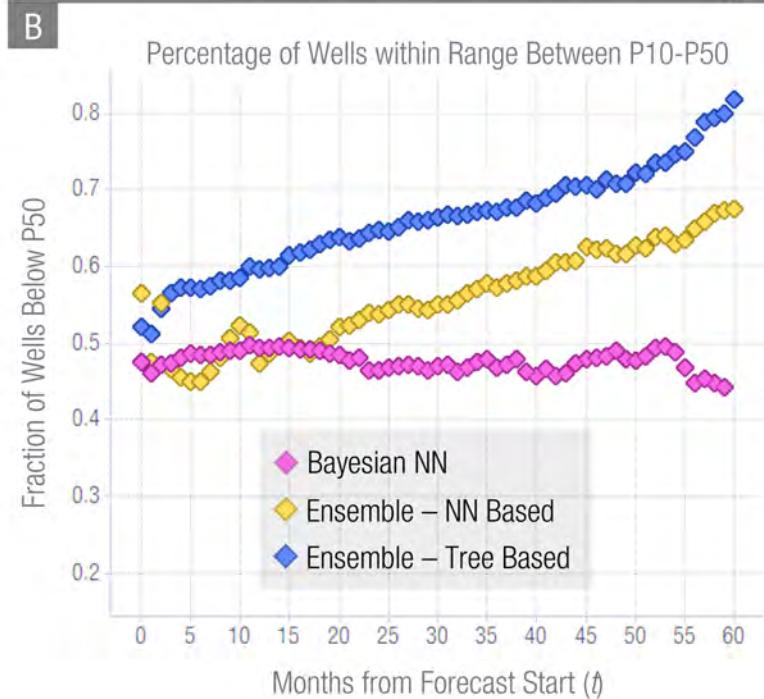
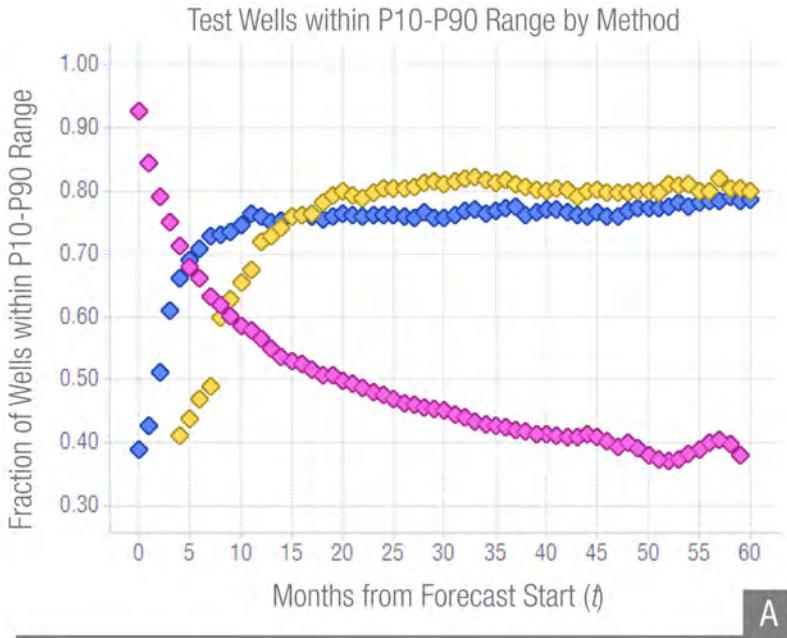


Figure 4-3: Percentile range results for each forecasting method. (A) depicts the percentage of wells with actual production within the P10-P90 range. The BNN and Ensemble methods show opposite effects, with the BNN capturing a significant portion of the wells initially and dropping to 40% as  $t$  increases. Ideal performance is approximately 80%, favouring the ensemble methods. (B) isolates for the percentage of wells below P50 and within the P10-P90 range. Ideally, this value would be 50% to have equal skew, preferring the BNN method.

Table 4.1: Summary of the evaluation criterion for each method binned into 10 month forecasting periods. MAPE and RMSE prefer the BNN after the  $20 < t < 30$  period, while OSR<sup>2</sup> does not significantly differ until  $50 < t < 60$ .

Model	Mean Absolute Percent Error (MAPE)					
	$0 < t < 10$	$10 < t < 20$	$20 < t < 30$	$30 < t < 40$	$40 < t < 50$	$50 < t < 60$
Automated DCA	2%	5%	8%	11%	14%	17%
Bayesian Neural Network	3%	5%	7%	9%	10%	11%
Ensemble - NN Based	3%	6%	8%	10%	12%	15%
Ensemble - Tree Based	2%	5%	8%	10%	12%	14%
Model	Out of Sample R <sup>2</sup> (OSR <sup>2</sup> )					
	$0 < t < 10$	$10 < t < 20$	$20 < t < 30$	$30 < t < 40$	$40 < t < 50$	$50 < t < 60$
Automated DCA	98%	91%	84%	81%	75%	67%
Bayesian Neural Network	98%	91%	85%	82%	79%	79%
Ensemble - NN Based	98%	91%	84%	81%	75%	67%
Ensemble - Tree Based	98%	92%	87%	84%	80%	75%
Model	Root Mean Squared Error (RMSE)					
	$0 < t < 10$	$10 < t < 20$	$20 < t < 30$	$30 < t < 40$	$40 < t < 50$	$50 < t < 60$
Automated DCA	7.3	23.3	39.9	59.1	82.0	111.3
Bayesian Neural Network	10.4	23.3	36.2	49.3	63.0	75.3
Ensemble - NN Based	11.7	27.7	40.3	52.6	68.2	85.5
Ensemble - Tree Based	9.2	24.7	39.3	53.9	71.5	94.0

ated a broader range, while the Neural Network was more closely aligned with actual production and has more evenly distributed residuals. Overall, the EV and P90 forecasts overestimated production, defining a more extensive spread than observed by the residuals. This result suggests that these methods created such a **vast probabilistic output** that it was highly likely to have wells fall within it. Ideally, this range would be as **small as possible** while still capturing the **most likely** degree of uncertainty, as a model that consistently suggests unrealistic production profiles will not be trusted and loses utility. Consequently, before concluding what approach is preferred for probabilistic estimations, individual well forecasts were inspected to identify trends that explain the emergent behaviour of the aggregate result.

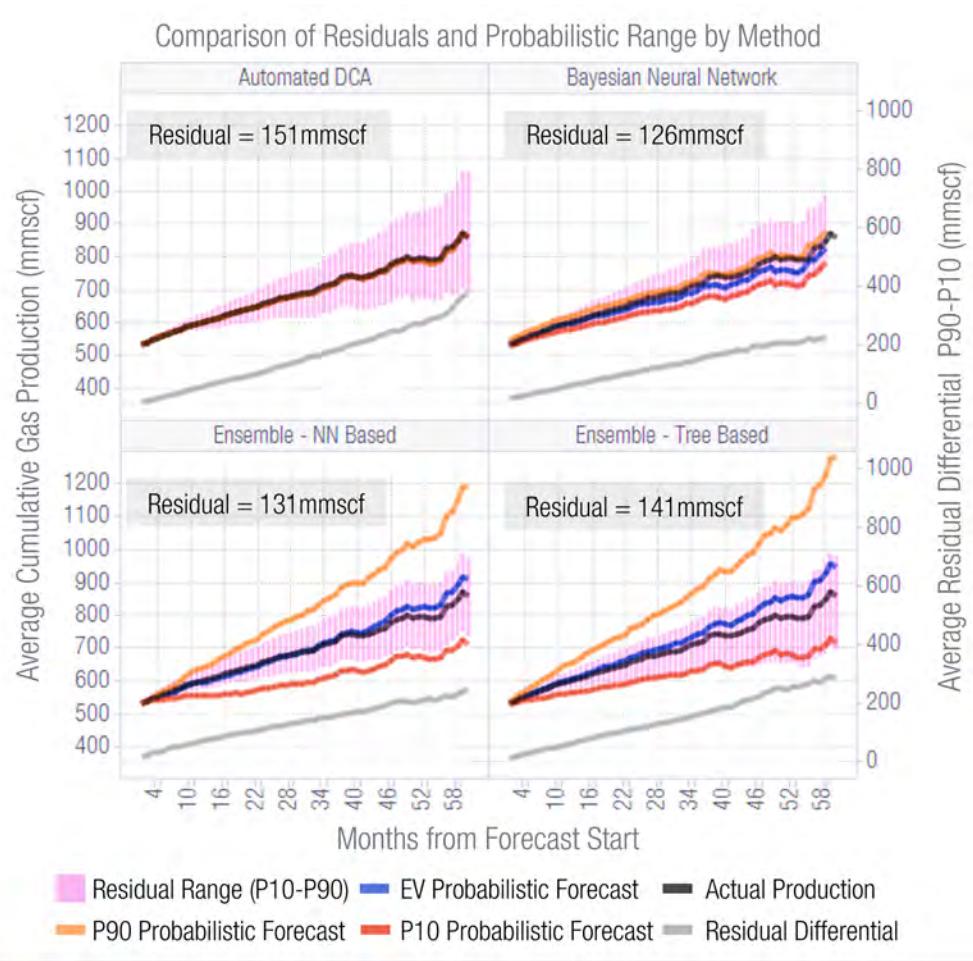


Figure 4-4: DCA was balanced with equal residuals above and below but had the largest residual range. BNN provided a narrow forecast range while still capturing the P10 residual within the bounds of the forecast. The P90 residual was above the corresponding forecast, indicating it was overly conservative. Both Ensemble methods captured the P10 residual appropriately but were too optimistic for the P90.

#### 4.1.2 Indicative Individual Well Plots

Inspecting the individual well production profiles provides an alternative to the aggregate analysis. A random subset of 100 wells was evaluated to identify common findings qualitatively and generate hypotheses of algorithm trends. Metrics were then defined to quantify these trends in aggregate, ensuring that these observations were representative of the entire dataset. In the context of systems analysis, this type of evaluation is equivalent to decomposing the problem into a lower-level system, then aggregating results to identify the higher-level trend. *Indicative wells* were selected

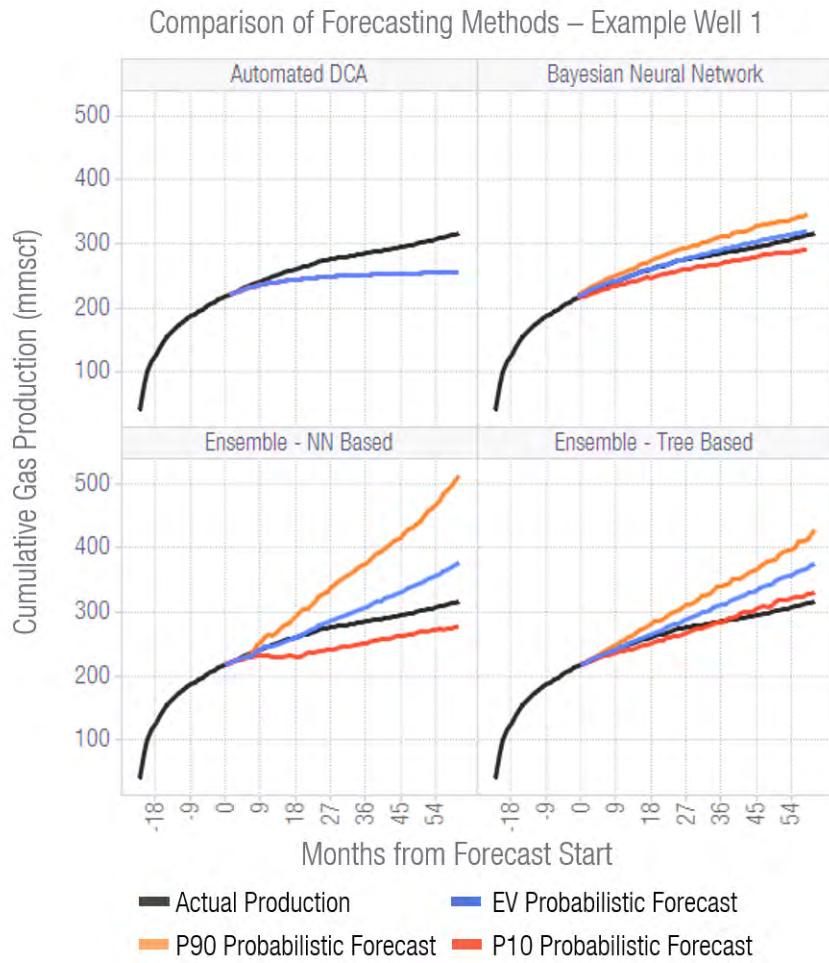


Figure 4-5: Well 1 exhibited exemplary behaviour for the BNN method by comparing the actual profile and the probabilistic range. For the Ensemble-NN based approach, a divergent forecast was encountered. The P10 forecast has periods of negative production followed by a steep increase in the P90 near month 50. The tree-based methodology also demonstrated cyclical tendencies. Generally, the Ensemble methods created cone-shaped envelopes, rather than the smooth profiles of the BNN.

to represent the observed characteristic trends; three profiles are presented in this section, with additional examples provided in the [Appendix B.1](#). Well 1, shown in [Figure 4-5](#), provides a reference for the BNN methodology working as anticipated with credible curvatures for each forecast. Generally, the ensemble-based methods also exhibited reasonable trends for the EV forecast; however, the P10 and P90 predictions displayed divergent behaviour. This effect created cone-shaped envelopes, which were unlikely to match actual production profiles.

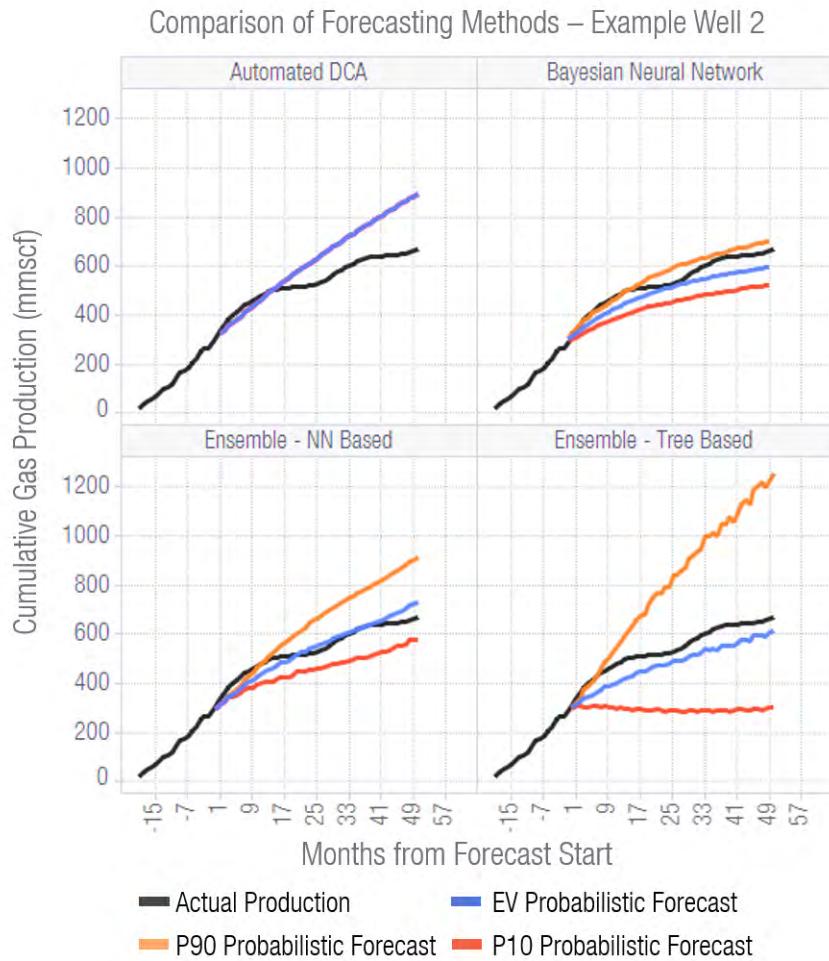


Figure 4-6: Well 2 had realistic ranges defined for both the BNN and Ensemble-NN approaches. However, the Ensemble-Tree based method had a massive uncertainty range that varied by a factor of 2. For the P10 forecast, negative production was encountered, which does not seem warranted based on the historical production trend.

Well 2, in [Figure 4-6](#), also exhibited similar trends to Well 1. In this case, clear divergent behaviour occurred for the Ensemble-Tree based method with consistent oscillations and negative production for the P10 estimate.

[Figure 4-7](#) provides an example of the BNN forecast failing to capture actual volumes within the probabilistic range. Despite the historic production appearing to flatten, a sudden spike in rates occurred. This unexpected signature appeared in the dataset to a larger degree than anticipated and requires further investigation.

Based on this analysis, the curvature of the production profiles generated from the

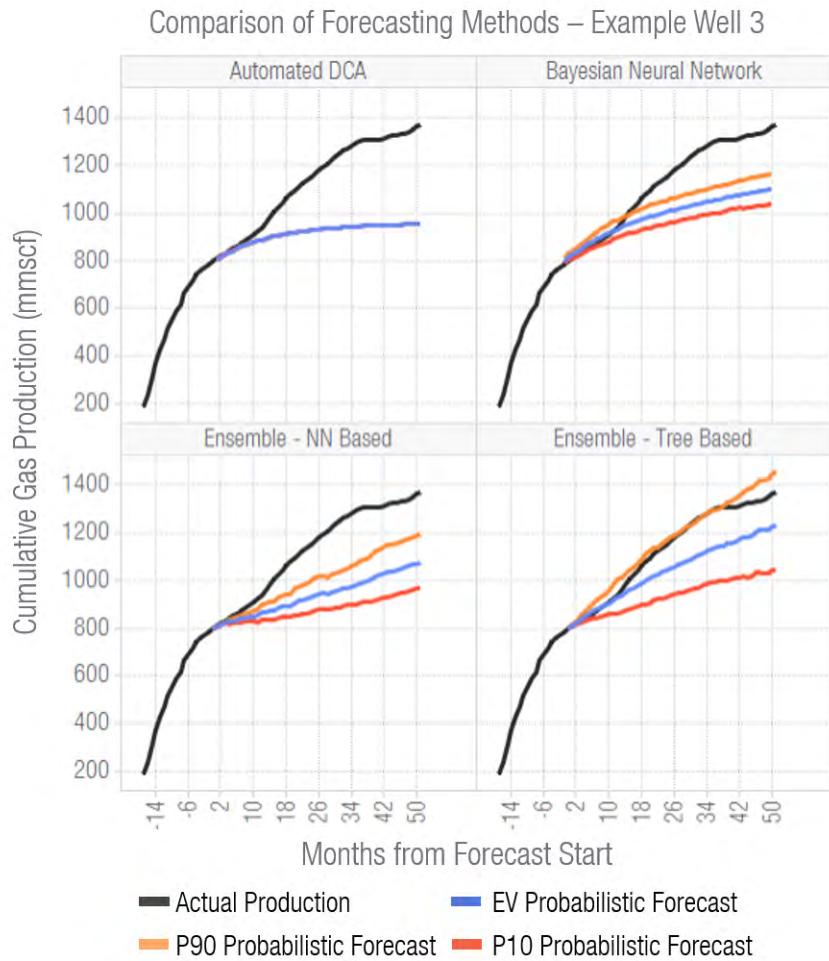


Figure 4-7: Well 3 provides an example of actual values falling outside of the BNN forecast range. The DCA, BNN, and Ensemble-NN based approaches all predicted significantly less cumulative volumes than the well achieved. However, the Ensemble-Tree based approach captured this increase due to a large P10-P90 spread.

BNN methodology seemed more realistic compared to the other methods; this claim was tested quantitatively using cosine angle ( $\cos \theta$ ) from [Equation 4.2](#), where  $y$  and  $\hat{y}$  are arrays of the actual and predicted production for each well, respectively. Note that only the vector shapes are compared and are independent of distance [75]. The maximum possible score of  $\cos \theta$  is 1; however, to emphasize the differences between the methods,  $1 - \cos \theta$  was calculated, with a **smaller score** indicating a **better**

Table 4.2: Angular Similarity of the probabilistic curves for each forecast and model class. The P10, mean, and P90 results are presented for every forecast to indicate performance variance across the dataset. The BNN reliably had the closest angular similarity to actual production profiles.

Model Curve Shape	Mean Forecast 1-cos( $\theta$ )			P10 Forecast 1-cos( $\theta$ )			P90 Forecast 1-cos( $\theta$ )		
	P10	Mean	P90	P10	Mean	P90	P10	Mean	P90
Automated DCA	3.7E-03	1.6E-03	1.9E-05						
Bayesian Neural Network	2.1E-03	1.0E-03	2.1E-05	2.7E-03	1.3E-03	3.0E-05	2.0E-03	9.8E-04	2.3E-05
Ensemble - NN Based	2.8E-03	1.3E-03	5.2E-05	5.2E-03	2.4E-03	1.2E-04	8.1E-03	3.6E-03	2.8E-04
Ensemble - Tree Based	2.4E-03	1.1E-03	5.3E-05	4.8E-03	1.9E-03	4.5E-05	8.8E-03	3.6E-03	3.1E-04

match with actual profiles:

$$\text{Angular Similarity} = \cos \theta = \frac{\mathbf{y} \cdot \hat{\mathbf{y}}}{|\mathbf{y}| |\hat{\mathbf{y}}|} = \frac{\sum_{i=1}^n y_i \hat{y}_i}{\sqrt{\sum_{i=1}^n y_i^2} \sqrt{\sum_{i=1}^n \hat{y}_i^2}}. \quad (4.2)$$

A summary of the statistical results for the test dataset are presented in [Table 4.2](#). The mean, P10, and P90 scores are reported for each probabilistic forecast method to demonstrate how strongly curvature varied between wells. Regardless of the metric, **BNN provided the best angular similarity**. The **Ensemble-tree based approach** had the **second-highest results**. Finally, the frequency of unexpected spikes was estimated using a box plot analysis on the BNN residuals. The result of this process indicated that approximately **7% of wells** were positive outliers, defined as 1.5 times the interquartile range and shown in [Appendix B-13](#). This percentage is the upper bound for wells exhibiting a spiking tendency.

### 4.1.3 Generating EUR Output Variable

The final component of the Existing Well forecast was to apply the models at the end of available historical production (June 2020) to forecast well EUR; see [Figure 1-5](#) for a visualization. Based on prior results, the BNN and Ensemble-Tree Based methods were consistently the two best algorithms. Therefore, to simplify the modelling

### Comparison EUR Forecasting Methods – Example Well 1

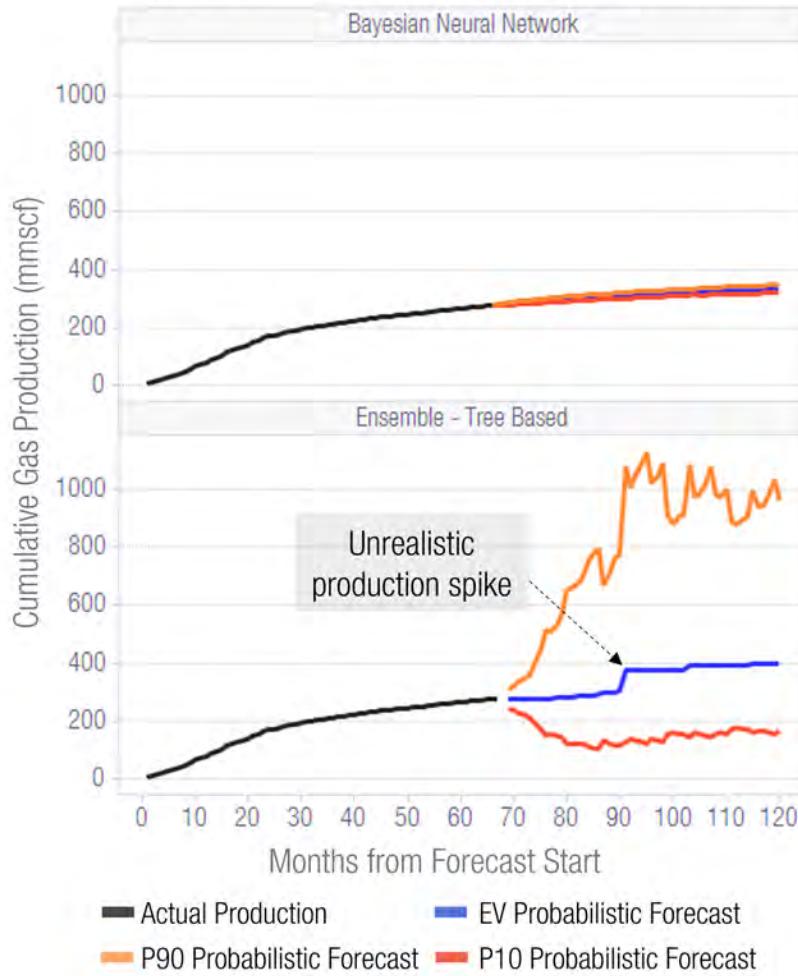


Figure 4-8: Extreme instability occurred for the ensemble method, generating an unrealistic spread in P10-P90 values. The range became so divergent that the graph scaling required to plot the ensemble forecast makes the BNN uncertainty range appear to be a single curve. The Ensemble also showed a spike in EV production around 90 months, potentially mimicking patterns seen in [Figure 4-7](#).

process, only these models were used to compare predicted volumes of month 120, corresponding with the longest normalized production month in the dataset. Utilizing this upper limit avoided imputation beyond observed values in the training set, which was discussed as a risk in [Section 1.4.3](#). Since the actual EUR is unknown, this section will qualitatively judge whether the profiles generated are reasonable. [Figure 4-8](#) demonstrates a typical well result.

[Figure 4-9](#) rescales the same well to illustrate the BNN forecast. This forecast was

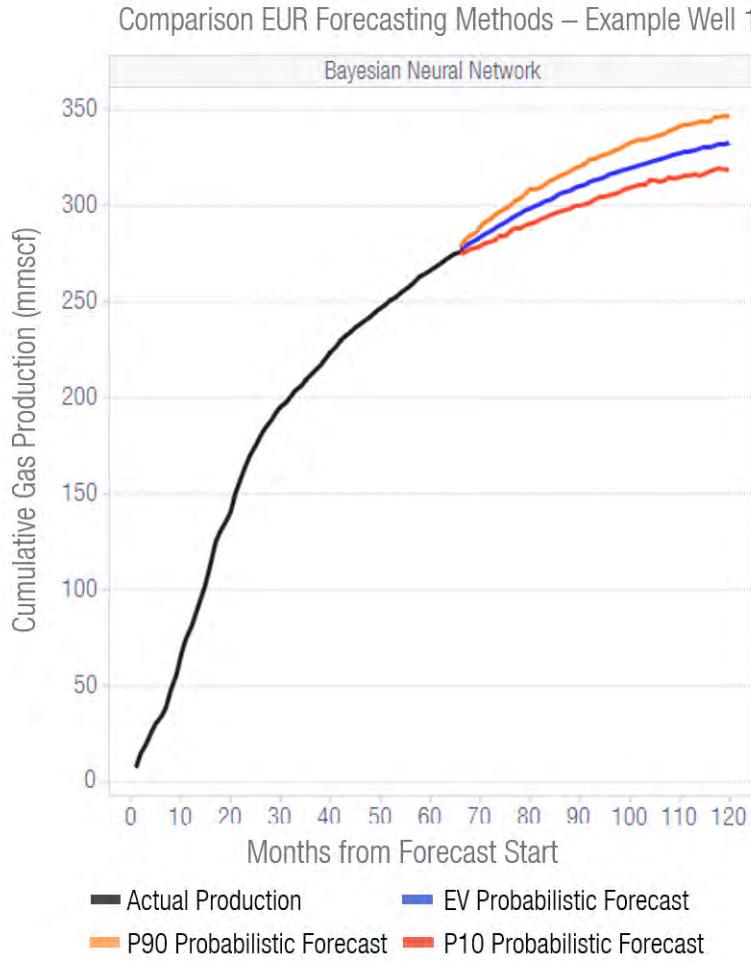


Figure 4-9: The BNN method rescaled for EUR Example Well 1 in [Figure 4-8](#). The curvature of each forecast was reasonable and matched the strong angular similarity results from [Table 4.2](#). A relatively narrow range should be expected for this period, as cumulative production plateaus over time, due to decreased well productivity.

better aligned with the expected production profile. The P10 and P90 forecasts also appeared to be reasonable, without the divergent effects seen by the ensemble method in [Figure 4-8](#). These results were consistent for the entire dataset, with a larger subset of examples provided in [Appendix B.2](#). Therefore, it is **recommended to use the BNN method to predict EUR**. Ensemble-based methods are too unstable in this configuration to probabilistically estimate end-of-life production.

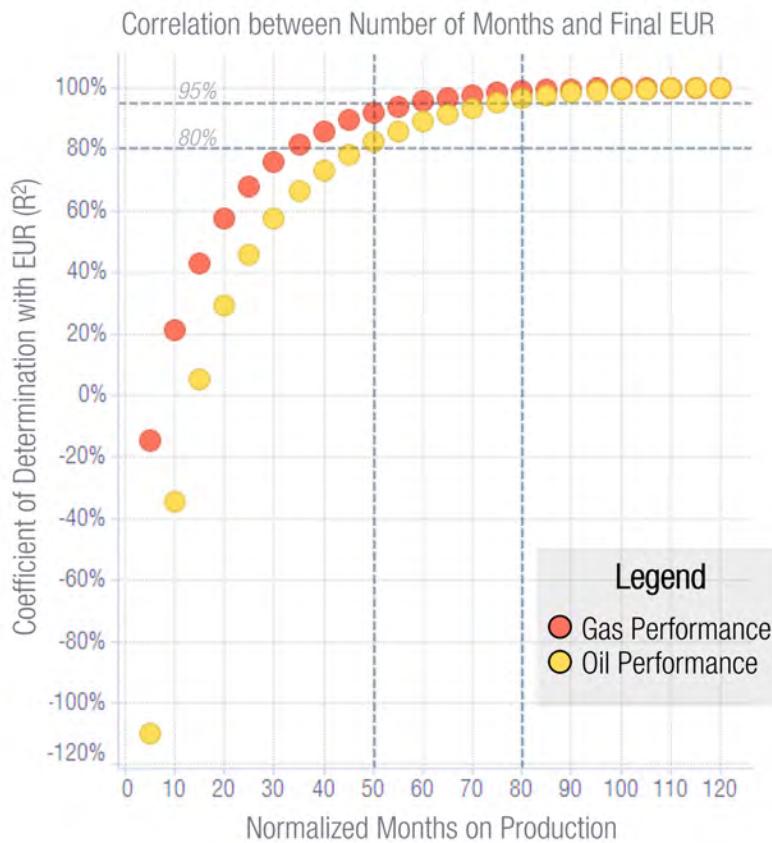


Figure 4-10: Calculating the correlation between normalized production months and EUR indicates that 80 months was ideal to have a 95%  $R^2$ . If data are limited, 48 months should provide a reasonable approximation with an 80% score.

### Oil Performance with the BNN

With the BNN selected as the preferred methodology, the same workflow was applied, substituting oil for gas as the target variable. Oil performance closely matched gas with an OSR<sup>2</sup> of 0.86 and MAPE of 5.2%. Appendix B.3 provides the full analysis conducted. Next Figure 4-10 was generated by calculating the correlation between the cumulative volume at various time steps to the well's EUR (production at month 120). 80 and 48 months were necessary to obtain an  $R^2$  of 95% and 80%, respectively. Recall that Figure 4-2 demonstrated how the MAPE increased based on forecast length. Consequently, these months will be used as a proxy for EUR on the Type Well workflow, limiting the potential amount of error introduced in the model.

#### 4.1.4 Existing Well Forecasting Summary and Conclusions

[Summary 4.1.1](#) provides a collection of key observations for the existing well forecasting process.

##### Summary 4.1.1: Existing Well Forecast Results

- Each probabilistic method improves upon DCA in terms of average error after 25 months of forecast. BNN had the strongest performance with 10% to 35% improvement in MAPE, which increased in significance over time.
- Ensemble methods provided a wider uncertainty range than BNN. This spread caused a larger percentage of wells to fall within the P10–P90 bounds.
- An ideal methodology provides the smallest range possible, while still capturing the most-likely degree of uncertainty; providing unrealistic outcomes will decrease utility and lose analyst trust.
- BNN can provide an accurate P10 forecast without inflating the uncertainty range; however, this method underestimates the P90 compared to actuals.
- BNN provided the most accurate curves reflecting the natural profile of the wells based on angular similarity. This property also extended to EUR.
- Up to 7% of wells potentially deviate from expectations and exhibit rapid increases in production late in time.
- Ensemble methods are not recommended for probabilistic EUR estimates due to significant forecasting instability and divergence.
- Overall, the BNN methodology is the best approach to generating a probabilistic forecast for both short-term and long-term (EUR) predictions.
- Normalized production at 80 months provides a reasonable indication of EUR, while 48 months should be sufficient for limited data.
- The BNN workflow can be applied to oil with only 1% reduced MAPE.

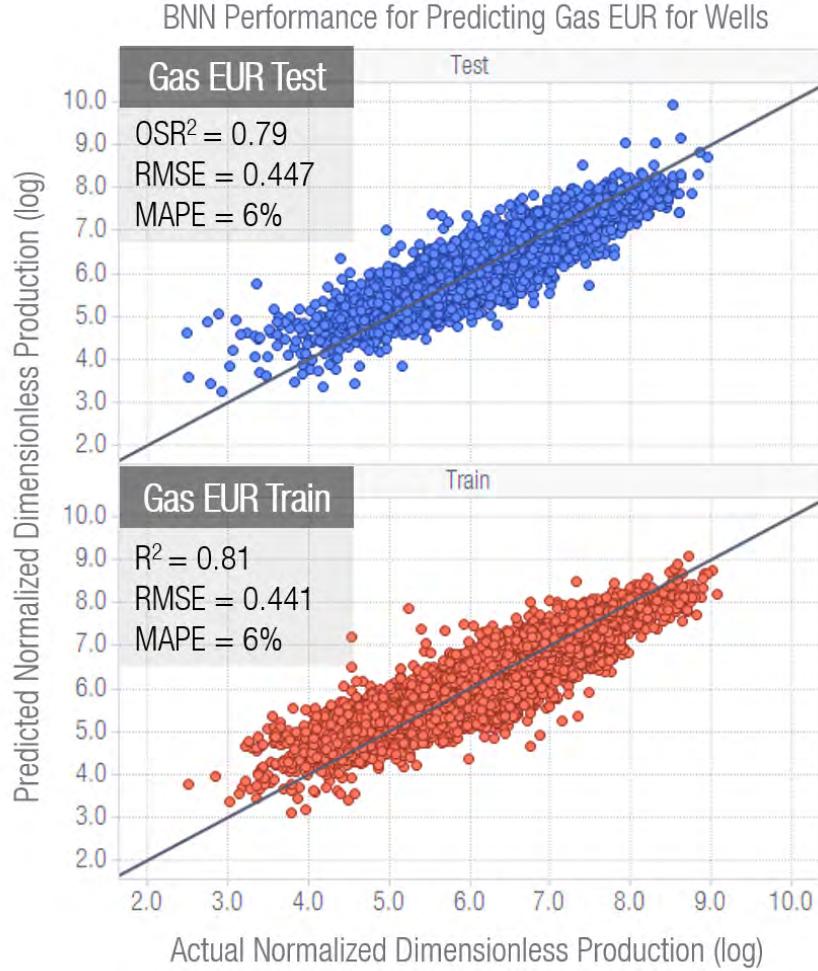


Figure 4-11: BNN Model performance at predicting well Gas EUR in train and test datasets. MAPE results were consistent with the Existing Well forecast, dropping by only 1%, despite the removal of dynamic lags.

## 4.2 Type Well Results

Next, the Type Well process can be evaluated by applying the BNN methodology to generate gas and oil predictions at month 80; this volume serves as the target variable for a model trained on the methodology from [Section 3.5](#). Notably, each well will be predicted on input features with dynamic lags removed and simple geologic variables populated. Once acceptable results are achieved, the model will be applied to grid cells to predict reservoir performance. Figures [4-11](#) and [4-12](#) demonstrate the results of this process for the gas and oil forecasts, respectively.

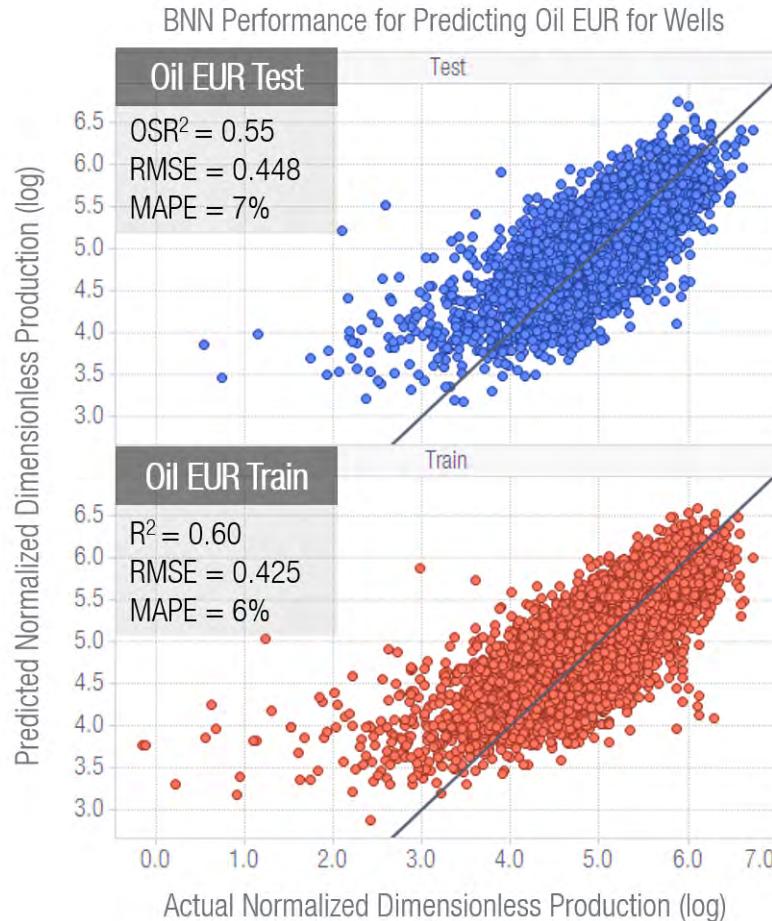


Figure 4-12: Validation plot of oil performance on the training and testing datasets. OSR<sup>2</sup> dropped significantly compared to gas (30%); however, this effect is likely caused by a more limited input range as discussed in [Section 3.1](#). MAPE stayed relatively consistent decreasing by 1%. The model seems to be skewed to over-predict oil production in low-production areas.

Overall, both the gas and oil predictions were reasonable, with approximately a 1% loss in MAPE. The OSR<sup>2</sup> results showed a more considerable decrease with oil (though from [Section 3.1](#), this metric was deemed to have secondary importance), and the primary indicators of MAPE and RMSE still maintained acceptable performance.

Now that results have been validated on replicating well performance, they can be applied to generic grid cells, assuming standardized engineering parameters. Figures [4-13](#) and [4-14](#) provide an indication of the expected mean result for each grid cell. However, there is a probabilistic range developed for each section as shown in

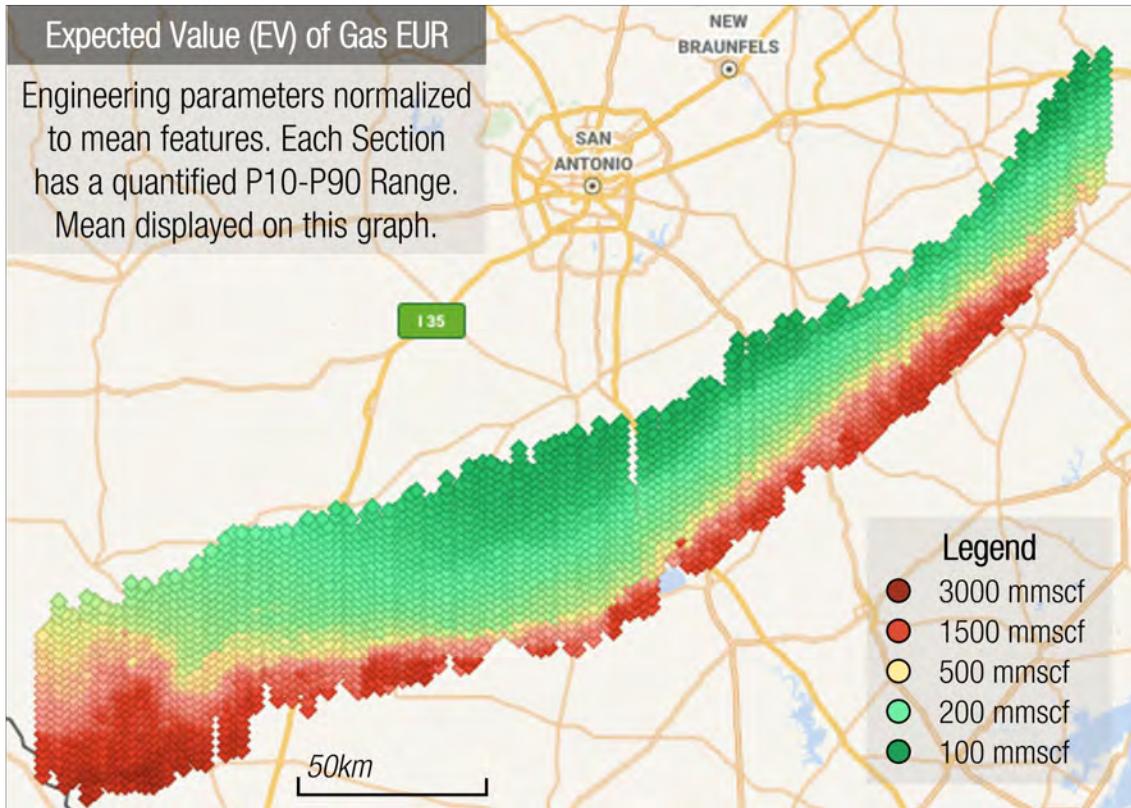


Figure 4-13: Gas Hotspot map illustrating the best performing areas in red and the lowest in green, with values in-between following a color gradient. Each grid cell corresponds with a  $2 \times 2 \text{ km}^2$  section for the mean (EV) EUR. Gas results showed smooth transitions progressing from high recovery to low, following the approximate inclination of the reservoir imputed from [Section 3.5](#). The corresponding maps for the predicted P10 and P90 EURs are provided in the [Appendix B.4](#).

[Appendix B.4](#). The gas results showed a smooth progression from maximum EUR in the southeast portion of the reservoir towards lower recovery in the northwest. This observation aligns with the expected CGR progression from dry gas to black oil.

Applying the same process to oil generates [Figure 4-14](#). This map shows the same general trend as [Figure 4-13](#) but has more localized regions compared to the smooth gas gradients. Note that the colors are reversed for maximum and minimum EUR, reflecting the convention of green for oil and red for gas.

The predictions were then clustered using GMM. Gas and oil probabilistic results were combined with the spatial input features to identify regions of similar performance. These areas can be interpreted to create polygon boundaries separating the

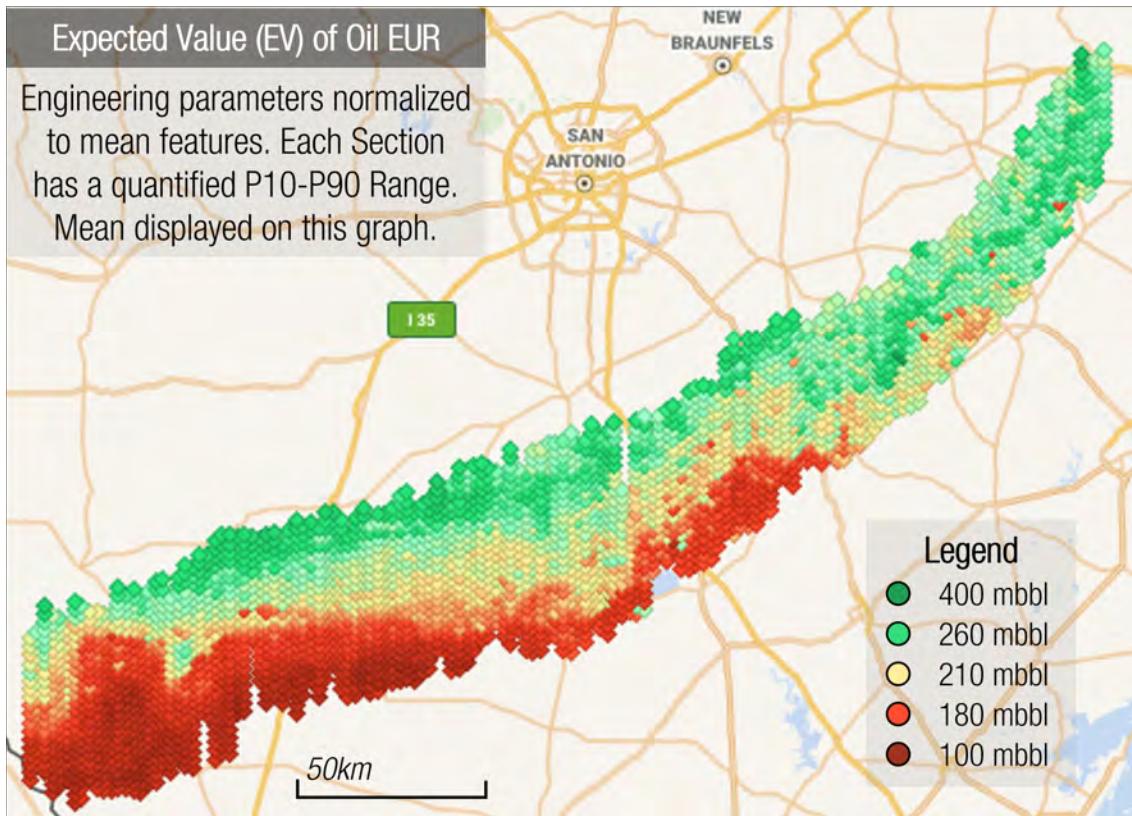
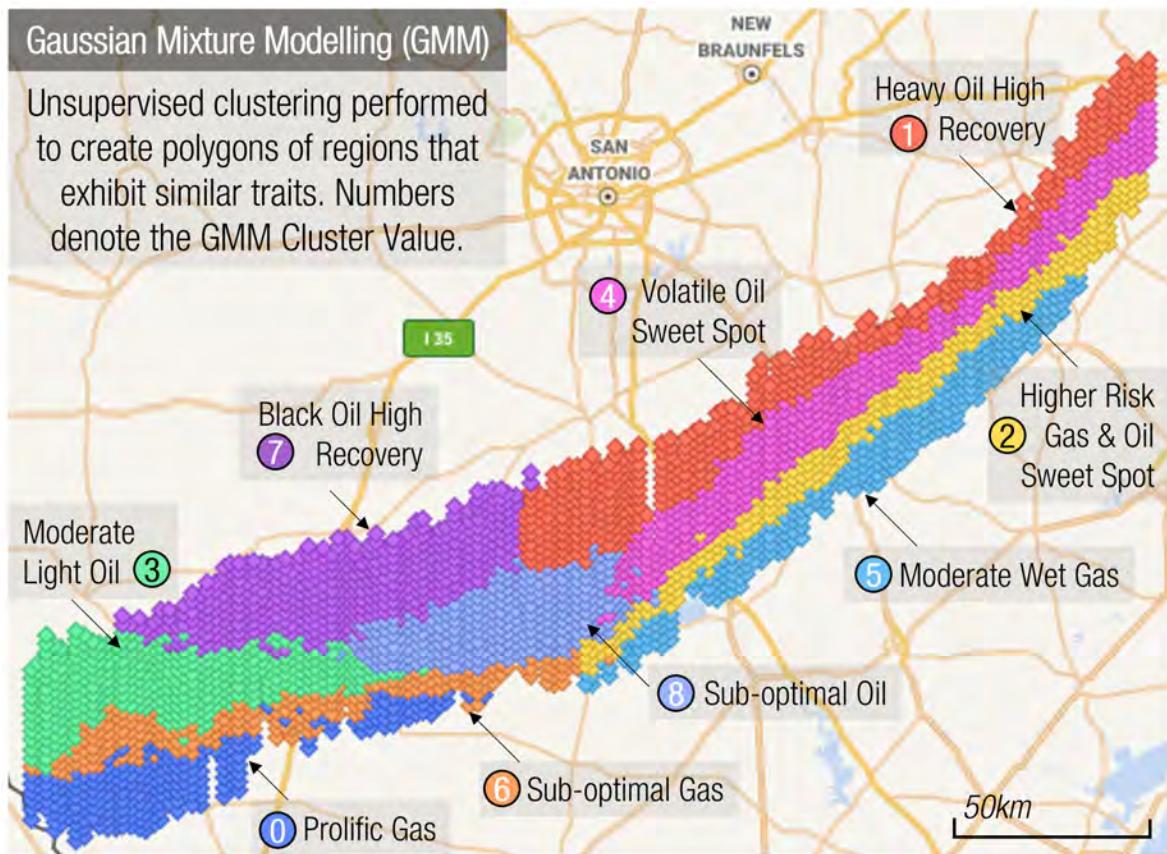


Figure 4-14: Oil Hotspot map illustrating the best performing areas in green and the lowest in red, with values in-between following a color gradient. Oil results were more volatile than gas, with rapid transitions between high- and low-production regions.

reservoir into discrete areas. A plot of the results is shown in Figure 4-15. Overall, gas exhibited similar trends across each probability range, with P90/P10 ratios consistently around 3. However, oil had a higher degree of variance, showing regions with relatively more upside (P90) or downside (P10) risk.

#### 4.2.1 Restricted Dataset Forecasting

Previous results have indicated how effective Probabilistic ML techniques could perform under favourable conditions with large well counts; however, in practice, this situation may not be possible for fields in an early stage of development. Consequently, a pragmatic question is how robust is the BNN algorithm with limited data. This question can be answered by artificially reducing the dataset size; specifically, any values after a chosen date are omitted. The date of December 31, 2015 was



cluster index	pred oil gravity	Gas (mmscf)				Oil (mbbl)			
		P10	EV (Mean)	P90	P90/P10	P10	EV (Mean)	P90	P90/P10
0	61.9	1021.3	1736.5	2950.8	2.9	57.3	124.1	268.0	4.7
1	36.7	64.6	122.3	231.9	3.6	154.1	252.4	413.7	2.7
2	51.7	297.9	526.0	930.7	3.1	126.1	227.4	412.1	3.3
3	50.2	294.6	524.3	935.7	3.2	102.2	196.1	377.2	3.7
4	43.9	136.8	248.3	451.1	3.3	138.7	237.2	405.4	2.9
5	59.8	704.4	1214.0	2092.7	3.0	102.4	199.7	389.3	3.8
6	58.4	572.5	986.4	1704.5	3.0	68.9	140.9	289.5	4.2
7	39.5	105.7	196.1	364.4	3.4	147.7	253.7	436.6	3.0
8	44.0	150.0	266.5	473.1	3.2	118.3	206.5	361.2	3.1

Figure 4-15: GMM Clustering Results dividing the Eagle Ford into regions denoted by Cluster Index. Each cluster has been labeled with a description, annotating the best and worst areas. Gas had consistent results across each range, while oil varied with areas with higher upside and downside risks seen in the fluctuating P90/P10 ratios.

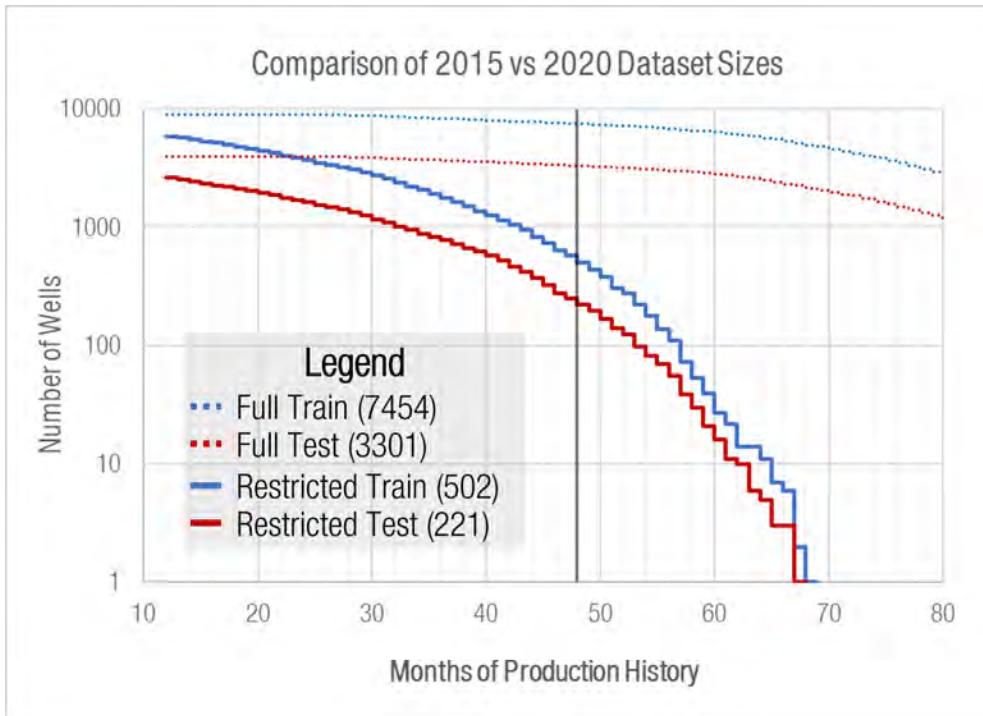


Figure 4-16: Plot of the full and restricted dataset size. The wells designated into training and testing remained consistent between these two analyses, maintaining a 70/30 split. At 13 months of production history, there were 5713 training wells compared to 9047 in the full dataset. By 48 months, this differential increased to be 7% of the 2020 data points. This gap raised the importance of the existing well forecasting to provide an accurate cumulative volume for the EUR analysis. Assuming an EUR based on month 80 is no longer possible, as no well in the restricted dataset has this production history length.

assumed as a cutoff for the restricted dataset, with Figure 4-16 demonstrating the reduction in well count. Crucially, **no wells** in the limited dataset produced **longer than 68 months**. Recall from Summary 1.4.2 that the longest producing well serves as the upper limit for the Existing Well forecast to avoid extrapolation errors. Therefore, the assumption of 80 months used in Section 4.2 is no longer plausible for this dataset. However, from Figure 4-10, **48 months** of production were sufficient to have a reasonable correlation with EUR ( $R^2 = 80$ ) and was chosen as the **target variable** for the restricted dataset. With this new cutoff, there was a 93% reduction for data available at 48 months, resulting in 502 training wells compared to 7454. Note that if a 90% correlation with EUR was desired, 60 production months are necessary; this choice would reduce the dataset to 27 wells and become impossible to train the BNN

Table 4.3: Summary of forecasting results for the Existing Well and Type Well methodologies using a BNN.

<b>Forecast Type</b>	<b>Dataset</b>	<b>Fluid</b>	<b>MAPE</b>	<b>OSR<sup>2</sup></b>	<b>RMSE</b>
Existing Well	Test	Gas	13%	0.73	0.12
		Oil	11%	0.78	0.08
Type Well (EUR)	Test	Gas	7%	0.78	0.49
		Oil	9%	0.54	0.41

adequately.

Following the workflow outlined in [Section 3.4](#), new models were generated using the restricted datasets. Since there are few wells with production at 48 months, significant weight was placed on the Existing Well forecasts to provide sufficiently accurate predictions, and enable the models to approximate the full dataset. [Table 4.3](#) summarizes the performance results of the modelling process. Note that the validation plots are provided in [Appendix B.5](#). Additionally, the Pearson Correlation between the full and restricted datasets was provided using [Equation 4.3](#), where the dataset pairs are represented by  $(x_i, y_i)$  and sample-mean values by  $\bar{x}$  and  $\bar{y}$  [82]:

$$\text{Correlation} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}}. \quad (4.3)$$

This equation measures the similarity between the predicted EUR on the reservoir grid cells, assuming a linear correlation between the full and restricted datasets. The **full dataset** is treated as the **ground truth** or the best performance reasonably expected for training on limited data. Therefore, the correlation between the restricted and full dataset predictions provides a rough approximation of how strongly the algorithm was able to identify the best relative performing regions of the reservoir. This approach ensures that the areal extent and number of predicted data points remain constant despite the considerable reduction in training data associated with the restricted dataset. A crossplot of the results are shown in [Figure 4-17](#). The restricted dataset had good results with a Pearson Correlation of 0.91 and 0.71 for gas and oil, respectively. Finally, the spatial map results comparing the full and restricted

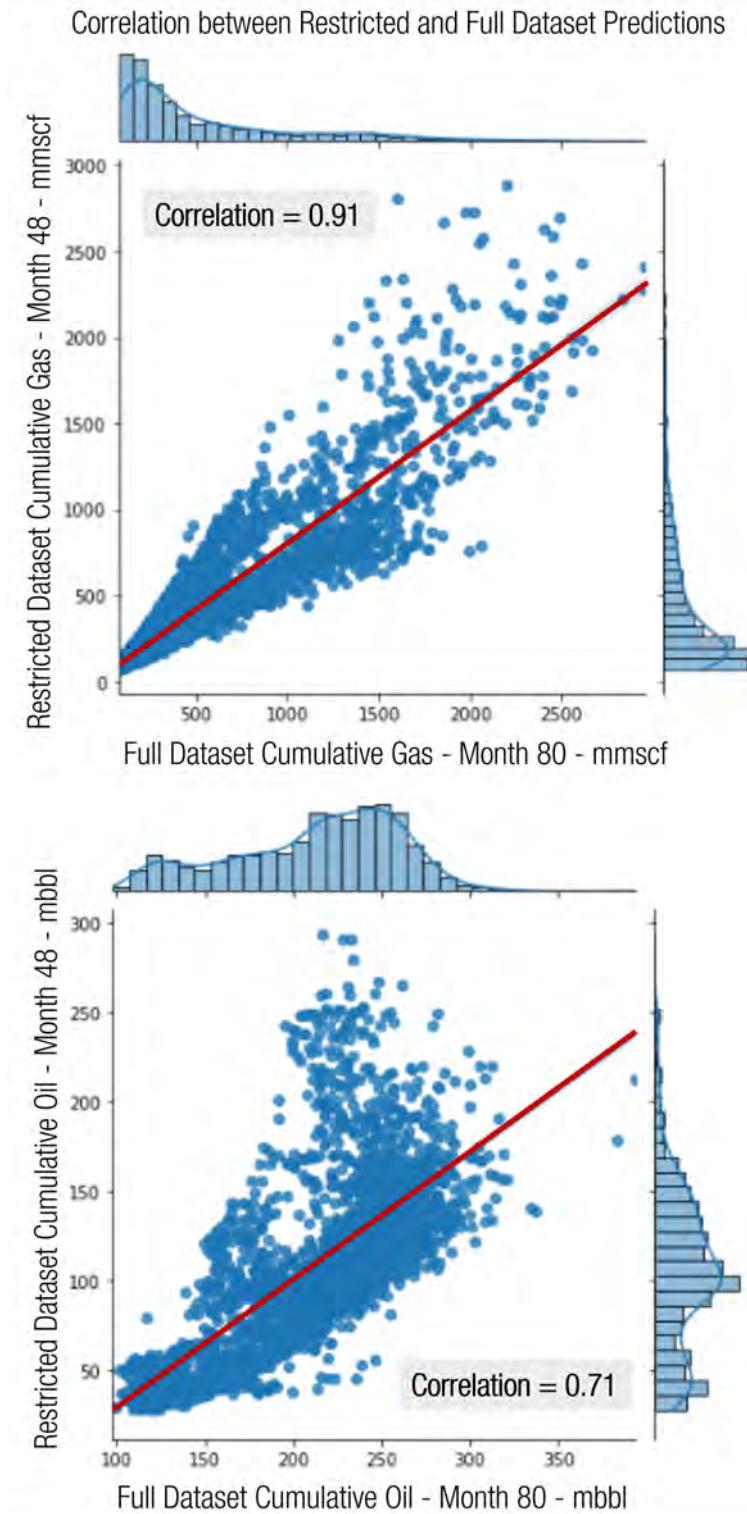


Figure 4-17: Predicted fluid volumes on the reservoir grid cells are compared in the full and restricted dataset. Note that the full dataset was predicting month 80, while the restricted was month 48. The diagonal line represents a best fit linear regression. Gas had a 20% higher correlation than oil indicating it was less negatively impacted by the reduced dataset size.

Table 4.4: Summary of the probabilistic results for the full and restricted datasets by GMM cluster. Shading indicates the relative ranking of values within each category, with green being the highest and red the lowest. Nearly the same clusters emerged for gas, which closely matched relative performance. Oil had more variation but was adequately aligned with the full dataset.

cluster index	Full Dataset - Month 80 - Gas (mmscf)				cluster index	Restricted Dataset - Month 48 - Gas (mmscf)			
	P10	EV (Mean)	P90	P90/P10		P10	EV (Mean)	P90	P90/P10
0	1021.3	1736.5	2950.8	2.9	5	860.2	1588.0	2934.5	3.4
1	64.6	122.3	231.9	3.6	4	56.1	110.6	218.0	3.9
2	297.9	526.0	930.7	3.1	2	359.5	674.8	1264.3	3.5
3	294.6	524.3	935.7	3.2	8	279.4	547.1	1073.1	3.8
4	136.8	248.3	451.1	3.3	7	115.1	220.4	422.2	3.7
5	704.4	1214.0	2092.7	3.0	6	287.9	522.8	953.1	3.3
6	572.5	986.4	1704.5	3.0	0	368.7	672.9	1230.2	3.3
7	105.7	196.1	364.4	3.4	3	85.1	169.0	335.8	3.9
8	150.0	266.5	473.1	3.2	1	109.3	206.9	391.2	3.6
cluster index	Full Dataset - Month 80 - Oil (mbbl)				cluster index	Restricted Dataset - Month 48 - Oil (mbbl)			
	P10	EV (Mean)	P90	P90/P10		P10	EV (Mean)	P90	P90/P10
0	57.3	124.1	268.0	4.7	5	18.6	36.5	71.5	3.8
1	154.1	252.4	413.7	2.7	4	71.6	131.3	240.9	3.4
2	126.1	227.4	412.1	3.3	2	104.3	201.8	390.6	3.7
3	102.2	196.1	377.2	3.7	8	46.2	86.6	162.4	3.5
4	138.7	237.2	405.4	2.9	7	69.2	128.5	239.0	3.5
5	102.4	199.7	389.3	3.8	6	56.7	109.7	212.5	3.7
6	68.9	140.9	289.5	4.2	0	25.9	49.9	96.1	3.7
7	147.7	253.7	436.6	3.0	3	72.6	132.4	241.7	3.3
8	118.3	206.5	361.2	3.1	1	51.5	95.1	175.4	3.4

datasets are shown in Figures 4-18 to 4-20.

Table 4.4 matches the closest polygon regions between the restricted and full datasets. The magnitude of the values between these datasets should be ignored as they are predicting different times (month 48 versus 80); instead, the critical metric is the **relative performance between regions**, as indicated by the **cell shading**. Note that the restricted model was free to select the number and placement of clusters based on the maximum point of curvature on the BIC plot (see Section 2.2.1). For gas, nearly the same relative performance between clusters was obtained. Oil exhibited more significant variance than gas, but the majority of polygons aligned appropriately with the full dataset; the most substantial differentiating factor was the refinement of clusters 2 and 7, as shown in Figure 4-20. The restricted dataset overestimated the oil recovery in the southern portions of this zone. In conclusion, taking into consideration the significant decrease in available data, the BNN method appears to

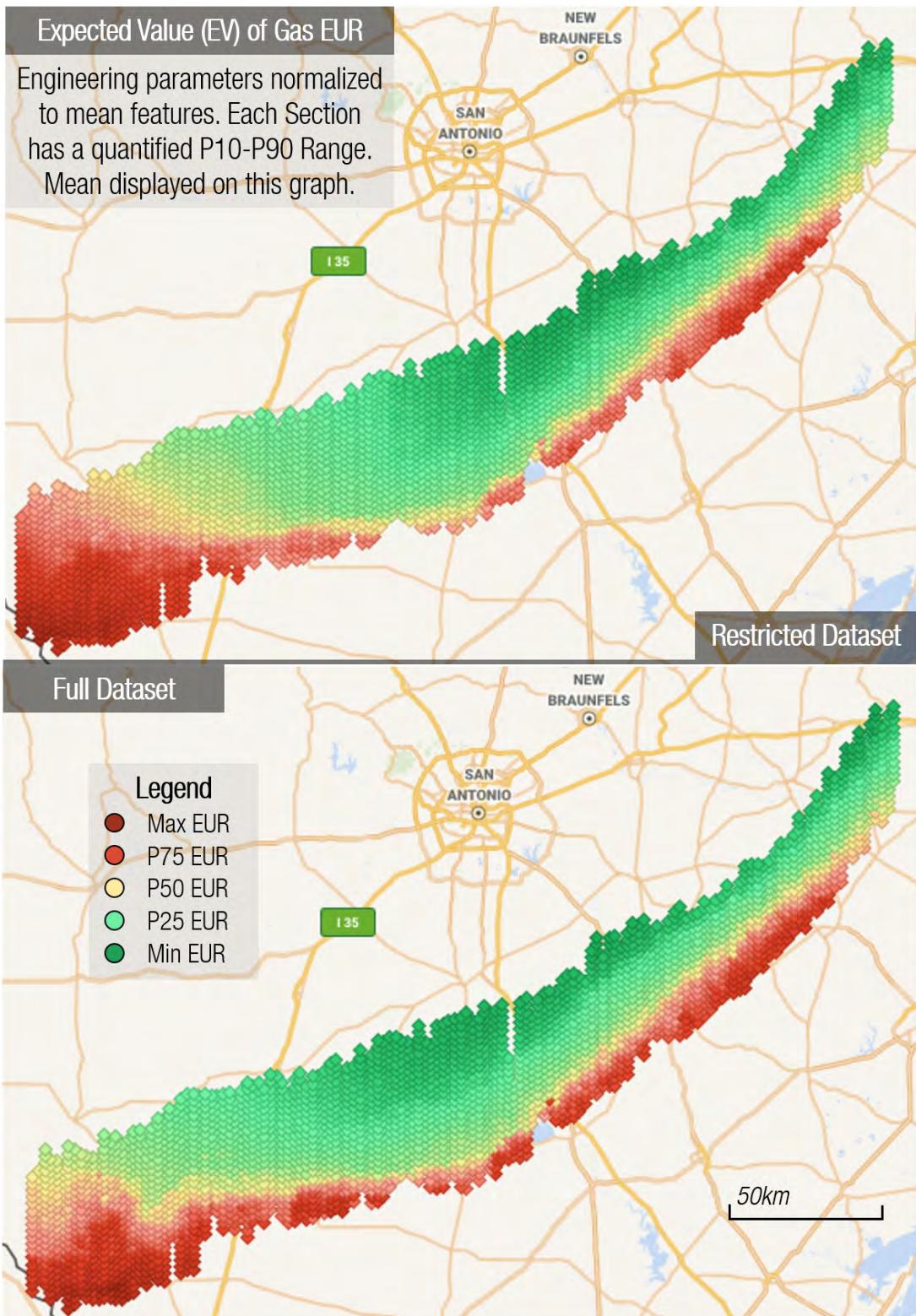


Figure 4-18: Gas spatial results have close alignment between datasets, with similar regions identified for maximum and minimum EUR. The main weaknesses of the restricted dataset are the EUR nuances in the center and southwest areas.

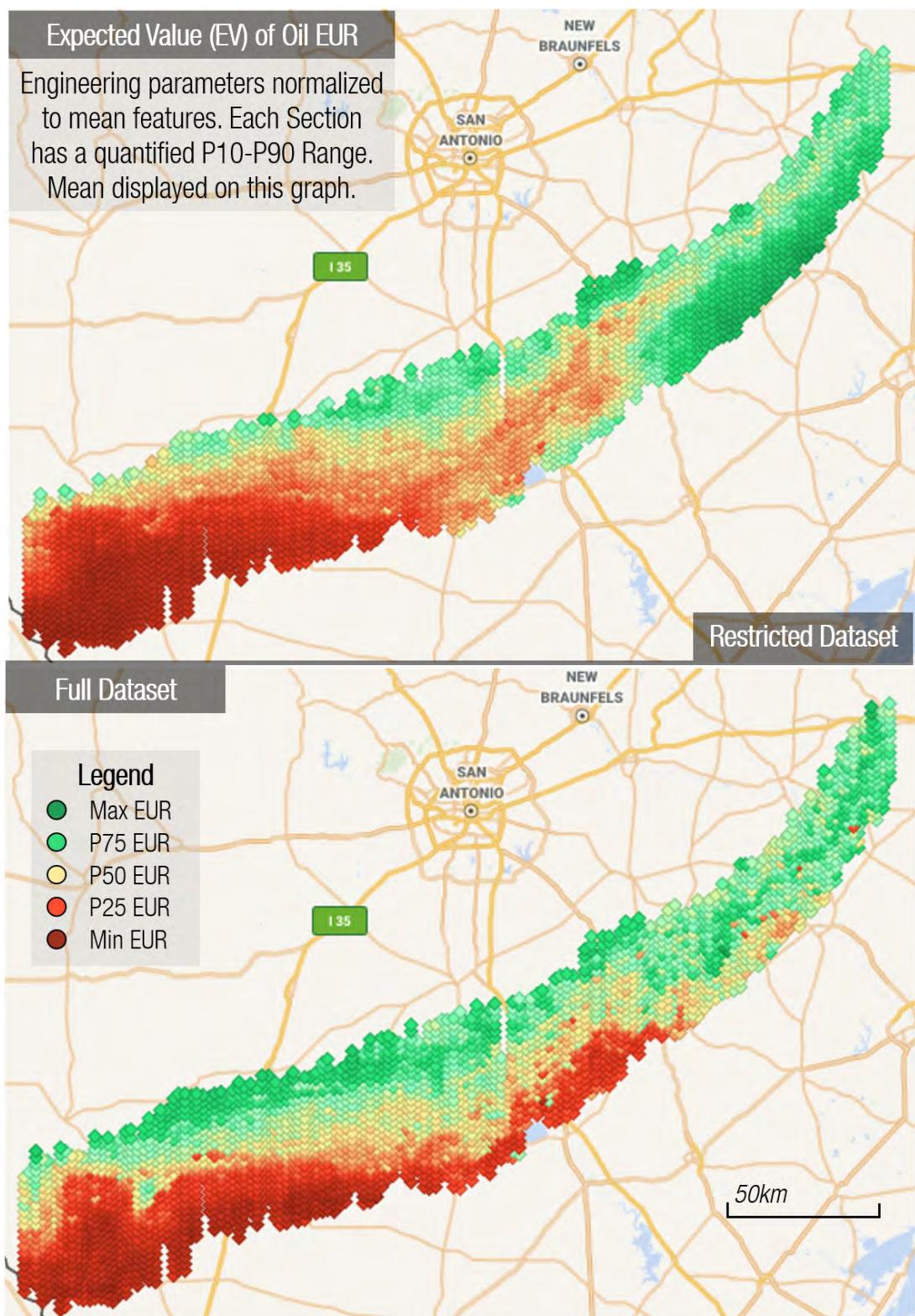


Figure 4-19: Oil spatial results were reasonable between the two maps, though this fluid had larger variances compared to gas. The maximum and minimum EUR occurred in similar regions; however, the restricted dataset overestimated production in the northeast.

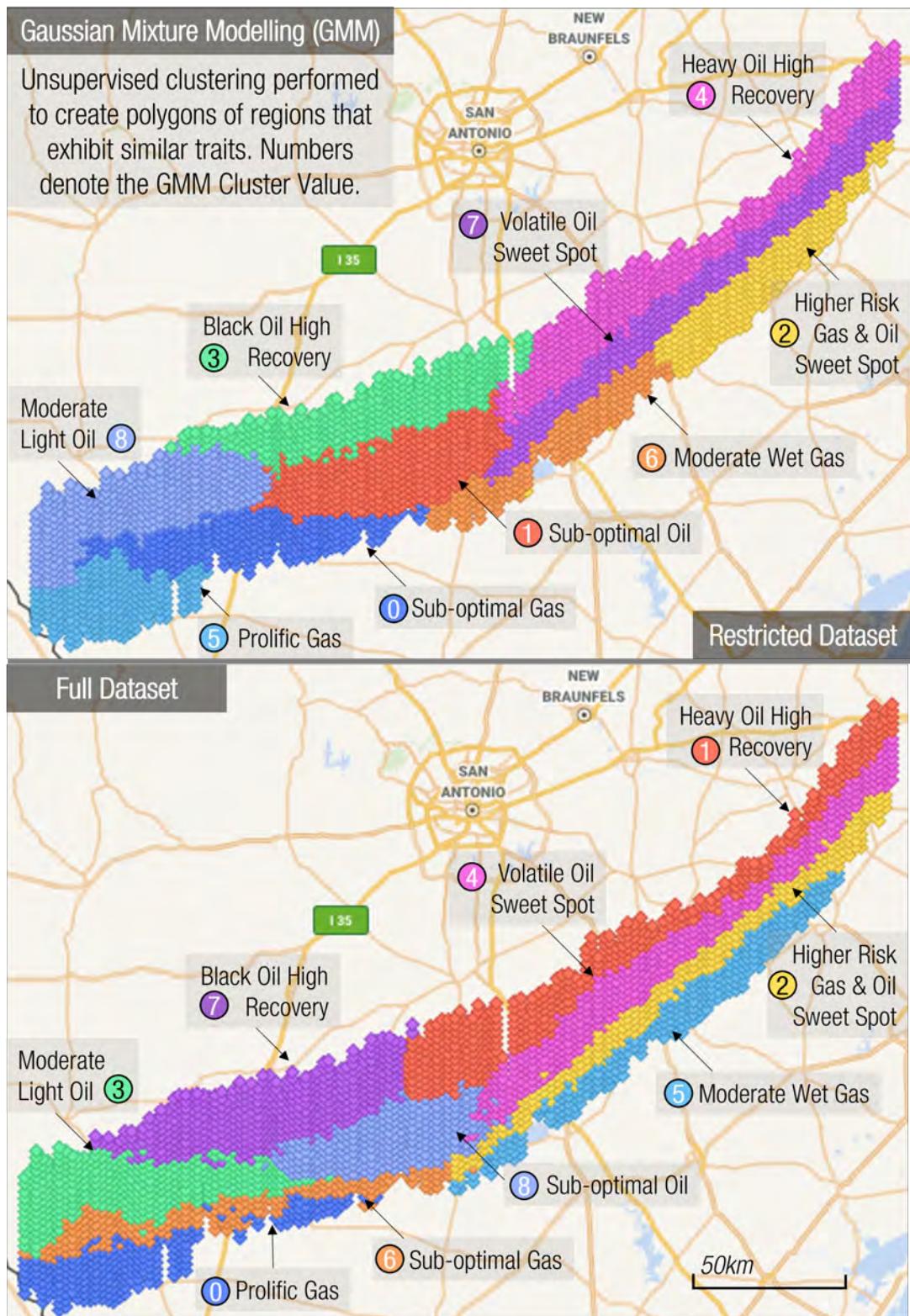


Figure 4-20: GMM Cluster Results showed similar results between datasets with the same number of regions identified. The full dataset provided additional refinement of the boundaries between clusters. Additionally, the location of cluster 2 is narrower compared to the restricted dataset, indicating an overestimation of the sweet spot.

be **robust** at estimating fields with **limited production**.

#### 4.2.2 Type Well Forecasting Summary and Conclusions

[Summary 4.2.1](#) provides a summation of the observations for the Type Well forecasting process.

##### Summary 4.2.1: Type Well Forecast Results

- Applying the BNN to forecast production at Month 80 had similar performance as the Existing Well workflow without the inclusion of dynamic lags. Both gas and oil performed reasonably well with only a 1% reduction in MAPE.
- Oil was less predictable than gas, inflating the recoveries in lower productivity areas and a degraded OSR<sup>2</sup> score.
- When applied to grid cells, gas had a smooth transition between high and low recovery areas. These regions had consistent results across P10, EV, and P90 probabilistic ranges.
- The oil hot spot map had more differences than that for gas, with localized regions and greater spread in P90/P10 ratios. This result suggests there are areas with higher degrees of upside or downside risk.
- The dataset was restricted to values before 2016 to test the algorithm robustness, resulting in a 93% reduction in wells at 48 months.
- Gas prediction accuracy was less negatively impacted by restricting data, exhibiting a 0.91 correlation with the prediction that used the full dataset on the reservoir grid cells, compared to 0.71 for oil.
- Performing GMM clustering on the restricted dataset still identified the gas and oil sweet spots derived from the full dataset analysis. However, the oil map overestimated the area of the sweet spot.

# Chapter 5

## Discussion and Conclusions

### 5.1 Existing Well Forecasting Discussion

This section discusses the critical observations identified in Chapter 4, providing commentary on the underlying mechanisms, and suggests improvements to address limitations. These conclusions will then be framed against the research questions from [Summary 1.1.1](#), repeated for convenience in [Table 5.1](#); an indicator of  $R_i$  will be provided in the discussion, relating the observations to research question  $i = 1, 2, 3$ . First, the Existing Well forecasting outcomes will be discussed for each probabilistic ML methodology before moving to the Type Well results and the restricted dataset.

Table 5.1: Recap of Research Questions

R1	Do probabilistic machine learning techniques improve well forecasting compared to existing methodologies in terms of accuracy, ease-of-use, and risk quantification?
R2	Can the most productive regions of the reservoir be identified with limited data, defined by under a thousand wells?
R3	Are computational power and practitioner demands sufficiently approachable to create a sustainable and repeatable workflow?

### 5.1.1 Ensemble Method Discussion

The Ensemble-based approach had improved performance over DCA after 25 months but had distinct limitations, shown in [Summary 4.1.1](#). In particular, this technique failed to match the BNN’s accuracy for EV predictions and generated divergent forecasts, resulting in unrealistic production profiles from [Section 4.1.3](#). The fundamental reason for these outcomes is the inherent tension between **forecast accuracy** and the creation of the **probabilistic range**. To understand this tension, first, the terms bias, precision, and accuracy require definition [43]. *Bias* measures the difference between the average and true results. *Precision* measures how tightly do repeated results vary amongst each other. Finally, *accuracy* combines bias and precision, measuring the closeness of results to the true value. [Figure 5-1](#) depicts these terms visually.

These concepts yield insight into the cause of the tension between uncertainty and accuracy. Suppose the probabilistic range was prioritized. This requirement would escalate the number of models and lowers precision to increase prediction spread. As demonstrated on [Figure 5-1](#), perfect accuracy can be obtained but may result in unrealistic uncertainty outcomes. In particular, the P10 forecast may decrease in cumulative volume to avoid introducing bias, which is physically impossible. Therefore, prioritizing a wide and plausible uncertainty range by reducing precision could generate prediction bias, dropping forecast accuracy. Conversely, if accuracy were prioritized, precision would be increased, narrowing the probabilistic range. The net effect could generate overconfident uncertainty estimations. Though not impossible to strike a balance, this approach would require significant tuning for each application and may be erroneous when new data are added. From [Summary 1.4.1](#), the strength of the Ensemble is utilizing simple models that have reasonable accuracy; attempting to satisfy both these needs resulted in a sub-optimal outcome.

Additionally, independent calculations between models were not possible with this approach. [Figure 3-6](#) shows that when  $t$  is incremented, the base, low, and high volumes for the next run are calculated. Therefore, if an extreme outlier is created

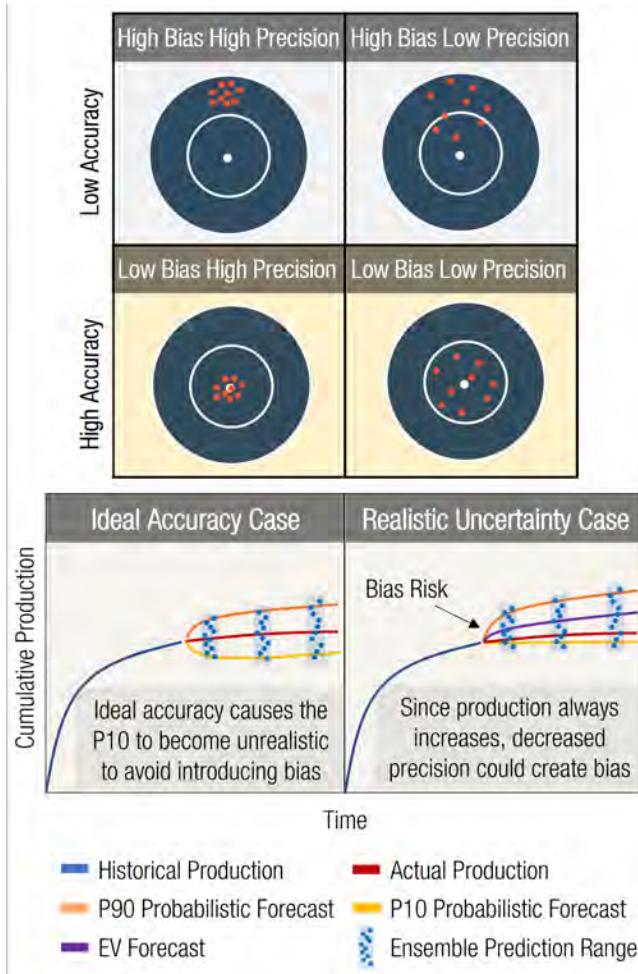


Figure 5-1: The terms bias, precision, and accuracy are depicted graphically. Tension emerges between the uncertainty range and accuracy when applied to production forecasting. Maintaining ideal accuracy could cause the P10 forecast to drop in cumulative volume but should always be monotonically increasing. Conversely, creating a realistic uncertainty range could increase the risk of bias and decrease accuracy.

in any prediction, it could shift the entire distribution. This risk is magnified as a skewed distribution affects the dynamic lags, compounding error for all subsequent predictions. Due to the numerous calculations in the Monte Carlo simulation, outliers are likely frequently encountered, making the forecast prone to divergence; this tendency causes the percentiles identified to be more likely the P1 and P99 rather than the P10 and P90. The same effect is not present in the BNN methodology, as each Monte Carlo run simulates an independent production forecast (see [Figure 3-11](#)); the percentiles are only calculated after the entire simulation is complete. The ensem-

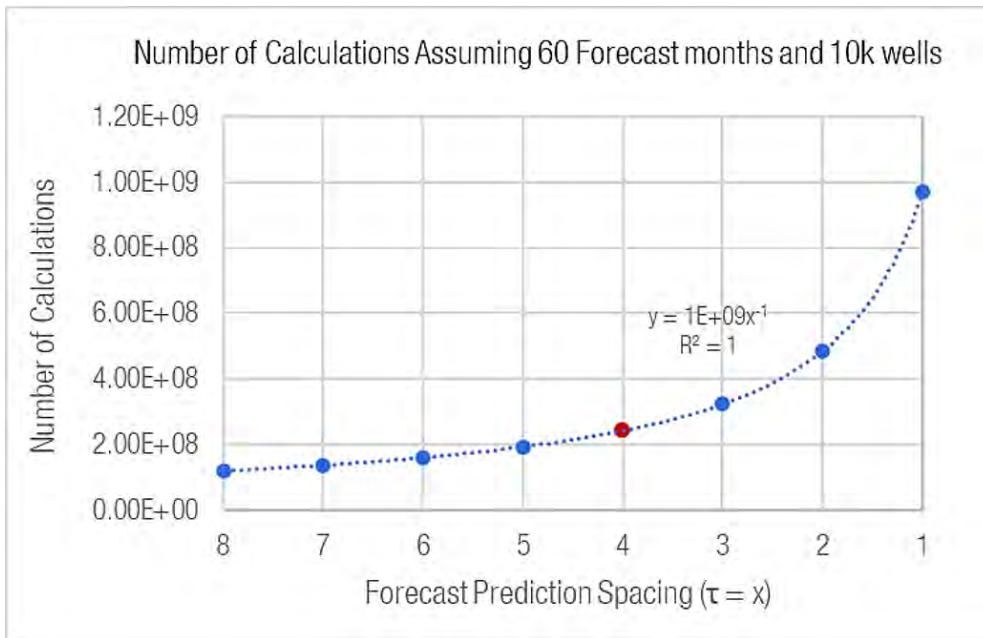


Figure 5-2: Reducing  $\tau$  from an increment of 4 to 1 substantially increases the number of models and generates an additional 800 million predictions, more than the total predictions at  $\tau = 4$ . This change can improve curve smoothness but at a substantial cost of complexity and computational time.

ble method relies on multiple models to generate uncertainty, so the same structure would not be possible.

Figure 3-6 also reveals the cause of the cyclic behaviour exhibited by the indicative well in Figure 4-6. Models are created at increments of  $\tau = 4n$ , so when  $t$  encounters the next multiple of 4, a new batch of predictions shift the distribution. This choice was made to improve workflow sustainability (R3), as the system complexity drastically increases with the number of models to handle the sheer amount of data generated. Figure 5-2 demonstrates how calculations would grow by an order of magnitude if  $\tau = 1$  were chosen, significantly increasing calculation time.

### Ensemble Improvements and Research Question Recommendations

Decoupling the tension between accuracy and the probabilistic range does not seem sustainable, as generally, an improvement in one negatively impacts the other. Therefore, it is recommended to simplify the workflow to provide an interpretable benchmark for the EV forecast of the BNN, leveraging the simplicity of the Tree-

Based methods to validate that a more complex model is working as anticipated. This conclusion aligns with (R3), as verification and validation are vital components to ensure a sustainable workflow that analysts can trust.

### 5.1.2 Bayesian Neural Network (BNN) Method Discussion

The BNN had the best overall performance of all methodologies. It had the lowest average MAPE score of 6%, which was 10-35% better than the DCA and Ensemble approaches after 25 months. The BNN forecasts for P10, EV, and P90 also had the most plausible curvature based on the angular similarity to actual production profiles. These results can be attributed to the architectural decisions for this algorithm. [Figure 3-8](#) demonstrates how the BNN was trained on the entire shape of the curve; thus, this structure captures the knowledge of the well's entire life cycle and improves angular similarity. The same design could not be made for the Ensembles, as numerous predictions are necessary to generate dissimilar forecasts, creating the probabilistic range. This effect captures the fundamental reason for improved performance, as the requirements of the model can be decoupled, unlike the Ensemble methods. [Figure 3-9](#) shows how the BNN predicts a distribution rather than a single value. Specifically, a mean and standard deviation are fit, capturing both accuracy and spread. The Ensembles in this application are trained only to fit a mean, requiring heuristics to introduce noise and approximate the uncertainty of the entire system. To put it more simply, the BNN is fit to satisfy both needs while the Ensemble satisfies only one.

Also, the structure of the BNN makes it less prone to outliers, as each Monte Carlo run is calculated independently. For example, if a P99 or P1 outcome is predicted, it is captured only within that run and does not affect all subsequent predictions like the Ensemble approach. Additionally, since distribution sampling occurs randomly, it would be rare to choose high or low percentiles consistently. Instead, most runs will have a mixture of percentiles sampled, resulting in a more balanced prediction and a fairer representation of the likely probabilistic range. Note that the choice was made to calculate percentile timeslices for each forecast month. Another approach can be taken by selecting representative P10 and P90 runs based on the final forecast

outcomes. These representative profiles can potentially provide more plausible curves, as limited numbers of data points within a time slice may not produce a statistically valid distribution [64]. However, [Table 4.2](#) demonstrates how the BNN has equal or better angular similarity than all other forecasts; this result is caused by the 1000 Monte Carlo simulations, which smooth out forecast noise, as shown in [Figure 3-11](#). Therefore, this time-slice approach seems appropriate based on the realistic curvatures and reduced complexity compared to identifying representative runs.

From a systems perspective, these characteristics make the BNN better suited at handling the defined problem; however, the BNN still has some limitations, particularly for the breadth of the gas and oil probabilistic ranges. [Figure 4-4](#) depicts how the BNN represents the P10 component of the gas forecast accurately while being conservative for the P90 prediction. The explanation behind the success of the P10 range can be attributed to [Figure 3-9](#). The usage of the softplus activation function ensured that volumes should be typically greater than or equal to zero, avoiding the negative production profiles observed in [Figure 4-6](#). This approach also allows the model to predict small changes in cumulative volumes since the activation function can generate low magnitude values and avoid a zero gradient, which turns off neurons [56, 75]. To understand the P90 tendencies, evaluating the distribution of the residuals offers clues. [Figure 5-3](#) plots the shape of the residuals at two different months of normalized production. Generally, the residuals fit a Cauchy distribution with a relatively narrow spread of values. This distribution is provided only for visual reference, with the key takeaway being approximate symmetry about the origin.

These observations hold until month 50, where a change in the distribution shape occurs. Four years into production is relatively late into a well's life cycle, as [Figure 4-10](#) demonstrated an  $R^2$  of 0.8 with EUR. What could be the cause of this effect? One plausible hypothesis is that month 50 is around the time of sizable interventions or engineering projects. An emerging trend for shale reservoirs is refracturing existing wells [58]. Instead of drilling a new well, an old asset can undergo a new fracture stimulation, resulting in an increase in EUR at a reduced cost. Also, as described in [Section 1.2.4](#), a field will progress from low well counts in exploration and appraisal

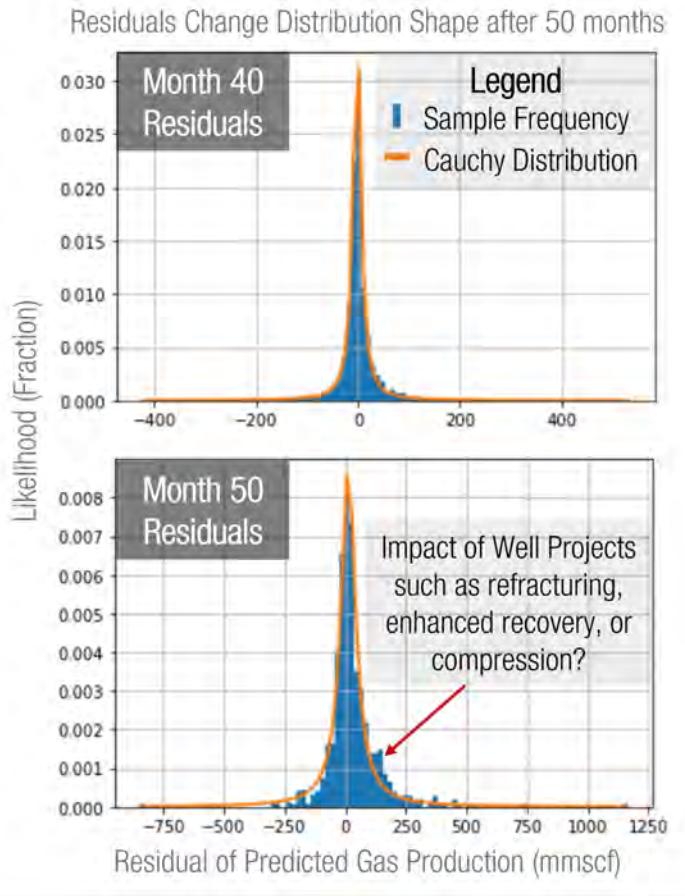


Figure 5-3: Plotting the residuals at different normalized months reveals a change to the distribution shape emerging after 50 months of production. Could operational or well intervention projects be responsible for this effect?

before large-scale development of the region. When there are enough wells, large-scale infrastructure projects can be implemented, as the capital burden spreads across the network. This effect allows numerous projects to become economical, such as compression to drop pipeline pressures or cyclic gas injection to improve hydrocarbon recovery (see [Appendix B-12](#) for a well exhibiting this behaviour) [61]. Therefore, the distribution shape change occurring at 50 months could coincide with a higher probability of one or more of these projects being implemented. However, these external events are beyond the algorithm's scope and should not be considered when predicting well production based on its initial conditions. If this hypothesis is valid, it would partially explain underestimating the P90 forecast. The residual in [Figure 4-4](#) represents the 90<sup>th</sup> percentile; thus, it captures an **extreme of the distribution**

with only 10% of wells above. If external events are happening on 7% of producers from [Figure B-13](#), this metric would be heavily influenced by this factor.

Another explanation is required to explain the trend of the average actual production falling between the P90 and EV. A more ubiquitous phenomenon must be affecting the well population to have an impact on the average. An additional hypothesis can be formed when combining the gas and oil probabilistic results. [Figure B-24](#) illustrates how the oil residuals follow the opposite pattern of gas and are more likely to be optimistic. This result suggests that there is something related to fluid behaviour dictating this effect. The first consideration is scale, as the size of pore throats in the rock matrix can be of a similar magnitude to the gas and oil molecules, shown in [Figure 5-4](#). With these diameters, natural filtering within the reservoir occurs, as the relatively large oil molecules are less likely to find connected passages wide enough to escape the matrix [5, 19].

Second, there is considerable capillary pressure ( $P_c$ ) generated, as demonstrated by [Equation 5.1](#):

$$P_c = \frac{2\gamma \cos \theta}{r}. \quad (5.1)$$

This equation is a function of fluid interfacial tension ( $\gamma$ ), contact angle ( $\theta$ ), and pore throat radius ( $r$ ) [23]. Since  $r$  is minuscule, the magnitude of  $P_c$  can be on the order of 10,000 psi. Generating a pressure differential between the well and reservoir to overcome this value would be impossible; thus, the liquid can become immobile within portions of the matrix [19, 23].

Third, there are known fluid-behaviour changes as the reservoir pressure drops from initial conditions, demonstrated by [Figure 5-5](#).  $A$ ,  $B$ , and  $C$  denote different initial starting points, roughly equivalent to the initial oil gravity input feature. All three exist in a single state at these conditions, namely, *gas* for  $B$  and  $C$ , and *oil* for  $A$ . As fluid  $A$  is produced, the reservoir pressure depletes following a constant temperature vertical line (isothermal) until it hits the bubble point. At  $A'$ , gas forms within the reservoir and increases in concentration as pressure drops. For  $B$ , the opposite effect occurs, with oil manifesting in the reservoir when it hits the dew point

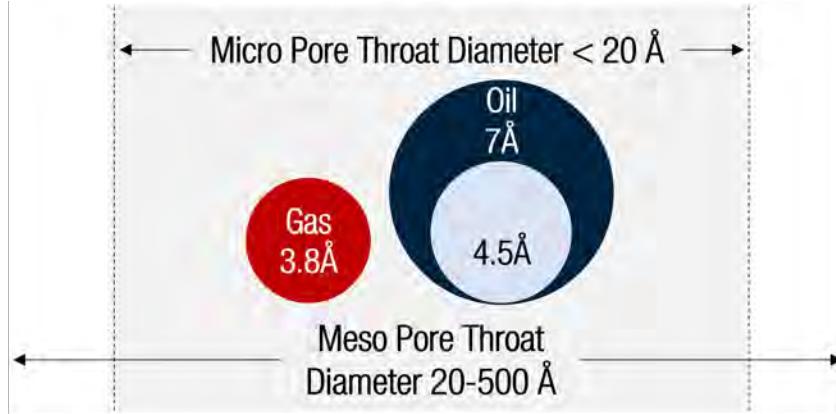


Figure 5-4: Relative diameters of gas, oil, and pore throat in Angstroms. Shale has significant variation in diameters from micro to meso pores. The large size of oil relative to gas makes it less likely to be produced through portions of the reservoir with small pores [5, 53]. These values are based on research from [32, 44, 53].

line ( $B'$ ).  $C$  is the only fluid in temperatures beyond the phase envelope and will always be in a gas phase in the reservoir. When  $A$  and  $B$  are below the saturation lines (dew point and bubble point), they operate in a 2-phase region where gas and oil coexist. Within this state, the production of one fluid impedes the flow of another; this effect is called relative permeability [71].

Each of the three foregoing effects will compound in the reservoir, creating the emergent effect of increased gas and decreased oil production over time.

## BNN Improvements and Research Question Recommendations

Several improvement opportunities exist for the BNN method. First is removing wells with late time spikes from the dataset; this change would eliminate noise generated by external projects, improving model results and yielding more conclusive insight on the extent of the phenomenon. These profiles can be identified with shape detection methods such as a convolutional neural network (CNN) or measuring cosine angle similarity to wells identified with the characteristic behaviour. Next, to limit the effects of fluid properties, several adjustments can be made. A promising scenario would be incorporating additional operating pressure and fluid properties as inputs into the workflow. These values can be used in a physics-based simulation

### Phase Envelope Examples for Various Reservoir Types

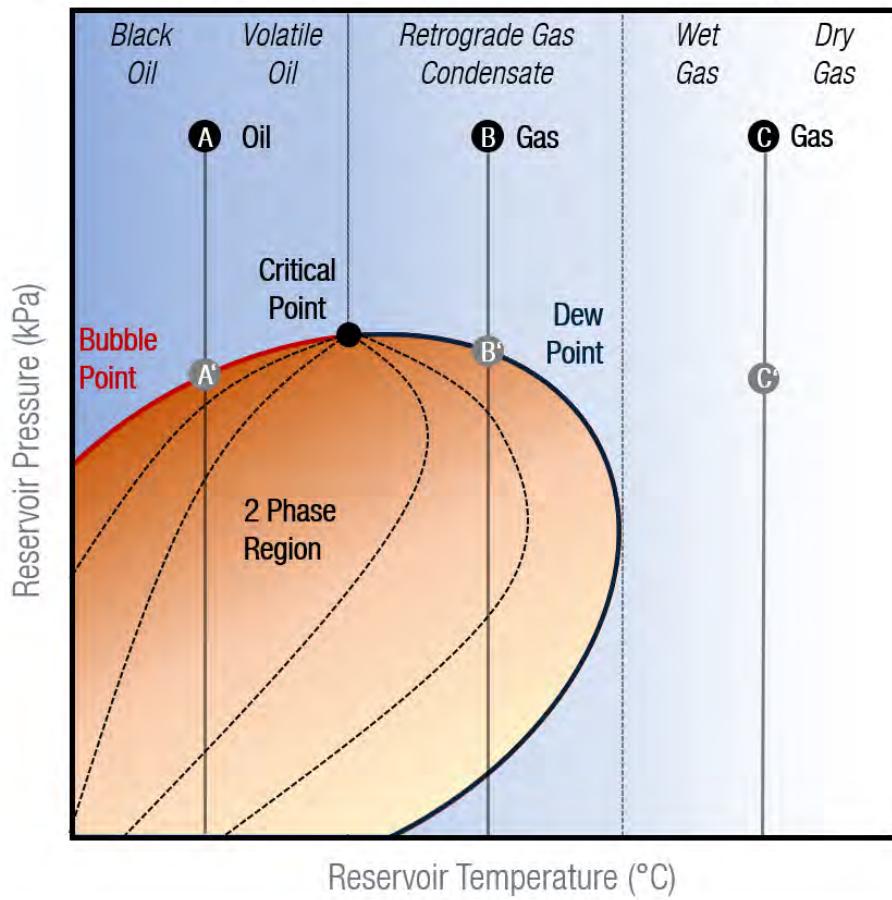


Figure 5-5: Phase envelope of different reservoir fluids. Complex flowing behaviour exists for gas condensate and volatile oil, which will likely result in a decrease in the oil volumes produced when in the 2-Phase region, based on research [71, 74].

to calculate the proportion of gas and liquid at the reservoir conditions. Finally, a Kalman filter [57] can blend the uncertainty distributions from the simulation and ML approaches as illustrated in Figure 5-6. The final distribution provides a way to determine a system state that cannot be measured directly and should generate outcomes more aligned with actuals. This approach is similar to the Fetkovic RTA methodology restricting the  $b$  curvature value so that it falls under permissible bounds (see Section 1.2.3).

However, the weakness of this approach is that the data necessary to run the simulation are unlikely to be available in large enough quantities. Another solution is

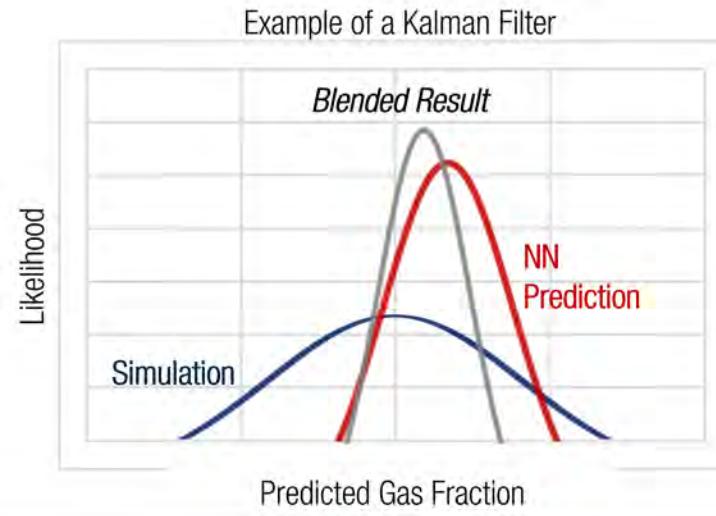


Figure 5-6: An example of a Kalman Filter applied which blends calculations from the NN prediction and simulation model. The blended result should steer results towards a realistic system state and avoid under-prediction tendencies.

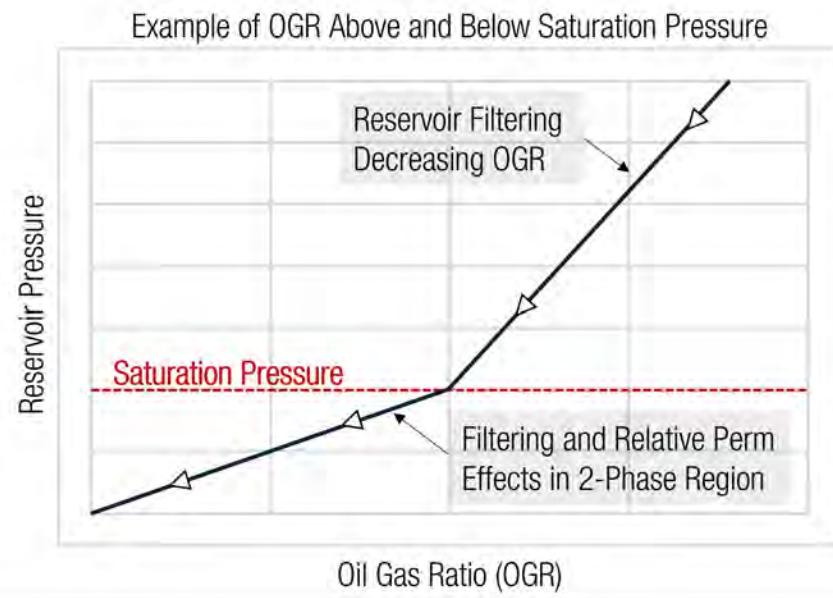


Figure 5-7: Oil Gas Ratio (OGR) as a function of Reservoir Pressure. Pressure decreases from right to left, with a cusp occurring once the reservoir hits the saturation point. The 2-phase region creates a larger change in OGR per unit of pressure due to compounding factors. Based on research [45].

to forecast gas and oil production simultaneously. These predictions allow the ratio of oil to gas (OGR) to be calculated and approximate the fluid behaviour. [Figure 5-7](#) demonstrates the expected OGR relationship with pressure. If the first derivative of this value is taken, with respect to pressure or time, there should be a detectable slope change after saturation pressure is encountered; this change can inform the model that the well is in the 2-phase regime and it should include this impact on the gas and oil predictions. These additions allow the fluid properties to be adjusted dynamically and improve the predicted probabilistic range.

When framed against the research questions, the BNN improves significant components of existing processes (R1). BNN forecast accuracy was the highest out of all methods tested, and BNN is relatively fast to train and implement (less than an hour for 10,000 wells on a personal computer). Though opportunities exist to improve the probabilistic range, the BNN enables new functionality, providing analysts with the capability of considering risk and making informed decisions. The biggest weakness of the BNN is the creation of an accessible and sustainable workflow (R3). Specifically, the complexity of the BNN theory could pose a barrier of entry, as practitioners may not want to utilize a methodology if they do not understand how it operates. However, this effect can be mitigated through several methods. A model can be designed based on subject matter expertise (SME) from a community of practice to eliminate the risk of incorrect application. These SMEs can offer a short course or mentoring to practitioners, building their competencies. Additionally, using simpler tree-based models can flag erroneous values from the EV output of the BNN and initiate appropriate intervention.

## 5.2 Type Well Forecasting Discussion

Applying the BNN to the Type Well process resulted in accurate EUR predictions for wells in the test dataset, with a MAPE of 6% for gas and 7% for oil. These results provided confidence to estimate reservoir performance by forecasting grid cells with standardized engineering parameters. The hotspot maps generated for gas exhibited

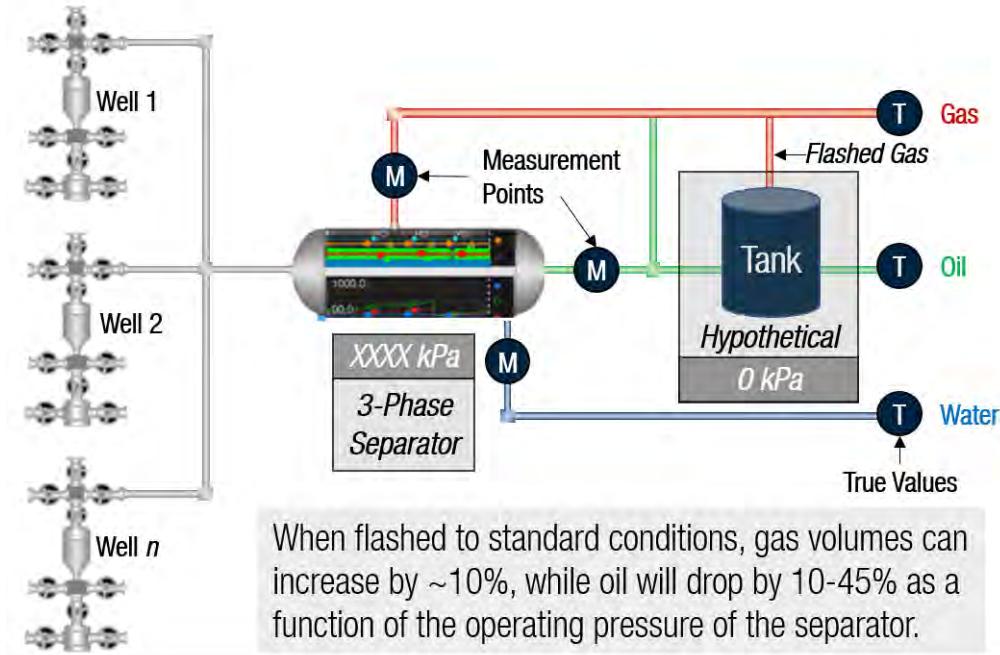


Figure 5-8: Since the separator operates at pressure, the oil stream will have a portion flash to gas when reduced to standard conditions (shrinkage factor). This conversion is often omitted and can account for a 10%–45% loss in volume.

smooth profiles while oil had increased volatility with more localized regions. These results are explainable for two primary reasons. First, there are considerably more mechanisms that can result in reduced oil recovery from the reservoir, as described by [Section 5.1.1](#) and [Summary 1.2.3](#). As a result, oil is more sensitive to geological properties; the best performing regions must have a series of favourable conditions coincide. Additionally, any heterogeneities with unfavourable properties can dominate and limit productivity for an area that is otherwise conducive to production. The net result should lead to more localized clustering, as variations in geological parameters create areas with more upside or downside potential. Second, there is more uncertainty in measuring oil production, with [Figure 5-8](#) demonstrating a typical separator layout. Here, a series of  $n$  wells flow to a central separator, which measures the group's production. These volumes are then prorated back to each entity, based on well tests. However, a common mistake is failing to convert oil volumes from the separator to standard conditions ( $25\text{ }^{\circ}\text{C}$  and  $0\text{ kPag}$ ); this process is also called shrinkage factor [\[15\]](#). Since the separators deliver volumes to plants, they typically operate between

1000–8000 kPa, depending on the network configuration. Therefore, the oil volumes measured at point  $M$  will be under high pressure and temperature. Since [Figure 5-5](#), showed that gas and liquid content varies as a function of pressure, if this fluid drops to standard conditions, it will generate additional gas and less oil. This calculation can be modelled by adding a hypothetical atmospheric tank; however, this process is an additional step that may be omitted depending on the operator. Shrinkage factor can have a considerable effect on liquid measurements, ranging from a 10% to a 45% reduction. If this effect occurs for a subset of wells, essentially random noise would be introduced into the dataset, generating more localized and heterogeneous areas.

The variation in oil production explains the more erratic P90/P10 ratio results from [Figure 4-15](#). However, these results can be evaluated using best practices from the *Unconventional Resource Assessment and Valuation* course from Rose and Associates to see if they span an appropriate range [9]. This course recommends that wells be grouped into areas following a P90/P10 ratio between 2.5 to 5. The GMM cluster results satisfy this range with gas consistently around 3, while oil varied between 2.7 and 4.7; therefore, these clusters fall within a reasonable level of variation.

Next, the shape and locations of the BNN clusters were compared to an independent study of the Eagle Ford reservoir. The BNN process utilized induction to analyze well production and predict the best portions of the reservoir. Conversely, *Deloitte Insights* published articles in 2019, which follow the exact opposite process; namely, they started from geological measurements to identify the best areas, then predicted where maximum production is expected [66]. This study utilized a geological database for the Permian, Bakken, Appalachian, and Eagle Ford shale gas reservoirs. The data contained numerous inputs unavailable for this thesis, such as formation thickness, gamma-ray, bulk density, neutron porosity, and deep resistivity. Sanderson et. al then calculated average formation quality across these plays using Equations 5.2 and 5.3 [65]:

$$Z_i = \frac{X_i - \min X}{\max X - \min X}; \quad (5.2)$$

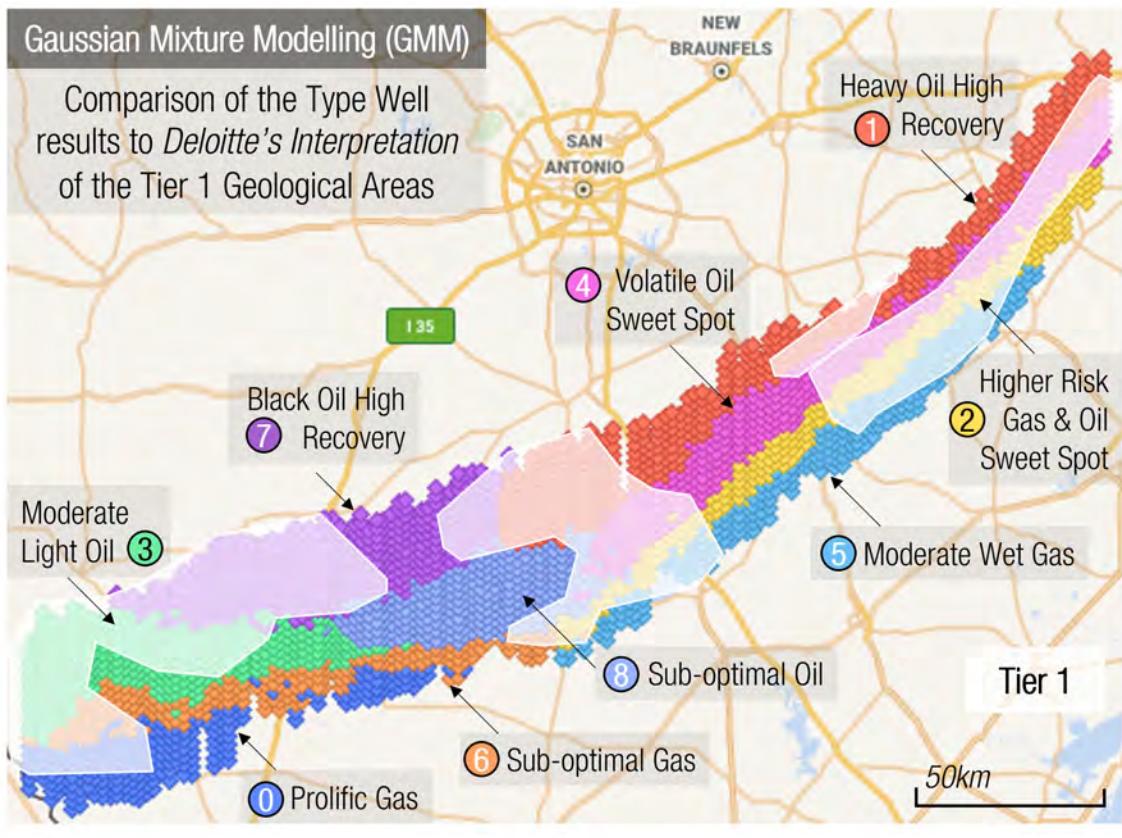


Figure 5-9: GMM Map with Tier 1 interpreted areas highlighted in white. These regions align with the best-predicted clusters of 1, 2, 3, 4, and 7. The suboptimal clusters (6 and 8) are also supported by the shape of the Tier 1 boundaries, which curve around these areas.

$$\text{Reservoir Quality Score} = \frac{1}{n} \sum_{i=1}^n Z_i = \frac{\frac{1}{n} \sum_{i=1}^n X_i - \min X}{\max X - \min X}. \quad (5.3)$$

These equations take the well's measurement for a metric ( $X_i$ ), then compares them to the maximum and minimum values observed in these four reservoirs. This process generates a relative score, averaged across all metrics ( $n$ ). The final result gives a relative ranking of the aggregate reservoir quality from the petrophysical properties.

Finally, a spatial map was generated based on categorizing the best wells into Tier 1 areas [65]. Figure 5-9 overlays the *Deloitte Insights* interpretations with the GMM map. Results show close agreement, identifying the strongest producing clusters and excluding the weakest. The largest discrepancy between maps is the extent of the Tier 1 areas. Deloitte's assessment utilized an undisclosed geospatial technique to create

clusters of data points; thus, with more sparse data this methodology created more localized zones as there was limited interpolation between wells. An improvement suggested by the authors is to utilize the data to create a reservoir model of each feature, allowing the geospatial analysis to span greater distances without well control. However, if these changes result in the same outcome, it could support adjusting the GMM methodology to select a value beyond the maximum point of curvature (see [Figure 2-8](#)). Though this alteration would not represent maximum algorithm efficiency, the best portions of the reservoir could be more clearly delineated, and provide practical benefits to field development.

## **EUR Improvements and Research Question Recommendations**

In conclusion, the BNN approach for predicting EUR across the reservoir provides numerous improvements compared to existing methodologies (R1). Instead of having discrete processes, the BNN provides a single workflow that determines normalization factors, existing well forecasts, and spatial results. This approach offers the distinct advantage of quantifying uncertainties and directly incorporating them into the P10 and P90 range. Additionally, the clusters of highest reservoir quality aligned with the independent *Deloitte Insights* study of Tier 1 areas. This observation flags a potential improvement area. Suppose these geological data can be acquired or mapped by subject matter experts. In that case, they can be included in the BNN model and improve both Type and Existing Well forecast accuracy. This combination of different techniques could also result in unique findings that could be missed by following one approach in isolation.

The BNN should also generate a sufficiently accessible workflow (R3). This model can be run monthly to generate a continuous prediction of the best regions, as it eliminates numerous discrete and manual processes. Currently, these Type Well processes are onerous and are typically completed once a year. Therefore, predictions are slow to respond to new data and can lead to erroneous decisions made on outdated information. This weakness would be significantly reduced by employing a BNN. However, the main challenge of adopting this technique is theory complexity. This weakness

can be potentially offset by performing a traditional statistical methodology to validate the output. These techniques are not mutually exclusive, and the production forecast at different points in the pipeline can be used as the foundation for existing workflows. Once there is sufficient evidence that the BNN is working appropriately, the previous techniques could be eliminated or used for unit testing.

### 5.2.1 Restricted Dataset Discussion

Finally, a restricted dataset was utilized to evaluate the methodology's robustness to smaller well counts and address research question (R2). For Existing Well forecasts, the BNN achieved a MAPE score of 13% for gas and 11% for oil. Overall these values are reasonable, considering there was a 93% reduction in the well count. However, additional insight can be gained by [Figure 5-10](#), which plots the average MAPE results spatially. This figure highlights notable areas where there is reduced model performance; these regions typically coincide with either sparsely distributed wells or proximity to the field's boundaries. There are numerous potential reasons to explain this behaviour. Recall that the well's location is an input feature that clusters wells into regions (see [Section 2.2.1](#)). More examples within an area will improve knowledge transfer from nearby producers that likely have similar geological properties. However, there are other confounding factors. An area with decreased well density could be caused by a lack of regional infrastructure or a less active operator that owns the mineral rights. The net effect is that the wells drilled in these regions will have fewer months on production and smaller well counts; thus, the model has reduced training data and is more prone to error.

Regardless of the cause of the error, a few strategic choices can be made to improve model accuracy. For example, the BNN could be applied only in areas with sufficient well control. Additionally, the splitting assumptions for training and testing can be adjusted to fit a restricted dataset more optimally, such as increasing the percentage of training data and splitting wells spatially to avoid unequal distributions. These choices will maximize the amount of data used for training and only apply the model in areas where favourable conditions exist. This concept is supported by the comparison

of MAPE in [Figure 5-10](#). Notable regions identified as having low accuracy are corrected in the full dataset when more wells are introduced. Additionally, the regions in the restricted and full datasets with sufficient density have roughly similar accuracy. Employing the BNN selectively to specific areas could also positively affect the Type Well workflow for limited data, which is more susceptible to outliers, as each point carries greater weight if there are fewer points overall.

The Type Well methodology also scaled appropriately with reduced data. Gas performed the strongest with a 0.91 correlation with the full dataset and closely aligned hotspot map. Comparatively, oil had a lower performance with a 0.71 correlation and higher degree of variance when spatially analyzed. This variation is likely attributed to the fluid and measurement complexities associated with oil, discussed in [Section 5.2](#). Additionally, some regions are sparsely populated, leading to greater model extrapolation. [Figure 4-19](#) demonstrated that the BNN overpredicted the size of the sweet spot of cluster 2; this effect could be driven by the strong performance of surrounding wells, which were extrapolated farther than appropriate. As previously discussed, this workflow should ideally be combined with geological mapping; this need is even greater for restricted datasets as the geology interpretation can function as an additional constraint, limiting extrapolation errors. Finally, the results of the restricted GMM plot were compared against the *Deloitte Insights* Tier 1 mapping, as shown in [Figure 5-11](#). These results appear reasonable, highlighting close alignment with the sweet-spot areas and excluding the sub-optimal clusters.

## **Restricted Data Improvements and Research Question Recommendations**

Findings support that the BNN method has potential to identify the best portions of the reservoir with limited well counts (R3). [Section 1.2.4](#) described how this situation is a considerable problem for existing methodologies, as new fields have insufficient data and violate traditional statistical assumptions necessary for proper application. The BNN provides an additional tool to handle these potential shortcomings; however, this method could be improved by incorporating geological interpretations and data from other reservoirs. A common practice is identifying analogous

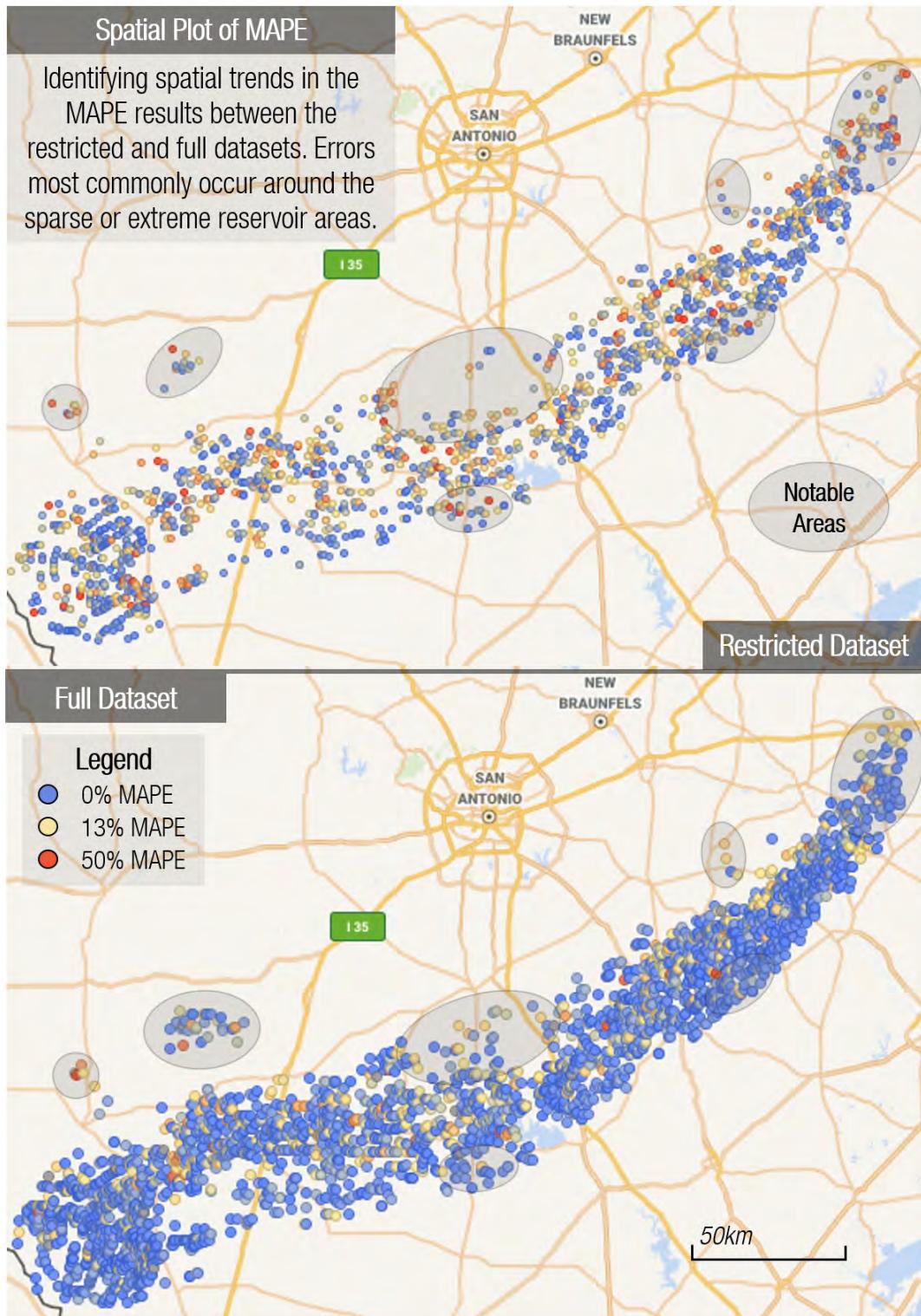


Figure 5-10: Average MAPE results were plotted spatially for the restricted and full datasets. Red, yellow, and blue represent the highest, average, and lowest degree of error, respectively, with a color gradient for values in-between. Most of the areas with the highest error are located in areas with lower well density or near reservoir boundaries; these regions improve in the full dataset when wells are added.

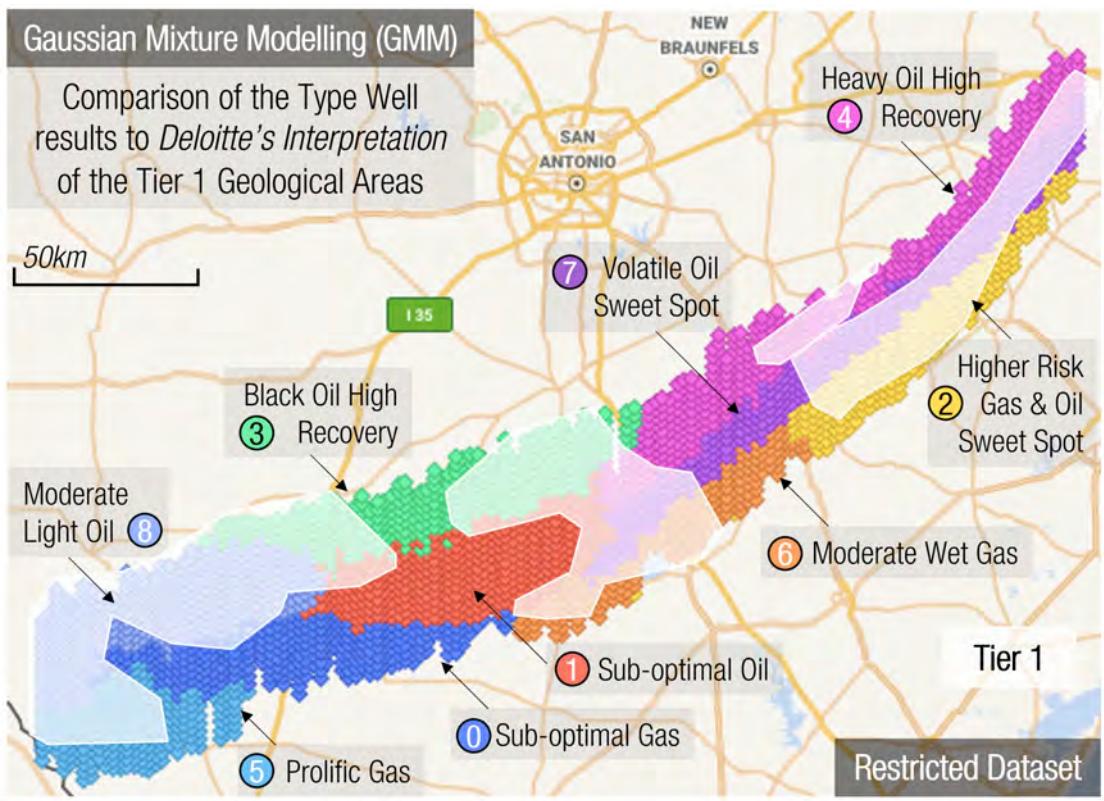


Figure 5-11: GMM Map with Tier 1 interpreted areas highlighted in white. Similar to in [Figure 5-9](#), these regions align with the best-predicted clusters of 2, 3, 4, 7, and 8. The suboptimal 0 and 1 clusters are notably excluded from the Tier 1 areas.

reservoirs that can be modified to predict expected performance for new fields. The same methodology could be applied to the BNN. Wells with similar geology, fluid types, and engineering designs from an established play like the Eagle Ford can be trained to generate a base model. The neural-network weights can be frozen, capturing the analogue’s information, then hidden layers can be added to the network to adjust for the nuances of the new play. This approach of knowledge transfer has been effective in other applications, such as convolutional neural networks, as a method to improve performance on limited data [11].

### 5.3 Conclusions

This thesis sought to improve well profitability by quantifying the underlying uncertainty of the forecasting process using probabilistic machine learning techniques.

This process was challenging, as there have been decades of analysis and practice on production forecasting, with the best methodologies hitting a limit for data availability and engineering knowledge. The solution proposed here relies on modern research into Bayesian Neural Networks to quantify the aleatory and epistemic uncertainties, typically omitted or obscured in deterministic production forecasts. Thorough processing and feature engineering were conducted to address data availability, introducing static and dynamic features that captured the geospatial, production, and engineering well properties. The Bayesian Neural Network also demonstrated strong capabilities to empirically account for complex reservoir phenomena and isolate the optimal geological areas by induction. These results were robust even with a restricted dataset containing a 93% reduction in data points at 48 months. Finally, the conclusions of the research questions are presented in [Summary 5.3.1](#) and [5.3.2](#).

#### Summary 5.3.1: Existing Well Forecast Conclusions to Research Questions

1. Do probabilistic machine-learning techniques improve well forecasting compared to existing methodologies in terms of accuracy, ease-of-use, and risk quantification? **Yes:**
  - The BNN improves mean average percent error of EV forecasts by 10%–35% over existing methods after 25 months of forecast. Compared to alternative methods, the BNN-generated profiles did more closely resemble, based on cosine angle, the curves of actualized production.
  - This model is rapidly trained and deployed, requiring less than an hour of computational time to forecast 10,000 wells on a personal computer. Speed could be further improved, if necessary with parallelization or distributed computing.
  - The method provides new functionality by estimating the maximum-likelihood P10 and P90 uncertainty range; these values allow analysts to consider upside and downside risks in decision making.
2. Can the most productive regions of the reservoir be identified with limited

data, defined by less than 1,000 wells? **Yes:**

- Results had reasonable prediction accuracy with 11%–13% MAPE, even after a substantial reduction of 93% of the data.
  - These results varied spatially, and if selectively applied to areas with sufficient data density, performance would roughly match the full dataset (5%–6% MAPE).
3. Are computational power and practitioner demands sufficiently accessible to create a sustainable and repeatable workflow? **Yes:**
- From purely an analysis perspective, the data availability and computational power are approachable. However, this question poses a workforce-management challenge; a strategy is required to create a culture willing to adopt unfamiliar methodologies.
  - Potential barriers to adoption can be reduced by implementing a community of practice to educate analysts, and leveraging simpler techniques to validate model outputs.

#### Summary 5.3.2: Type Well Forecast Conclusions to Research Questions

1. Do probabilistic machine learning techniques improve well forecasting compared to existing methodologies in terms of accuracy, ease-of-use, and risk quantification? **Yes:**
- The BNN provides a single methodology capable of integrating uncertainties from Existing Well forecasts, engineering feature normalization, and spatial mapping into the probabilistic distribution.
  - Model results were able to reproduce the predicted EURs with 6%–7% MAPE, based on the initial static inputs of the well.
  - Unsupervised learning using GMM clustering identified regions of optimal reservoir quality, which closely aligned with independent analyses.

2. Can the most productive regions of the reservoir be identified with limited data, defined by less than 1,000 wells? **Yes:**

- Gas and oil hotspot mapping matched reasonably with the full dataset. Oil had a larger degree of variance but can be potentially improved with more sophisticated PVT analyses or merging gas and oil predictions into a single model. This workflow should be tested to other reservoirs to validate that the conclusions hold; however, the likelihood of success is promising based on the demonstrated robustness with decreased data.
- The final GMM clustering closely resembled the full dataset and Tier 1 mapping, supporting that this methodology is sufficiently robust to apply on limited data.
- The BNN can delineate the best portions of the reservoir, which is a limitation of traditional techniques due to violations of statistical principles (non-stationarity, heteroscedasticity, and limited data).

3. Are computational power and practitioner demands sufficiently accessible to create a sustainable and repeatable workflow? **Yes:**

- The methodology is capable of replacing numerous discrete analyses that are required for traditional methods. This approach provides the capability of turning a process conducted yearly into a continuous prediction that can rapidly respond to new data.
- Cultural support can be gained by validating findings with traditional methods and incorporating them into unit testing. Importantly, implementation of this workflow must be made with consultations with stakeholders to ensure that their needs are being satisfied. Without gaining this support, the system is unlikely to succeed even if proven to be more accurate than existing methods.

THIS PAGE INTENTIONALLY LEFT BLANK

# Chapter 6

## Future Analysis

### 6.1 Summary of Improvements to Current Workflow

This thesis demonstrated how Existing and Type Well production can be predicted probabilistically utilizing a Bayesian Neural Network (BNN). However, as with most research projects, significantly more analysis could be pursued. This section summarizes the improvements on the current workflow, then describes additional work that can build off of this foundation. First, a summary of improvements is presented in [6.1.1](#) based on the discussion from Chapter 5.

#### Summary 6.1.1: BNN Workflow Improvements

- Simplify the Ensemble model approach to generate an EV forecast, utilizing the same input feature structure as the BNN. The output of this process can be used for unit testing to validate that the BNN workflow is working as anticipated.
- Utilize a convolutional neural network or angular similarity to identify the wells with external projects. Remove these entities from the dataset to improve forecast accuracy and optimize the probabilistic range.
- If possible, obtain supplementary operating pressure and fluid analysis data. Combine these data into a fluid simulation model and utilize a Kalman filter

to predict the likely fraction of gas and oil. Conversely, if unavailable, refactor the code to predict both gas and oil simultaneously. Calculate the oil gas ratio (OGR) and utilize the derivative of OGR with respect to pressure or time as a dynamic lag input feature. These solutions should provide insights into the fluid state and correct gas and oil tendencies to align with aggregate trends.

- Set up a community of practice with subject matter experts to deploy and improve the algorithm. Provide training for other analysts to build competencies and remove barriers of entry.
- Incorporate geological mapping interpretations as input features. These values can improve the clustering of analogous wells and serve as a constraint to limit extrapolation errors. These values are impactful indicators of well performance, based on prior research, and should improve model performance.
- Create a deployable modelling pipeline that can execute the entire workflow monthly. Provide the output of the process in a readily accessible database that analysts can easily consume.
- Acquire data from analogous reservoirs to create a base model when data are limited in the reservoir of interest. These analog sources can augment the dataset and transfer knowledge from established plays with longer production histories, informing late-life volumes.

## 6.2 Future Applications

Lastly, this thesis will conclude with some examples of future applications. The Type Well process described previously was used to generate a probabilistic estimate of ultimate recovery; however, this decision was made to provide a manageable thesis scope. The entire process can be expanded to generate a complete production profile

### Example of Generating a Complete Type Well Forecast

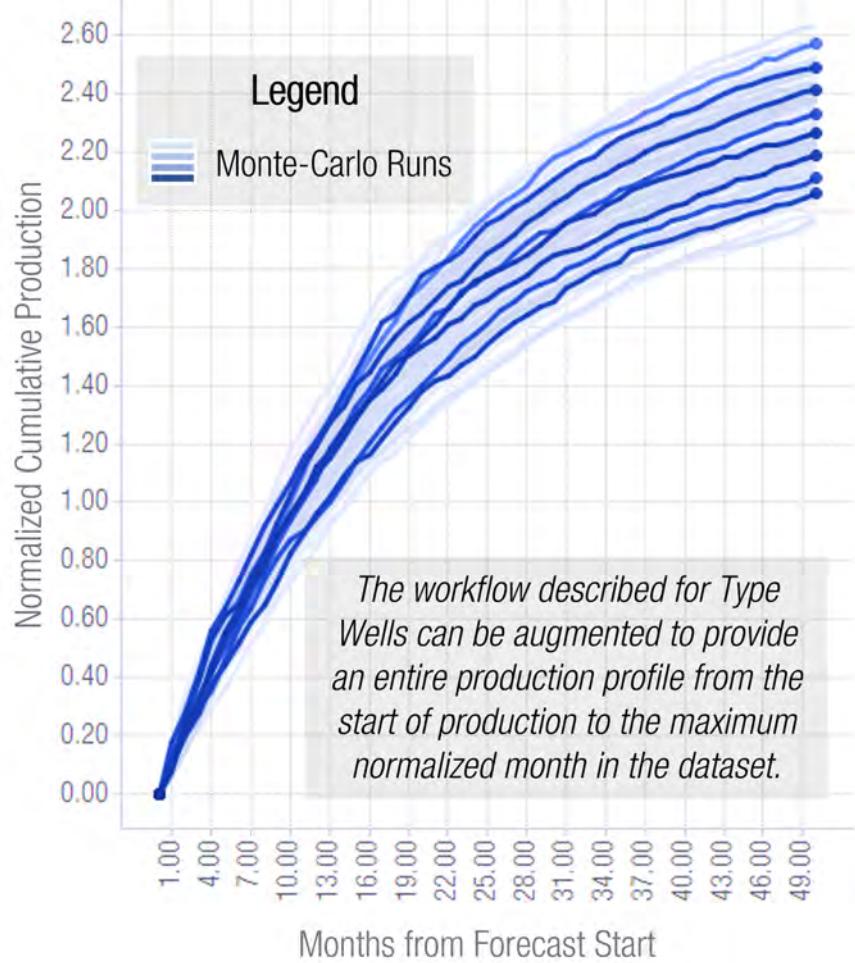


Figure 6-1: An illustrative example of a Type Well forecast profile. Instead of generating only an estimate of EUR, the methodology can be modified to generate predictions spanning from the start of production and extending to the end. After validating, the methodology can recreate the profiles of existing wells, the algorithm will be applied on grid cells; this process will create an expected forecast for new wells drilled within a specified area and engineering design.

shown in [Figure 6-1](#). A Monte Carlo simulation can be performed following a similar process as the Existing Well forecast to generate this plot. Precisely, the algorithm will predict an incremental volume per month starting from zero production. This value will update the dynamic inputs and generate a plausible production profile for each run. These profiles can then be time-sliced to create probabilistic P10 and P90 forecasts.

Example of a Tradespace Analysis to Select Preferred BOD

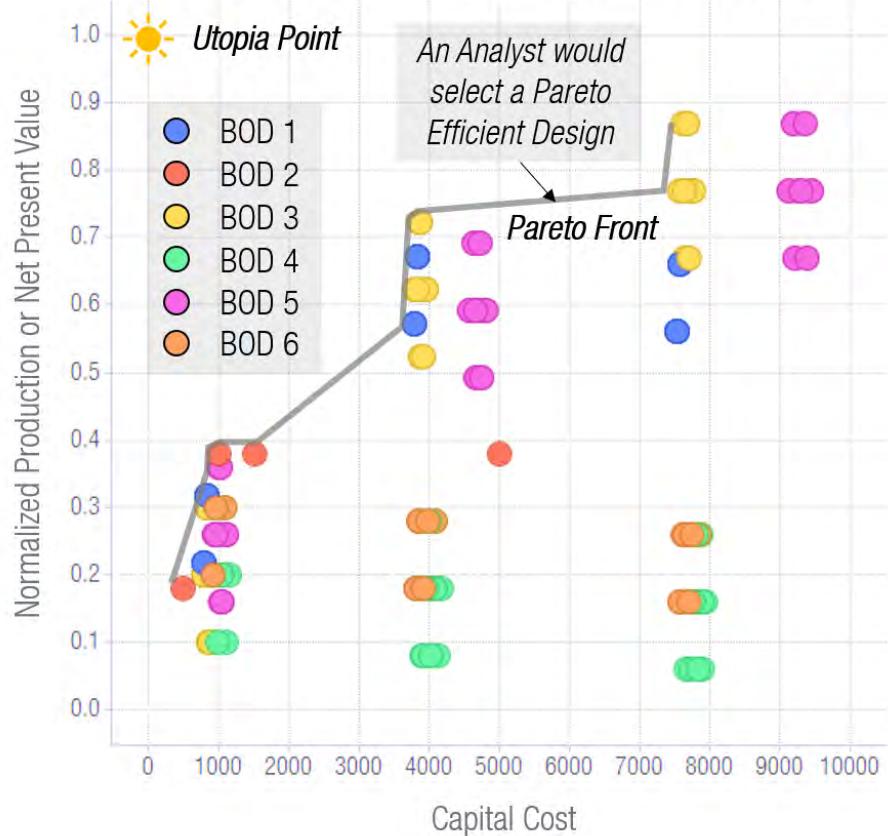


Figure 6-2: The Tradespace analysis highlights the impact of engineering parameters on EUR and cost. The grey line indicates Pareto efficient results, which have more utility than designs with lower costs. The Utopia point signifies the ideal outcome. An analyst selects the preferred architecture to achieve a target EUR or cost.

The Type Well process can also be improved by considering the impact of engineering parameters on recovery. The best geological areas were identified by applying the trained BNN model on generic grid-cells with an average engineering design; however, as discussed in [Section 1.2.4](#) the reservoir characteristics can be exploited through the basis of design (BOD). Therefore, varying engineering setpoints within the model will predict the impact of those architectural decisions on EUR. [Figure 6-2](#) demonstrates an example of this analysis, also known as a Tradespace. Plotting the tradeoff between production and capital cost generates a series of Pareto efficient solutions. This process allows engineers to select the design combination that balances cost and production, tailoring the specific BOD based on the well's reservoir location.

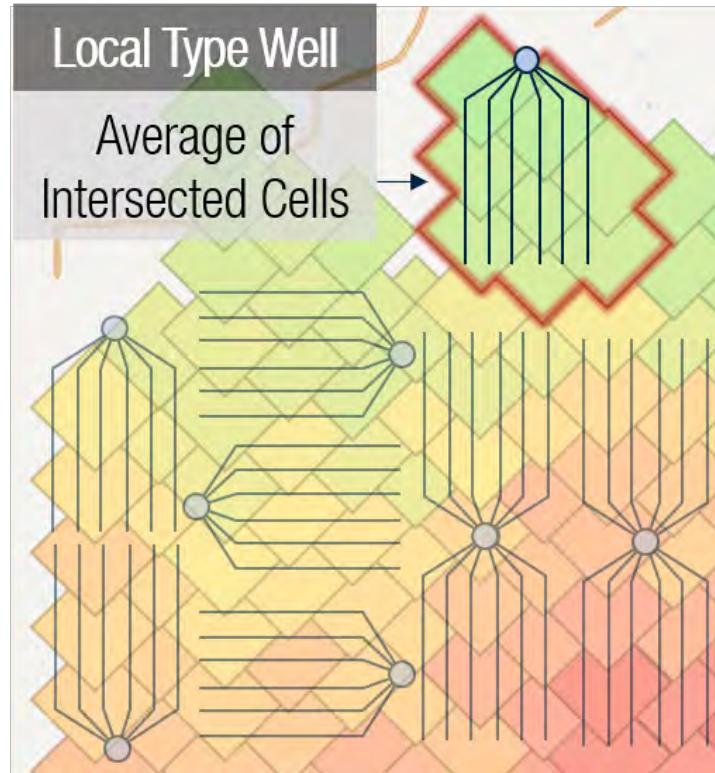


Figure 6-3: An example of how local Type Wells can be formed by intersecting the well layouts with nearby grid cells. This process can be repeated for each planned pad, capturing the nuances of the area, which can rapidly change in shale reservoirs over short distances.

Finally, the full Type Well production profile and tailored engineering design concepts can be blended to provide a localized Type Well forecast. A conventional workflow defines a large polygon boundary, with all the wells within it consisting of the same forecast. The disadvantage of this approach is that this boundary could extend farther than appropriate, resulting in erroneous predictions. [Figure 6-3](#) provides an example of how this effect can be mitigated. Suppose the horizontal wellbore layouts are superimposed over the grid cell locations; inputting the planned well lengths, BOD, and spatial location into the BNN generates a **local forecast** capturing the area's nuances and providing the most accurate estimate of production. If these improvements are implemented, a truly robust and powerful workflow will emerge. This process will help to pierce the veil of uncertainty and improve well profitability.

THIS PAGE INTENTIONALLY LEFT BLANK

# Appendix A

## Tables

### A.1 Cross-Validation

Note that the number of estimators was held constant for the entire cross-validation process, as this value will consistently improve model performance i.e., varying this number will always maximize it and extend the computational time.

Table A.1: XGBoost Hyperparameter tuning after 5-fold cross validation. The selected parameter after the grid-search process is highlighted in blue.

Architecture Decisions	XGBoost Hyperparameters			
	Option 1	Option 2	Option 3	Option 4
n_estimators	500			
max_depth	2	4	6	10
colsample_bytree	0.1	0.5	0.8	1
eta	0.3	0.2	0.1	0.05

Table A.2: Random Forest Hyperparameter tuning after 5-fold cross validation. The selected parameter after the grid-search process is highlighted in blue.

Architecture Decisions	Random Forest Hyperparameters			
	Option 1	Option 2	Option 3	Option 4
n_estimators	500			
max_depth	5	10	20	30
min_samples_leaf	1	3	5	
min_samples_split	2	3	5	10

Table A.3: LightGBM Hyperparameter tuning after 5-fold cross validation. The selected parameter after the grid-search process is highlighted in blue.

Architecture Decisions	LightGBM Hyperparameters			
	Option 1	Option 2	Option 3	Option 4
n_estimators	500			
num_leaves	32	300	1024	
max_depth	1	5	8	10
reg_lambda	0	0.25	0.5	
reg_alpha	0	0.25	0.5	

Table A.4: Feedforward Neural Network Hyperparameter tuning after 5-fold cross validation. The selected parameter after the grid-search process is highlighted in blue. Since this model is used in the Ensemble method, 3 nodes are tested to provide greater variances between models.

Architecture Decisions	Feedforward Neural Network			
	Option 1	Option 2	Option 3	Option 4
epochs	300			
layers	1	2	3	4
activation_function	sigmoid	relu	swish	tanh
nodes	32	n	64	100
batch_size	32	150	300	

Table A.5: BNN Hyperparameter tuning after 5-fold cross validation. The selected parameter after the grid-search process is highlighted in blue.

Architecture Decisions	Bayesian Neural Network			
	Option 1	Option 2	Option 3	Option 4
epochs	300			
dense_layers	1	2	3	4
dense_activation_func	sigmoid	relu	swish	tanh
prob_activation_func	linear	softplus		
nodes	32	n	64	100
optimizer	SGD	Adam		

## A.2 Oil Existing Well Forecast Results

Table A.6: Summary of the evaluation criterion for each method binned into 10-month forecasting periods. MAPE and RMSE prefer the BNN after the  $20 < t < 30$  period, while OSR<sup>2</sup> does not significantly differ until  $50 < t < 60$ . Oil results exhibited similar behaviour as gas but had degraded performance notable after  $40 < t < 50$

Model	Mean Absolute Percent Error (MAPE)					
	$0 < t < 10$	$10 < t < 20$	$20 < t < 30$	$30 < t < 40$	$40 < t < 50$	$50 < t < 60$
Automated DCA	2%	5%	8%	11%	14%	17%
Automated DCA - Oil	1%	3%	5%	7%	10%	13%
BNN - Gas	3%	5%	7%	9%	10%	11%
BNN - Oil	2%	4%	6%	8%	10%	13%
Ensemble - NN Based	3%	6%	8%	10%	12%	15%
Ensemble - Tree Based	2%	5%	8%	10%	12%	14%
Model	Out of Sample R2 (OSR2)					
	$0 < t < 10$	$10 < t < 20$	$20 < t < 30$	$30 < t < 40$	$40 < t < 50$	$50 < t < 60$
Automated DCA	98%	91%	84%	81%	75%	67%
Automated DCA - Oil	99%	93%	82%	74%	66%	59%
BNN - Gas	98%	91%	85%	82%	79%	79%
BNN - Oil	99%	91%	82%	76%	71%	68%
Ensemble - NN Based	98%	91%	84%	81%	75%	67%
Ensemble - Tree Based	98%	92%	87%	84%	80%	75%
Model	Root Mean Squared Error (RMSE)					
	$0 < t < 10$	$10 < t < 20$	$20 < t < 30$	$30 < t < 40$	$40 < t < 50$	$50 < t < 60$
Automated DCA - Oil	1.4	4.6	8.1	12.0	16.6	21.6
BNN - Oil	2.2	5.8	9.0	12.4	16.3	20.1

THIS PAGE INTENTIONALLY LEFT BLANK

# Appendix B

## Figures

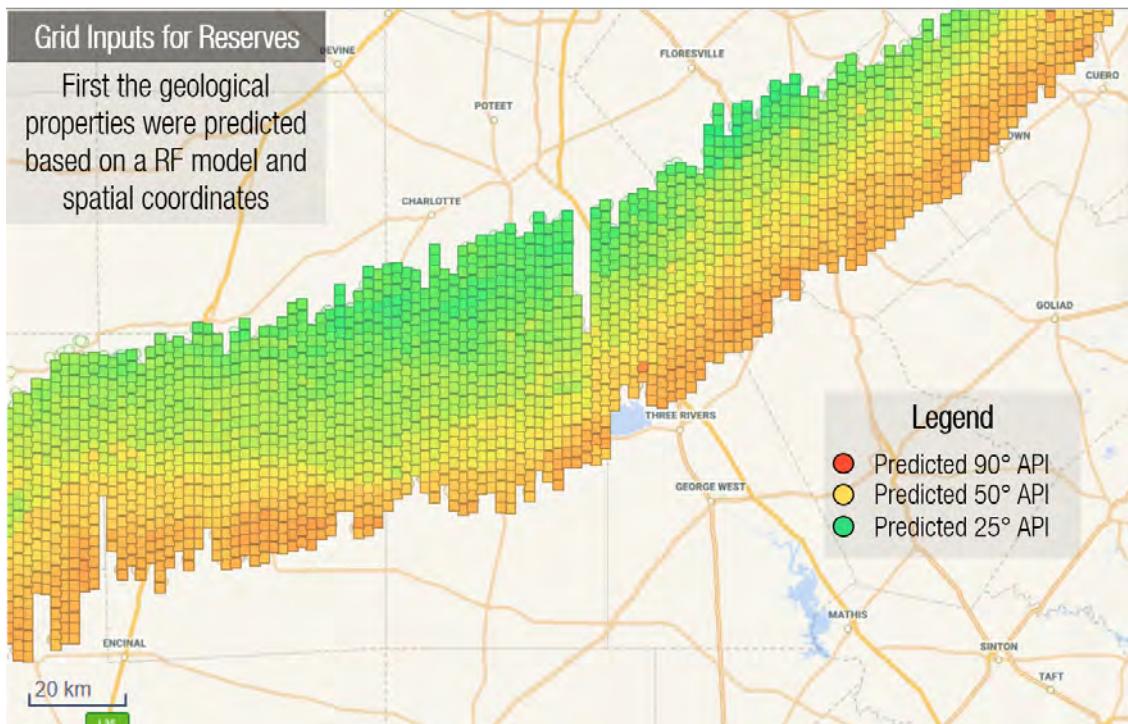


Figure B-1: Plot of the Eagle Ford split into 4 km<sup>2</sup> sections. This grid was formed from the maximum and minimum longitude and latitude of nearby wells. A trained RF model was then applied to each grid cell to predict TVD and oil gravity (represented by the gradient from red to green).

```

model = Sequential([
    Dense(units=x_train_final.shape[1], input_shape=(x_train_final.shape[1],), activation='relu'),
    tfpl.DenseVariational(units=2,
        make_prior_fn=prior, A           Input features connected to dense
        make_posterior_fn=posterior, B   layer with RELU activation function
        kl_use_exact = True,
        kl_weight=1/x_train_final.shape[0]),
    tfpl.DistributionLambda(
        lambda t: tfd.Normal(loc=1e-5 + tf.math.softplus(0.01*t[..., :1]),      Probabilistic Weights to
        scale=1e-3 + tf.math.softplus(0.01 * t[..., 1:]))), must be > 0
]) Aleatory Uncertainty                                         Approximate Bayesian Inference

model.compile(loss=nll, optimizer=Adam(learning_rate=0.001)) C Negative Log Likelihood with
model.summary()                                                 Kullback–Leibler Divergence

def prior(kernel_size,bias_size,dtype=None): A
    n = kernel_size+bias_size
    prior_model = Sequential([
        tfpl.DistributionLambda(
            lambda t:tfd.MultivariateNormalDiag(loc=tf.zeros(n),scale_diag=tf.ones(n)))
    ])
    return prior_model                                         Fixed Multivariate Normal Distribution
                                                               Layer with mean of 0 and std of 1

def posterior(kernel_size,bias_size,dtype=None): B
    n = kernel_size+bias_size
    #want to Learn the distribution
    posterior_model = Sequential([
        tfpl.VariableLayer(tfpl.MultivariateNormalTriL.params_size(n),dtype=dtype),
        tfpl.MultivariateNormalTriL(n)
    ])
    return posterior_model                                     Multivariate Normal Distribution
                                                               Layer with Tunable parameters
                                                               following KL Divergence to
                                                               match true posterior

def nll(y_true, y_pred): C
    return -y_pred.log_prob(y_true)                         Loss function based on the negative log
                                                               likelihood. KL divergence term accounted
                                                               for within the Dense Variational Layer.

```

Figure B-2: Code summary of the main structure of the BNN. The epistemic uncertainty is captured with a Dense Variational Layer, with the specified prior and posterior functions. The exact KL divergence function is utilized since this method uses common distribution types. The KL weight term incorporates the number of data points to average the error of the loss function. Aleatory Uncertainty is captured by a normal distribution with mean and standard deviation following a softplus activation function. Adam (Adaptive Moment Estimation) is utilized as the optimizer with the loss specified as being the negative log-likelihood (this function automatically including the KL Divergence score).

	sumsquare_error	aic	bic	kl_div
lognorm	975.017153	-17.588036	136.527528	inf
norm	988.484046	46.635328	133.398947	inf

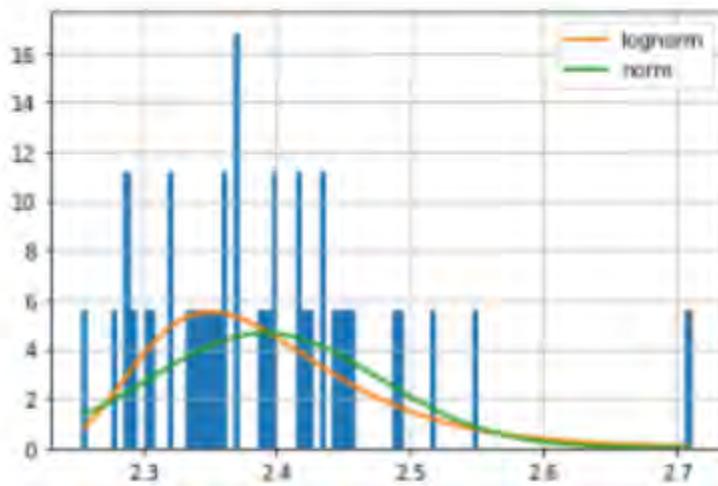


Figure B-3: Validation that the result of the ensemble-based method follows a lognormal distribution at every time step. This outcome matches with expectations from theory [78]

## B.1 Well Plots - Existing Well Forecasting

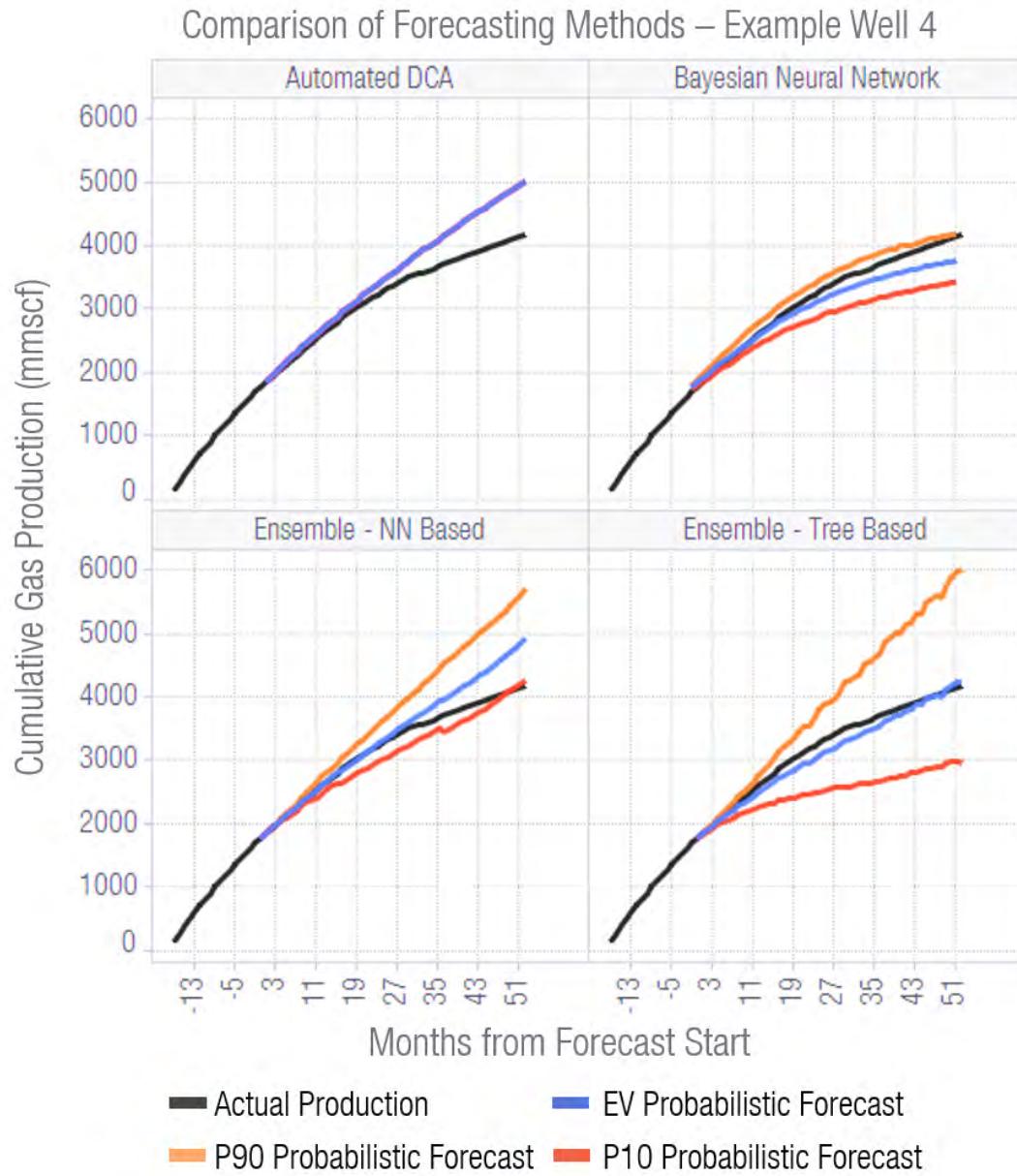


Figure B-4: Well 4 exhibits actual results that fall within the P10–P90 range for the BNN. Both Ensemble methods exhibit a characteristic funnel shape to the prediction curves.

### Comparison of Forecasting Methods – Example Well 5

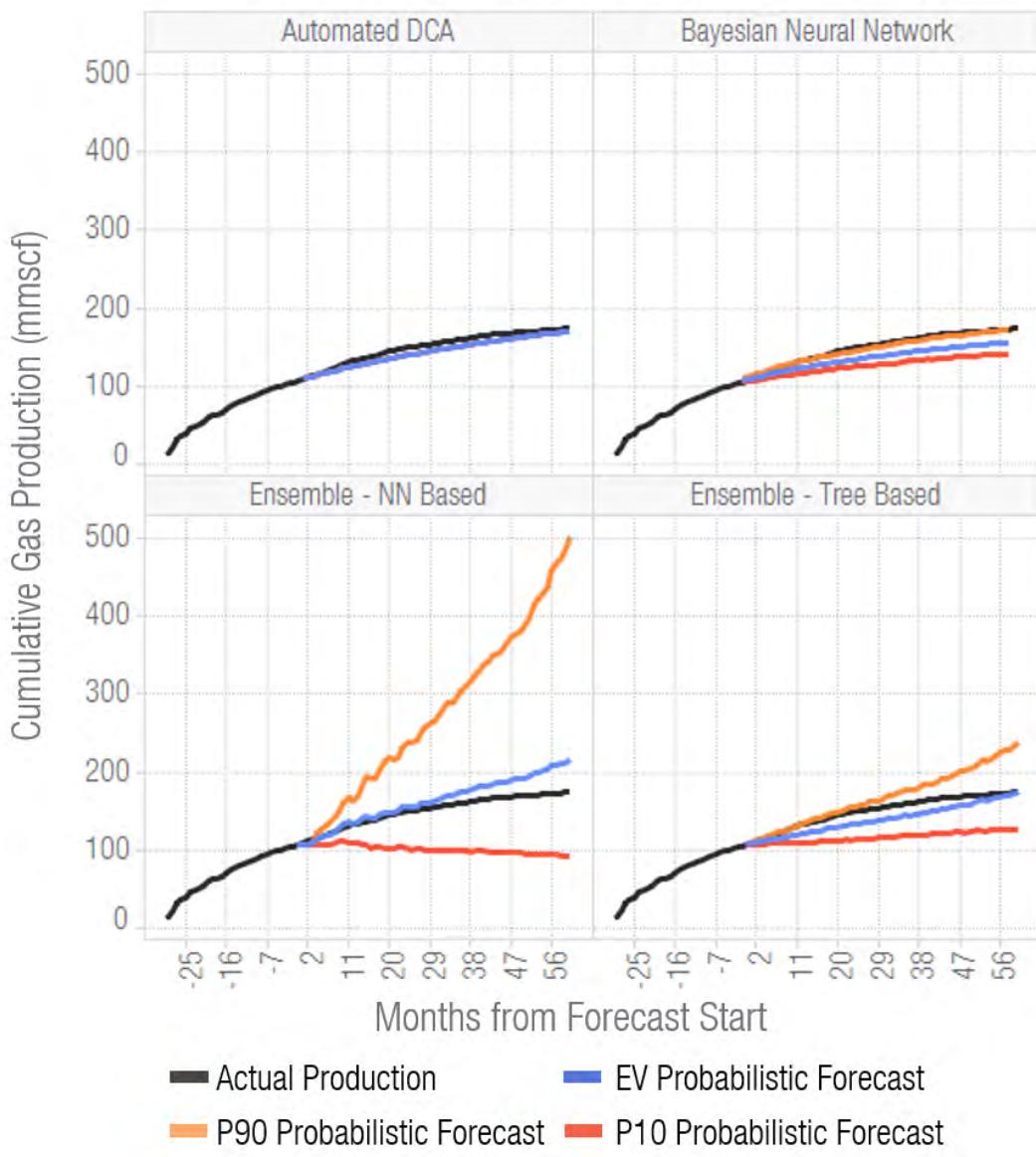


Figure B-5: Well 5 exhibits a case where the BNN forecast is closely aligned with the P90. The Ensemble-NN approach shows extreme divergence with the P10 growing more negative and the P90 having upwards acceleration. The Ensemble-Tree based result is more reasonable but also has a funnel-shaped prediction. DCA prediction matches well with actuals.

## Comparison of Forecasting Methods – Example Well 6

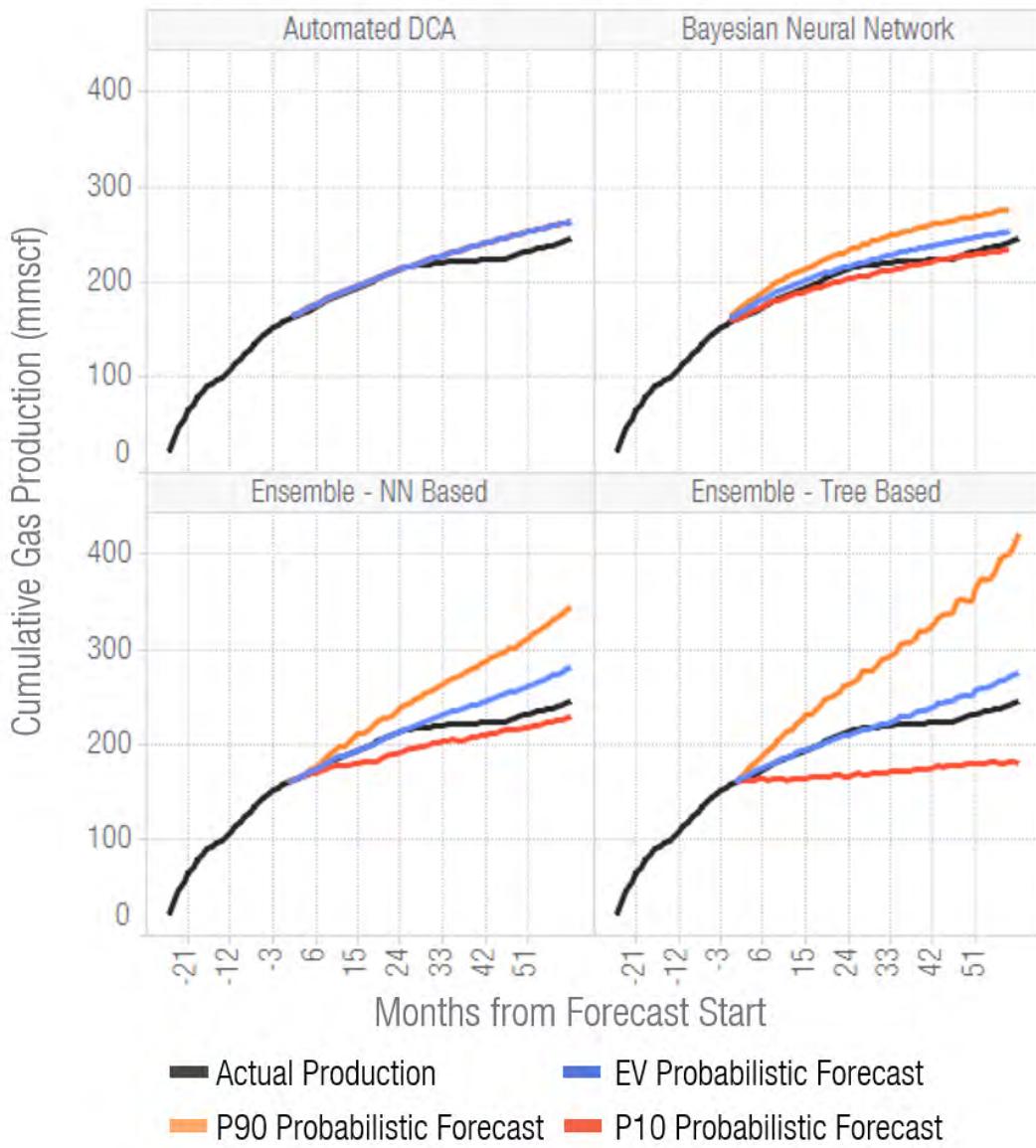


Figure B-6: Well 6 corresponds roughly with the P10 prediction for the BNN. Both Ensemble methods exhibit a characteristic funnel shape to the prediction curves. Significant oscillatory behaviour can be identified with the Ensemble-Tree based method. DCA prediction is similar to the BNN EV forecast, but does not provide a probabilistic range.

### Comparison of Forecasting Methods – Example Well 7

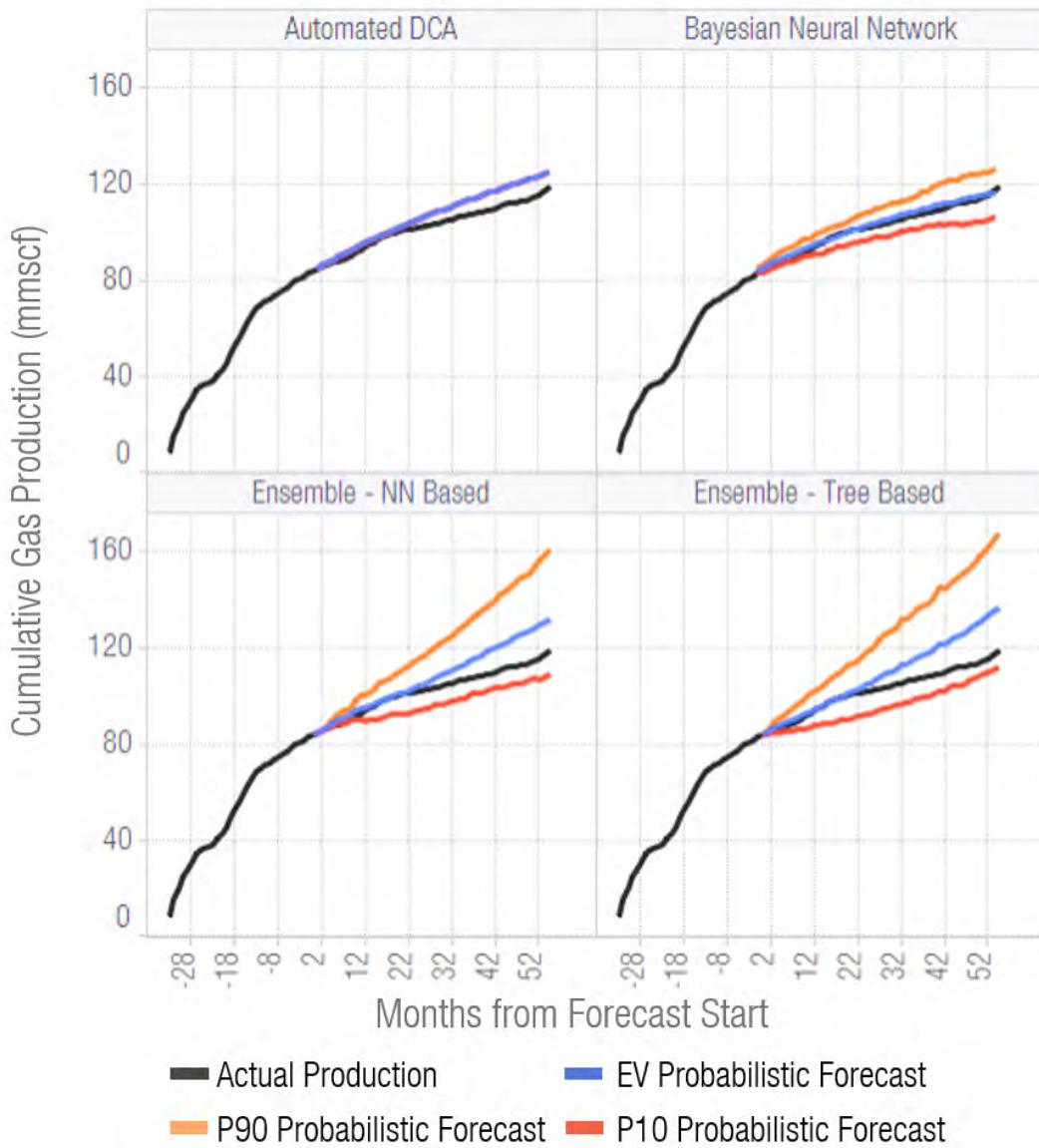


Figure B-7: Well 7 has close alignment between actuals and the EV for the BNN. Both Ensemble methods exhibit a characteristic funnel shape to the prediction curves. DCA matches reasonably well with the prediction but overpredicts the final volume.

## Comparison of Forecasting Methods – Example Well 8

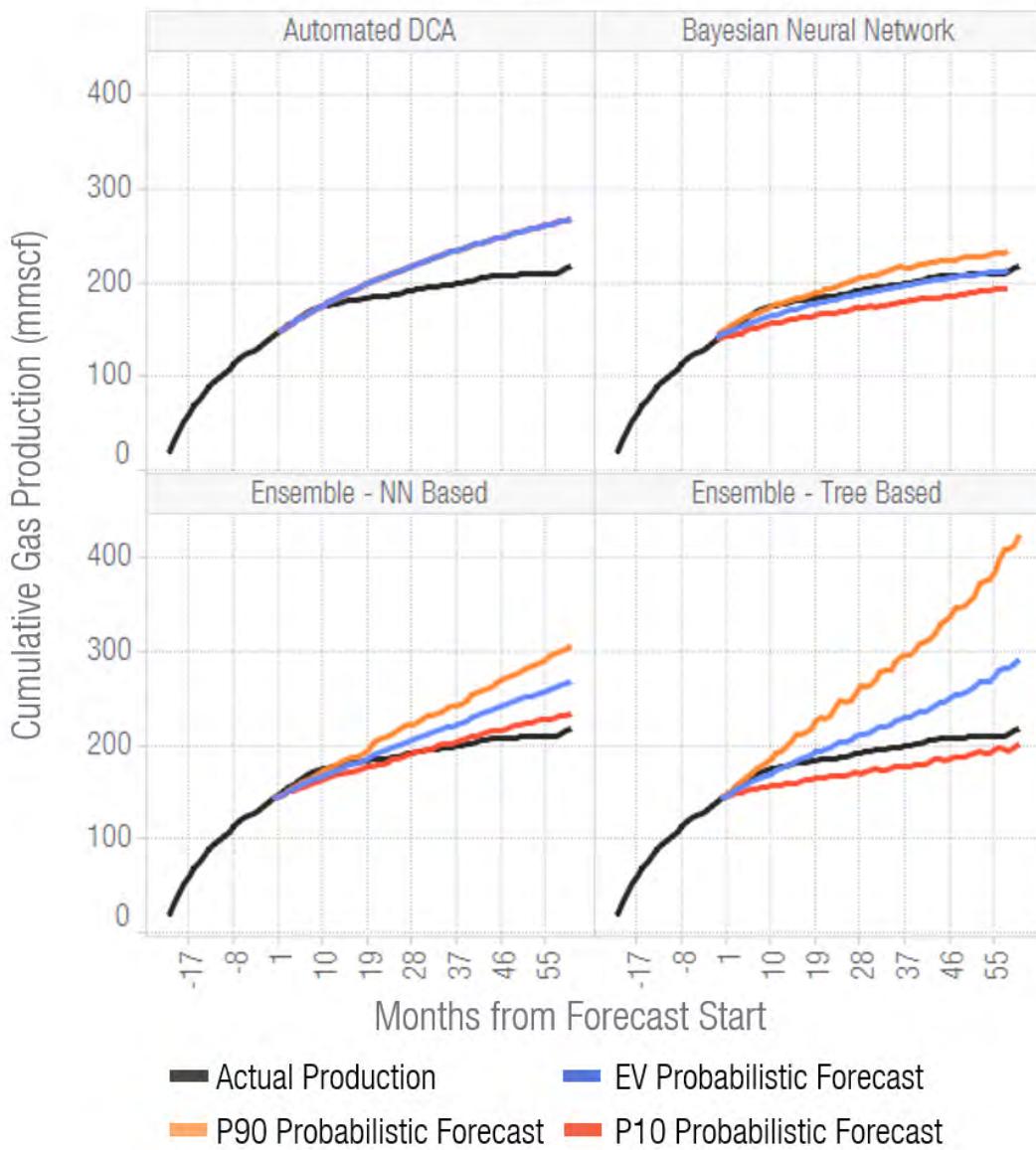


Figure B-8: Well 8 demonstrates strong results for the BNN. Strongly divergent and oscillatory behaviour occurs for the Ensemble-Tree based method. The Ensemble-NN method appears like an unrealistic profile. The DCA forecast considerably overestimates production.

### Comparison of Forecasting Methods – Example Well 9

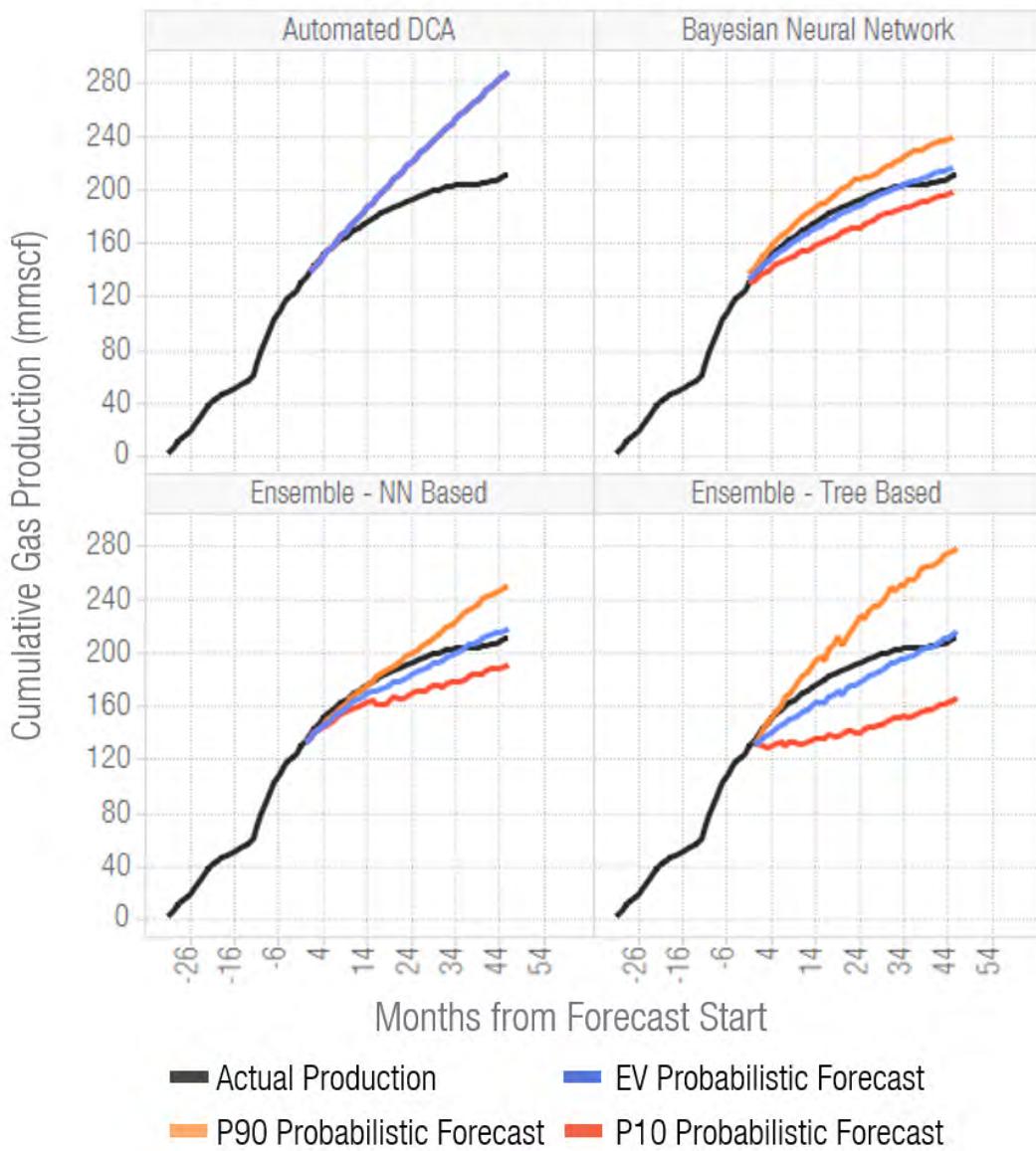


Figure B-9: Well 9 matches well with the BNN forecast. Both Ensemble methods exhibit significant prediction oscillations. The EV and P90 curvatures look plausible; however, the P10 is unrealistic. The DCA approach significantly overestimates production.

## Comparison of Forecasting Methods – Example Well 10

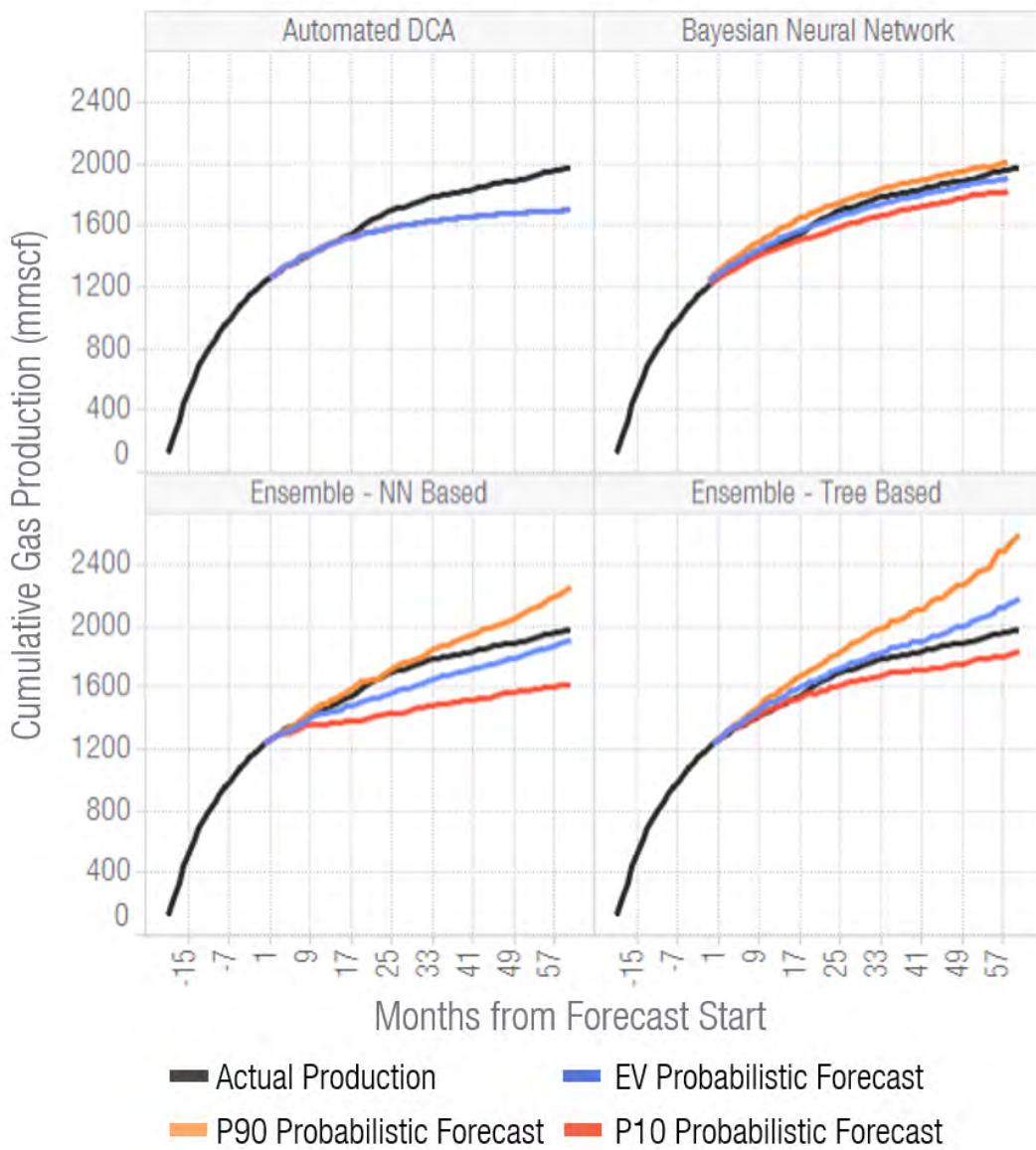


Figure B-10: Well 10 closely matches with the BNN. The Ensemble-Tree based method shows signs of instability with production accelerating near the end of production. This effect would be unlikely to occur. The DCA prediction appears reasonable but underestimates production.

## Comparison of Forecasting Methods – Example Well 11

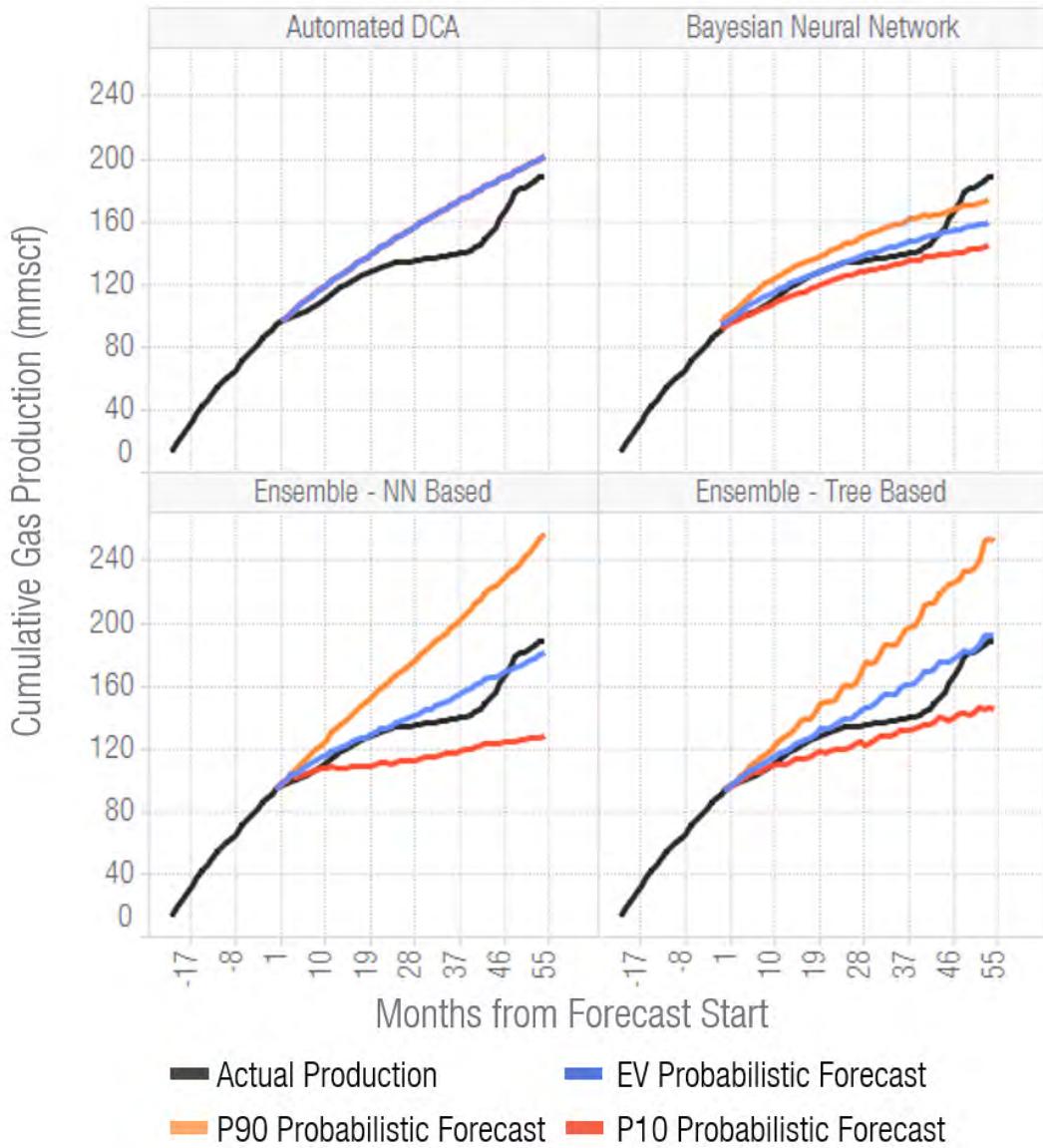


Figure B-11: Well 11 exhibits a late time shift in production volumes which does not align with expectations and could align with an external project implemented. This change causes the forecast to fall outside of the BNN probabilistic range. The forecast falls within the range for the Ensembles, though, this prediction appears broad. The profile did not align closely before to the step change.

## Comparison of Forecasting Methods – Example Well 12

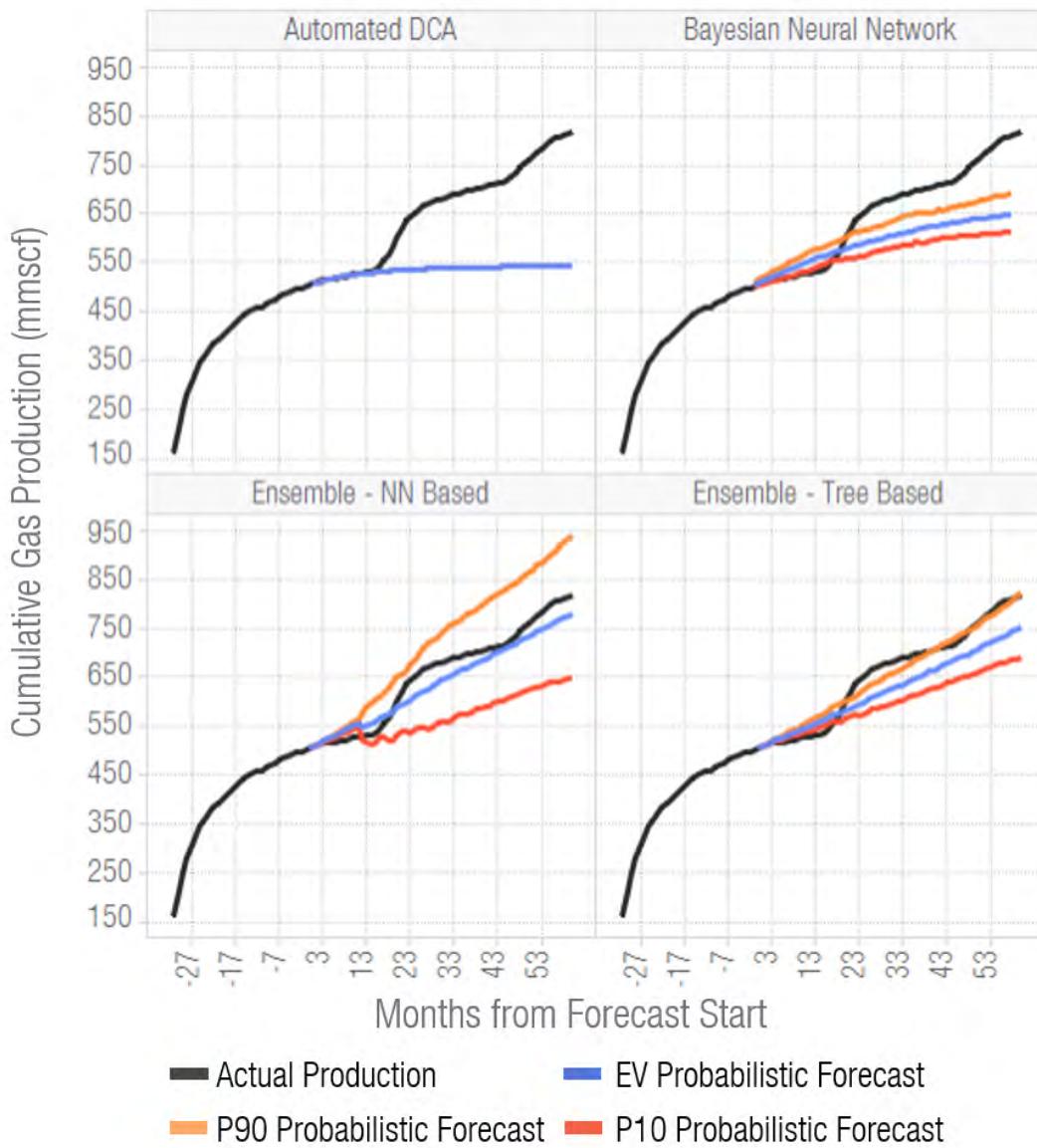


Figure B-12: Well 12 exhibits very uncharacteristic behaviour with two late time spikes in production followed by a plateau. This trend could indicate a cyclic gas injection project which injects gas into the reservoir, followed by a period of production. This cycle is repeated resulting in multiple spikes in volumes. The spread in the Ensemble methods are able to capture this well; however, they do not match appropriately with the data like the BNN and DCA methods, until after the spikes.

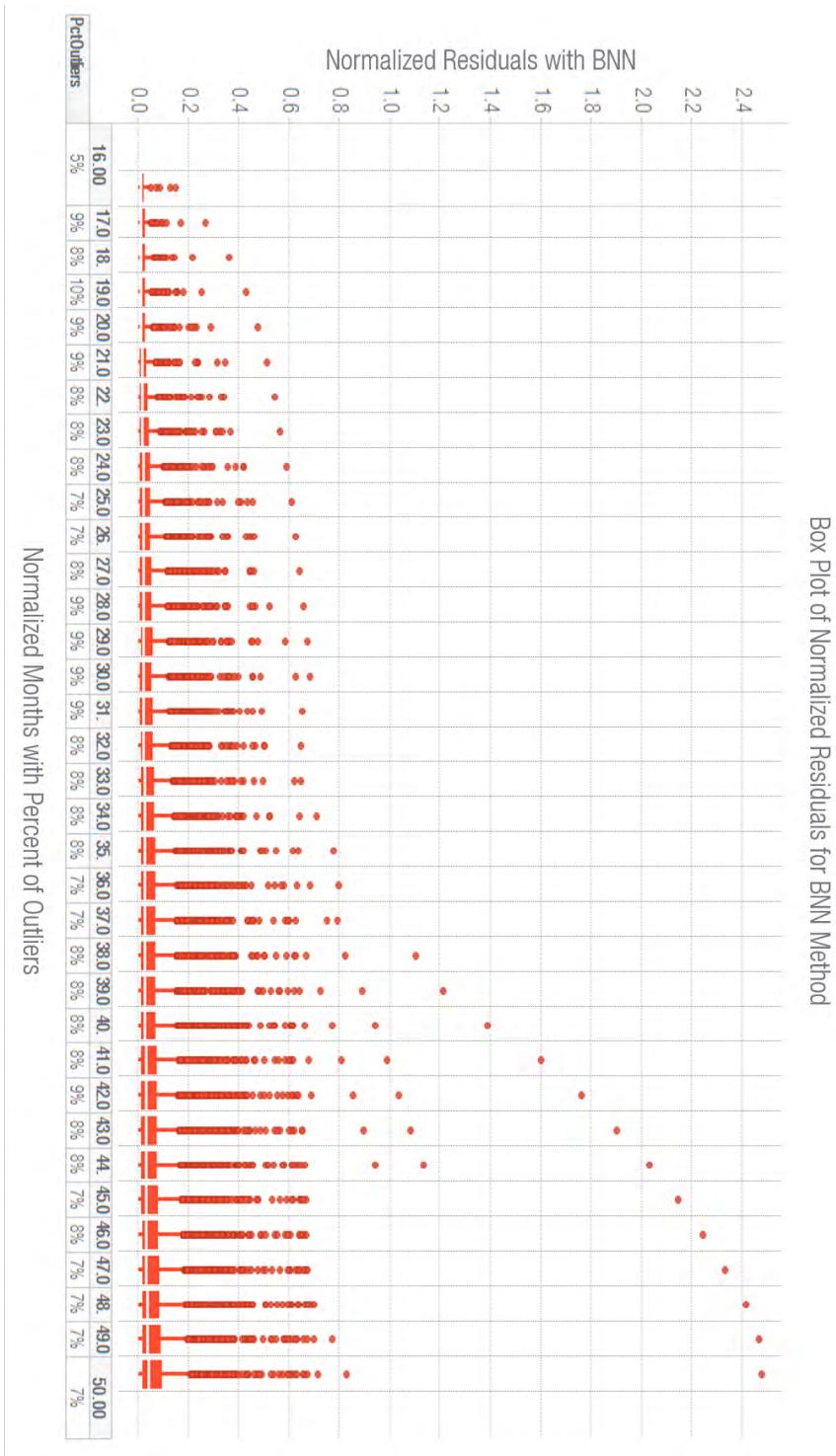


Figure B-13: A highlevel estimate of the number of wells exhibiting divergent behaviour was provided by filtering to the wells with positive residuals. The BNN has consistently demonstrated that it will not overpredict wells; thus, strongly positive residuals are the candidates that could show late-time step changes. Approximately 7% of wells were identified as being outliers based on a 1.5IQR.

## B.2 Well Plots - EUR Forecast

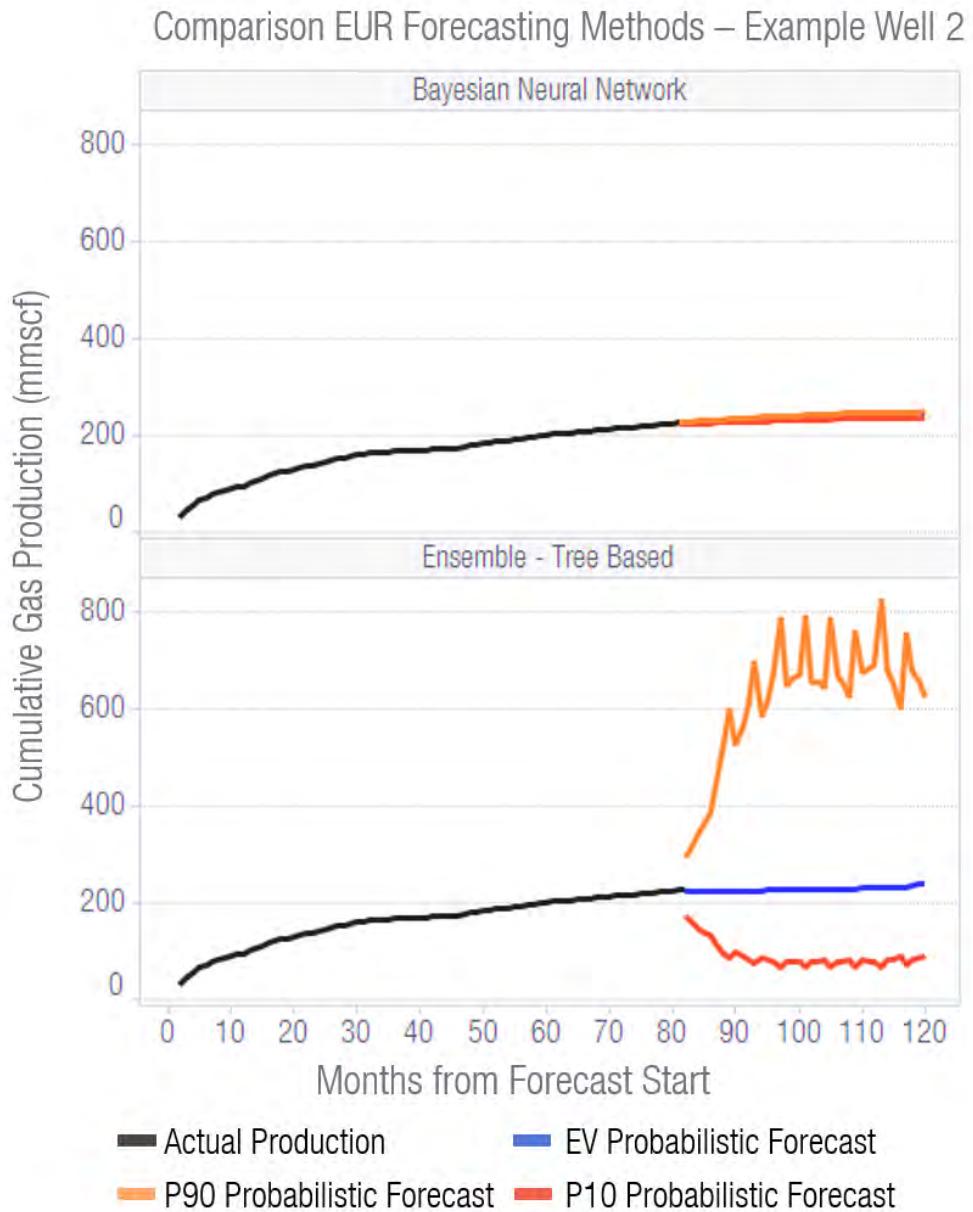


Figure B-14: Well 2 shows extreme instability for the ensemble method, generating an unrealistic spread in P10–P90 values. The range becomes so divergent that the uncertainty in the BNN method becomes indistinguishable at the same scale. The Ensemble also has extreme production oscillations for the P10–P90 range.

## Comparison EUR Forecasting Methods – Example Well 2

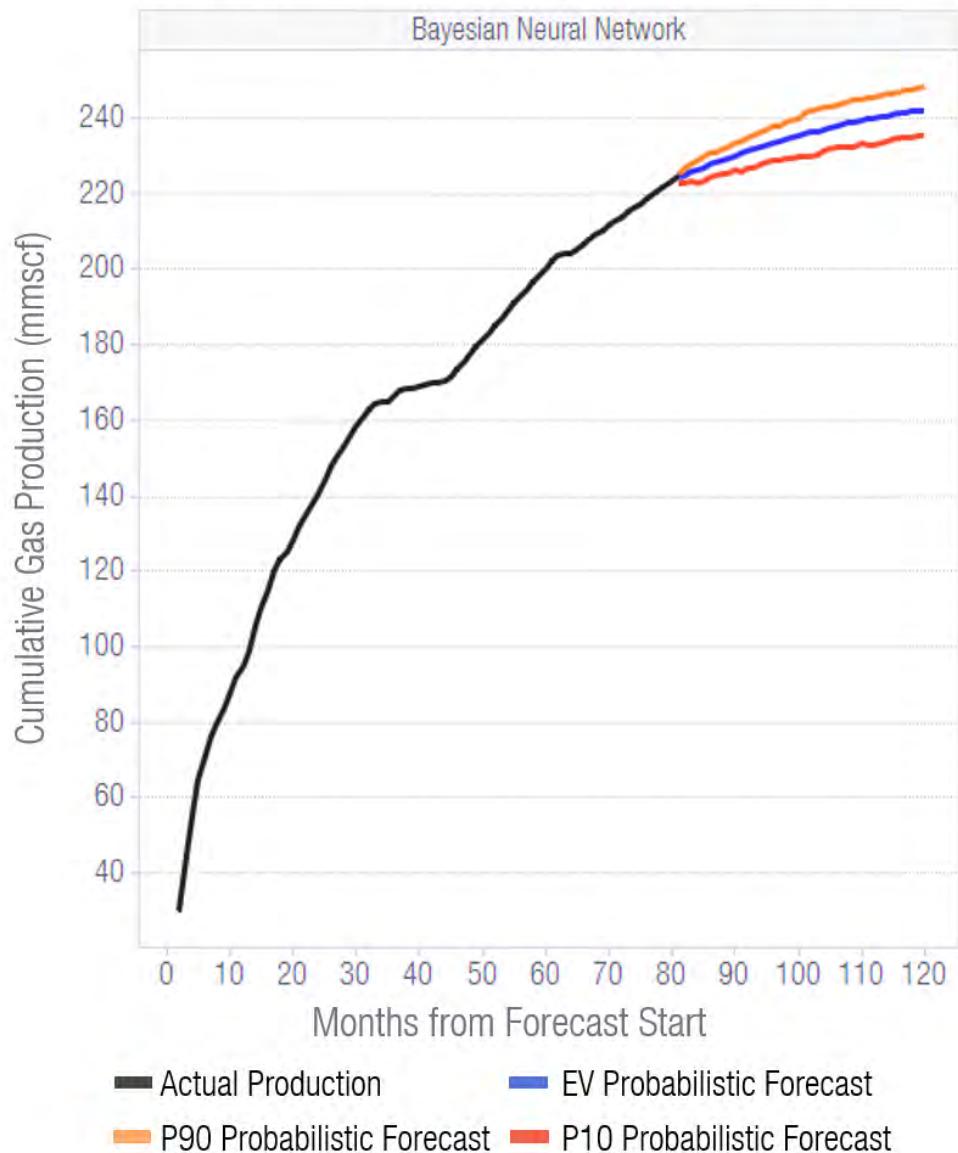


Figure B-15: The BNN method rescaled for Well 2. The curvature of each forecast appears plausible. Overall the uncertainty range is low but is reasonable based on how late into the well's lifecycle the prediction takes place.

### Comparison EUR Forecasting Methods – Example Well 3

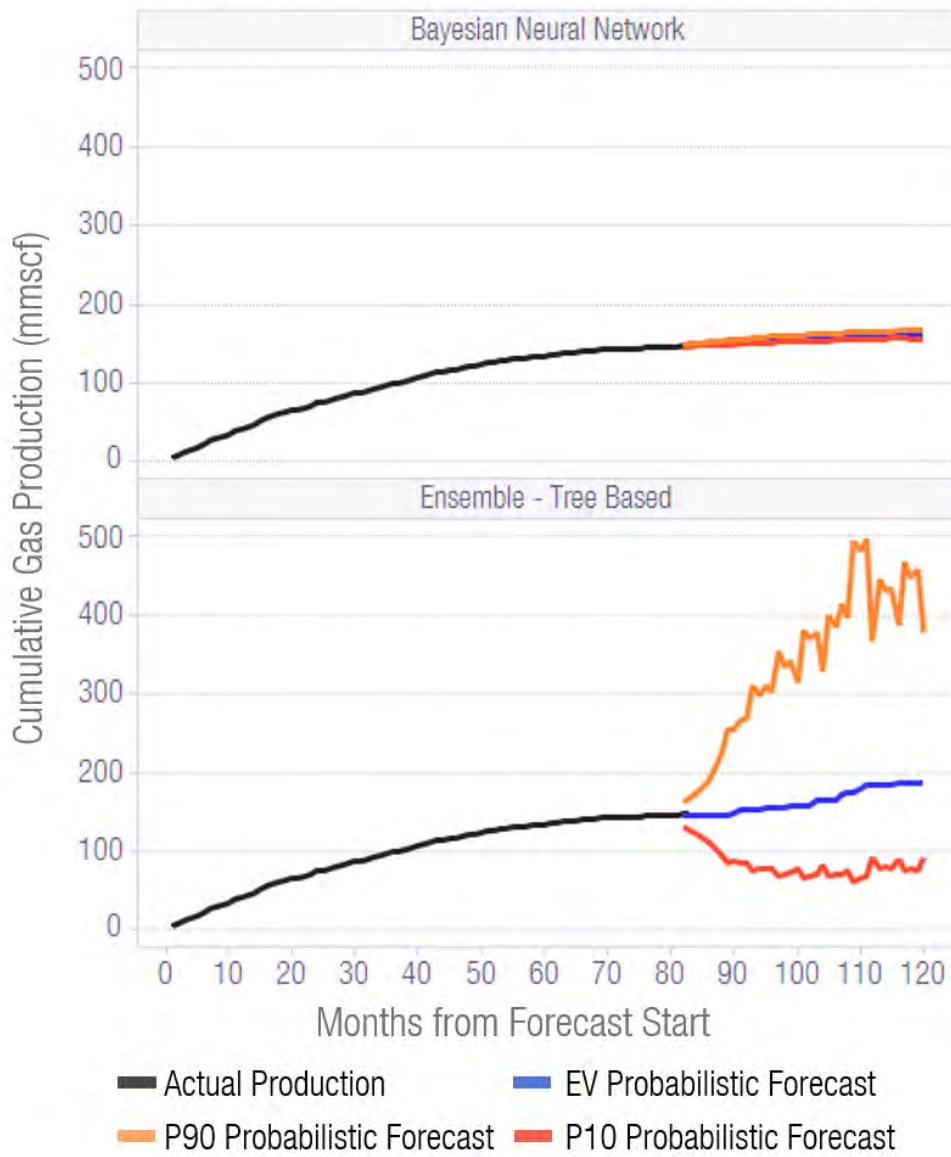


Figure B-16: Well 3 shows extreme instability for the ensemble method, generating an unrealistic spread in P10–P90 values. The range becomes so divergent that the uncertainty in the BNN method becomes indistinguishable at the same scale. The Ensemble also has extreme production oscillations for the P10–P90 range.

### Comparison EUR Forecasting Methods – Example Well 3

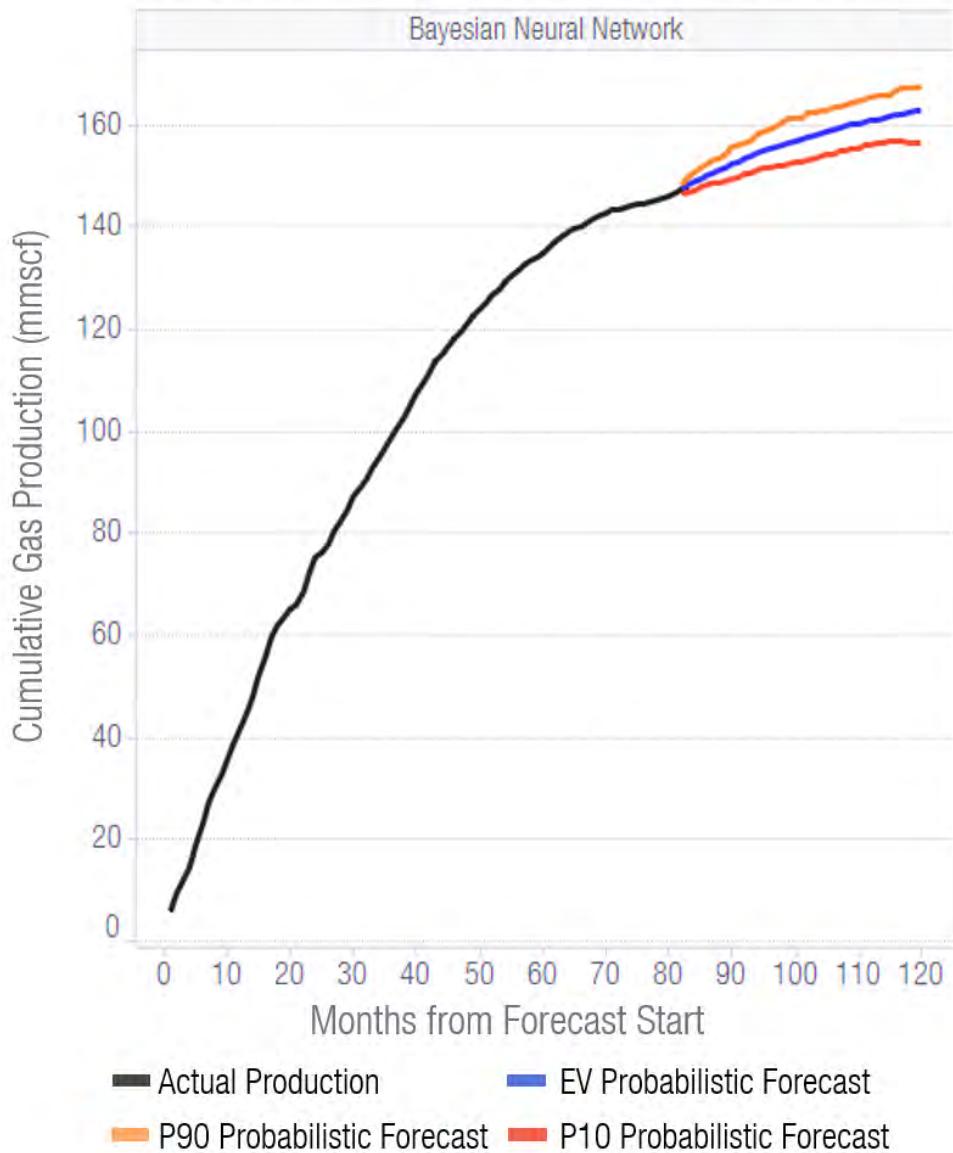


Figure B-17: The BNN method rescaled for Well 3. The curvature of each forecast appears plausible. Overall the uncertainty range is low but is reasonable based on how late into the well's lifecycle the prediction takes place.

### Comparison EUR Forecasting Methods – Example Well 4

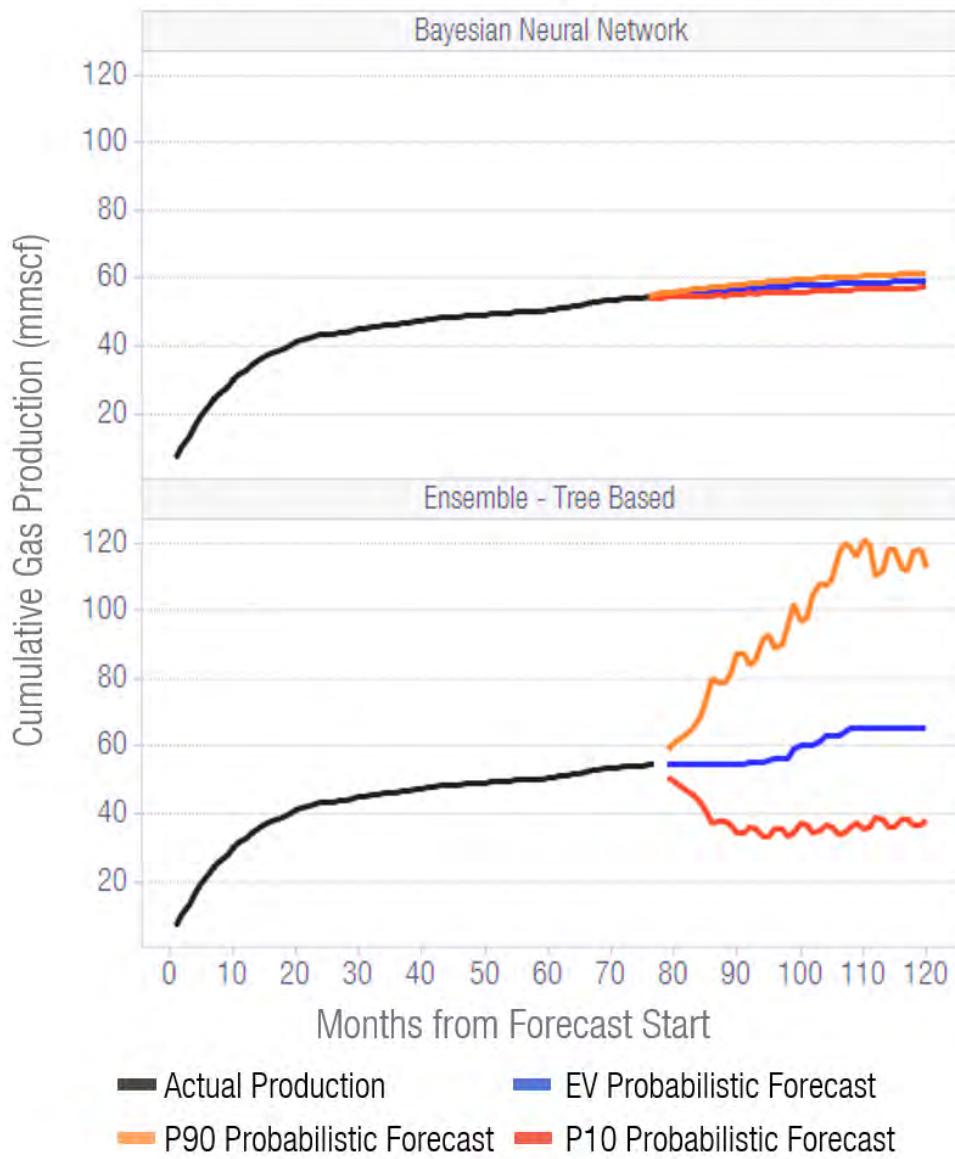


Figure B-18: Well 4 shows extreme instability for the ensemble method, generating an unrealistic spread in P10–P90 values. The range becomes so divergent that the uncertainty in the BNN method becomes indistinguishable at the same scale. The Ensemble also has extreme production oscillations for the P10–P90 range.

### Comparison EUR Forecasting Methods – Example Well 4

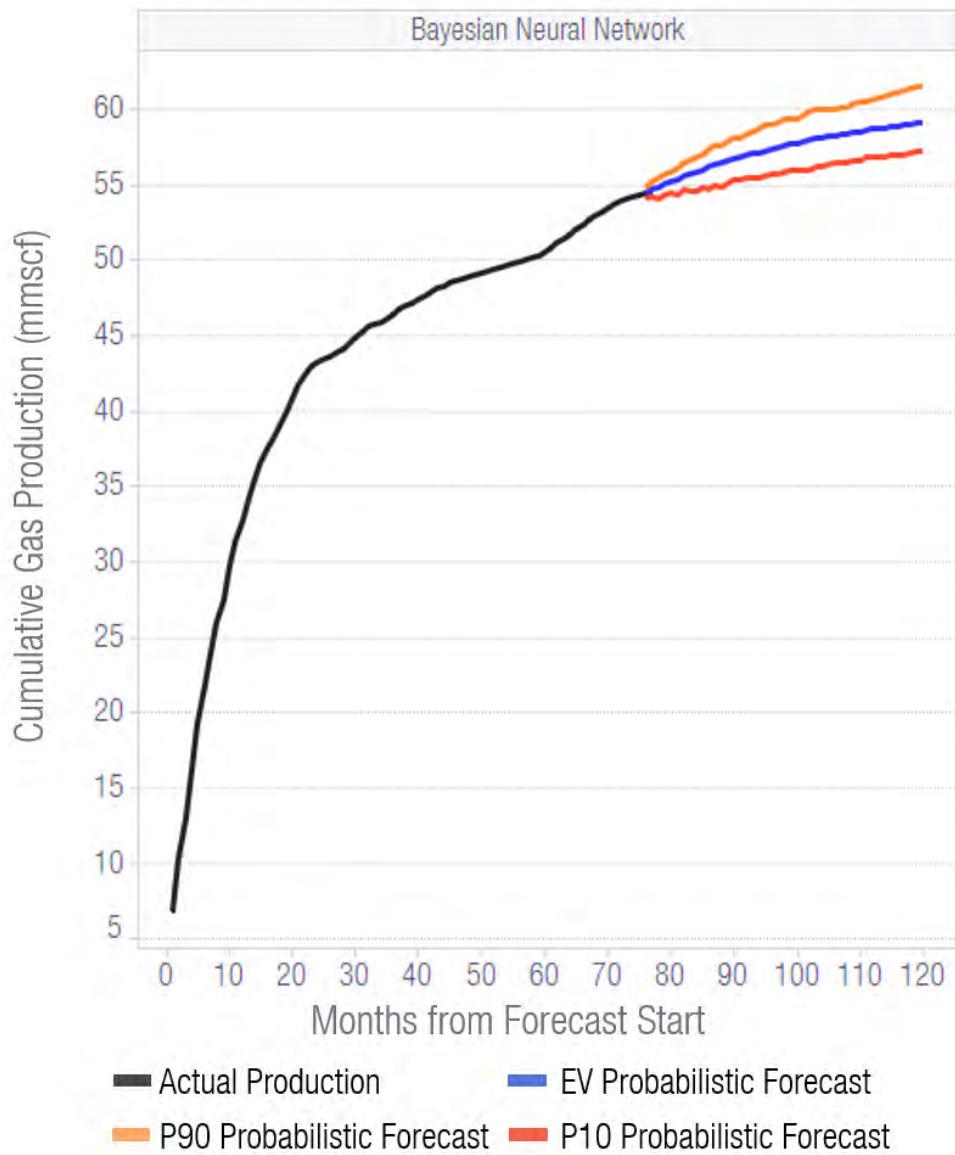


Figure B-19: The BNN method rescaled for Well 4. The curvature of each forecast appears plausible. Overall the uncertainty range is low but is reasonable based on how late into the well's lifecycle the prediction takes place.

### B.3 Oil Existing Well Forecasts

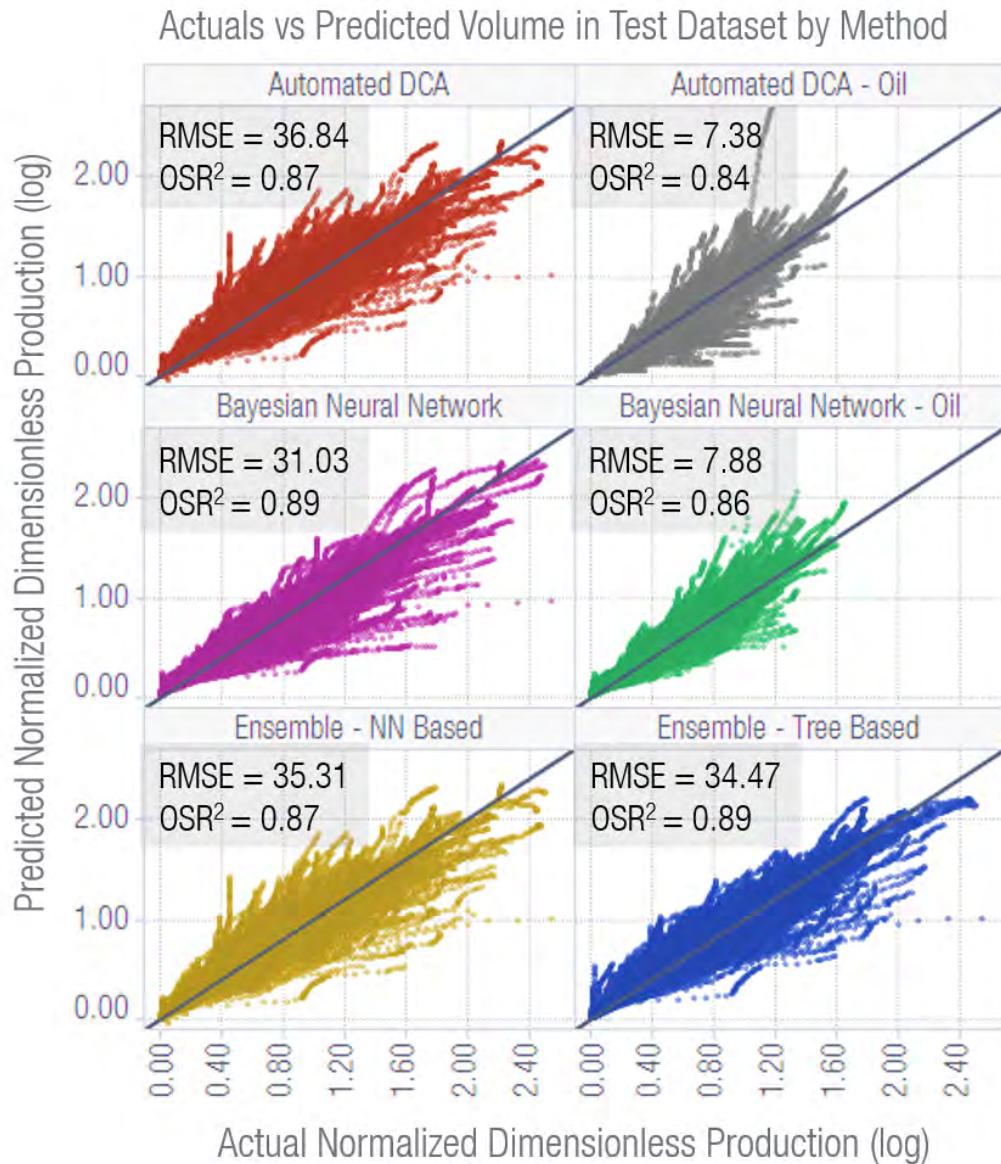


Figure B-20: Validation plot demonstrating model performance on unseen test data. The normalized dimensionless volumes are presented to allow plotting to occur on the same scale. Overall the BNN had the best results for all metrics; however, this method had limited differences in terms of accuracy with the DCA method for oil.

### Comparison of MAPE in Test Dataset by Method

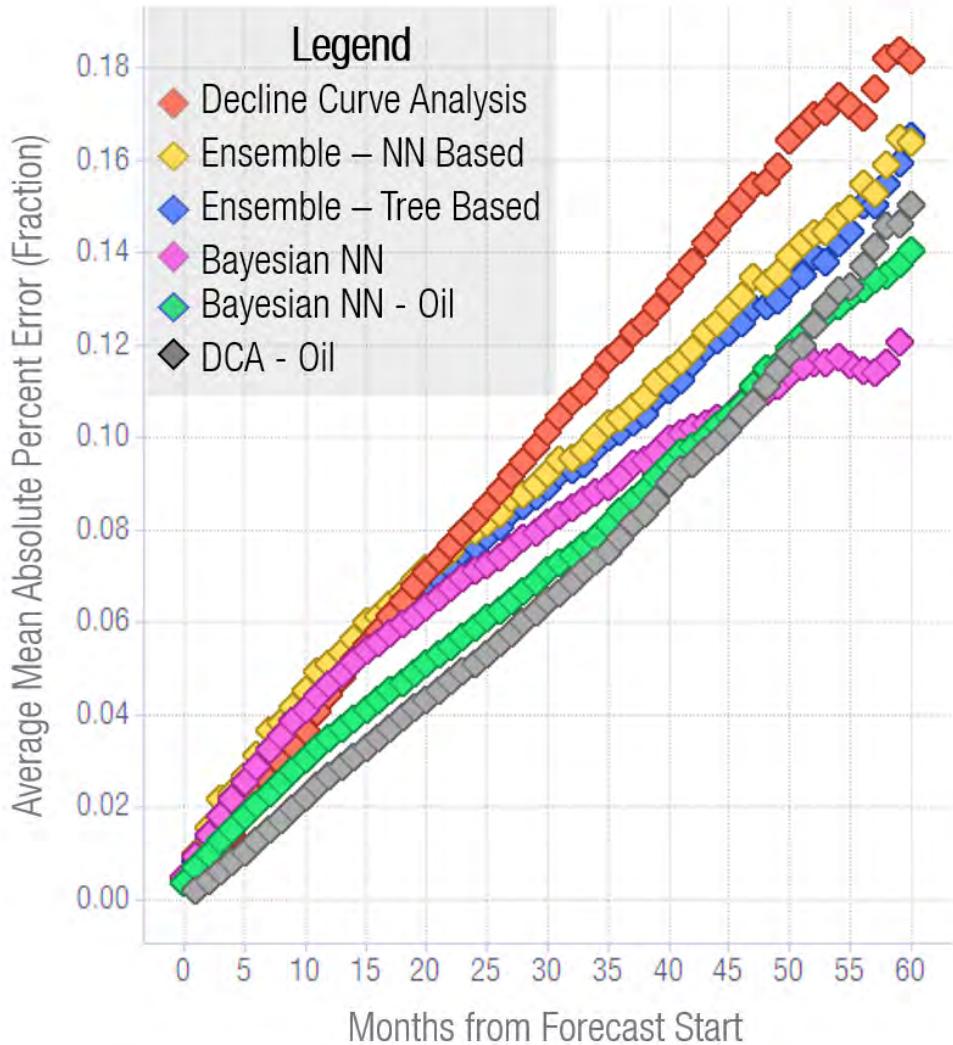


Figure B-21: Plot of MAPE versus the start of forecast ( $t$ ). All Probabilistic ML algorithms demonstrate improvements over DCA after 25 months for gas. The BNN has the strongest results, which increase in significance with  $t$ . Oil performance was roughly equivalent with DCA, with both of these methods outperforming the gas results until month 50 of the forecast.

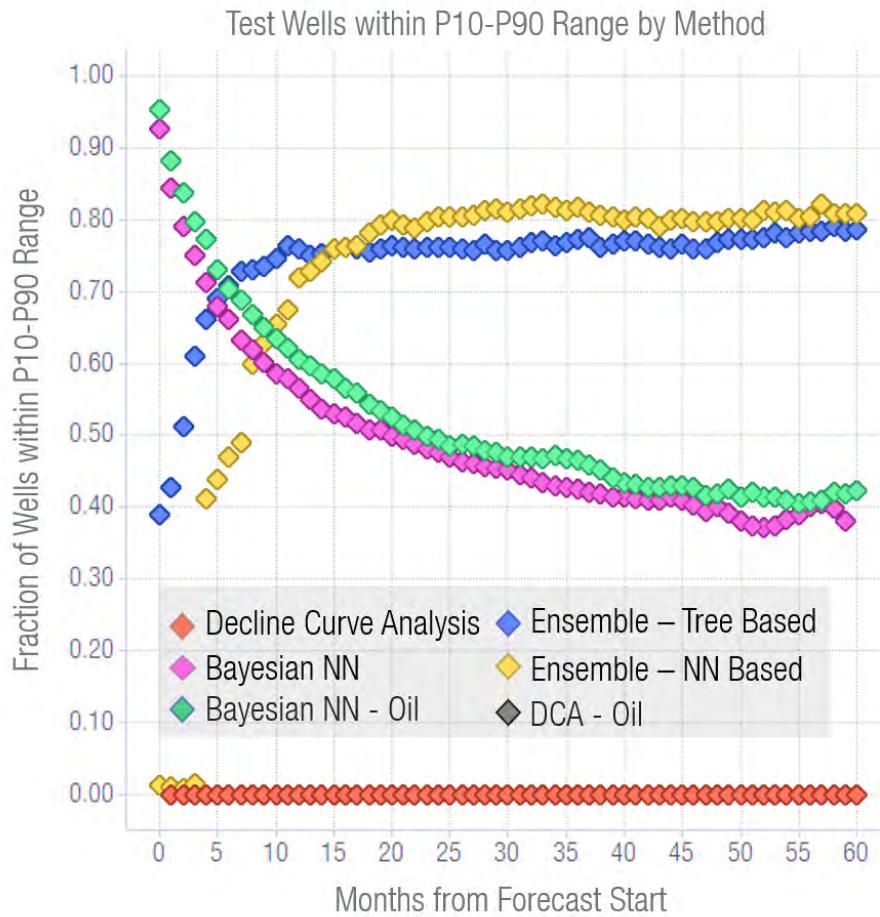


Figure B-22: Percentile range results for each forecasting method, depicting the percentage of wells with actual production within the bounds of the P10–P90 range. The BNN and Ensemble methods show opposite effects, with the BNN capturing a significant portion of the wells initially and dropping to 40% as  $t$  increases. Results between gas and oil in the BNN were roughly equivalent.

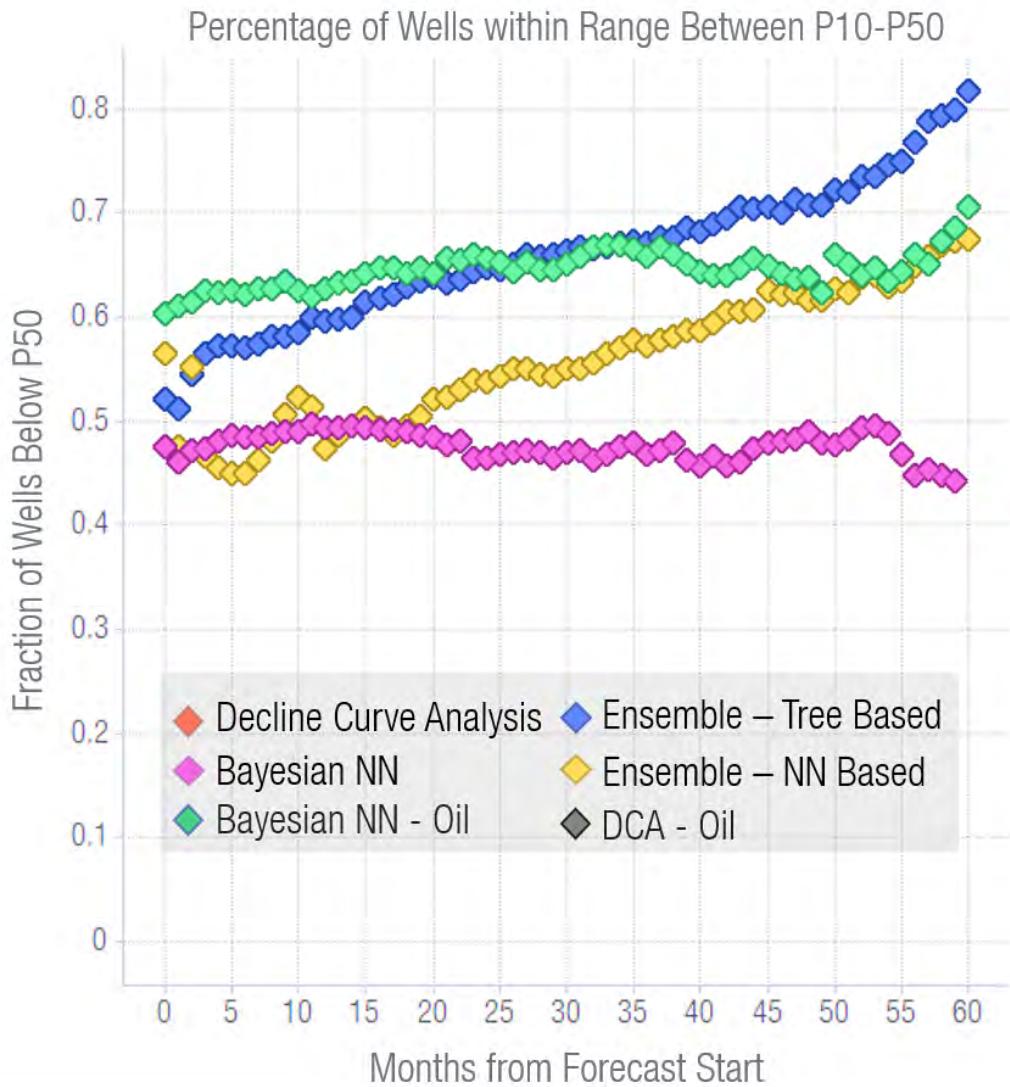


Figure B-23: Percentile-range isolating for the percentage of wells below P50 and within the P10–P90 range. Ideally, this value would be 50% to have equal skew, preferring the BNN method for gas. The BNN applied to oil had similar results as the Ensemble methods.

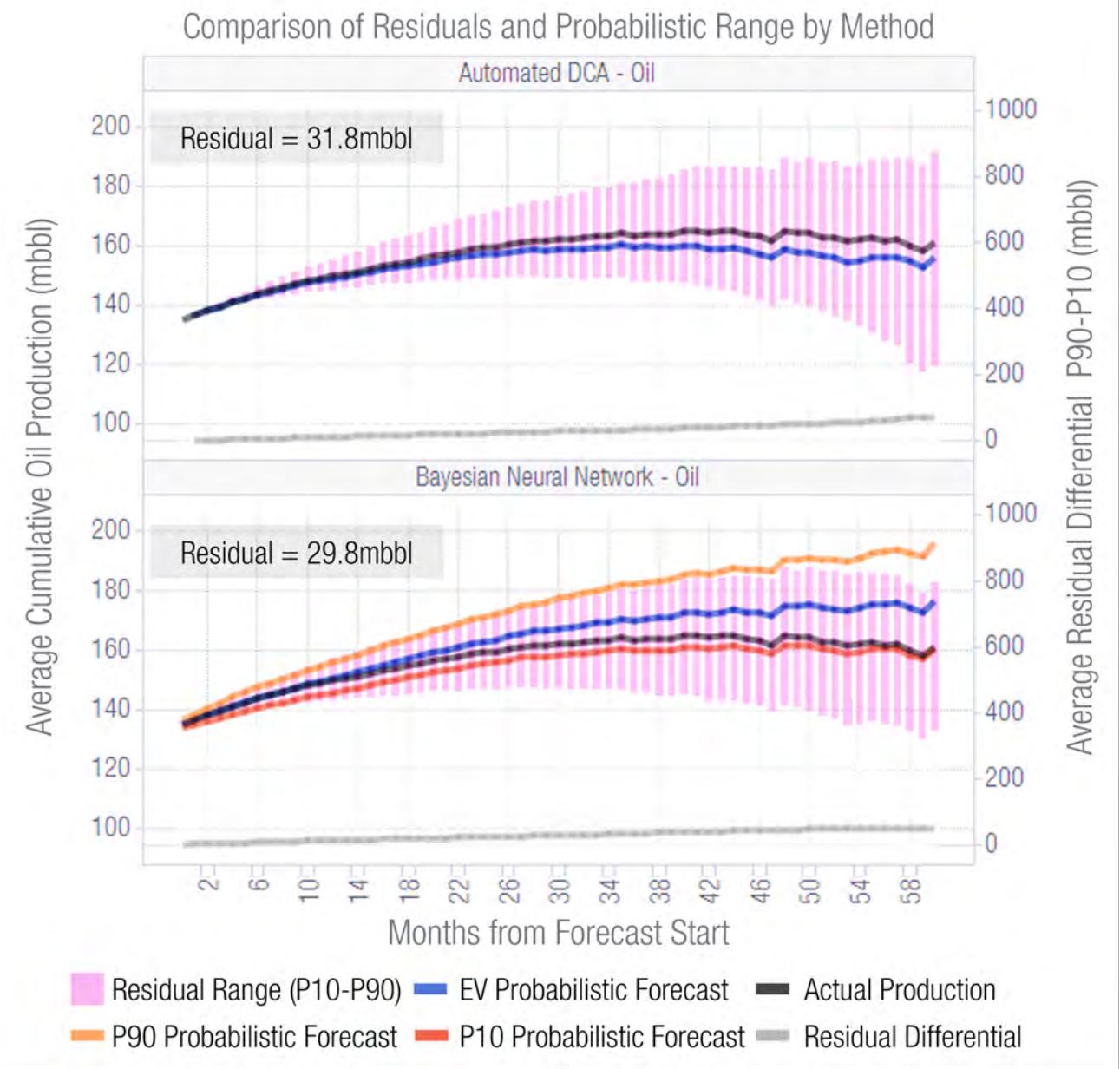


Figure B-24: BNN reasonably captures the P90 result within the residual range. Oil production seems to be skewed to overestimate actuals, which fall within the P10 EV of the forecast in aggregate. The residuals are also negatively skewed below the P10 forecast indicating that this prediction is optimistic. These results are the opposite behaviour of the gas forecast, and indicate that some large-scale PVT effects are creating modelling error.

## B.4 Sweet Spot Mapping Full Dataset

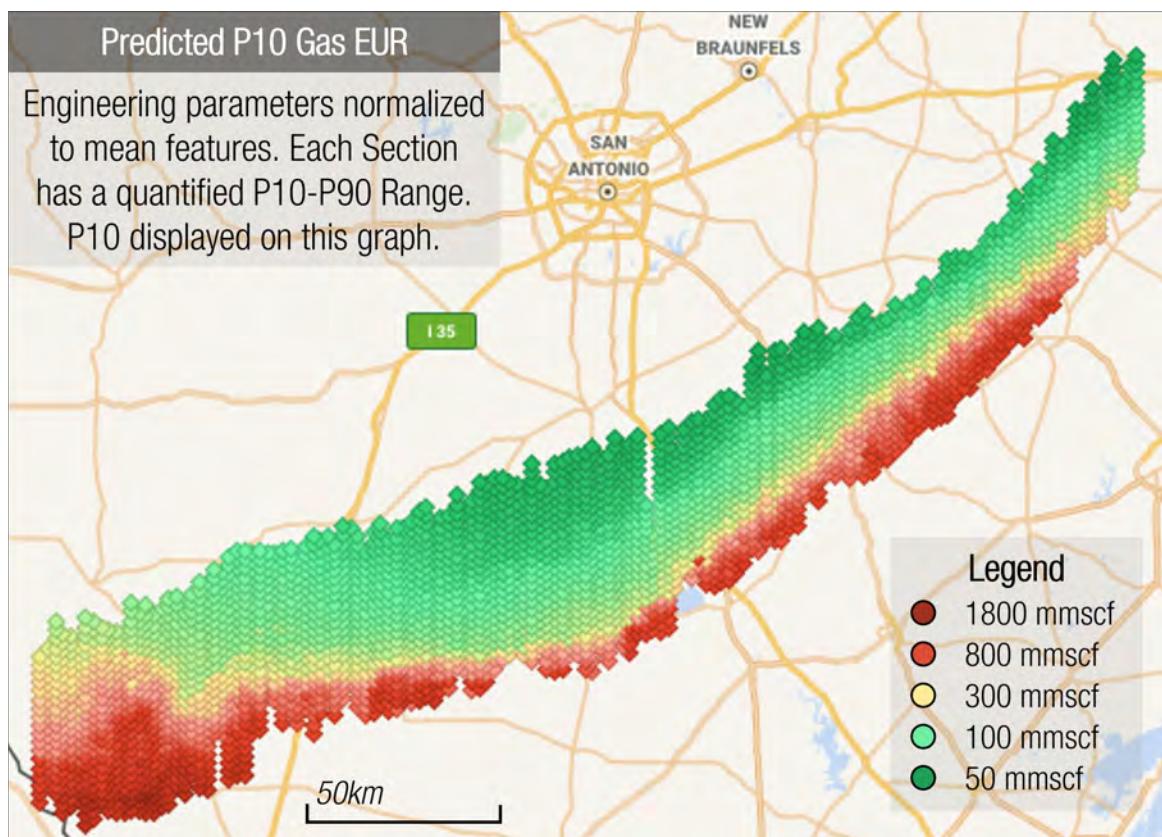


Figure B-25: Gas Hotspot map illustrating the best-performing areas in red and the worst in green for the P10 forecast. Each grid cell corresponds with a  $2 \times 2 \text{ km}^2$  section for the mean EUR; however, each section also has a probabilistic estimate for P10 and P90, following a lognormal distribution. Gas results show smooth transitions progressing from high recovery to low, following the approximate inclination of the reservoir. Visually, this map is nearly identical to the EV as the relative performance differences for each section are similar across the field; this result is due to a consistent P90/P10 ratio.

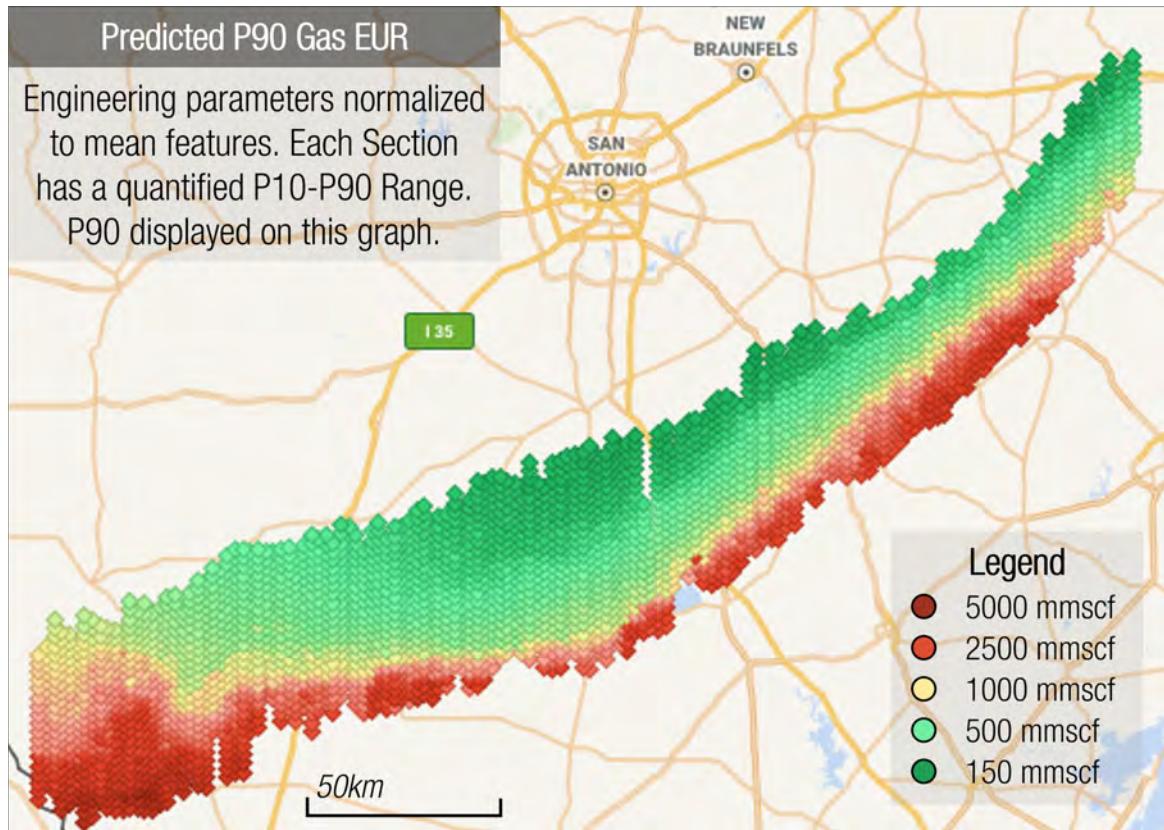


Figure B-26: Gas Hotspot map illustrating the best-performing areas in red and the worst in green for the P90 forecast. Each grid cell corresponds with a  $2 \times 2 \text{ km}^2$  section for the mean EUR; however, each section also has a probabilistic estimate for P10 and P90, following a lognormal distribution. Gas results show smooth transitions progressing from high recovery to low, following the approximate inclination of the reservoir. Visually, this map is nearly identical to the EV as the relative performance differences for each section are similar across the field; this result is due to a consistent P90/P10 ratio.

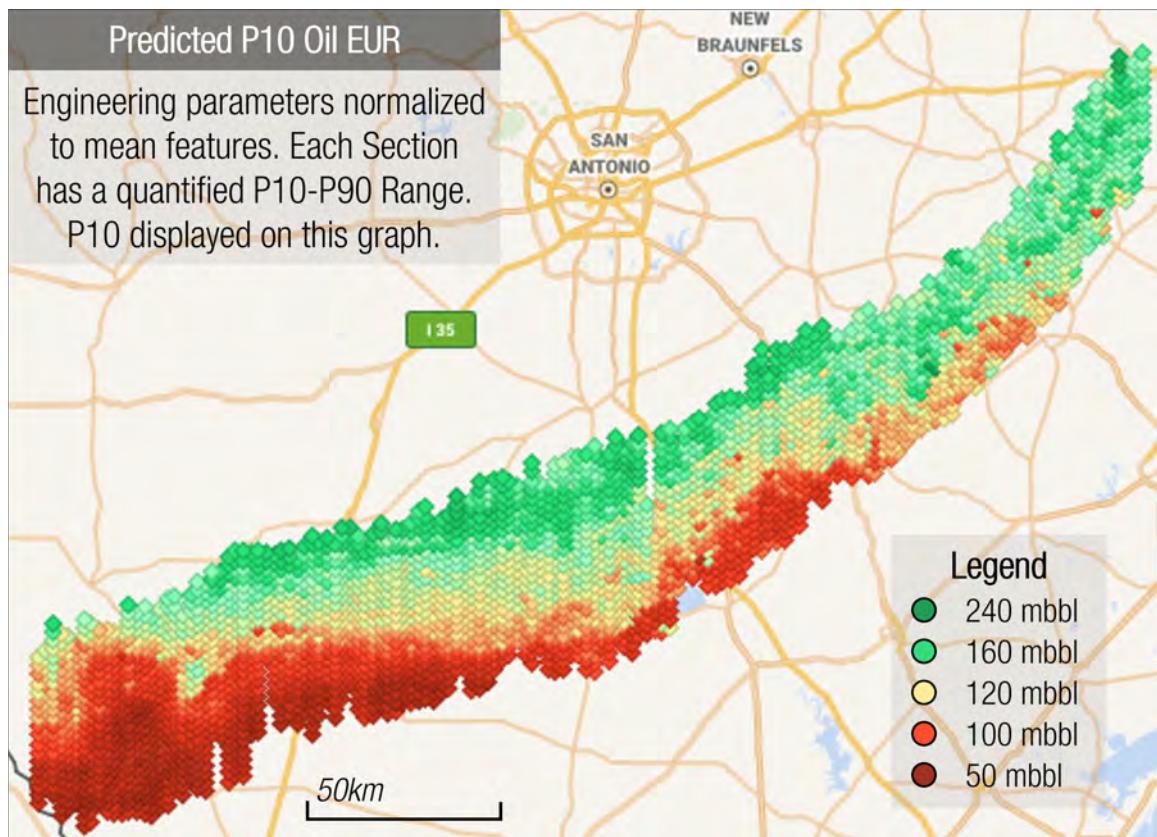


Figure B-27: Oil Hotspot map illustrating the best-performing areas in green and the worst in red for the P10 forecast. Oil results show more volatility compared to gas, with harsher transitions between high and low productivity regions. Visually, this map does differ compared to the EV with localized areas with relative upside and downside risks.

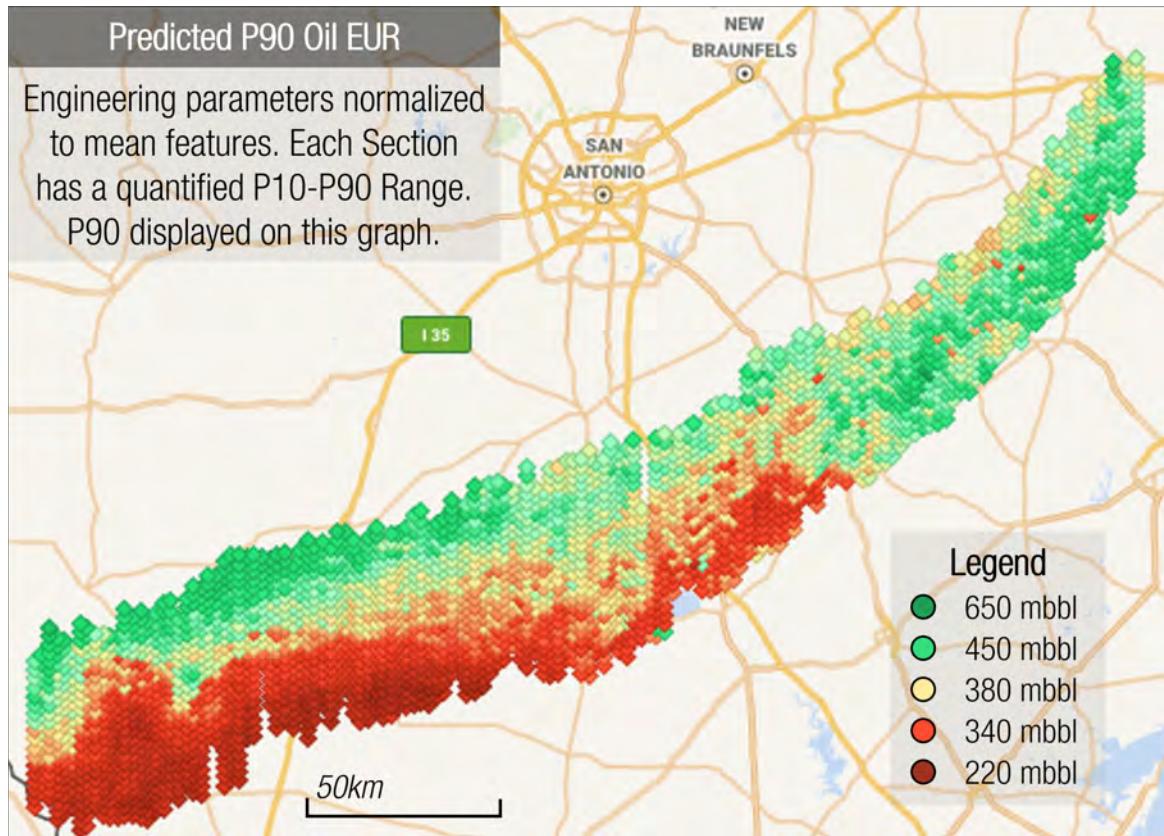


Figure B-28: Oil Hotspot map illustrating the best-performing areas in green and the worst in red for the P90 forecast. Oil results show more volatility compared to gas, with harsher transitions between high- and low-productivity regions. Visually, this map does differ compared to the EV with localized areas with relative upside and downside risks.

## B.5 Restricted Dataset Forecasts

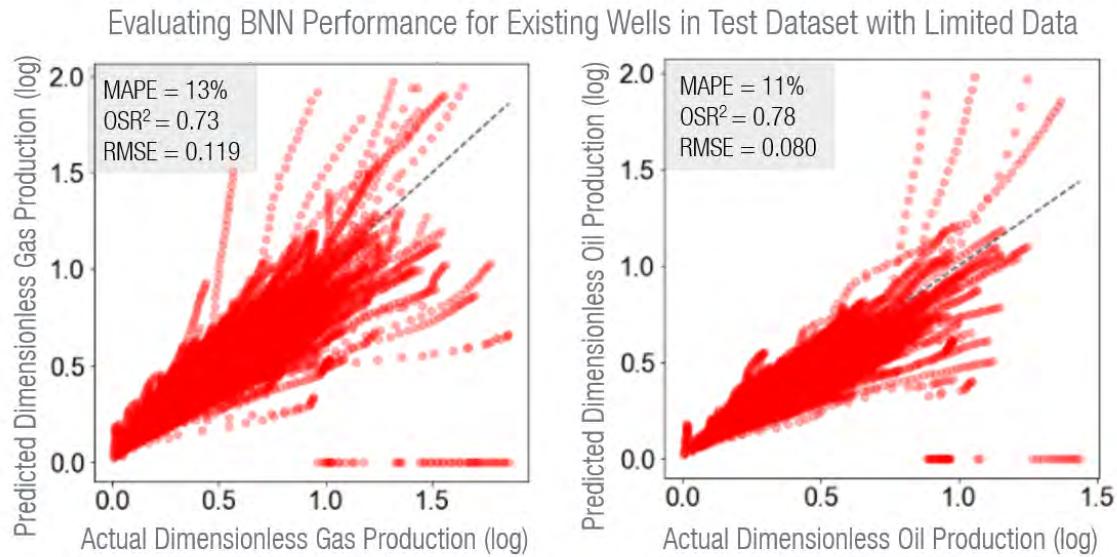


Figure B-29: BNN Model performance at predicting well Gas production on data limited to be before 2016. This plot demonstrates the test results for gas and oil, with only the initial static and dynamic lags prior to forecast start provided. MAPE results were consistent with the full dataset but increased by approximately 6-8%.

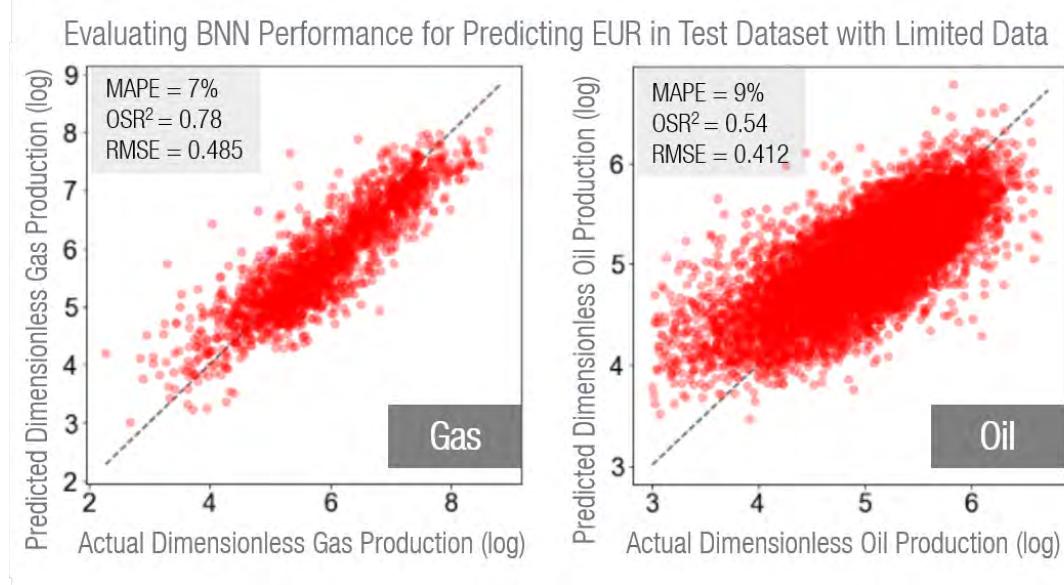


Figure B-30: BNN Model performance at predicting well Gas production on data limited to come from before 2016. This plot demonstrates the test results for gas and oil, without the inclusion of dynamic lags. MAPE results were consistent with the full dataset but increased by approximately 2%–4%.

# Bibliography

- [1] Heteroscedasticity.
- [2] Christopher Ahlberg. Spotfire: An Information Exploration Environment. *SIGMOD Rec.*, 25(4):25–29, December 1996. Place: New York, NY, USA Publisher: Association for Computing Machinery.
- [3] Maher J. Alaboodi and Shahab D. Mohaghegh. Conditioning the Estimating Ultimate Recovery of Shale Wells to Reservoir and Completion Parameters. September 2016. SPE-184064-MS.
- [4] Alexander Amini and Ava Soleimany. Deep Learning Lecture Notes (6.S191), January 2021.
- [5] Masoud Alfi, Cheng An, Yang Cao, Bicheng Yan, Maria Barrufet, and John Killough. *Pore Size Variability and Sieving Effect in Liquid Shale—A Multiple Permeability Approach and Eagle Ford Case Study*. February 2017.
- [6] Algorithmia. Introduction to loss functions, April 2018.
- [7] Alexander Amini, Wilko Schwarting, Ava Soleimany, and Daniela Rus. Deep evidential regression. *Advances in Neural Information Processing Systems*, 33, 2020.
- [8] J.J. Arps. Analysis of Decline Curves. *Transactions of the AIME*, 160(01):228–247, December 1945.
- [9] Rose and Associates, Jim Gouveia, and Bryce Kalynchuk. Unconventional Resource Assessment and Valuation. Calgary, Canada, 2016. Rose and Associates.

- [10] Y.. Bansal, T.. Ertekin, Z.. Karpyn, L.. Ayala, A.. Nejad, Fnu Suleen, O.. Balogun, D.. Liebmann, and Q.. Sun. Forecasting Well Performance in a Discontinuous Tight Oil Reservoir Using Artificial Neural Networks. April 2013. SPE-164542-MS.
- [11] Bart Van Parys. The Analytics Edge Course Notes, 2020.
- [12] Bill Vail. *Monograph 3, Guidelines for the Practical Evaluation of Undeveloped Reserves in Resource Plays*, volume Monograph 3. Society of Petroleum Evaluation Engineers, 2010.
- [13] T.A. Blasingame, J.L. Johnston, and W.J. Lee. Type-Curve Analysis Using the Pressure Integral Method. April 1989. SPE-18799-MS.
- [14] Braden Bowie. Machine Learning Applied to Optimize Duvernay Well Performance. March 2018. D021S008R003.
- [15] Chris Carpenter. Method Quantifies Separator-Oil Shrinkage. *Journal of Petroleum Technology*, 72(12):48–49, December 2020.
- [16] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight Uncertainty in Neural Network. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, pages 1613–1622. PMLR, June 2015.
- [17] Tianqi Chen and Carlos Guestrin. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD ’16, pages 785–794, New York, NY, USA, 2016. ACM. event-place: San Francisco, California, USA.
- [18] Nidhan Choudhuri, Subhashis Ghosal, and Anindya Roy. Bayesian Methods for Function Estimation. In D.K. Dey and C.R. Rao, editors, *Handbook of Statistics*, volume 25, pages 373–414. Elsevier, January 2005.

- [19] C.R. Clarkson. Production data analysis of unconventional gas wells: Review of theory and best practices. *International Journal of Coal Geology*, 109-110:101–146, April 2013.
- [20] Crude Oil Logistics Committee. Forecasting Procedures, January 2021.
- [21] Cosma Shalizi. Lecture 10: F-Tests, R<sup>2</sup>, and Other Distractions, October 2015.
- [22] Edwin Crow and Kunio Shimizu. *Lognormal Distributions : theory and applications*. Marcel Dekker, 1988.
- [23] Felipe Cruz, Ali Tinni, Steve Brackeen, Carl Sondergeld, and Chandra Rai. Impact of Capillary Pressure on Total Porosity in Unconventional Shales. July 2020. D023S026R006.
- [24] Anh N. Duong. Rate-Decline Analysis for Fracture-Dominated Shale Reservoirs. *SPE Reservoir Evaluation & Engineering*, 14(03):377–387, May 2011.
- [25] Eric Rebentisch and Tom Toner. System Design and Management Lecture Notes (EM412), January 2021.
- [26] Eric W. Weisstein. Log Normal Distribution.
- [27] ESRI. How Kriging works, 2016.
- [28] IHS Fekete. Rate Transient Analysis Poster, 2013.
- [29] M.J. Fetkovich. Decline Curve Analysis Using Type Curves. *Journal of Petroleum Technology*, 32(06):1065–1077, June 1980.
- [30] Francis M. O’Sullivan and Justin B. Montgomery. The Salient Distribution of Unconventional Oil and Gas Well Productivity, December 2015.
- [31] Randy Freeborn and Boyd Russell. Creating More-Representative Type Wells. *SPE Economics & Management*, 8(02):50–58, April 2016.

- [32] JOSE LUIS GARCIA-GUTIERREZ and Georgina Laredo. Molecular size evaluation of linear and branched paraffins from the gasoline pool by DFT quantum chemical calculations. *Fuel*, 83:2183–2188, January 2004.
- [33] Abbas Ghandi, Sonia Yeh, Adam Brandt, Kourosh Vafi, Hao Cai, Michael Wang, Bridget Scanlon, and Robert Reedy. *Energy Intensity and Greenhouse Gas Emissions from Crude Oil Production in the Eagle Ford Region: Input Data and Analysis Methods*. September 2015.
- [34] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.
- [35] Siavash Hakim Elahi. A Novel Workflow for Oil Production Forecasting using Ensemble-Based Decline Curve Analysis. September 2019. D022S090R001.
- [36] Dongkwon Han, Sunil Kwon, Hanam Son, and Janghyun Lee. Production Forecasting for Shale Gas Well in Transient Flow Using Machine Learning and Decline Curve Analysis. November 2019. D022S028R003.
- [37] J. D. Hunter. Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007. Publisher: IEEE COMPUTER SOC.
- [38] ICE NGX. Nominations & Scheduling, 2021.
- [39] IHS Fekete. Volumetric Analysis Theory.
- [40] IHS Fekete. Traditional Decline Analysis Theory, 2014.
- [41] IHS Markit. IHS Markit Eagle Ford Database, 2020.
- [42] Plotly Technologies Inc. Collaborative data science, 2015. Place: Montreal, QC  
Publisher: Plotly Technologies Inc.
- [43] AOAC INTERNATIONAL. Accuracy, Trueness, Error, Bias, Precision, and Uncertainty: What Do These Terms Mean?, March 2015.

- [44] A.F. Ismail, K.C. Khulbe, and T. Matsuura. *Gas Separation Membranes: Polymeric and Inorganic*. Springer International Publishing, 2015.
- [45] Ivan Olea, Adam Chin, and Louis Mattar. Drawdown Management Literature Review to Investigate Changing Fluid Ratio, April 2017.
- [46] JAMES CODY STATTON. *Application of the Stretched Exponential Production Decline Model to Forecast Production in Shale Gas Reservoirs*. MASTERS OF SCIENCE, Texas A&M University, May 2012.
- [47] Jessica Yung. MSE as Maximum Likelihood, June 2018.
- [48] John Lee. Type Well Analysis – Complexities and Analytic Techniques. May 2017.
- [49] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pages 3149–3157, Red Hook, NY, USA, 2017. Curran Associates Inc. event-place: Long Beach, California, USA.
- [50] Olalekan Keshinro, Yetunde Aladeitan, Olugbenga Oni, Jemimah-Sandra Samuel, and Jaja Adagogo. Improved Decline Curve Analysis Equations – Integration of Reservoir Properties into Arps Equation. August 2018. SPE-193419-MS.
- [51] Kevin Webster. Probabilistic Deep Learning with TensorFlow 2, 2020.
- [52] Y. Kocoglu, M. E. Wigwe, G. Sheldon, and C. Watson. Machine Learning Based Decline Curve - Spatial Method to Estimate Production Potential of Proposed Wells in Unconventional Shale Gas Reservoirs. July 2020. D013S011R006.
- [53] Utpalendu Kuila, Manika Prasad, and Hossein Kazemi. Assessing Knudsen flow in gas-flow models of shale reservoirs, May 2013.

- [54] J.F. Lea, H.V. Nickens, and M. Wells. *Gas Well Deliquification: Solutions to Gas Well Liquid Loading Problems*. Gulf drilling guides. Elsevier Science, 2003.
- [55] Boxiao Li, Travis C. Billiter, and Timothy Tokar. Significant Error Reduction in Machine-Learning Decline Curve Analysis for Unconventional Reservoirs. July 2020. D013S003R001.
- [56] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaqiang Zheng. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. 2015.
- [57] Matlab. Kalman Filtering, 2021.
- [58] Riley McFall, Kevin Mullen, Jason Baihly, Garrett Lindsay, and Jung Shin. Refracturing in the Eagle Ford Shale: One Operator’s Quest to Identify and Rank Candidates, Minimize Well Interference, and Understand Variability of Results. July 2017. URTEC-2691375-MS.
- [59] Patrick Miller, Keane Dauncey, and James Gouveia. Appropriately Characterizing Uncertainty in Estimated Ultimate Recovery for Unconventional Type Wells. October 2020. D031S026R008.
- [60] Oliver Wilson. *Machine Learning for Well Rate Estimation: Integrated Imputation and Stacked Ensemble Modeling*. Master of Engineering in System Design and Management, MASSACHUSETTS INSTITUTE OF TECHNOLOGY, September 2020.

- [61] Daniel Orozco, Alfonso Fragoso, Karthik Selvan, Graham Noble, and Roberto Aguilera. Eagle Ford Huff ‘n’ Puff Gas-Injection Pilot: Comparison of Reservoir-Simulation, Material Balance, and Real Performance of the Pilot Well. *SPE Reservoir Evaluation & Engineering*, 23(01):247–260, February 2020.
- [62] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [63] Boyd Russell and Randy Freeborn. A Practical Guide to Unconventional Petroleum Evaluation Part 2. November 2013. SPE-167215-MS.
- [64] Boyd Russell, Randy Freeborn, and Wayne Keinick. A Practical Guide to Unconventional Petroleum Evaluation. October 2012. SPE-158867-MS.
- [65] Scott Sanderson, Andrew Slaughter, and Tom Bonny. Deciphering the performance puzzle in shales, October 2019.
- [66] Scott Sanderson, Andrew Slaughter, and Tom Bonny. The Eagle Ford basin playbook: Focusing on capital efficiency, October 2019.
- [67] SECURITIES AND EXCHANGE COMMISSION. Modernization of Oil and Gas Reporting, 2009.
- [68] AKASH SINGH. Anomaly detection for temporal data using Long Short-Term Memory (LSTM). Master’s thesis, Royal Institute of Technology, STOCKHOLM, SWEDEN, August 2017.
- [69] Pavel Sountsov, Chris Suter, Jacob Burnim, and Joshua V. Dillon. Regression with Probabilistic Layers in TensorFlow Probability, March 2019.
- [70] Stanley Robertson. Generalized Hyperbolic Equation. *Society of Petroleum Engineers*, (SPE-18731-MS), August 1988.
- [71] T.A. Blasingame. Introduction to Phase Behavior, 2014.

- [72] The pandas development team. *pandas-dev/pandas*: Pandas, February 2020.
- [73] Tensorflow. TensorFlow Probability, 2021.
- [74] R.E. Terry, J.B. Rogers, and B.C. Craft. *Applied Petroleum Reservoir Engineering*. Prentice Hall, 2015.
- [75] Tonio Buonassisi, Danny Ren, and Liu Zhe. Applications of Machine Learning in Engineering Science (2.S986) Course Notes, September 2020.
- [76] Peter P. Valkó and W. John Lee. A Better Way to Forecast Production from Unconventional Gas Wells. September 2010. SPE-134231-MS.
- [77] Guido Van Rossum and Fred L. Drake. *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA, 2009.
- [78] Bert Van Schaeybroeck and Stéphane Vannitsem. A Probabilistic Approach to Forecast the Uncertainty with Ensemble Spread. *Monthly Weather Review*, 144(1):451–468, January 2016. Place: Boston MA, USA Publisher: American Meteorological Society.
- [79] Vincenzo Lavorini. Gaussian Mixture Model clustering: how to select the number of components (clusters), November 2018.
- [80] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

- [81] Michael Waskom, Olga Botvinnik, Drew O’Kane, Paul Hobson, Saulius Lukauskas, David C. Gemperline, Tom Augspurger, Yaroslav Halchenko, John B. Cole, Jordi Warmenhoven, Julian de Ruiter, Cameron Pye, Stephan Hoyer, Jake Vanderplas, Santi Villalba, Gero Kunter, Eric Quintero, Pete Bachant, Marcel Martin, Kyle Meyer, Alistair Miles, Yoav Ram, Tal Yarkoni, Mike Lee Williams, Constantine Evans, Clark Fitzgerald, Brian, Chris Fonnesbeck, Antony Lee, and Adel Qalieh. mwaskom/seaborn: v0.8.1 (September 2017), September 2017.
- [82] Weisstein, Eric W. Correlation Coefficient, 2021.
- [83] J.. Yip, P.. Miller, and T.. Branter. Predicting Economic Value in Unexplored Regions of an Unconventional Liquids Rich Play. March 2018. D021S008R004.