# Multi-horizon Well Performance Forecasting with Temporal Fusion Transformers

**Eduardo Maldonado-Cruz1, Michael J. Pyrcz1,2**

1 – Hildebrand Department of Petroleum and Geosystem Engineering, Cockrell School of Engineering, The University of Texas at Austin

2 – Department of Geological Sciences, Jackson School of Geosciences, The University of Texas at Austin

**Keywords**: Deep Learning, Temporal Fusion Transformer, Machine Learning, Interpretability, Explainable AI, Multi-well forecasting, History matching

# Introduction

Accurately predicting fluid flow is critical for making informed decisions about the development of subsurface resources such as groundwater, geothermal energy, and oil and gas. Forecasting provides valuable insights into complex rock and fluid systems, helping to optimize well productivity. (water, hydrocarbon, and heat), maximize resource recovery, and maximize project economics. Accurate forecasting is essential to operate, design, and develop oil and gas fields (Liu et al., 2020).

Traditional production forecasting methods include physics-based reservoir flow simulation models, semi-analytical models, decline curves, and more recently, surrogate flow models based on data-driven methods. The surrogate flow model is a function that approximates reservoir flow simulation and is utilized to reduce computational expenses. This approach facilitates timely and informed development decision-making. All these methods base their forecasts on a subset of historical production data. However, production forecasting depends on multivariate reservoir parameters and complex non-linear fluid flow behavior that is challenging to capture through conventional forecasting techniques (Zhou Q. et al., 2014; Cunningham C. et al., 2012; Mohammadmoradi et al., 2018; Hui G. et al., 2021; Esmali S. et al., 2012).

The use of numerical simulation with reservoir flow simulation models represents the most prevalent method for forecasting petroleum field production. Nevertheless, the accuracy and precision of said models is contingent upon the quality of the geological model as well as the input parameters utilized to match with actual reservoir performance. (Ostojic et al., 2012; Xu et al., 2018). The creation of a geological model involves several underlying assumptions that include boundary conditions, fluid compressibility, reservoir heterogeneity, and data utilized to perform grid upscaling (Michelevichuius et al., 2002). Despite best efforts to match the model with actual reservoir performance, the unavailability of rock and fluid properties poses a significant challenge to achieving accurate production forecasting. Short-term events may not be accurately forecasted by a reservoir simulator (i.e., changes in operation controls), particularly in reservoirs with high heterogeneity (Parvizi et al., 2017; Sun et al., 2019).

Semi-analytical and analytical approaches utilize simplified flow physics to build closed-form solutions. These approaches are used for forecasting well production forecasting and incorporate multiple properties related to fluid flow in the porous media. There are no convergence issues compared to numerical flow simulation, and they are more computationally efficient, especially for large reservoir models (Wang et al., 2019; Zongxiao et al., 2016). A drawback of these methods is that due to the simplified physics, they are unable to include the effect of production constraints, provide general solutions for any possible forecast scenario, include static properties such as position, and are limited to one-at-a-time well analysis (Rahuma et al., 2013; Clarkson, 2013; Mattar and Anderson, 2003).

Traditional decline curve analysis (DCA) uses empirical formulas of production decline patterns to fit single-well production data (Li et al., 2022). DCA models are founded on straightforward mathematical principles, requiring only a handful of parameters, and calibration of these parameters require only production data (Arps, 1945; Gentry, 1972). However, various decline curve models can be applied to conventional and unconventional wells (Pickup., 2013). Moreover, determining the optimal model for use can be a complex task as multiple models may fit a dataset with similar precision yet provide different forecasts. Selecting a single model for probabilistic decline curve analysis (DCA) may result in underestimating the uncertainty in a production forecast (Fanchi, 2010; Belyadi et al., 2019).

More recently, data-driven approaches based on surrogate flow models to perform production forecasts have been demonstrated (Kubota and Reinert, 2019; Davtyan et al., 2020; Liu et al., 2020; Zhong et al., 2020; Chahar et al., 2022; Cao et al., 2022; Iskandar and Kurihara, 2022). These approaches only consider field response data to perform forecasts (i.e., historical oil rate) and suffer similar limitations as DCA and semi-analytical approaches. They are unable to include the effect of production constraints, are limited to one-at-a-time well analysis, and selecting a single model for probabilistic forecasting can underestimate the uncertainty in a production forecast (Werneck, R. de O., 2022; Li et al., 2022). In addition, these approaches are typically black-box models with low model interpretability, which is the difficulty to understand and explain how a model makes its forecasts or decisions. This is important because it builds trust in the model and its outputs and enables users to identify potential biases, errors, or limitations (Carvalho et al., 2019). A method to improve model interpretability through feature importance is SHapley Additive exPlanations (SHAP), proposed by Lundberg and Lee (2017). The SHAP explanation method computes Shapley values that come from cooperative game theory and are used to distribute the prediction among the predictor features as a linear additive model. For example, SHAP provides intuitive feature importance through treeSHAP, a variant particularly suited to tree-based machine learning models like decision trees, to offer comprehensible feature importance metrics.

Standard machine learning approaches to build surrogate flow models use deep neural networks (DNN) (e.g., recurrent neural networks and long short-term memory networks). These surrogates are trained to generate forecasts for predefined horizons at each time step, and the architectures typically rely on a sequence-to-sequence model, where the sequence is a series of numbers or events recorded over time (e.g., production rate). However, these surrogates use only the target response feature to generate future forecasts of the target response feature (e.g., historical production rate and pressure measurements), even when numerous time-varying and static features exist (e.g., location, depth, scheduled operations), omitting essential information for the forecasting problem. In addition, long sequences suffer from the vanishing gradient problem (Hochreiter, 1998). This problem occurs when there are multiple layers in the DNN, and the model loses the ability to propagate useful gradient information from the output layer back to the input and early hidden layers of the model.

The transformer architecture (Vaswani et al., 2017) solves the vanishing gradient problem using a multi-head attention-based architecture for sequential data for natural language processing (Wolf et al., 2020; Yang et al., 2020) and image processing (Deng et al., 2022; Pirnay and Chai, 2022, Xiang et al., 2023; Liu and Jung, 2021). The multi-head attention is a critical component of the transformer architecture, where the attention mechanism is implemented as self-attention. An input sequence, , of elements with predictor features, is projected using three trainable weight matrices , and called query, , key, , and value , (see Equation 1), which are used to compute attention scores for each position in the sequence described in Equation 2, to extract feature representations.

|  |  |  |
| --- | --- | --- |
|  |  |  |

The dot product in equation 2 determines the contribution of the position to the final representation of the series (Vaswani et al., 2017).

|  |  |  |
| --- | --- | --- |
|  |  |  |

Where softmax denotes a row-wise softmax normalization function, and is the dimension of the attention vector. Each element in the attention vector depends on all other elements in the same row. The most relevant vectors are assigned the highest weights through the softmax operation. Using the attention mechanism, the model “attends” positions with the highest weights of the input sequence in parallel instead of relying on a recurrent structure to learn long and short-term relationships across different time steps (Lin et al., 2022), and then concatenates the resulting matrices before multiplying them by a trainable weight matrix to obtain the final multi-head attention matrix.

|  |  |  |
| --- | --- | --- |
|  |  |  |

The Temporal Fusion Transformer (TFT) is a variant of the Transformer architecture. It is a novel attention-based architecture that learns temporal relationships at various scales. The TFT is designed to forecast multi-objective time series, data sequences from multiple sources, and temporal dependencies (Lim et al., 2021). Given unique measured points in a time series dataset, each element is associated with a set of static predictor features (i.e., position), dynamic predictor features and response features (i.e., oil rate, water rate, or BHP) at each time step . The time-dependent predictor features are divided into observed predictor features which are measured at each step and are unknown in the future (i.e., oil rate, water rate, or BHP), and known predictor features , which are known in the future (i.e., scheduled injection rate). Then each forecast using quantile regression is given by Equation 4:

|  |  |  |
| --- | --- | --- |
|  |  |  |

Where is the th forecasted quantile of the step forecast at time , is the TFT model, and the sub-index is a finite look-back window.

The Temporal Fusion Transformer (TFT) uses an encoder-decoder architecture. The encoder part of the TFT takes the time series data and processes it in a sequence of encoder layers. Each encoder and decoder layer has multiple self-attention heads that attend to different parts of the input sequence. The output of the encoder is a set of hidden states, one for each time step in the input sequence (Lim et al., 2021). A hidden state is a vector of values representing a sequence that contains information about the temporal relationships between the predictor features (Jurafsky and Martin, 2009). These hidden states are the inputs of the decoder part of the TFT.

The decoder is responsible for generating the forecasts. It consists of a sequence of decoder layers. The decoder input is hidden states from the encoder and processes them using self-attention. The output of the decoder is the input of a feedforward neural network that generates the final forecasts for each time step in the forecast (Sutskever et al., 2014).

The TFT proposes a modified multi-head attention given by Equation 5, where the multi-head attention shares values in each head by aggregating all attention heads, (Lim et al., 2021). This representation is necessary because attention weights alone are not indicative of a particular predictor feature importance (Tay et al., 2021).

|  |  |  |
| --- | --- | --- |
|  |  |  |

Machine learning-based surrogate flow models have demonstrated low forecast errors during testing. However, these models are deterministic, do not include critical static predictor features, are only able to analyze one well at a time, and are prone to autocorrelation, leading to shifted forecasts due to high correlation with the most recent available observations at different times (Werneck, R. de O., 2022; Li et al., 2022). Furthermore, the lack of interpretability of results reduces confidence in broader applications for development decision support (Gurina et al., 2022).

To improve current surrogate flow model approaches for forecasting, we propose a novel and general workflow to generate a TFT-based surrogate flow model. Through the problem formulation, we include static (i.e., depth, position) and dynamic (i.e., injection rate, operational constraints) predictor features, forecast across multiple wells, and uncertainty estimation through the quantile loss function. We propose two static predictor features to include location, and to improve model interpretability, we calculate SHAP values to relate forecasts to attention head weights at each time step. Our proposed TFT-based surrogate flow model performs forecasts of oil and water rate and injection pressure, provides interpretability and allows the identification of persistent temporal patterns, predictor feature importance, and detection of significant events during forecasting.

The methodology section presents our novel workflow to construct a TFT-based surrogate flow model, the proposed location-based predictor features, the evaluation metrics, the training, testing, and hyperparameter tuning details, and our method to evaluate the attention head weights. The results section presents the results on the maximum forecast length and the training and testing results with our TFT-based surrogate flow model over the three constructed scenarios. We present the limitations and considerations of the models and concisely discuss the attention weights and predictor feature importance for every scenario.

# Methodology

Our proposed workflow augments conventional reservoir flow simulation, decline curve analysis, and analytical methods with a TFT-based surrogate flow model by providing forecasts of bottom-hole injection pressure, water, and oil rate. Figure 1 shows the inputs, outputs, and the TFT-based surrogate flow model schematically.

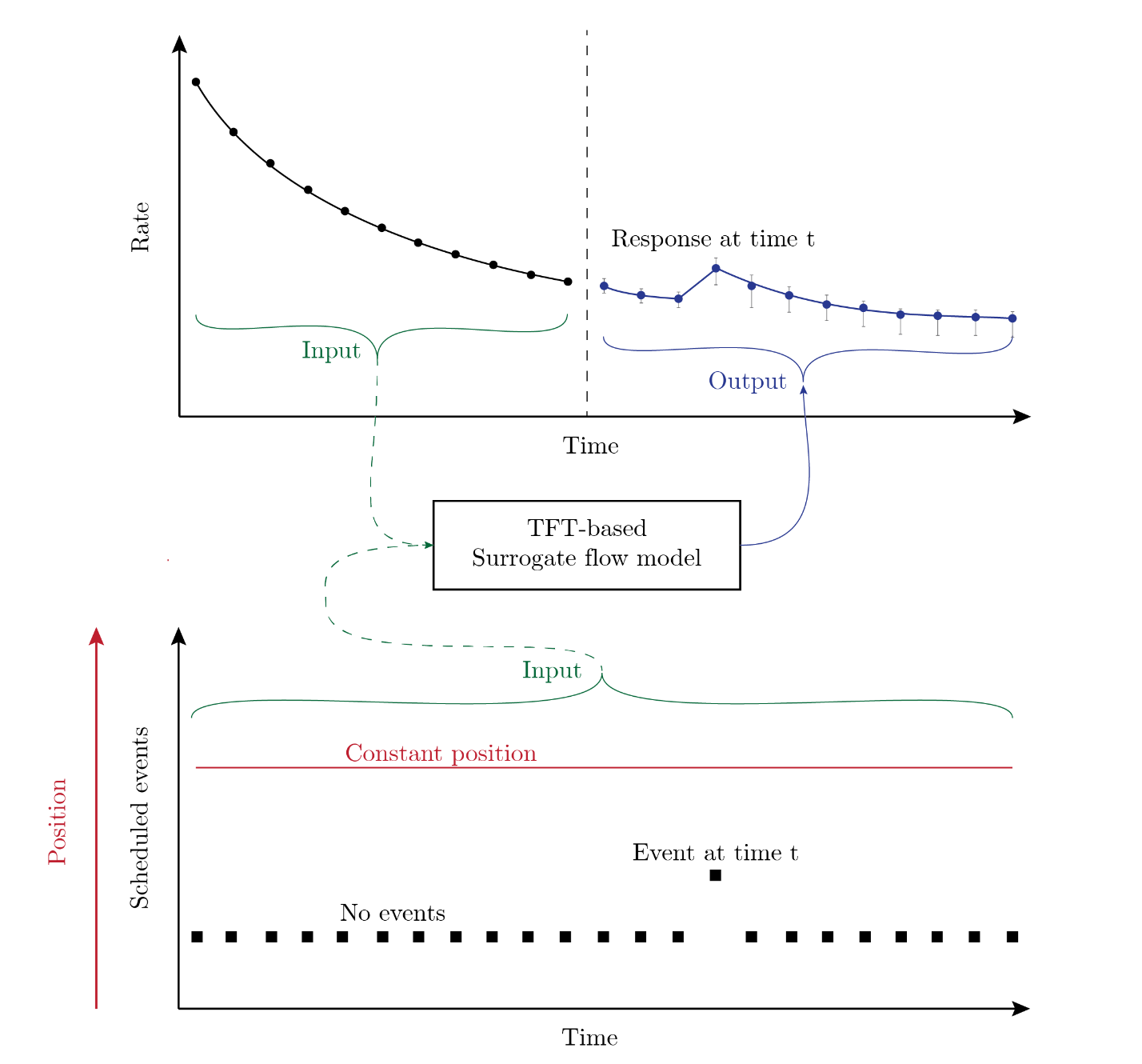


Figure .- Schematic of the forecasting problem containing static and dynamic predictor features.

To construct a TFT-based surrogate flow model for forecasting with uncertainty and interpretable outputs, we propose the following algorithm:

**Input**: Time-dependent predictor features (i.e., oil rate, pressure measurements, scheduled injection rates), static features (i.e., position, distance to injector well), and response features (oil rate, water rate, and injection pressure).

**Output**: TFT-based surrogate flow model for forecasting with uncertainty and interpretable outputs.

1. Prepare the dataset and perform data pre-processing. Remove anomalies, inconsistent observations, and erroneous measurements, add a time index and predictor features.
2. Create a baseline benchmark using the last observed value from the series.
3. Perform hyperparameter importance to select the hyperparameters to tune.
   1. For combination of hyperparameters:
      1. Perform model training with selected hyperparameters.
      2. Scale the predictor features dynamically during training.
      3. Select the combination of hyperparameters that minimize the loss function.
4. Perform model training using the quantile loss function and the selected hyperparameters from step 3.
   1. Scale the predictor features dynamically during training.
5. For every time step in the testing dataset:
   1. Extract attention head weights
   2. Interpret predictions using SHAP and calculate Shapley values.
   3. Return the average of the absolute Shapley values for every feature.

**Return** TFT-based surrogate flow model with interpretable outputs.

## Location-based predictor features

To include spatial predictor features in the dataset, we propose the following predictor features to relate the formation flow capacity () and storage capacity () with the well location. First, we define a fixed point with coordinates ( ) (i.e., the position of the injector well, the shortest distance to a fault). Then using equations 6 and 7, we calculate the proposed predictor features for every well in the dataset.

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

where is the absolute permeability, is the thickness, and is the porosity of well . and are coordinates where the subindex refers to the well number and subindex refers to the fixed point.

## Model evaluation metrics

A common occurrence in previous works is that forecasts of pressure and rate are shifted by the lag used during training caused by autocorrelation. This phenomenon occurs when the input sequence is not informative enough to produce reliable forecasts (Werneck, R. de O., 2022; Li et al., 2022). Then, the neural network repeats the last observed value. We use this concept to create two basic benchmark metrics to outperform based on the Mean Absolute Error (MAE) and Symmetric Mean Absolute Percentage Error (SMAPE) metrics in equations 8 and 9 (Kubota and Reinert, 2019; Cao et al., 2016; Sun et al., 2018).

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

where is the testing data value at step , is the testing data value at step and M is the total number of testing data values.

We evaluate the TFT-based surrogate flow model results with the MAE in equation 10 and the SMAPE in equation 11, which measures the percentage error of the forecasted values.

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

where is the forecasted value, and is the testing data value.

## Model training and testing

We train the TFT-based surrogate flow model for 100 epochs with the training dataset containing static and dynamic predictor features to make short-term forecasts of water rate, oil rate, and water injection pressure. The selection of the number of epochs is empirical, based on observing the error versus epoch curves over multiple model training runs for a specific problem. The training of the model uses a train-test methodology (Beleites and Salzer, 2008; Beleites et al., 2013; Xu and Goodacre, 2018). During training, we dynamically standardize the input data to avoid the look-ahead bias.

We use the quantile loss function to forecast with uncertainty. This loss function results in non-symmetrical confidence intervals compared to previous approaches (Maldonado-Cruz and Pyrcz, 2021; Maldonado-Cruz and Pyrcz, 2022). To evaluate the uncertainty of a non-symmetrical distribution, we use the interquartile range (IQR), the difference between the 75th percentile and the 25th percentile. The IQR measures the spread of the middle 50% of the data. During training, the output sequence uses the quantile loss function for training over the maximum forecast length using equation 12.

|  |  |  |
| --- | --- | --- |
|  |  |  |

Where is a quantile, is the forecasted value and is the testing data value. If we select and perform training with the quantile loss, then the MAE function has its minimum at the median.

## Hyperparameter tuning.

To select which hyperparameter to tune, we first quantify hyperparameter importance by using functional analysis of variance, this method partitions the observed variation into components due to each of its hyperparameters (Hutter et al., 2014). The hyperparameters selected for tuning are dropout, learning rate, attention head size, gradient clipping, and hidden size of the network. Dropout is a regularization technique used in deep learning to prevent overfitting (the model fits noise or idiosyncrasies) by randomly setting neurons to zero during training (Hastie et al., 2009). The learning rate controls the step size the optimization algorithm takes during the training process and determines how quickly the model updates its parameters in response to the error or loss function. The attention head size refers to the number of attention heads used in the model. Larger attention head sizes can capture more complex relationships but require more computing and memory to train and run. Gradient clipping prevents the gradients from becoming too large during the training of a neural network (Goodfellow et al., 2016). Finally, the hidden size of a neural network refers to the number of neurons in the hidden layer of the network (LeCun et al., 2015). This hyperparameter determines the capacity of the network to learn complex patterns and relationships from the data. The rest of the hyperparameters maintain their default values.

## Interpretability analysis of attention heads

To improve interpretability, we propose using SHAP values to relate forecasts to attention head weights at each time step. We extract the attention head weights from the encoder and decoder at every time step of the testing dataset. Each attention head has a dimension of , by , where is the number of predictor features, and is the size of the encoder length (look-back periods) or prediction length (forecast periods), each with corresponding attention weights. We then use a tree-based model (i.e., random forest) to compute SHAP values (Lundberg et al., 2018), which relate the forecasts to attention head weights for each time step. Tree-based models allow fast and accurate calculation of Shapley values through tree pruning to determine all necessary marginal contributions from a single model. Plotting attention head weights at each time step provides a basic interpretation, but as problem complexity and the number of features increase, we need a robust method such as Shapley values to attribute feature importance. The underlying principle of SHAP feature importance is based on the size of the absolute Shapley values, with greater values indicating greater importance. To derive the overall importance, the average of the absolute Shapley values is computed for each predictor feature across the entire dataset, as depicted in Equation 13.

|  |  |  |
| --- | --- | --- |
|  |  |  |

Where is the global importance of feature , are the Shapley values of the feature and is the total number of time steps in the testing data. We use the original definition proposed by Lim et al. (2021) to evaluate the static predictor feature importance. By examining Shapley values of the attention head weights at every time step, one can determine the impact of predictor features on forecasting, thereby improving model interpretability and enabling the practitioner engineer to identify significant features for forecasting and events that may affect well productivity.

# Results and discussion

The proposed surrogate flow model, is a valuable tool for advanced diagnostics by forecasting, identifying patterns, and anticipating well responses. This section presents the results for the optimum encoder length (look-back periods), prediction length (forecast periods), and the results from training and testing the TFT-based surrogate flow model with three synthetic datasets. We opt for a synthetic training dataset to mitigate the impact of exogenous events unrelated to the forecasting problem (i.e., defective measurements and human errors) and to provide comprehensive testing over diverse set of known truth models. Additionally, we discuss the results and interpretability of the attention weights.

## Scenario construction

To generate synthetic geological models of porosity () and permeability (), Sequential Gaussian Simulation (SGS) was employed. This was implemented using the open-source Python package GeostatsPy (Pyrcz et al., 2021), which utilizes a reimplementation of the original GSLIB Fortran Geostatistical Library (Deutsch and Journel, 1992). The spatial distribution of the features is defined by their variogram, which is utilized in univariate and spatial modeling. The porosity realization is first computed and subsequently used to determine the corresponding collocated co-kriging permeability realization. The Pearson correlation coefficient is assumed to be 0.9, and the permeability realization is confined within the bounds of the porosity realization. The univariate permeability distribution ranges from 1 μD to 1.2 D, while the univariate porosity distribution varies from 0 to 0.2 in fraction. A representative example of the generated permeability () and porosity () realizations is depicted in Figure 2.

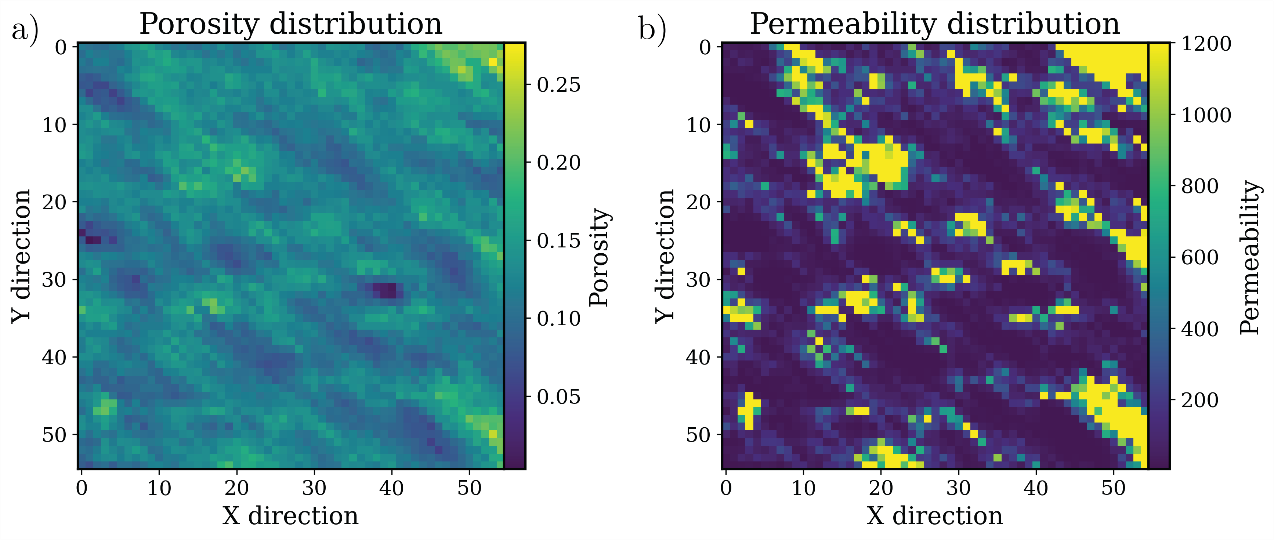


Figure .- An illustration of the porosity and permeability distributions employed to generate the training datasets is presented in subfigure a) porosity distribution and (b) the permeability distribution.

Subsequently, a finite differences reservoir flow simulator (Aziz and Settari, 1979) is employed to generate a fit-for-purpose production dataset. We explore primary and secondary recovery scenarios using the porosity and permeability realizations from SGS. The simulation model operates by controlling injection wells through the specification of time-varying water injection rates, while production wells are regulated using time-varying well bottom-hole pressure (BHP). To create three distinct scenarios, the initial pressure at the top of the reservoir is set to 3800 psi, and the initial water saturation is fixed at 0.20.

## Scenario description

Scenario A explores a primary depletion example comprising production data from 7 producer wells under constant bottom-hole pressure conditions. The objective is to forecast the oil rate. All wells are located along a constant value in the x-axis, as observed in Figure 3. We hypothesize that the importance of the static predictor features will be higher for the y-position feature. We add scheduled events during simulation time to represent: no events, changes in choke size, and minor well repairs. These changes are represented by changing the bottom-hole pressure by constant values of 10 and 100 psi., respectively. In Figure 4, we show the production history from Well #5 and the events from the production history.

Chart

Description automatically generated

Figure .- Well distribution along the simulation grid. All wells are producing under constant bottom-hole pressure from the same formation. We maintain a constant well location along the x-axis.

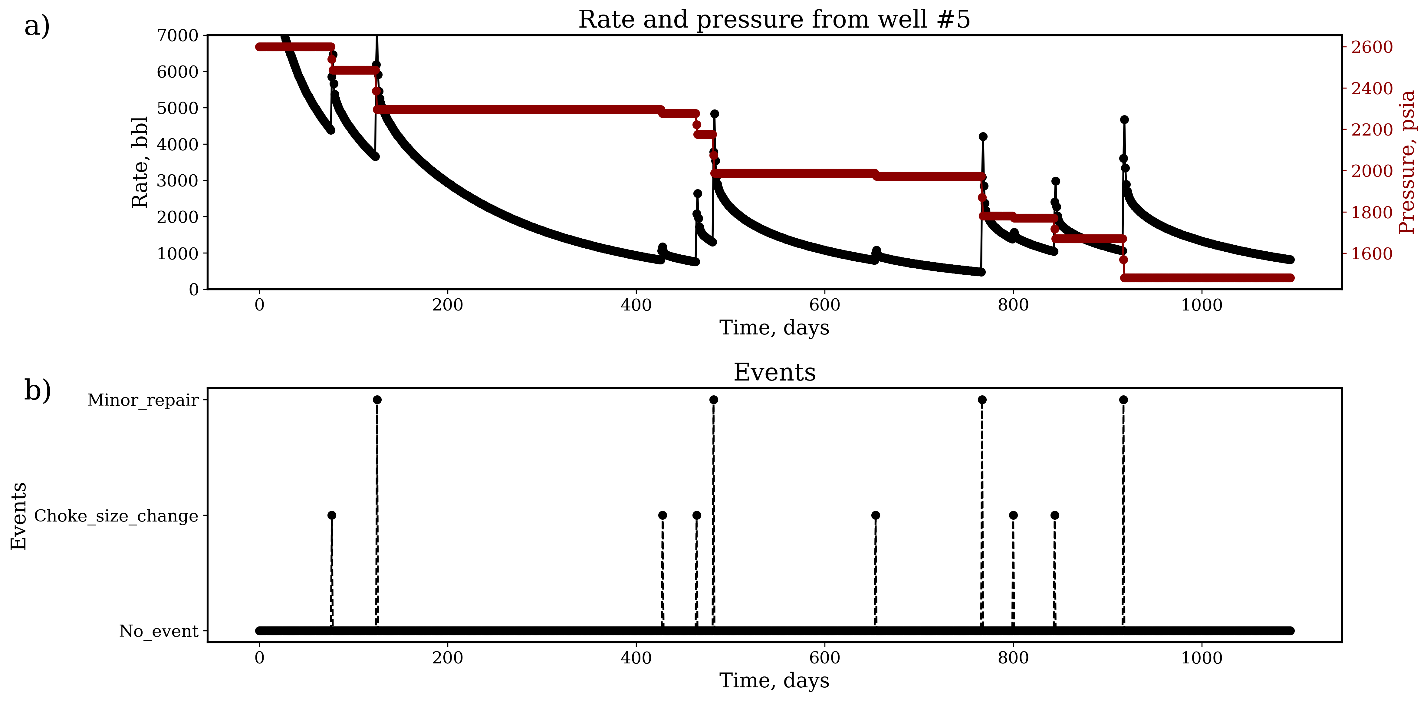


Figure .- Production history of Well #5 from scenario A. In subfigure a) we show the production history for 2000 days, while in subfigure b) we show the type and time of the random events that occur in the production history.

Scenario B is a replica of scenario A, except that we add random Gaussian noise to the oil rate. The objective is to forecast oil rates in the presence of noisy data and to show that attention head weights and Shapley feature importance are not affected by noisy measurements.

Scenario C explores a secondary recovery example and comprises production and injection data from a simple 5-spot injection pattern, as observed in Figure 5. The objective is to forecast the oil rate, water rate, and water injection pressure to show the applicability of our workflow to a more complicated problem with more predictor and response features. In Figure 6, we show the oil and water production history of well P1 in black and blue colors. In red, we plot the water bottom-hole injection pressure from well I1.

Graphical user interface

Description automatically generated

Figure .- Schematic plot of well location for a 5-spot injection pattern. P refers to producer wells, and I to injector wells.

Chart, line chart

Description automatically generated

Figure .- Production history of well P1 from the simulator, including the oil rate (black line), the water rate (blue line), and bottom-hole injection pressure (red line) for well I1.

For all three scenarios, the reservoir flow simulation model is run for 1100 days, from which we obtain static and dynamic predictor features, for example, oil rate, water rate, water cut, bottom-hole pressure, porosity, permeability, thickness, initial pressure, well name, position, production layer, and formation. The data from the numerical simulator is transformed into a tabular form where each row can be identified with a time step and a corresponding response feature. Table 1 shows a truncated example of a dataset used to train the TFT-based surrogate flow model. The dataset consists of multiple entries with corresponding static and dynamic predictor features to forecast pressure, water, and oil rate.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Well** | **Oil rate** | **Water rate** | **Pressure** | **Porosity**  **Fraction** | **Permeability**  **mD** | **Feature kh** | **Feature**  **frac** | **Date** | **…** | **Thickness** | **Formation** |
| 0 | 598.4 | 0 | 3800.00 | 0.099 | 0.64 | 0.05456 | 0.0013 | 1/1/2018 | … | 43 | 1 |
| 0 | 501.8 | 0.008 | 3714.90 | 0.099 | 0.64 | 0.05456 | 0.0013 | 1/2/2018 | … | 43 | 1 |
| 0 | 490.3 | 0.010 | 3687.55 | 0.099 | 0.64 | 0.05456 | 0.0013 | 1/3/2018 | … | 43 | 1 |
| 0 | 483.4 | 0.012 | 3671.39 | 0.099 | 0.64 | 0.05456 | 0.0013 | 1/4/2018 | … | 43 | 1 |
| … | … | … | … | … | … | … | … | … | … | … | … |
| 4 | 213.5 | 2561 | 3679.51 | 0.066 | 0.18 | 0.00510 | 0.0051 | 10/9/2020 | … | 33 | 1 |
| 4 | 213.2 | 2562 | 3679.38 | 0.066 | 0.17 | 0.00510 | 0.0051 | 10/10/2020 | … | 33 | 1 |
| 4 | 212.9 | 2563 | 3679.25 | 0.066 | 0.17 | 0.00510 | 0.0051 | 10/11/2020 | … | 33 | 1 |

Table .- Truncated example of the dataset. We include a mixture of static, dynamic and engineered predictor features.

## Results on the optimum encoder and decoder prediction length

Using the datasets from scenarios A, B, and C, we perform a grid search to determine the optimum encoder length (training time window, look-back periods) and decoder prediction length (testing time window, forecast periods). We follow the following algorithm:

Input: Time-dependent predictor features, static features, and response features

Output: Optimum encoder and decoder prediction length

1. Using the datasets from Scenarios A, B, and C, separate the dataset between training and testing. We use the last 180 days of oil rate production history for testing.
2. For every encoder length and decoder prediction length combination
   1. Perform hyperparameter importance to select the hyperparameters to tune.
      1. For combination of hyperparameters:
         1. Perform model training with selected hyperparameters.
            1. Scale the predictor features dynamically during training.
         2. Select the combination of hyperparameters that minimize the quantile loss function.
      2. Return combination of hyperparameters
   2. Perform model training using the combination of hyperparameters
      1. Scale the predictor features dynamically during training.
   3. Perform model testing and return the average MAE for the encoder and decoder prediction length combination.

Return Optimum encoder and decoder prediction length.

For encoder length, we evaluate from 10 to 210 days in steps of 10, and for decoder prediction length, we evaluate from 2 to 42 days in steps of 2. We evaluate models to determine the encoder and decoder prediction lengths that minimize the error metric in equation 10. Figure 7 shows the search space colored by the average MAE metric. We observe low-error forecasts in models with small decoder prediction lengths. However, these models are not helpful as the forecast window (decoder prediction length) is too small. We would opt for models with long forecast windows and small encoder lengths with the lowest MAE.

Background pattern

Description automatically generated

Figure .- Search space colored by the mean absolute error. We select the encoding length and decoder prediction length that minimizes the MAE metric.

## Evaluation of scenario A

Using the dataset from scenario A, we evaluate the performance of our proposed location predictor features in equations 8 and 9. We train models A1 and A2 following the proposed workflow in the methodology section and use the MAE and SMAPE metrics in equations 2 and 3 as error metrics to evaluate the model performance on testing data. We use the last 180 days from the production history as testing data and the rest of the history as training data. Model A1 uses the location predictor features, and Model A2 does not include the location predictor features. Figure 8 summarizes the error distributions for Models A1 and A2 based on the MAE and SMAPE metrics. We observe that using the proposed predictor features in equations 8 and 9 improves the generalization ability of the TFT-based surrogate flow model. Table 2 summarizes the hyperparameters selected for the TFT-based surrogate flow model used for scenario A.

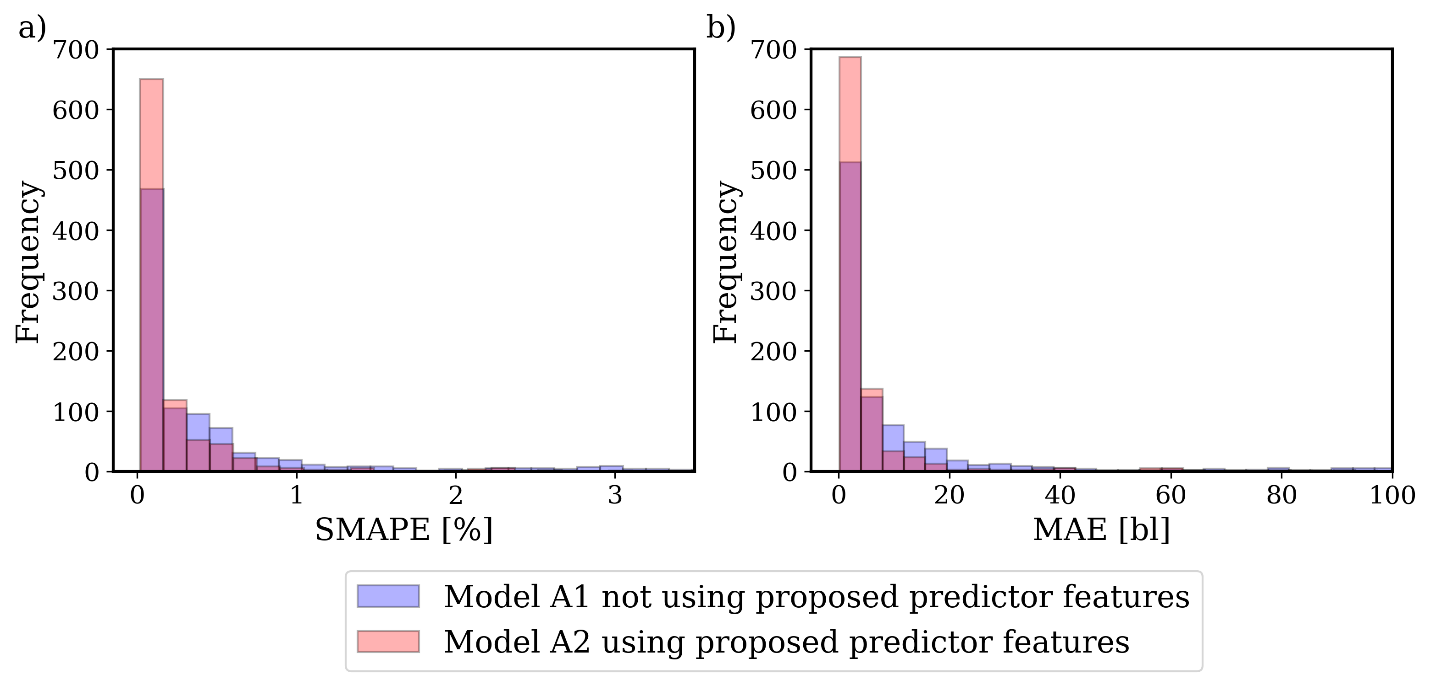


Figure .- Error distribution over the testing dataset using our proposed predictor features (red) and not using our proposed predictor features (blue). In a) we show the distribution of SMAPE, while b) shows the distribution of the MAE.

Table .- Hyperparameters used for the TFT-based surrogate flow model for scenario A.

|  |  |
| --- | --- |
| **Hyperparameter** | **Value** |
| gradient\_clip\_val | 0.9939 |
| hidden\_size | 90 |
| Dropout | 0.178735 |
| hidden\_continuous\_size | 65 |
| attention\_head\_size | 2 |
| learning\_rate | 0.01366059 |
| Epochs | 100 |

Using model A1, we construct Figure 9, which shows a histogram with the testing performance results, and Figure 10 shows examples of the forecasts obtained by the TFT-based surrogate flow model over Well #2. We select this well because the forecasting period contains no events, resulting in a smooth curve with little uncertainty in the forecasted rates by the TFT-based surrogate flow model. Figure 11 presents the IQR for every forecasted value presented in Figure 10.

Chart, histogram

Description automatically generated

Figure .- Error distribution over the testing dataset from scenario A. In a) we show the distribution of SMAPE, while b) shows the distribution of the MAE.

Chart

Description automatically generated

Figure .- Forecasting with the TFT-based surrogate flow model. In a) we use the last 180 days of the production history for testing. In this figure we show the forecasts for time step 1020. The red points are the forecasted data, and the black points are the measured values. In shadows of grey, we show the P10 and P90 confidence intervals. The confidence intervals for this figure are small, indicating little uncertainty in future forecasts. In subfigure b) we show the events for the period. We observe no events in this period.

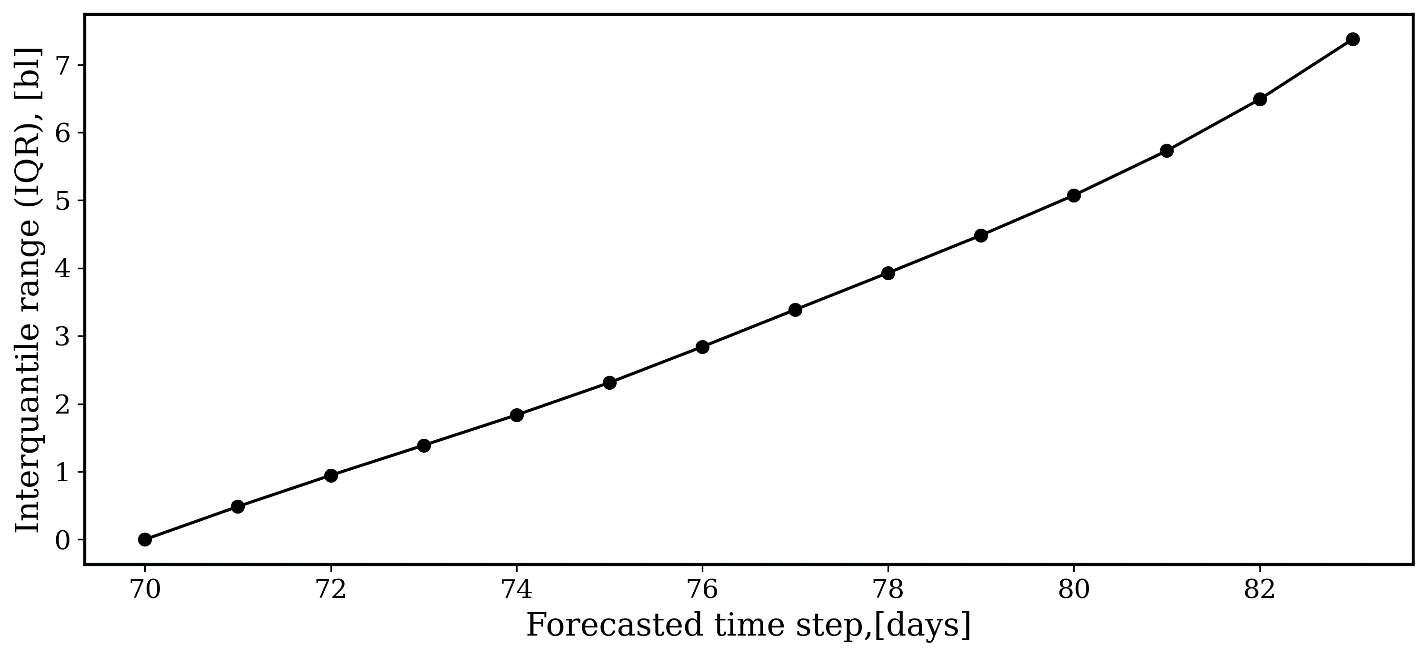


Figure .- To evaluate the uncertainty in the resulting non-symmetrical distribution at every time step, we use the IQR. The IQR measures the spread of the middle 50% of the data at every time step. For the case presented in Figure 10, we observe small values of the IQR.

To complement Figure 10, we plot the encoder and decoder attention head weights and the events predictor feature in Figure 12. We observe equal attention weights over periods with no events for the encoded part of the time series. However, for the decoder, the initial forecasted rates must be consistent with the observed historical production. Resulting in greater attention weights for the first forecasted steps decreasing until zero.

Chart, line chart

Description automatically generated

Figure .- In subfigure a) we show the oil rate forecast and encoder-decoder attention weights; in subfigure b) we show the events predictor feature. The attention head weights show a consistent trend of high attention at the beginning of the forecasts and decrease until the last time step.

Figure 13 shows a forecasting period at time 1020 for Well #5 containing a scheduled choke size change. We observe an increase in the forecasted oil rate resulting from the change in choke size. Compared with Figure 10, the forecasted uncertainty increases, as observed with the confidence intervals and the IQR for every time step, as shown in Figure 14. To complement Figure 13, we include Figure 15 with the attention head weights. This figure shows increased attention to sharp changes corresponding to events for the decoder part. Greater attention weights are expected as we would pay more attention to changes in the schedule for our forecasts.

Diagram

Description automatically generated

Figure .- Example of oil forecasting for a well with scheduled events a) The black line represents the start of the forecast. The red points are the forecasted oil rate, and the black points are the measured rates. In shadows of grey, we show the P10 and P90 confidence intervals. In subfigure b) we show a change in choke size in this period.

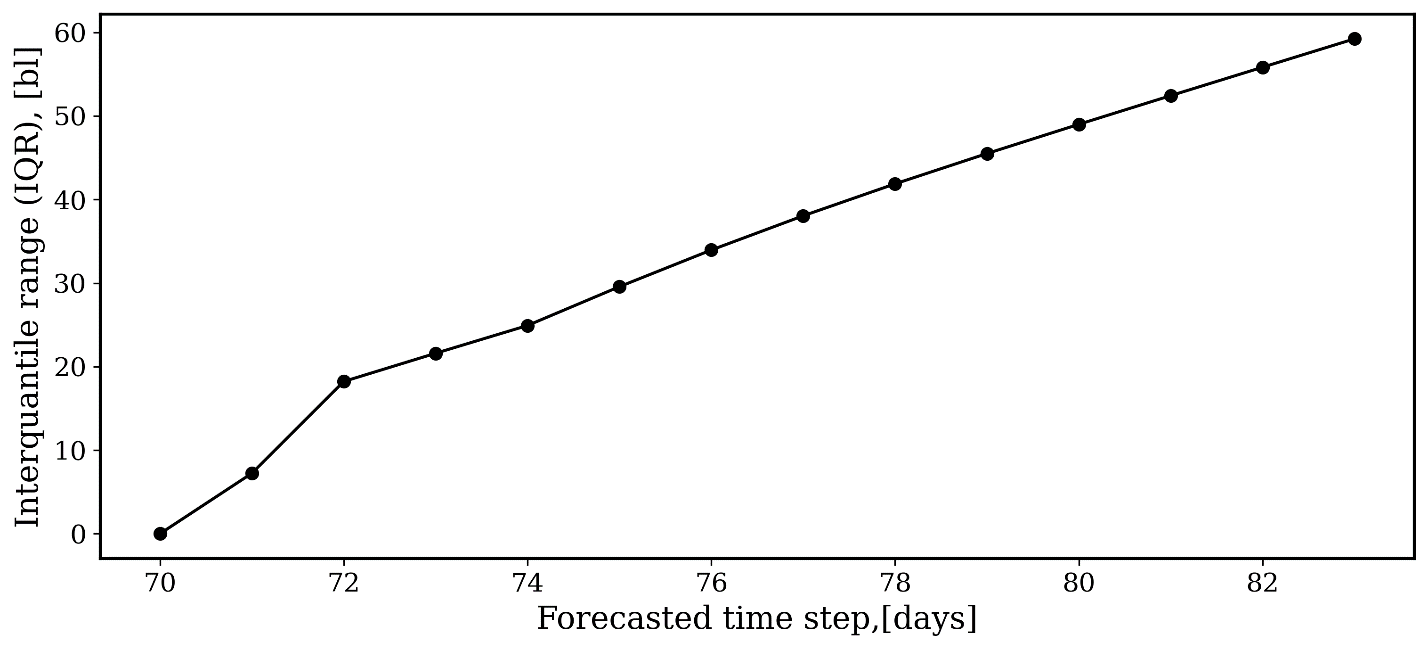


Figure .- IQR shows the spread of the middle 50% of the data at every time step for the case presented in Figure 13.

Diagram

Description automatically generated with low confidence

Figure .- Production forecast with rate, encoder-decoder attention weights, and events predictor feature. The attention head weights show consistent high attention weights to sharp changes corresponding to events for the decoder part.

Figure 16 presents the static predictor feature importance to interpret the attention head weights. There is no prior knowledge of the importance of position, but position becomes the most critical static predictor feature. As position changes, so do the initial porosity and permeability of the well.

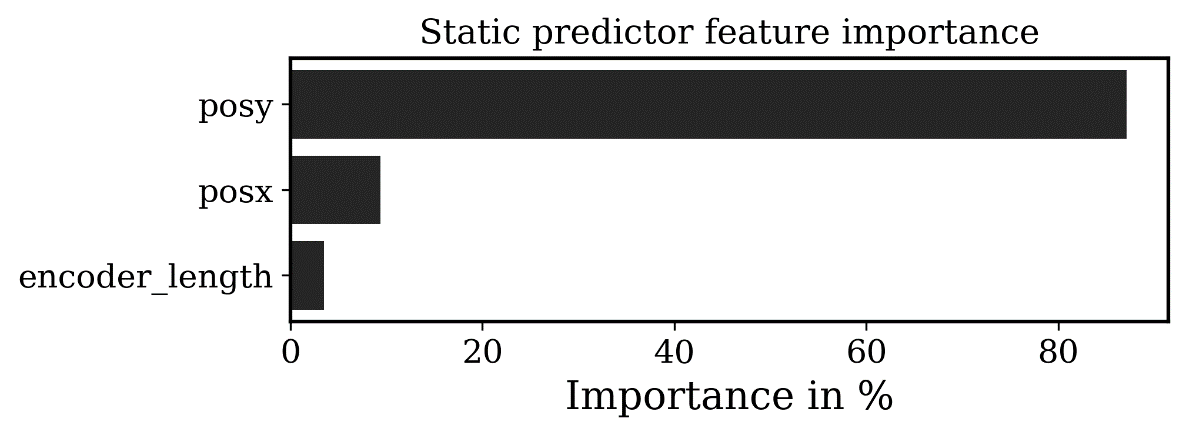


Figure .- Attention weights given to static predictor features. There is no prior knowledge of the importance of the position.

Figure 17 shows the Shapley importance of the encoder predictor features calculated with Equation 13. During encoding, the rate and the operation at the previous time steps are the most critical predictor features, followed by the location predictor features presented in equations 6 and 7.

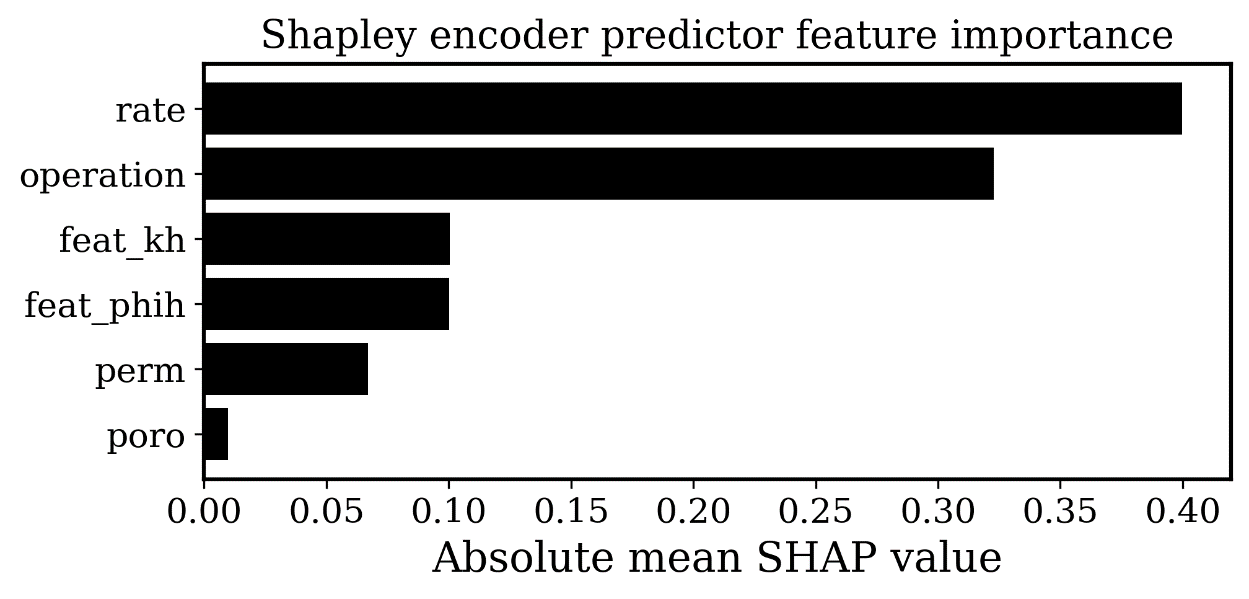


Figure .- Importance of the encoder predictor features. Operation is the most critical predictor feature, followed by the proposed location predictor feature.

In Figure 18, we show the Shapley importance of the decoder predictor features. The scheduled operation is the most crucial predictor feature during the decoding step, creating a sharp change in the forecasted rate. The following important predictor feature is permeability, which dominates the rate response. In Table 3, we include a summary of the error metrics for individual wells and field production.

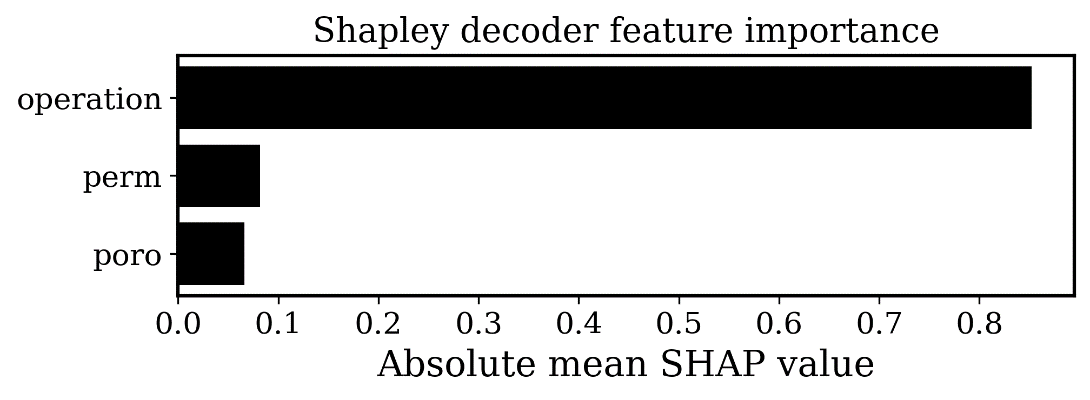


Figure - Importance of the decoder predictor feature. The scheduled event is the most important predictor feature during the decoding process, followed by the permeability.

Table .- SMAPE and MAE for all wells in scenario A using the TFT-based surrogate flow model.

|  |  |
| --- | --- |
| **Scenario A** | |
| Metric | Value |
| Benchmark MAE | 28.6119 |
| Benchmark SMAPE | 2.8 % |
| MAE | 15.81 |
| SMAPE | 0.75 % |

## Evaluation of scenario B

In scenario B, the rate measurements from the dataset contain Gaussian random noise to show that interpretability is not affected by noise from the measurements, thus increasing the applicability of the workflow to real-world scenarios. We use the last 180 days from the production history as testing data and the rest as training data. Table 4 summarizes the hyperparameters selected for the TFT-based surrogate flow model used for scenario B.

Table .- Hyperparameters used for the TFT-based surrogate flow model for scenario B.

|  |  |
| --- | --- |
| **Hyperparameter** | **Value** |
| gradient\_clip\_val | 0.9739 |
| hidden\_size | 95 |
| dropout | 0.171725 |
| hidden\_continuous\_size | 70 |
| attention\_head\_size | 2 |
| learning\_rate | 0.01266315 |
| epochs | 100 |

We select Well #5 to generate Figure 19. In subfigure a), we present the maximum encoding and decoder prediction length from the production history at 1020 days and the forecasted rate by the TFT-based surrogate flow model. In subfigure b), we show the scheduled events of the dataset where there is an event during this time frame. In subfigure c), we include the Shapley importance of the encoder. There is greater attention to past rates and the operation feature, as observed in scenario A. In subfigure d), we show the importance of the decoder, where operation becomes the most important feature during decoding, followed by permeability, which dominates the flow rate. Figure 19 shows that the Shapley predictor feature importance is consistent in the presence of noisy data. Figure 20 presents the IQR at every time step for the case presented. We observe that the IQR does not change in the presence of noisy measurements. Table 5 summarizes the error metrics for individual wells and field production. We complement our results with Figure 21, showing the attention head weights in the presence of noisy data. We observe equal attention over steady periods and higher attention weights over scheduled events.

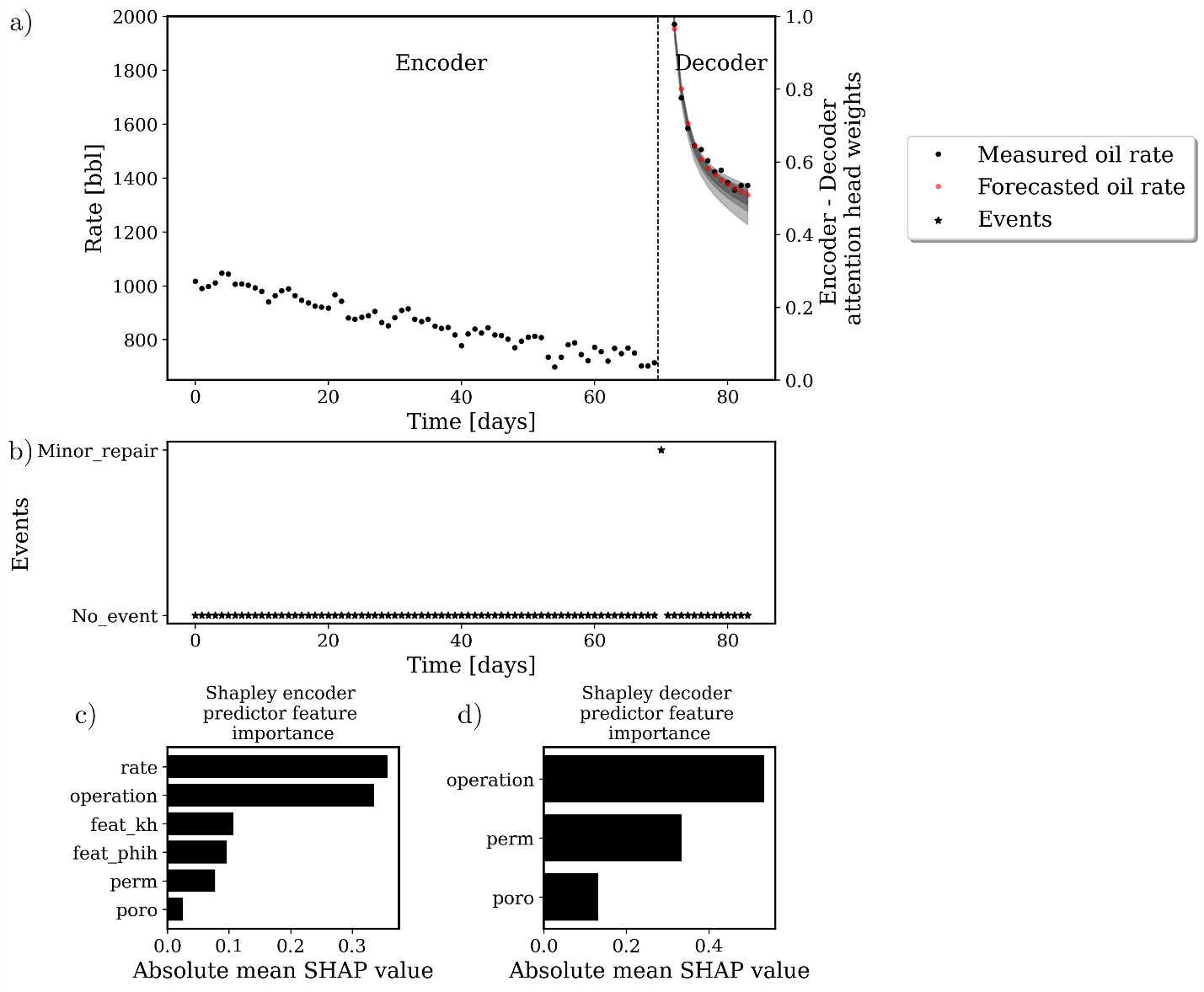


Figure .- Forecasting using the TFT-based surrogate flow model for Well #5 with scheduled events a) The black dashed line represents the start of the forecast. The red points are the forecasted data, and the black points are the measured values. In shadows of grey, we show the P10 and P90 confidence intervals. In subfigure b) we show the scheduled events of the dataset; in subfigure c) we include the importance of the encoder and d) decoder predictor features.

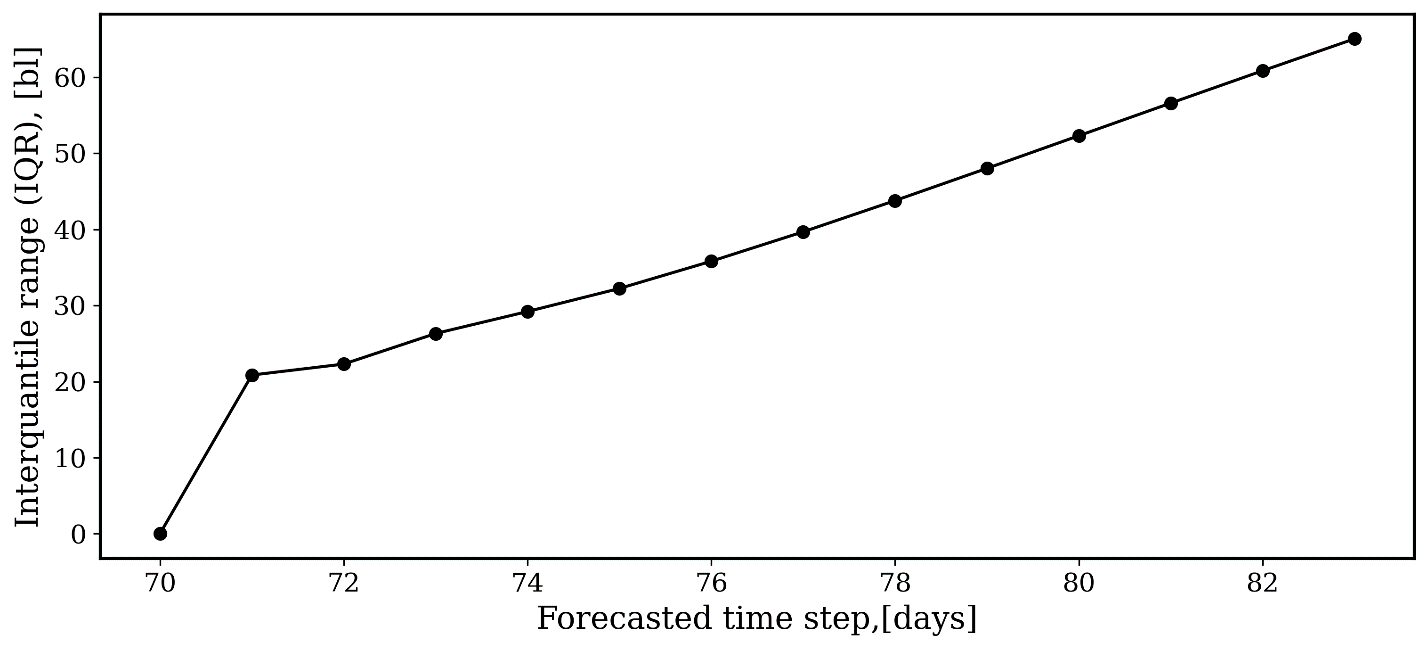


Figure - IQR at every time step for the case presented in Figure 19. We observe that the IQR is not affected by noisy rate measurements.

Table .- SMAPE and MAE for all wells in Scenario B using the TFT-based surrogate flow model.

|  |  |
| --- | --- |
| **Scenario B** | |
| Metric | Value |
| Benchmark MAE | 285.51 |
| Benchmark SMAPE | 16.5 % |
| MAE | 23.63 |
| SMAPE | 1.33 % |

Chart

Description automatically generated

Figure .- Rate forecasting, encoder-decoder attention weights, and events predictor feature. The attention head weights show consistently high attention weights to sharp changes corresponding to events for the decoder part in the presence of noisy measurements.

## Evaluation of scenario C

Using the dataset from scenario C, we follow the workflow described in the methodology section and train a TFT-based surrogate flow model. For training, we use the entire production history of well P1 and the last 180 days of production and injection data from wells P2-P4 and I1 for testing. The rest of the production and injection history is used for training. Table 6 presents the hyperparameters selected for the model used in scenario C.

Table .- Hyperparameters used for the TFT-based surrogate flow model for scenario C.

|  |  |
| --- | --- |
| **Hyperparameter** | **Value** |
| gradient\_clip\_val | 0.0246711 |
| hidden\_size | 61 |
| dropout | 0.19559 |
| hidden\_continuous\_size | 59 |
| attention\_head\_size | 3 |
| learning\_rate | 0.073 |
| epochs | 100 |

Figure 22 shows the water and oil rate forecast at 175 days for well P1. We observe a good agreement between the forecasted and observed rates. When forecasting water rate, the TFT-based surrogate flow model forecasts water production even when there is no information other than injection rates in the dataset. We plot the encoder and decoder attention head weights on the second axis of Figure 22, we observe high attention weights corresponding to the estimated water breakthrough. Table 7 includes the summary score results for the testing well.

Graphical user interface, histogram

Description automatically generated

Figure .- Water and oil rate forecast at 175 days for well P1. We observe a good agreement between the forecasted and observed rates. When forecasting water rate, the TFT-based surrogate flow model forecasts water production even when there is no information other than injection rates in the dataset. We include the events predictor feature to show that no events are specified for the timeframe.

Table .- SMAPE and MAE for all wells in scenario C using the TFT-based surrogate flow model.

|  |  |  |  |
| --- | --- | --- | --- |
| Scenario C | | | |
| Metric | Oil rate forecast error | Water rate forecast error | Bottom-hole pressure forecast error for injector well I1 |
| Benchmark MAE | 26.12 | 39.6 | 9.5 |
| Benchmark SMAPE | 1.16 % | 2.3 % | 3 % |
| MAE | 5.7 | 8.14 | 6 |
| SMAPE | 1.15 % | 2.2 % | 0.1 % |

To improve the interpretability of results in Figure 22, we construct Figure 23, where we plot the Shapley encoder and decoder predictor feature importance. In subfigure a), we show the Shapley encoder predictor feature importance at 175 days. The highest importance is given to the total oil rate, followed by the injector rate and our two proposed location-based predictor features. In subfigure b), we show the Shapley decoder predictor feature importance. The cumulative injection rate and well porosity have the highest importance. In subplot c), we show the static predictor feature importance where the position in has the highest predictor feature importance. The use of our proposed workflow and our results on decoder prediction length minimizes the importance of the encoder length. Figure 24 shows the forecasts of bottom-hole pressure for well I1 at 175 days. We observe high uncertainty in the estimates as indicated by the P10 and P90 confidence intervals. In a secondary axis, we include the attention head weights. Figure 23 shows the Shapley importance of the encoder and decoder predictor features. When forecasting injection pressure, the cumulative injection rate becomes the most critical predictor feature during decoding, followed by porosity. Finally, in Figure 25 subfigure a), we show the production history forecast of well P1. In black, we show the measured points; in red, we show the predicted values with the TFT-based surrogate flow model; in gray, we show the P10 and P90. To complement subfigure a), we plot the SHAP values in subfigure b). The SHAP values are used to measure the effect of the predictor features on the forecasted values at every time step.

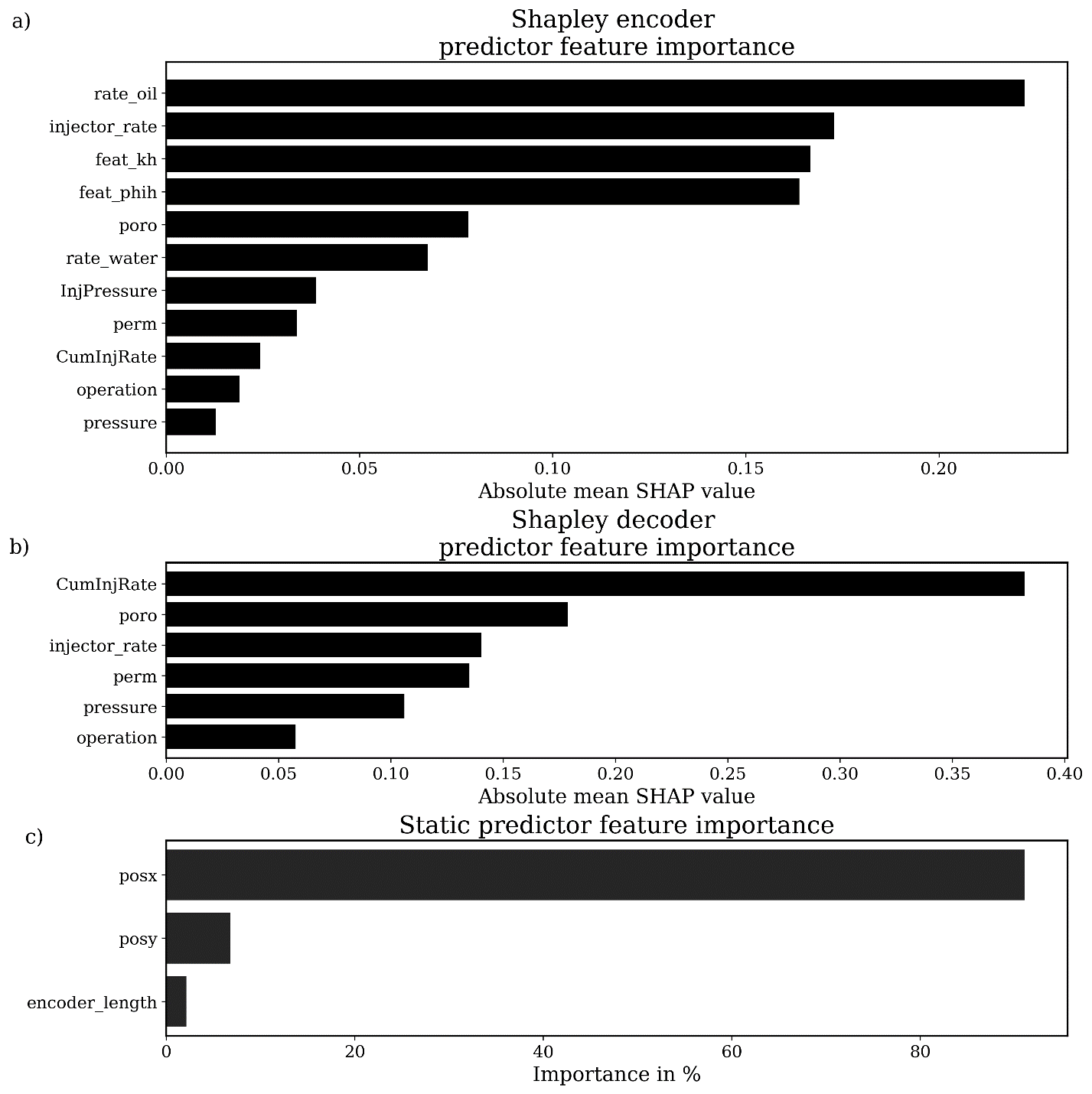


Figure .- Summary of predictor feature importance for scenario C, in a) we show the Shapley encoder predictor feature importance where the oil rate, injector rate, and the two proposed predictor features have the highest predictor feature importance. In b) we show the Shapley decoder predictor feature importance where the cumulative injection rate is the most important predictor feature. In c) we show the static predictor feature importance, where the position in x is the most important predictor feature.

Chart, histogram

Description automatically generated

Figure .- Bottom-hole forecasts of pressure of the injector well I1. We include in a secondary axis the attention head weights.

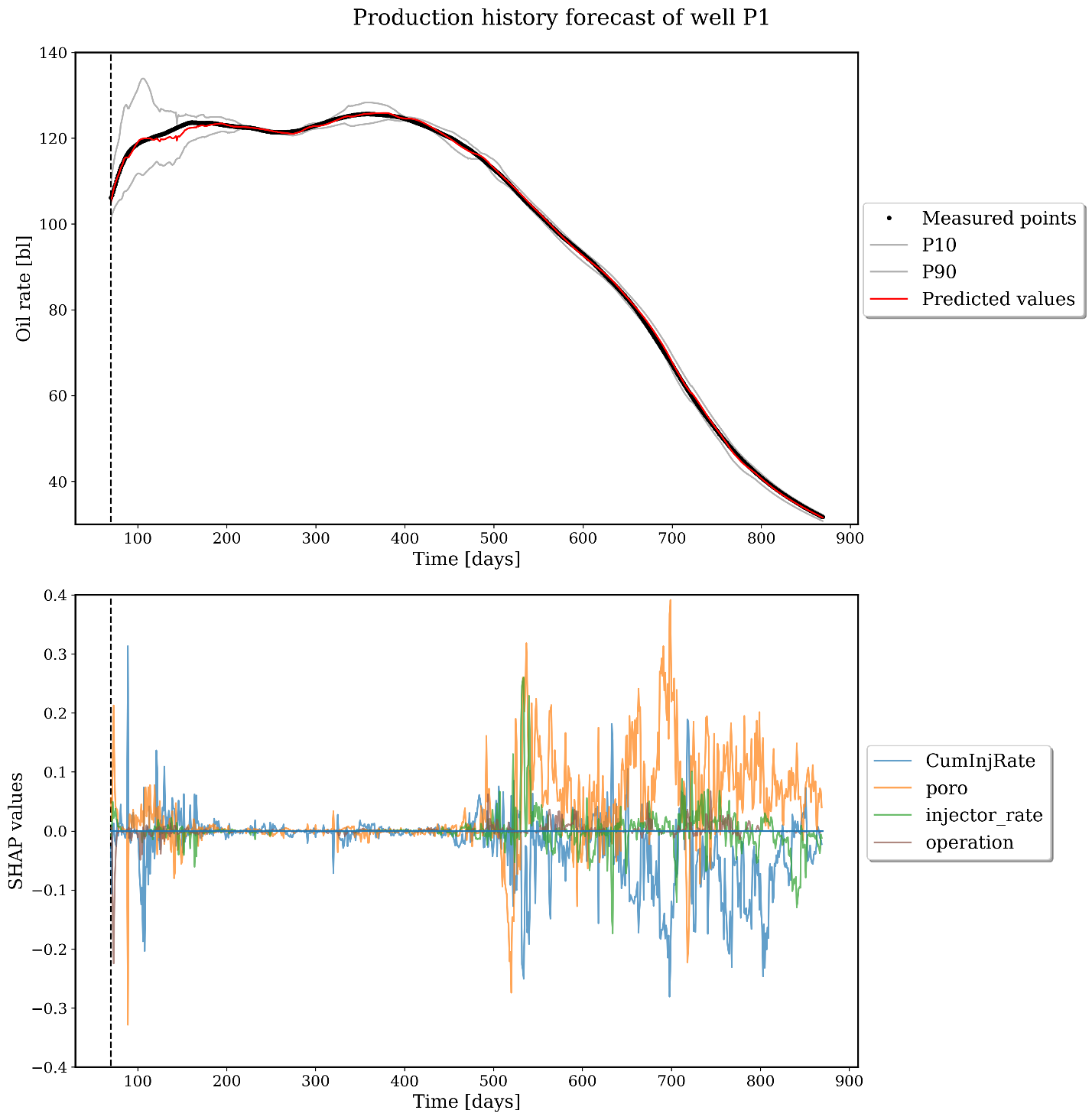


Figure – Production history forecast of well P1. Subfigure a) shows the production history forecast of well P1. In black, we show the measured points; in red, we show the predicted values; in gray, we show the P10 and P90. Subfigure b) shows the SHAP values for 4 predictor features.

# Conclusions

We propose a novel and general workflow to build a TFT-based surrogate flow model. Through the problem formulation, we include static (i.e., depth, position) and variable (i.e., injection rate, operational constraints) predictor features, forecast across multiple wells, and uncertainty estimation through the quantile loss function. To include location in the TFT-based surrogate flow model, we propose two static predictor features, and to improve model interpretability, we integrate SHAP values in the workflow to relate the forecasts to attention head weights at each time step. We also present a robust workflow to determine the best encoder and decoder prediction lengths. By utilizing our innovative workflow, the practitioner engineer can create surrogate flow models that account for uncertainty and accurately predict water and oil rates and pressures across multiple wells and steps. As a result, these models can be valuable tools for optimizing subsurface development decision-making and identifying globally important variables for forecasting and ongoing events that may impact well productivity. Future work could explore the implementation of the model in a real-world scenario to provide insights into subsurface dynamics during production. This could provide valuable information for optimizing subsurface development decision-making and identifying events that may impact well productivity. Additional work could include using reinforcement learning that could improve the accuracy and efficiency of the TFT-based surrogate flow model. A potential limitation of the proposed TFT-based surrogate model is that the accuracy of the surrogate flow model depends heavily on the quality and quantity of data used to train the model. Therefore, poor quality data or a lack of data could negatively affect the performance of the model.

Reproducibility

The workflow to construct the TFT-based surrogate flow model and interpretability of attention heads is publicly available on the author’s GitHub repositories (github.com/emaldonadocruz and github.com/GeostatsGuy).

Acknowledgements

The authors thank the industry partners of the DIRECT consortium at the University of Texas at Austin for supporting this work.

# References

Ahmadi, R., Shahrabi, J., & Aminshahidy, B. (2020). Forecasting multiple-well flow rates using a novel space-time modeling approach. Journal of Petroleum Science and Engineering, 191, 107027. https://doi.org/https://doi.org/10.1016/j.petrol.2020.107027

Arps, J. J. (1945). Analysis of Decline Curves. Transactions of the AIME, 160(01), 228–247. https://doi.org/10.2118/945228-G

Aziz, K., & Settari, A. (1979). Petroleum reservoir simulation. Applied Science Publishers. https://books.google.com/books?id=GJ5TAAAAMAAJ

Bao, T., Corgnet, B., Hanaki, N., Riyanto, Y. E., & Zhu, J. (2023). Predicting the unpredictable: New experimental evidence on forecasting random walks. Journal of Economic Dynamics and Control, 146, 104571. https://doi.org/https://doi.org/10.1016/j.jedc.2022.104571

Beleites, C., Neugebauer, U., Bocklitz, T., Krafft, C., & Popp, J. (2013). Sample size planning for classification models. Analytica Chimica Acta, 760, 25–33. https://doi.org/https://doi.org/10.1016/j.aca.2012.11.007

Beleites, C., & Salzer, R. (2008). Assessing and improving the stability of chemometric models in small sample size situations. Analytical and Bioanalytical Chemistry, 390(5), 1261–1271. https://doi.org/10.1007/s00216-007-1818-6

Belyadi, H., Fathi, E., & Belyadi, F. (2019). Chapter Seventeen - Decline curve analysis (H. Belyadi, E. Fathi, & F. B. T.-H. F. in U. R. (Second E. Belyadi (eds.); pp. 311–340). Gulf Professional Publishing. https://doi.org/https://doi.org/10.1016/B978-0-12-817665-8.00017-5

Cao, C., Jia, P., Cheng, L., Jin, Q., & Qi, S. (2022). A review on application of data-driven models in hydrocarbon production forecast. Journal of Petroleum Science and Engineering, 212, 110296. https://doi.org/https://doi.org/10.1016/j.petrol.2022.110296

Cao, Q., Banerjee, R., Gupta, S., Li, J., Zhou, W., & Jeyachandra, B. (2016). Data Driven Production Forecasting Using Machine Learning. In SPE Argentina Exploration and Production of Unconventional Resources Symposium (p. D021S006R001). https://doi.org/10.2118/180984-MS

Carvalho, D. V, Pereira, E. M., & Cardoso, J. S. (2019). Machine Learning Interpretability: A Survey on Methods and Metrics. Electronics, 8(8). https://doi.org/10.3390/electronics8080832

Chahar, J., Verma, J., Vyas, D., & Goyal, M. (2022). Data-driven approach for hydrocarbon production forecasting using machine learning techniques. Journal of Petroleum Science and Engineering, 217, 110757. https://doi.org/https://doi.org/10.1016/j.petrol.2022.110757

Clarkson, C. R. (2013). Production data analysis of unconventional gas wells: Review of theory and best practices. International Journal of Coal Geology, 109–110, 101–146. https://doi.org/https://doi.org/10.1016/j.coal.2013.01.002

Cunningham, C. F., Cooley, L., Wozniak, G., & Pancake, J. (2012). Using Multiple Linear Regression to Model EURs of Horizontal Marcellus Shale Wells. In SPE Eastern Regional Meeting (p. SPE-161343-MS). https://doi.org/10.2118/161343-MS

Davtyan, A., Rodin, A., Muchnik, I., & Romashkin, A. (2020). Oil production forecast models based on sliding window regression. Journal of Petroleum Science and Engineering, 195, 107916. https://doi.org/https://doi.org/10.1016/j.petrol.2020.107916

Deng, Y., Hui, S., Zhou, S., Meng, D., & Wang, J. (2022). T-Former: An Efficient Transformer for Image Inpainting. Proceedings of the 30th ACM International Conference on Multimedia, 6559–6568. https://doi.org/10.1145/3503161.3548446

Deutsch, C. V, & Journel, A. G. (1992). GSLIB: Geostatistical Software Library and User’s Guide. Oxford University Press. https://books.google.com/books?id=BiZOAQAAIAAJ

Esmaili, S., Kalantari-Dahaghi, A., & Mohaghegh, S. D. (2012). Forecasting, Sensitivity and Economic Analysis of Hydrocarbon Production from Shale Plays Using Artificial Intelligence &amp; Data Mining. In SPE Canadian Unconventional Resources Conference (p. SPE-162700-MS). https://doi.org/10.2118/162700-MS

Fanchi, J. R. (2010). 9 - Production Evaluation Techniques (J. R. B. T.-I. R. A. M. Fanchi (ed.); pp. 145–166). Gulf Professional Publishing. https://doi.org/https://doi.org/10.1016/B978-0-12-382088-4.00009-8

Gentry, R. W. (1972). Decline-Curve Analysis. Journal of Petroleum Technology, 24(01), 38–41. https://doi.org/10.2118/3356-PA

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.

Gupta, S., Fuehrer, F., & Jeyachandra, B. C. (2014). Production Forecasting in Unconventional Resources using Data Mining and Time Series Analysis. In SPE/CSUR Unconventional Resources Conference – Canada (p. D011S004R008). https://doi.org/10.2118/171588-MS

Gurina, E., Klyuchnikov, N., Antipova, K., & Koroteev, D. (2022). Making the black-box brighter: Interpreting machine learning algorithm for forecasting drilling accidents. Journal of Petroleum Science and Engineering, 218, 111041. https://doi.org/https://doi.org/10.1016/j.petrol.2022.111041

Hastie, T., Tibshirani, R., & Friedman, J. (2009). Neural Networks BT - The Elements of Statistical Learning: Data Mining, Inference, and Prediction (T. Hastie, R. Tibshirani, & J. Friedman (eds.); pp. 389–416). Springer New York. https://doi.org/10.1007/978-0-387-84858-7\_11

Hochreiter, S. (1998). The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 06(02), 107–116. https://doi.org/10.1142/S0218488598000094

Hui, G., Chen, S., He, Y., Wang, H., & Gu, F. (2021). Machine learning-based production forecast for shale gas in unconventional reservoirs via integration of geological and operational factors. Journal of Natural Gas Science and Engineering, 94, 104045. https://doi.org/https://doi.org/10.1016/j.jngse.2021.104045

Iskandar, U. P., & Kurihara, M. (2022). Time-Series Forecasting of a CO2-EOR and CO2 Storage Project Using a Data-Driven Approach. In Energies (Vol. 15, Issue 13). https://doi.org/10.3390/en15134768

Jurafsky, D., & Martin, J. H. (2009). Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Pearson Prentice Hall. https://books.google.com/books?id=fZmj5UNK8AQC

Kubota, L., & Reinert, D. (2019). Machine Learning Forecasts Oil Rate in Mature Onshore Field Jointly Driven by Water and Steam Injection. In SPE Annual Technical Conference and Exhibition (p. D021S020R003). https://doi.org/10.2118/196152-MS

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436–444. https://doi.org/10.1038/nature14539

Li, X., Ma, X., Xiao, F., Xiao, C., Wang, F., & Zhang, S. (2022). Time-series production forecasting method based on the integration of Bidirectional Gated Recurrent Unit (Bi-GRU) network and Sparrow Search Algorithm (SSA). Journal of Petroleum Science and Engineering, 208, 109309. https://doi.org/https://doi.org/10.1016/j.petrol.2021.109309

Li, Y., He, J., Zhou, X., Zhang, Y., & Baldridge, J. (2020). Mapping Natural Language Instructions to Mobile {UI} Action Sequences. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 8198–8210. https://doi.org/10.18653/v1/2020.acl-main.729

Lim, B., Arık, S. Ö., Loeff, N., & Pfister, T. (2021). Temporal Fusion Transformers for interpretable multi-horizon time series forecasting. International Journal of Forecasting, 37(4), 1748–1764. https://doi.org/https://doi.org/10.1016/j.ijforecast.2021.03.012

Lin, Y., Chen, K., Zhang, X., Tan, B., & Lu, Q. (2022). Forecasting crude oil futures prices using BiLSTM-Attention-CNN model with Wavelet transform. Applied Soft Computing, 130, 109723. https://doi.org/https://doi.org/10.1016/j.asoc.2022.109723

Liu, J., & Jung, C. (2021). Facial image inpainting using attention-based multi-level generative network. Neurocomputing, 437, 95–106. https://doi.org/https://doi.org/10.1016/j.neucom.2020.12.118

Liu, W., Liu, W. D., & Gu, J. (2020). Forecasting oil production using ensemble empirical model decomposition based Long Short-Term Memory neural network. Journal of Petroleum Science and Engineering, 189, 107013. https://doi.org/https://doi.org/10.1016/j.petrol.2020.107013

Maldonado-Cruz, E., & Pyrcz, M. J. (2022). Fast evaluation of pressure and saturation predictions with a deep learning surrogate flow model. Journal of Petroleum Science and Engineering, 212, 110244. https://doi.org/https://doi.org/10.1016/j.petrol.2022.110244

Maldonado-Cruz, E., & Pyrcz, M. J. (2021). Tuning machine learning dropout for subsurface uncertainty model accuracy. Journal of Petroleum Science and Engineering, 205, 108975. https://doi.org/https://doi.org/10.1016/j.petrol.2021.108975

Mattar, L., & Anderson, D. M. (2003). A Systematic and Comprehensive Methodology for Advanced Analysis of Production Data. In SPE Annual Technical Conference and Exhibition (p. SPE-84472-MS). https://doi.org/10.2118/84472-MS

Michelevichius, D., Minijos, N., & Zolotukhin, A. B. (2002). Evaluating Productivity of a Horizontal Well. In SPE International Thermal Operations and Heavy Oil Symposium and International Horizontal Well Technology Conference (p. SPE-79000-MS). https://doi.org/10.2118/79000-MS

Mohammadmoradi, P., Moradi, H. M., & Kantzas, A. (2018). Data-Driven Production Forecasting of Unconventional Wells with Apache Spark. In SPE Western Regional Meeting (p. D041S011R006). https://doi.org/10.2118/190098-MS

Ostojic, J., Rezaee, R., & Bahrami, H. (2012). Production performance of hydraulic fractures in tight gas sands, a numerical simulation approach. Journal of Petroleum Science and Engineering, 88–89, 75–81. https://doi.org/https://doi.org/10.1016/j.petrol.2011.11.002

Parvizi, H., Rezaei-Gomari, S., & Nabhani, F. (2017). Robust and Flexible Hydrocarbon Production Forecasting Considering the Heterogeneity Impact for Hydraulically Fractured Wells. Energy & Fuels, 31(8), 8481–8488. https://doi.org/10.1021/acs.energyfuels.7b00738

Pickup, G. E. (2013). 2 - CO2 storage capacity calculation using static and dynamic modelling (J. Gluyas & S. B. T.-G. S. of C. D. (CO2) Mathias (eds.); pp. 26–44). Woodhead Publishing. https://doi.org/https://doi.org/10.1533/9780857097279.1.26

Pirnay, J., & Chai, K. (2022). Inpainting Transformer for Anomaly Detection BT - Image Analysis and Processing – ICIAP 2022 (S. Sclaroff, C. Distante, M. Leo, G. M. Farinella, & F. Tombari (eds.); pp. 394–406). Springer International Publishing.

Pyrcz, M.J., Jo. H., Kupenko, A., Liu, W., Gigliotti, A.E., Salomaki, T., and Santos, J., (2021). Geostatistical methods from GSLIB: Geostatistical Library translated and reimplemented in Python. https://github.com/GeostatsGuy/GeostatsPy

Rahuma, K. M., Mohamed, H. A. M., Hissein, N., & Giuma, S. K. B. (2013). Prediction of Reservoir Performance Applying Decline Curve Analysis. International Journal of Chemical Engineering and Applications, 74–77.

Salles, R., Belloze, K., Porto, F., Gonzalez, P. H., & Ogasawara, E. (2019). Nonstationary time series transformation methods: An experimental review. Knowledge-Based Systems, 164, 274–291. https://doi.org/https://doi.org/10.1016/j.knosys.2018.10.041

Schulze-Riegert, R., & Ghedan, S. (2007). Modern Techniques for History Matching.

Sharma, V., Cali, Ü., Sardana, B., Kuzlu, M., Banga, D., & Pipattanasomporn, M. (2021). Data-driven short-term natural gas demand forecasting with machine learning techniques. Journal of Petroleum Science and Engineering, 206, 108979. https://doi.org/https://doi.org/10.1016/j.petrol.2021.108979

Sun, J., Ma, X., & Kazi, M. (2018). Comparison of Decline Curve Analysis DCA with Recursive Neural Networks RNN for Production Forecast of Multiple Wells. In SPE Western Regional Meeting (p. D041S012R009). https://doi.org/10.2118/190104-MS

Sun, Y., Ma, X., Guo, W., Jia, R., & Li, B. (2019). Numerical simulation of the short- and long-term production behavior of the first offshore gas hydrate production test in the South China Sea. Journal of Petroleum Science and Engineering, 181, 106196. https://doi.org/https://doi.org/10.1016/j.petrol.2019.106196

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. Advances in Neural Information Processing Systems, 27.

Tadjer, A., Hong, A., & Bratvold, R. B. (2021). Machine learning based decline curve analysis for short-term oil production forecast. Energy Exploration & Exploitation, 39(5), 1747–1769. https://doi.org/10.1177/01445987211011784

Tay, Y., Bahri, D., Metzler, D., Juan, D.-C., Zhao, Z., & Zheng, C. (2021). Synthesizer: Rethinking Self-Attention for Transformer Models. In M. Meila & T. Zhang (Eds.), Proceedings of the 38th International Conference on Machine Learning (Vol. 139, pp. 10183–10192). PMLR. https://proceedings.mlr.press/v139/tay21a.html

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is All you Need. In I. Guyon, U. V Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), Advances in Neural Information Processing Systems (Vol. 30). Curran Associates, Inc. https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

Vincent, G., Corre, B., & Thore, P. (1999). Managing Structural Uncertainty in a Mature Field for Optimal Well Placement. SPE Reservoir Evaluation & Engineering, 2(04), 377–384. https://doi.org/10.2118/57468-PA

Wang, W., Fan, D., Sheng, G., Chen, Z., & Su, Y. (2019). A review of analytical and semi-analytical fluid flow models for ultra-tight hydrocarbon reservoirs. Fuel, 256, 115737. https://doi.org/https://doi.org/10.1016/j.fuel.2019.115737

Xiang, H., Zou, Q., Nawaz, M. A., Huang, X., Zhang, F., & Yu, H. (2023). Deep learning for image inpainting: A survey. Pattern Recognition, 134, 109046. https://doi.org/https://doi.org/10.1016/j.patcog.2022.109046

Xu, F., Yu, W., Li, X., Miao, J., Zhao, G., Sepehrnoori, K., Li, X., Jin, J., & Wen, G. (2018). A Fast EDFM Method for Production Simulation of Complex Fractures in Naturally Fractured Reservoirs. In SPE/AAPG Eastern Regional Meeting (p. D043S007R001). https://doi.org/10.2118/191800-18ERM-MS

Xu, Y., & Goodacre, R. (2018). On Splitting Training and Validation Set: A Comparative Study of Cross-Validation, Bootstrap and Systematic Sampling for Estimating the Generalization Performance of Supervised Learning. Journal of Analysis and Testing, 2(3), 249–262. https://doi.org/10.1007/s41664-018-0068-2

Yang, M., & Foster, J. T. (2022). Multi-output physics-informed neural networks for forward and inverse PDE problems with uncertainties. Computer Methods in Applied Mechanics and Engineering, 402, 115041. https://doi.org/https://doi.org/10.1016/j.cma.2022.115041

Zhong, Z., Sun, A. Y., Wang, Y., & Ren, B. (2020). Predicting field production rates for waterflooding using a machine learning-based proxy model. Journal of Petroleum Science and Engineering, 194, 107574. https://doi.org/https://doi.org/10.1016/j.petrol.2020.107574

Zhou, Q., Dilmore, R., Kleit, A., & Wang, J. Y. (2014). Evaluating gas production performances in marcellus using data mining technologies. Journal of Natural Gas Science and Engineering, 20, 109–120. https://doi.org/https://doi.org/10.1016/j.jngse.2014.06.014

Zongxiao, R., Xiaodong, W., Dandan, L., Rui, R., Wei, G., Zhiming, C., Jiaming, Z., Xiaojun, W., Heng, Z., & Zhaoguang, T. (2016). Semi-analytical model of the transient pressure behavior of complex fracture networks in tight oil reservoirs. Journal of Natural Gas Science and Engineering, 35, 497–508. https://doi.org/https://doi.org/10.1016/j.jngse.2016.09.006