University of Tehran

# Long-term Production Forecasting Using Multivariate Time Series Analysis and a Novel Residual 3D-CNN LSTM Model

***Presented by:***
***Milad Masroor***

***Supervisor:***
***Dr. Mohammad Emami Niri***

***Advisor:***
***Dr. Mozhdeh Sajjadi***

Summer 2022

Institute of Petroleum Engineering

University of Tehran

Institute of Petroleum Engineering

# Table of Content

University of Tehran

# Problem Definition

Institute of Petroleum Engineering

University of Tehran

Institute of Petroleum Engineering

- For decades, petroleum engineers and researchers have been looking for a reliable and straightforward approach to predicting the oil production of petroleum wells.
- **Conventional approaches** and **soft computing approaches**.
- The conventional methodologies: **analogy**, **volumetric**, **material balance**, **decline curve fitting**, and **reservoir simulation**.
- Soft computing approaches: **Machine Learning (ML) algorithms**
- **Deep Learning (DL) algorithms: CNNs (convolutional neural networks) and RNNs (recurrent neural networks)**
- **RNNs: LSTM (Long Short-Term Memory)** and **GRU (Gated Recurrent Unit)**.

University of Tehran

# Literature Review

Institute of Petroleum Engineering

University of Tehran

Institute of Petroleum Engineering

❖ According to the literature, production forecasting using artificial intelligence (AI)-based methods can be classified into three categories based on the type of data-driven forecasting technique they utilized: (1) traditional point-data regressor algorithms, (2) conventional ML algorithms, and (3) DL algorithms.

| Type of data-driven forecasting technique | Authors | Algorithms |
|---|---|---|
| Traditional point-data regressor algorithms | Aliyuda and Howell, 2019 | support vector machine (SVM) |
| | Guo et al., 2021 | multiple linear regression (MLR)<br>SVR<br>gaussian process regression (GPR) |

| Type of data-driven forecasting technique | Authors | Algorithms |
|---|---|---|
| Traditional ML algorithms | Klie, 2015 | radial basis function (RBF) |
| | Cao et al., 2016 | artificial neural network (ANN) |
| | Fulford et al., 2016 | Bayesian ML technique |
| | Jia and Zhang, 2016 | ANN |
| | Li and Han, 2017 | ANN |
| | Khan et al., 2019 | Artificial neuro fuzzy inference systems (ANFIS)<br>SVM<br>ANN |

University of Tehran

Institute of Petroleum Engineering

| Type of data-driven forecasting technique | Authors | Algorithms |
|---|---|---|
| DL algorithms | Li et al., 2022 | temporal convolutional networks (TCN)<br>LSTM |
| | Calvette et al., 2019 | LSTM<br>Bidirectional LSTM |
| | Bao et al., 2020 | Cascaded LSTM<br>EnKF enhanced LSTM<br>Standard LSTM |
| | Li et al., 2022 | CNN<br>PSO-CNN<br>LSTM<br>PSO-LSTM<br>CNN-LSTM<br>PSO-CNN-LSTM |
| | Zha et al., 2022 | Deep feedforward neural network (DNN)<br>RNN<br>CNN<br>LSTM<br>CNN-LSTM |

University of Tehran

# Innovation and Research Value

Institute of Petroleum Engineering

University of Tehran

❖ There are **three main shortcomings** associated with these studies:

1. **Not capable of predicting multiple time series features simultaneously.**

**Solution:** Combination of two time-series forecasting methods: multiple-output and multiple-step.

2. **Not considering the information of adjacent wells**

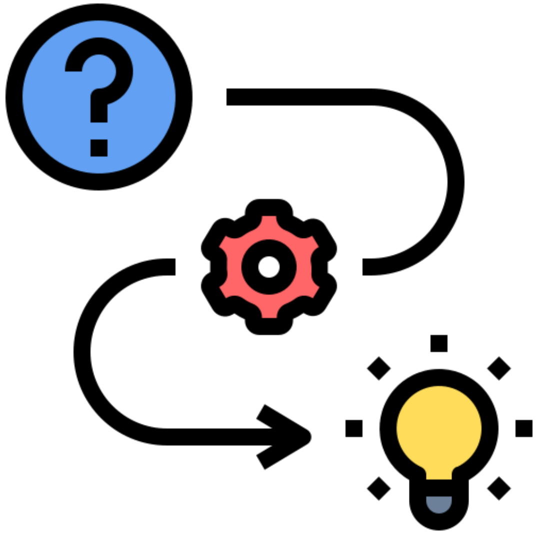
**Solution:** Using production and injection data, we have developed an artificial 3D feature image to determine the connection between wells.

3. **Their intelligent models lack residuals and deeper bottleneck structures.**

**Solution:** As a result, the residual and deeper bottleneck structures are intended to reduce the training time of the neural networks used in this study.

Institute of Petroleum Engineering

University of Tehran



# The General Process of Problem Solving

Institute of Petroleum Engineering

University of Tehran

1. The preparation of input and output data for neural networks
2. Analyze the data and perform any necessary pre-processing
3. Using production and injection data to construct an artificial 3D feature image
4. The training of recurrent and convolutional neural networks
5. Using optimization algorithms to optimize the hyper-parameters of neural networks
6. Reducing the training time for neural networks by using a series of specific architectures
7. Performance evaluation of trained networks
8. The performance of different networks is compared
9. Choosing the best network

Institute of Petroleum Engineering

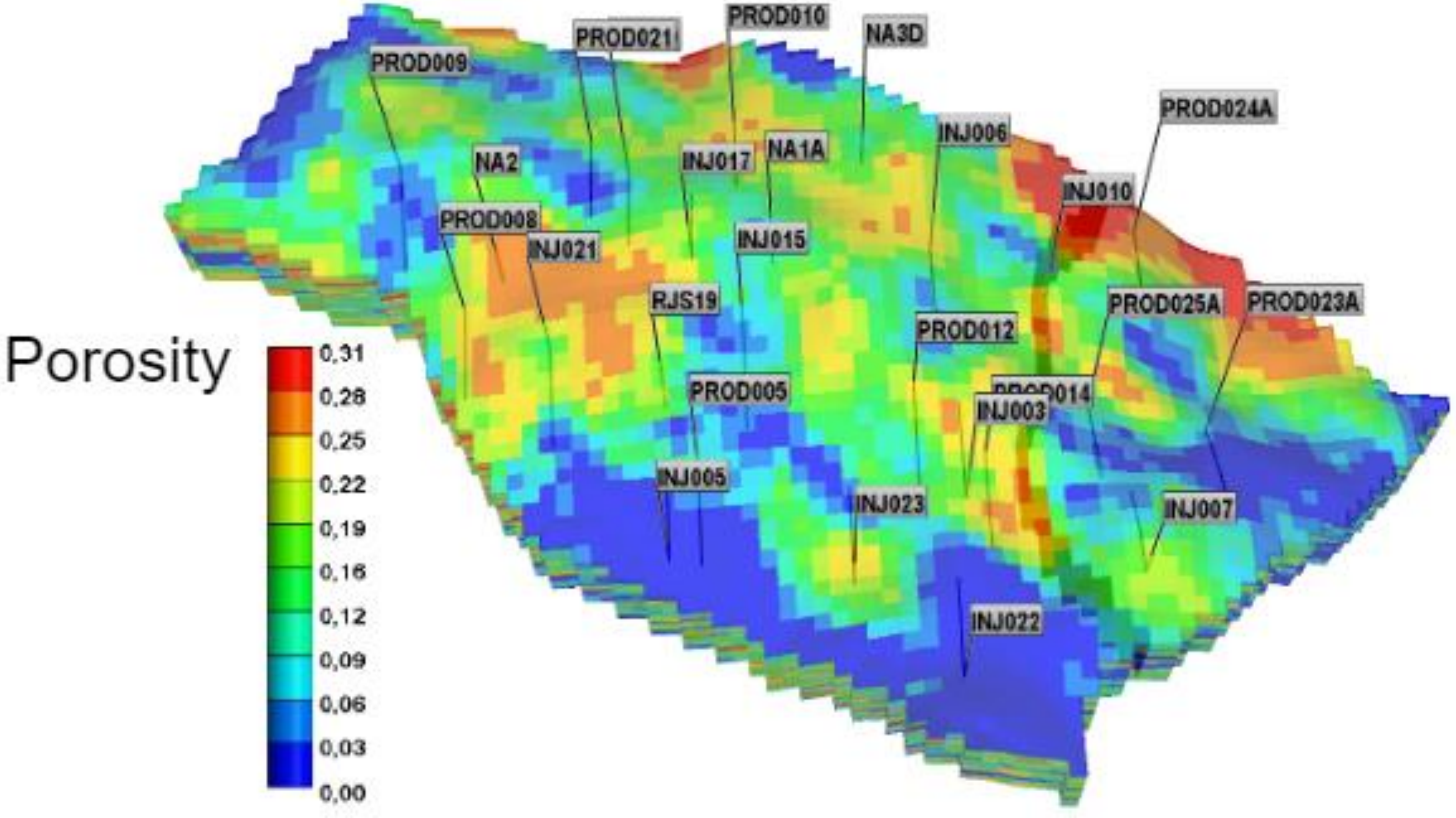
University of Tehran

# Description of the Synthetic Model

Institute of Petroleum Engineering

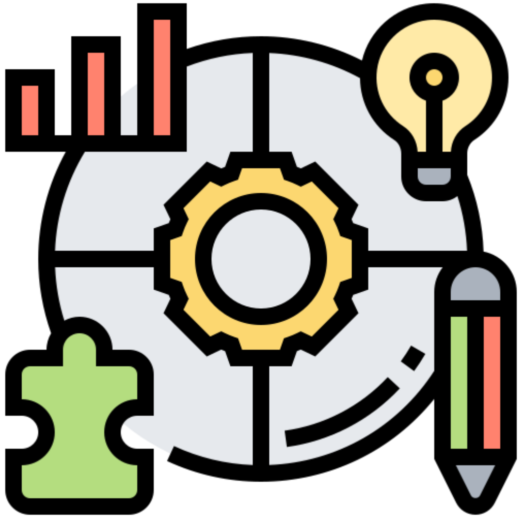
University of Tehran

- The final dataset is the **UNISIM-I benchmark**, created by the **UNISIM group** at the University of Campinas. This synthetic benchmark runs on the **CMG-IMEX** simulator.
- The UNISIM-I model is based on the geomodel of **Namorado Field**, located in the Campos Basin in Brazil.




Institute of Petroleum Engineering

University of Tehran



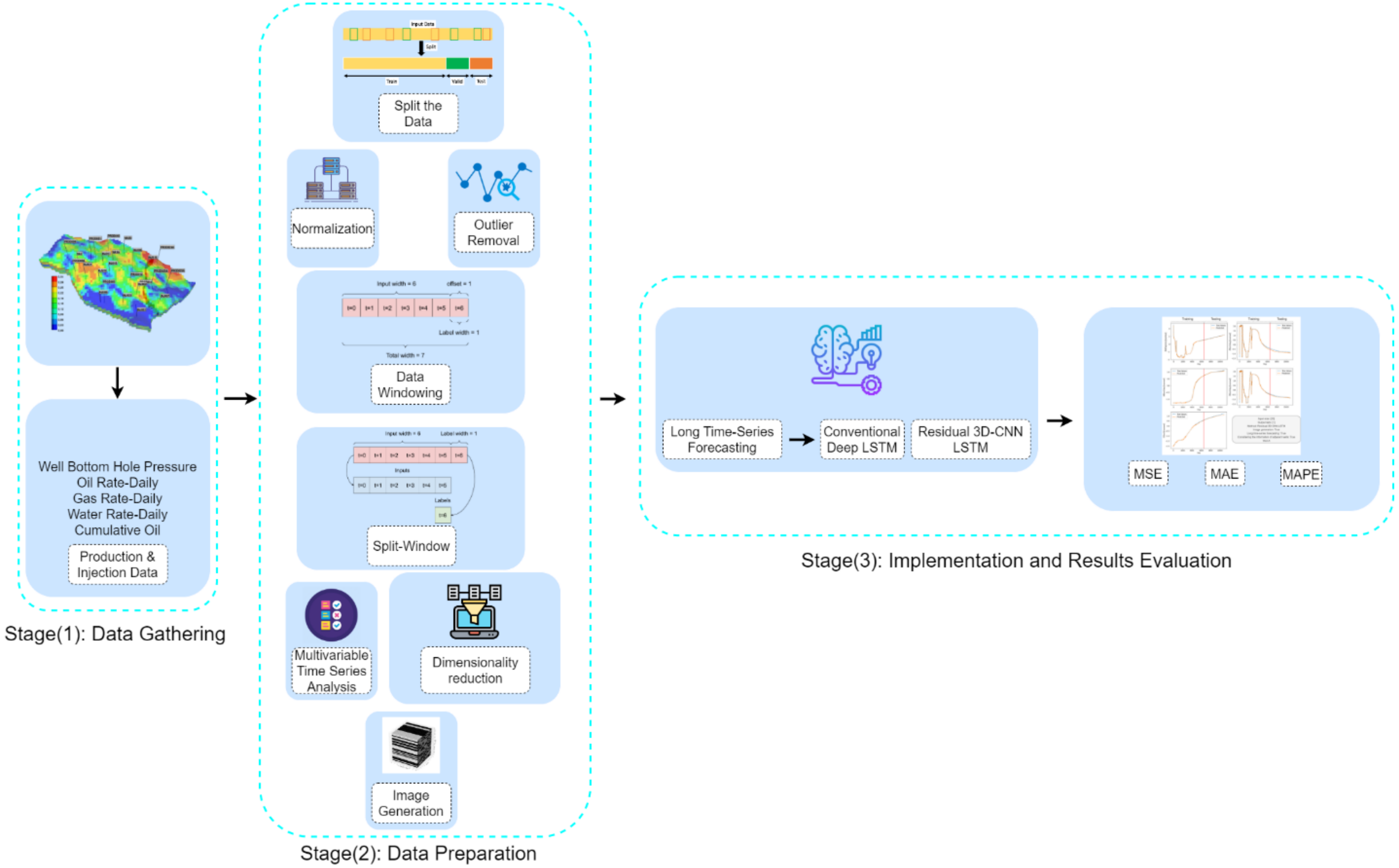
# Methodology

Institute of Petroleum Engineering

University of Tehran

Institute of Petroleum Engineering

❖ There are three main stages in the proposed workflow for analyzing and forecasting production data: data gathering and description (stage 1), data preparation (stage 2), and the implementation of (training) and evaluation of intelligent models (stage 3).



University of Tehran

Institute of Petroleum Engineering

## Stage (1): data gathering and description

- ❖ In this research, the production data of two wells (Well NA1A & Well NA3D) from the UNISIM-I synthetic model were used to build the final database.

| Data Category | Parameter, Unit | Definition | Source | Data type |
|---|---|---|---|---|
| Production | BHP, $kg/cm^2$ | Bottom Hole pressure | UNISIM-I: Synthetic Model | Continuous |
| Production | CO, $m^3$ | Cumulative Oil | UNISIM-I: Synthetic Model | Continuous |
| Production | OR, $m^3/day$ | Oil Rate-Daily | UNISIM-I: Synthetic Model | Continuous |
| Production | GR, $m^3/day$ | Gas Rate-Daily | UNISIM-I: Synthetic Model | Continuous |
| Production | WR, $m^3/day$ | Water Rate-Daily | UNISIM-I: Synthetic Model | Continuous |

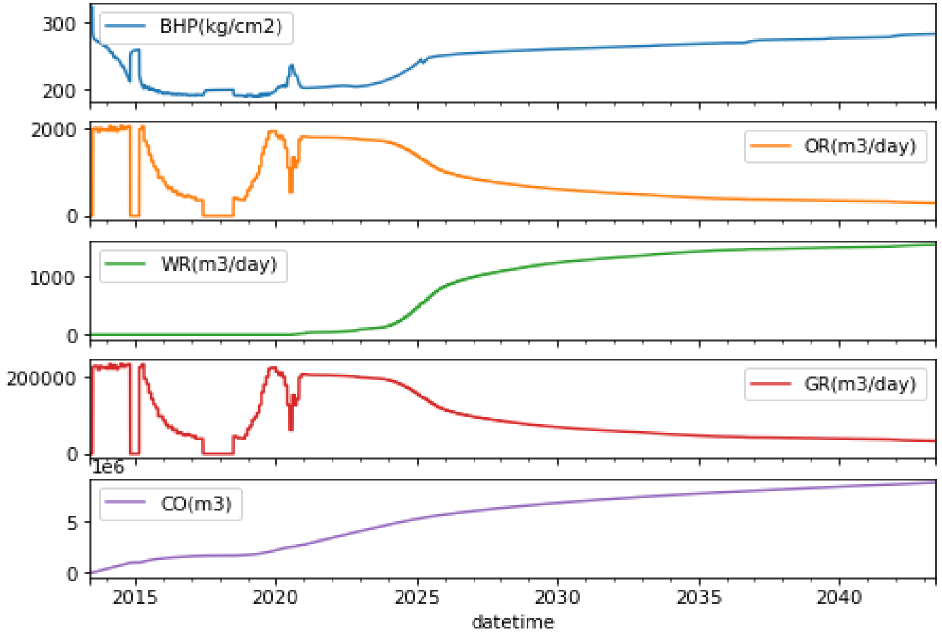
| Well Name, No. of Samples | Variable | Minimum | Maximum | Mean | STD |
|---|---|---|---|---|---|
| Well NA1A, 10957 | BHP ($kg/cm^2$) | 189.276413 | 3.243229e+02 | 2.456801e+02 | 3.161153e+01 |
| | CO ($m^3$) | 0 | 8.828875e+06 | 5.488739e+06 | 2.727175e+06 |
| | OR ($m^3/day$) | 0 | 2.071000e+03 | 8.057748e+02 | 5.906355e+02 |
| | GR ($m^3/day$) | 0 | 2.349680e+05 | 9.142010e+04 | 6.723840e+04 |
| | WR ($m^3/day$) | 0 | 1.547000e+03 | 8.225214e+02 | 6.473949e+02 |
| Well NA3D, 10957 | BHP ($kg/cm^2$) | 189.645813 | 3.287782e+02 | 2.228425e+02 | 2.690785e+01 |
| | CO ($m^3$) | 0 | 7.561194e+06 | 4.213243e+06 | 2.594496e+06 |
| | OR ($m^3/day$) | 0 | 1.973000e+03 | 6.900789e+02 | 4.721571e+02 |
| | GR ($m^3/day$) | 0 | 2.238440e+05 | 7.830664e+04 | 5.372015e+04 |
| | WR ($m^3/day$) | 0 | 1.542000e+03 | 7.742698e+02 | 6.419767e+02 |

University of Tehran

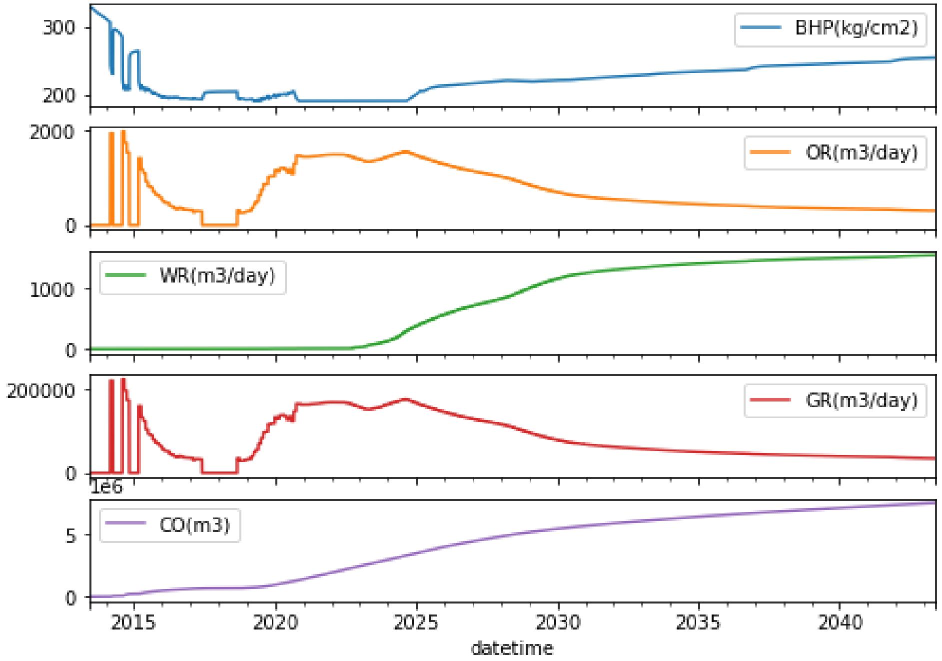
Institute of Petroleum Engineering

# Stage (1): data gathering and description

Well NA1A




Well NA3D



University of Tehran

Institute of Petroleum Engineering

## Stage (2): data preparation

❖ In this study, the following pre-processing steps are applied to the production dataset:

### 1. Split the data:

❖ We'll use a **(42%, 18%, 40%) (Long time-series forecasting)** split for the training, validation, and test sets. Note the data is **not** being randomly shuffled before splitting. This is for two reasons:

1. It ensures that chopping the data into windows of consecutive samples is still possible.
2. It ensures that the validation/test results are more realistic, being evaluated on data collected after the model was trained.

### 2. Normalize the data:

❖ It is important to scale features before training a neural network. Normalization is a common way of doing this scaling.

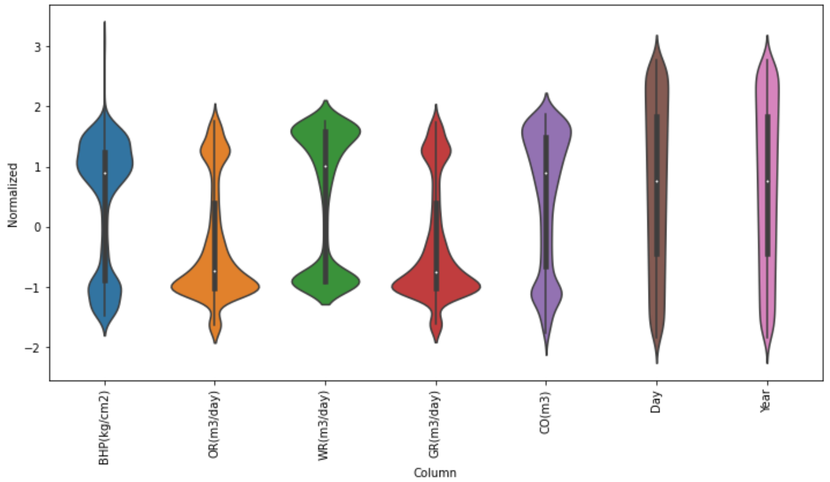
| Normalization Technique | Formula |
|---|---|
| Linear Scaling | x' = (x - x_{min}) / (x_{max} - x_{min}) |
| Clipping | If x > max, then x' = max. If x < min, then x' = min. |
| Log Scaling | x' = log(x) |
| Z-score | x' = (x - μ) / σ |

❖ Subtract the mean and divide by the standard deviation of each feature. The mean and standard deviation should only be computed using the training data so that the models have no access to the values in the validation and test sets.

University of Tehran

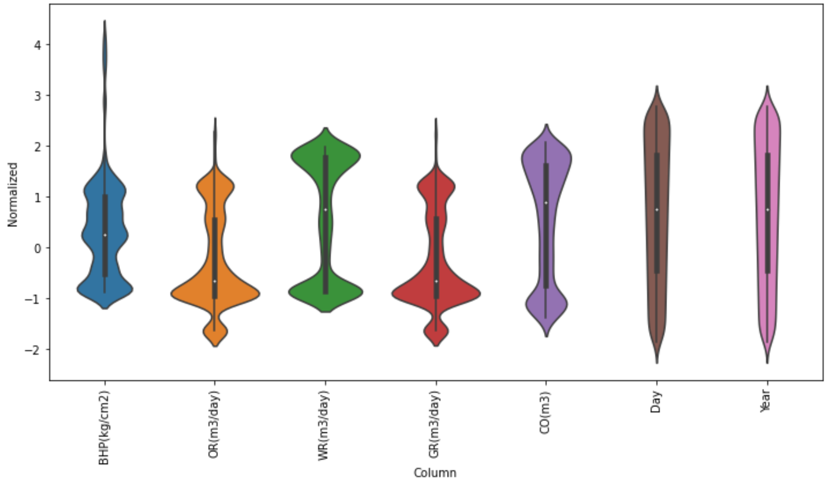
## 3. Outlier removal:

Well NA1A




Now peek at the distribution of the features. Some features do have long tails, but there are no obvious errors.

Well NA3D




Institute of Petroleum Engineering

University of Tehran

Institute of Petroleum Engineering

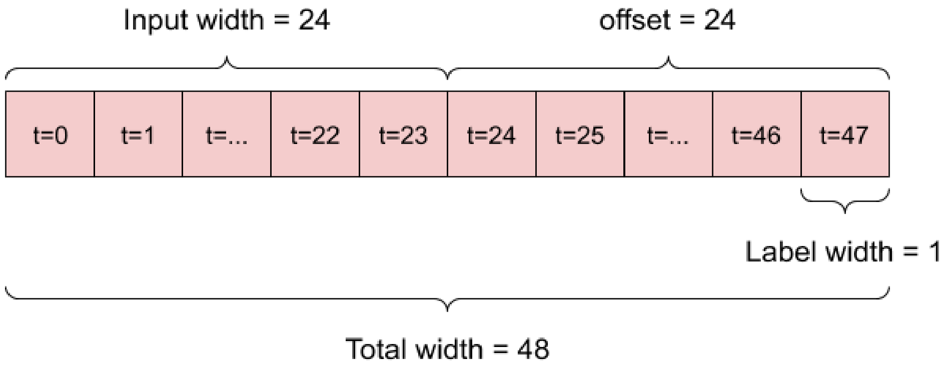
## 4. Data windowing:

❖ The main features of the input windows are:

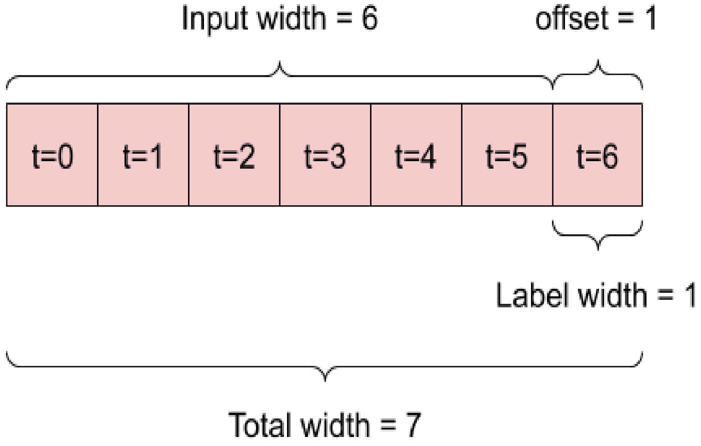
- The width (number of time steps) of the input and label windows
- The time offset between them.
- Which features are used as inputs, labels, or both.

❖ Depending on the task ((1) single-output or multiple-output predictions, (2) single-time-step or multiple-time-step predictions) and the type of model, we may want to generate different windows. Here are some examples:

Example 1




Example 2

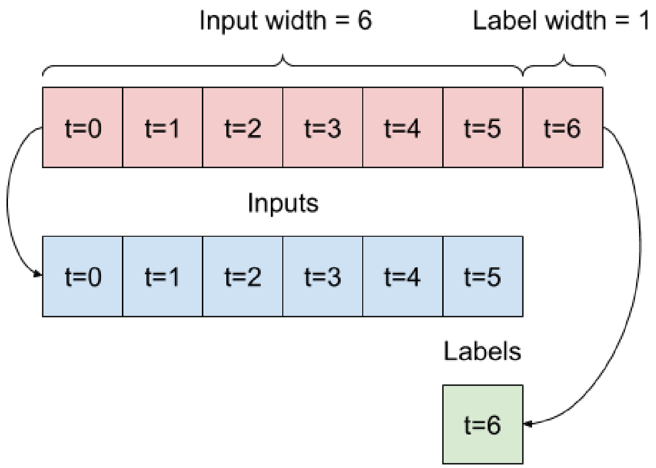


University of Tehran

Institute of Petroleum Engineering

5. **Spilt window:**

❖ Given a list consecutive inputs, the **split_window** method will convert them to a window of inputs and a window of labels. The above example will be split like this:




6. **Multivariable time series (MTS) analysis (involving the information of adjacent wells):**

a) **Pearson's correlation coefficient analysis:**

$$r = \frac{\sum_{i=1}^{n}(I_i - \bar{I})(P_i - \bar{P})}{\sqrt{\sum_{i=1}^{n}(I_i - \bar{I})^2}\sqrt{\sum_{i=1}^{n}(P_i - \bar{P})^2}}$$

❖ in which $I_i$ and $P_i$ are the injector and producer time series, respectively, $n$ is the length of the series, and $\bar{I}$ and $\bar{P}$ are the mean value of the series $I$ and $P$, respectively.

University of Tehran

## a) Pearson's correlation coefficient analysis:

| Adj wells | P5 | P8 | P9 | P12 | P14 | P21 | P24 | P25 | I3 | I5 | I6 | I7 | I10 | I15 | I17 | I19 | I21 | I22 | I23 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $r$ | 0.08 | 0.52 | 0.48 | 0.57 | 0.56 | 0.19 | 0.59 | 0.38 | 0.04 | 0.42 | 0.54 | -0.2 | 0.33 | 0.45 | -0.1 | 0.36 | -0.1 | 0.44 | 0.50 |

Based on Pearson's correlation coefficient analysis, the results are presented for well NA1A (P: Producer & I: Injector). Green highlights indicate the output of Pearson's correlation coefficient analysis for well A (Threshold = 0.4).

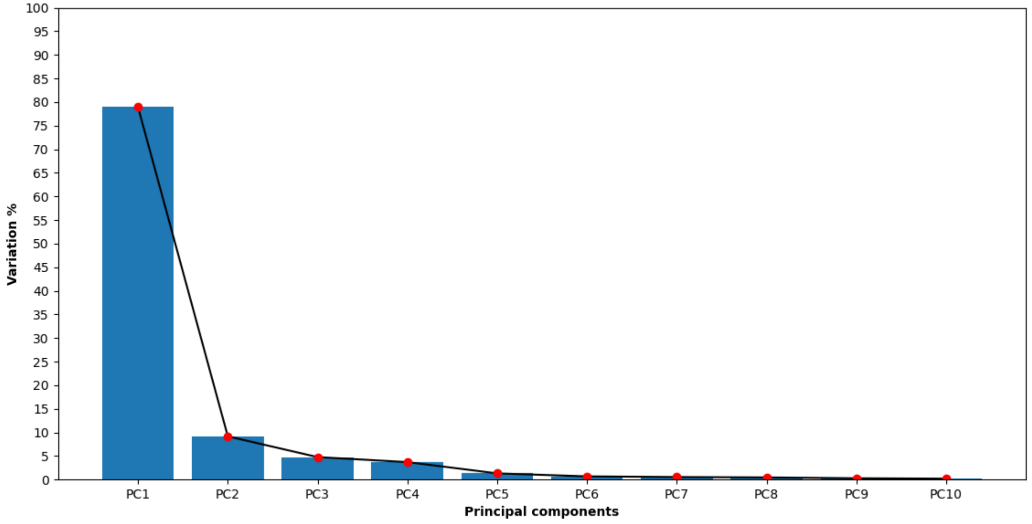
| Adj wells | P5 | P8 | P9 | P12 | P14 | P21 | P24 | P25 | I3 | I5 | I6 | I7 | I10 | I15 | I17 | I19 | I21 | I22 | I23 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $r$ | 0.13 | 0.43 | 0.77 | 0.75 | 0.70 | 0.42 | 0.56 | 0.38 | 0.29 | 0.65 | 0.68 | 0.13 | 0.62 | 0.73 | 0.26 | 0.58 | 0.27 | 0.58 | 0.76 |

Based on Pearson's correlation coefficient analysis, the results are presented for well NA3D (P: Producer & I: Injector). Green highlights indicate the output of Pearson's correlation coefficient analysis for well B (Threshold = 0.5).

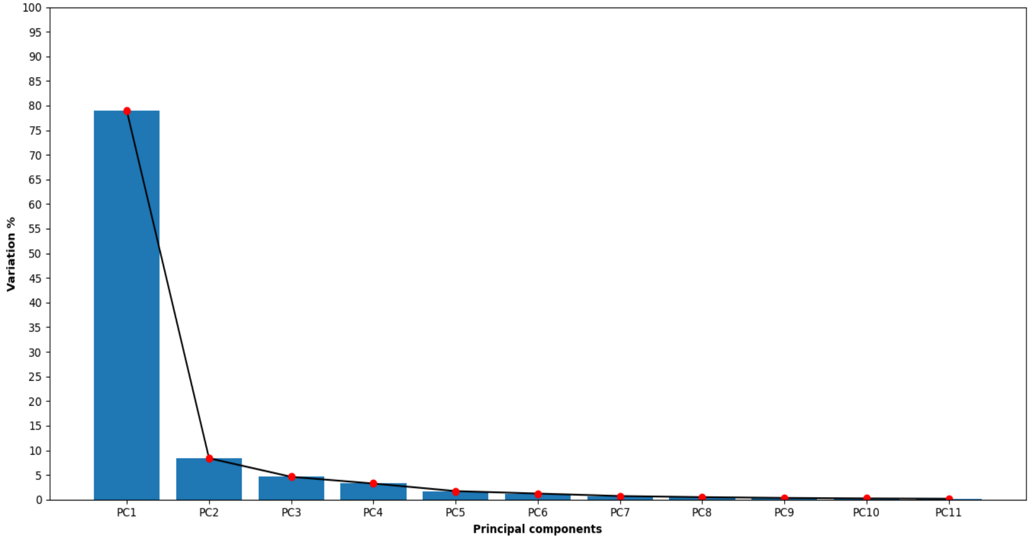
Institute of Petroleum Engineering

## b) Principal component analysis (PCA):

Well NA1A




Well NA3D



University of Tehran

Institute of Petroleum Engineering

## b) Principal component analysis (PCA):

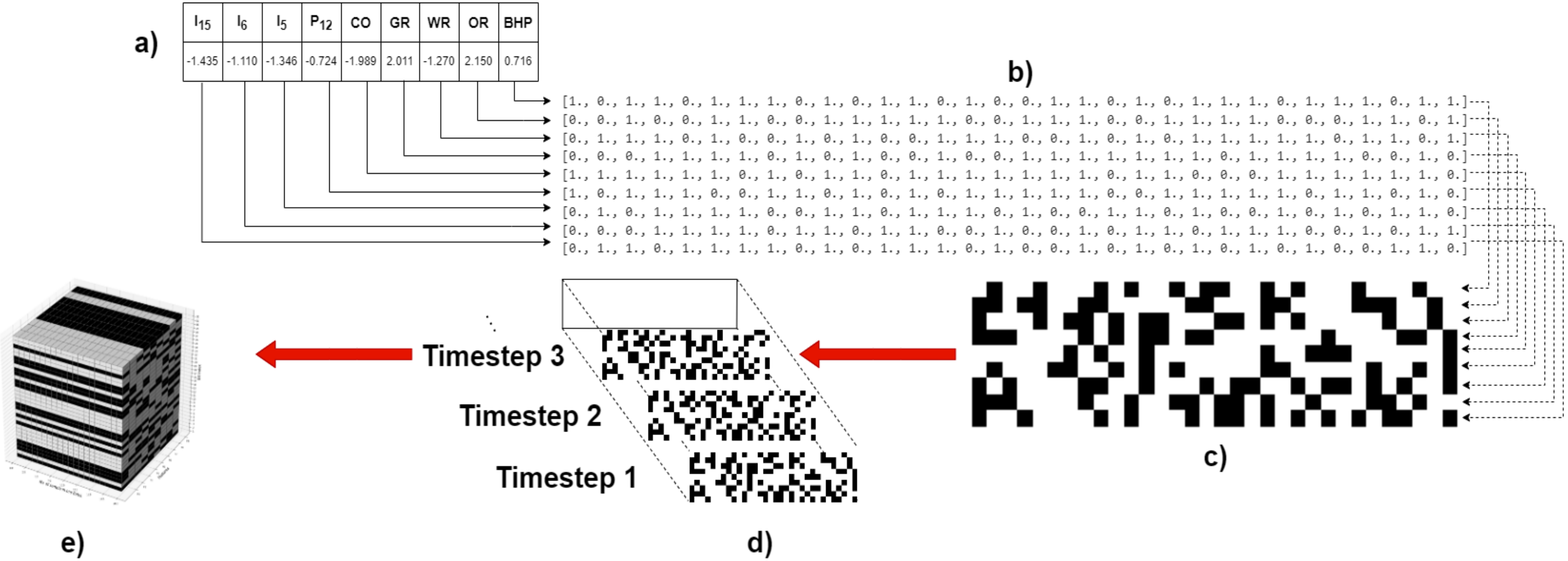
| Adjacent wells | $PC_1$ | $PC_2$ | $PC_3$ | $PC_4$ | $PC_5$ | Removal rate |
|---|---|---|---|---|---|---|
| P8 | 0.223449 | 0.646604 | 0.212113 | -0.13177 | 0.13683 | 80% |
| P9 | 0.328354 | -0.41006 | -0.10282 | -0.27167 | 0.275609 | 80% |
| P12 | 0.371525 | -0.11281 | 0.290257 | -0.37338 | -0.0953 | 40% |
| P14 | 0.256408 | -0.07383 | 0.252495 | -0.35841 | 0.088091 | 60% |
| P24 | 0.231705 | 0.475298 | 0.080153 | 0.105588 | 0.358978 | 60% |
| I5 | 0.353671 | -0.00682 | 0.174285 | 0.620862 | -0.45015 | 40% |
| I6 | 0.37683 | 0.25581 | -0.77976 | -0.20737 | -0.32648 | 40% |
| I15 | 0.378337 | -0.17084 | -0.14848 | 0.445487 | 0.57506 | 40% |
| I22 | 0.236927 | 0.038679 | 0.359533 | -0.03885 | -0.31215 | 80% |
| I23 | 0.344141 | -0.27111 | 0.017103 | 0.037138 | -0.14915 | 80% |

Well NA1A

| Adjacent wells | $PC_1$ | $PC_2$ | $PC_3$ | $PC_4$ | $PC_5$ | $PC_6$ | Removal rate |
|---|---|---|---|---|---|---|---|
| P9 | 0.346358 | -0.35452 | -0.03098 | -0.40134 | -0.33296 | 0.214447 | 16% |
| P12 | 0.37804 | -0.35008 | 0.168953 | 0.179229 | -0.25565 | -0.22829 | 33% |
| P14 | 0.215051 | -0.29576 | 0.161945 | 0.138129 | -0.28043 | -0.14134 | 83% |
| P24 | 0.200822 | 0.207254 | 0.399584 | 0.489795 | -0.00036 | -0.11506 | 66% |
| I5 | 0.327747 | -0.00203 | -0.16291 | 0.369094 | 0.417369 | 0.607714 | 33% |
| I6 | 0.331205 | 0.151448 | 0.637504 | -0.47357 | 0.439518 | -0.061 | 33% |
| I10 | 0.353239 | 0.050845 | -0.53222 | -0.07626 | 0.357322 | -0.62404 | 33% |
| I15 | 0.286393 | 0.034736 | -0.22167 | -0.12264 | -0.08432 | 0.278309 | 66% |
| I19 | 0.292321 | 0.704968 | -0.0872 | -0.01797 | -0.48283 | 0.016501 | 66% |
| I22 | 0.202126 | -0.17679 | 0.059754 | 0.397409 | 0.052919 | -0.13743 | 83% |
| I23 | 0.316742 | -0.25761 | -0.0899 | -0.09059 | 0.083964 | 0.097989 | 83% |

Well NA3D

## 7. Image generation:

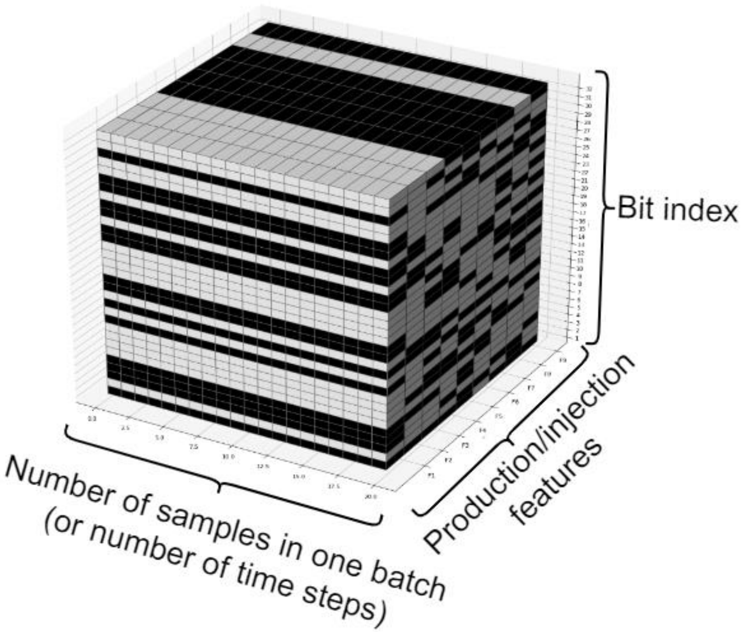


University of Tehran

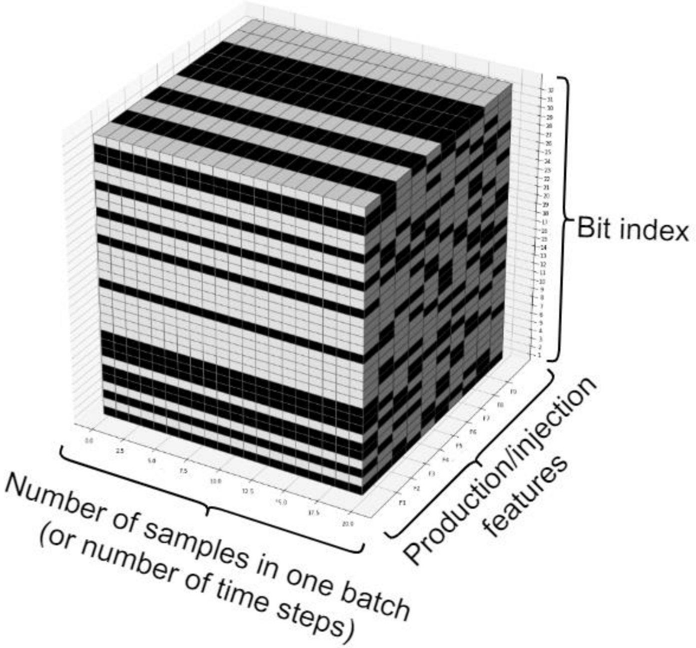
Institute of Petroleum Engineering

## 8. Image generation:

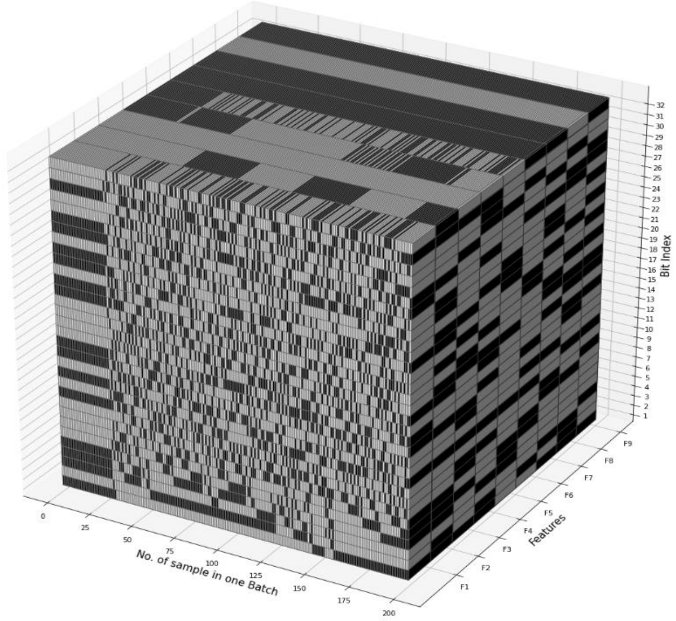
Well NA1A




Well NA3D




Example for 200 time-step



University of Tehran

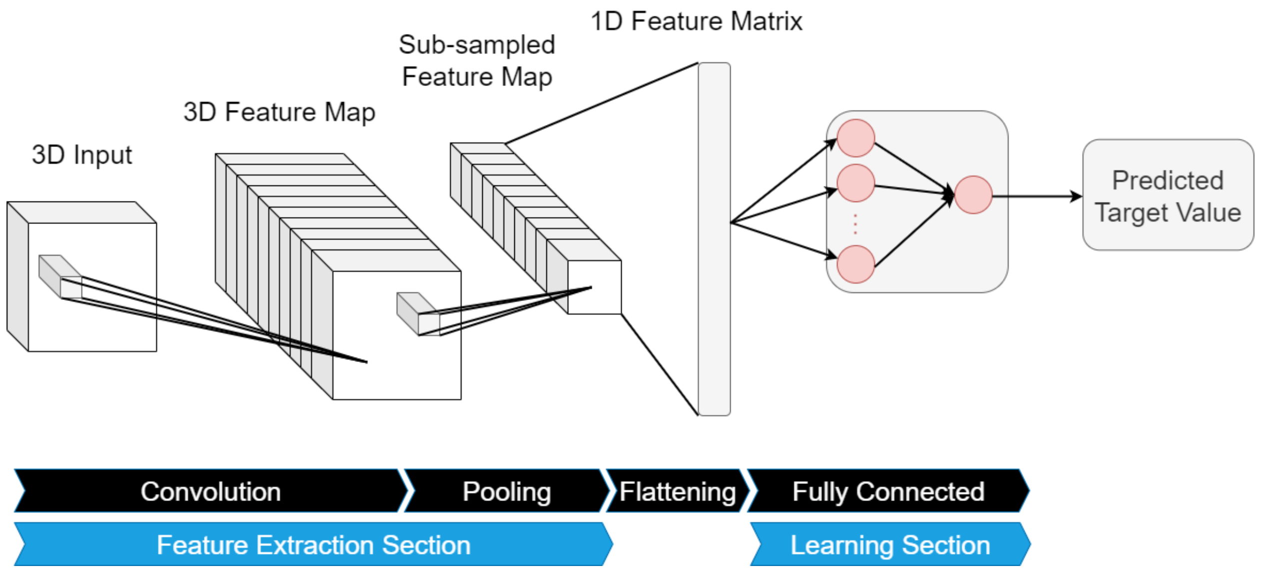
# Deep Artificial Neural Networks

Institute of Petroleum Engineering

University of Tehran

Institute of Petroleum Engineering

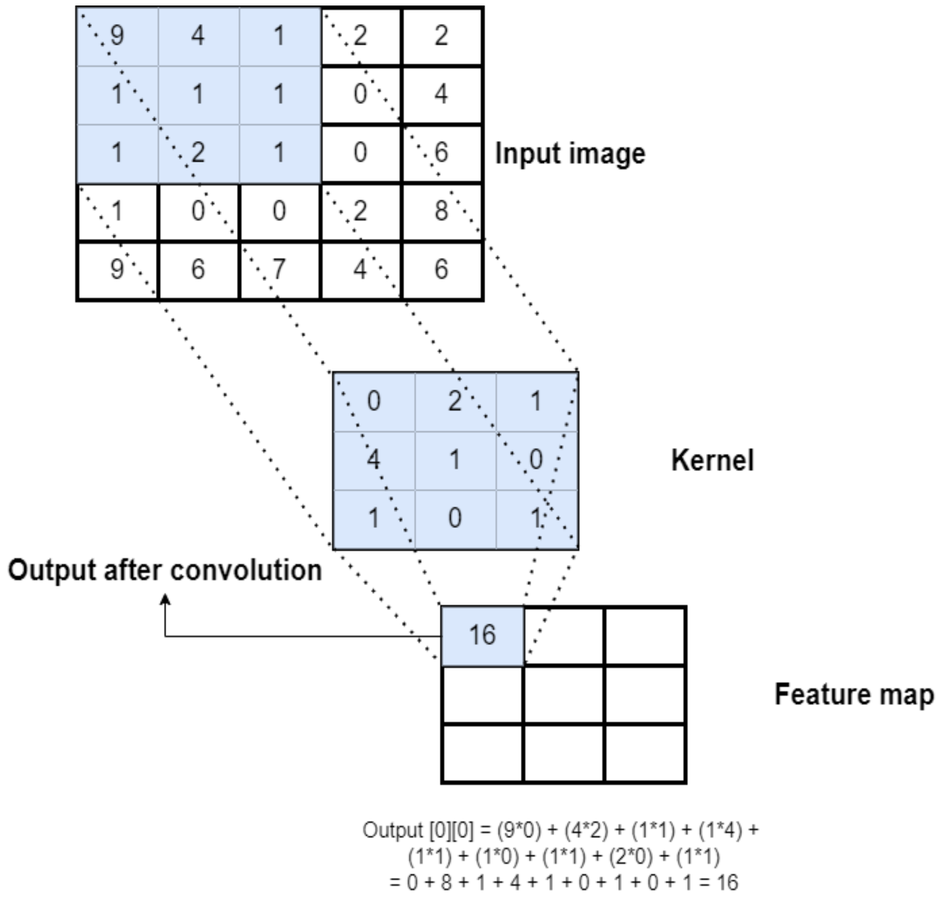
1. **Convolutional Neural Network (CNN):** CNNs are the most **popular class of deep learning structures**, and their widespread application is mostly related to **image processing tasks**. CNN architecture is composed of three main layer types: **(1) Convolutional layer, (2) subsampling (pooling or downsampling) layer, and (3) Fully-connected (FC) layer.**



University of Tehran

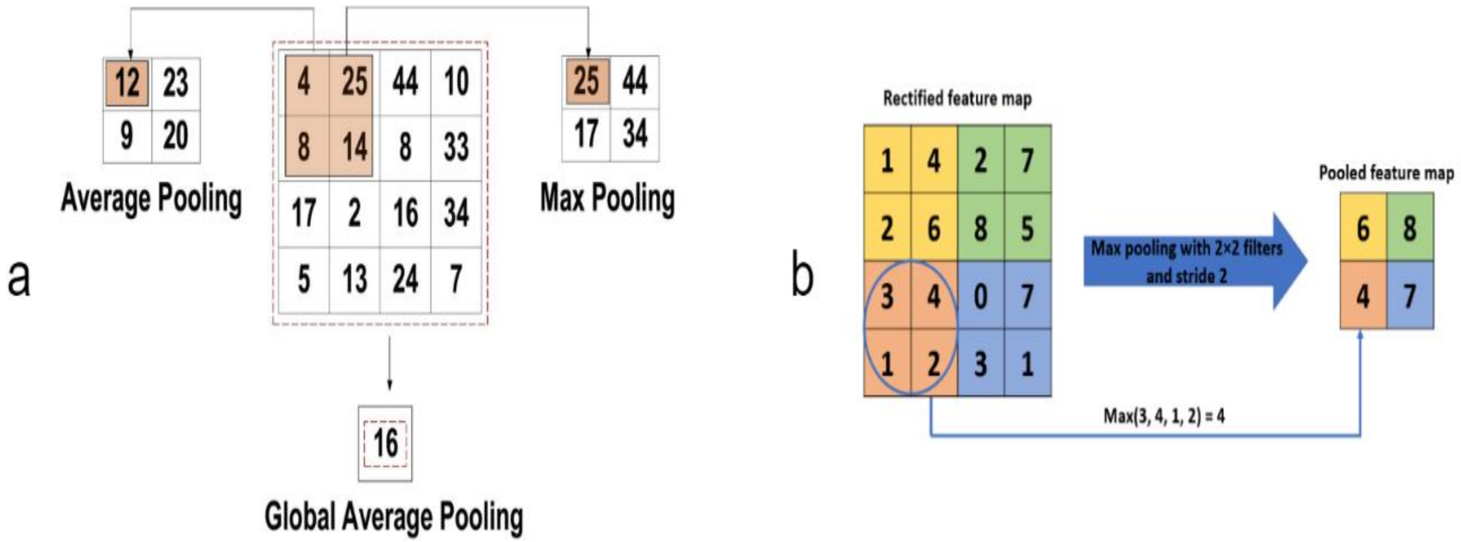
Institute of Petroleum Engineering

- ***Convolutional layer:*** the basic building block of the CNN is the convolutional layer, which consists of three components, **input data, kernel, and feature map**.

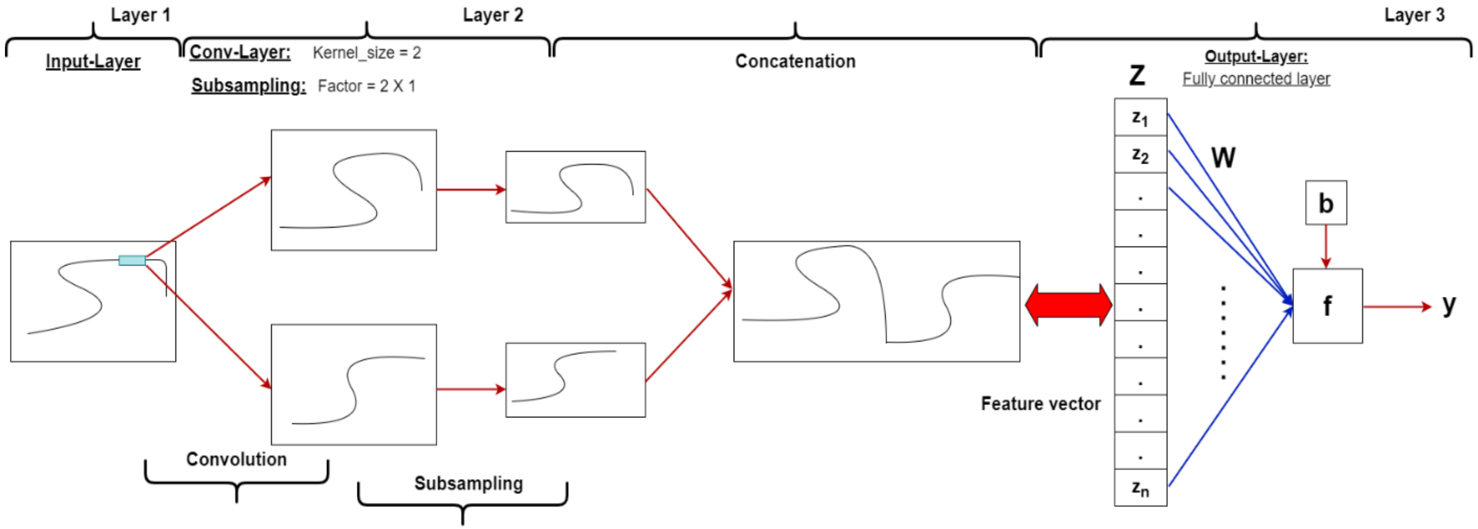


University of Tehran

- ***Subsampling layer:*** the subsampling layer aims to **decrease the number of parameters and reduce the dimensionality of the input**.




- ***Fully-connected layer:***

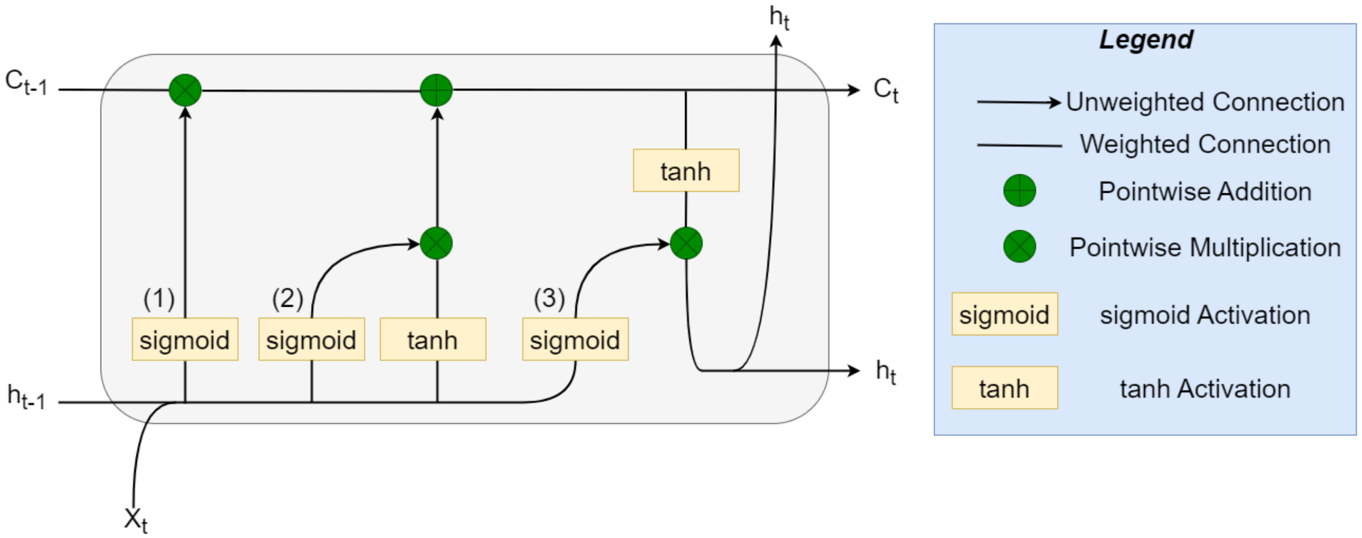



Institute of Petroleum Engineering

University of Tehran

Institute of Petroleum Engineering

2. **Recurrent Neural Network (RNN):** There are two main recurrent layers in the literature: **LSTM (Long Short-Term Memory)** and **GRU (Gated Recurrent Unit)**. LSTM is a recurrent layer that remembers previous steps using three gates that manipulate information.




$X_t$ represents the input at instant $t$, $C_{t-1}$ and $C_t$ are the memory from the previous LSTM cell and current cell, and $h_{t-1}$ and $h_t$ are the output of the previous cell and the output of the actual cell. The gates of this cell are enumerated as follows: (1) is the forget gate, (2) is the input gate, and (3) is the output gate. The forget gate removes irrelevant information from the previous steps. The input gate, which enables the input information to be accumulated, is responsible for updating the cell memory. Finally, the output gate is capable of shutting off the cell's output.

University of Tehran

3. **CNN-LSTM model:** It is important to consider three main factors when selecting a neural network for forecasting: (1) the complexity of the data, (2) the accuracy required in forecasting, and (3) the characteristics of the input data. It is appropriate to use **LSTM** when the input data consists of a chain of **time series**. **CNNs** or **feed-forward neural networks** are excellent options if the **complexity of the data** is taken into account. When the input data is in the **form of images**, **CNNs** are the best choice.

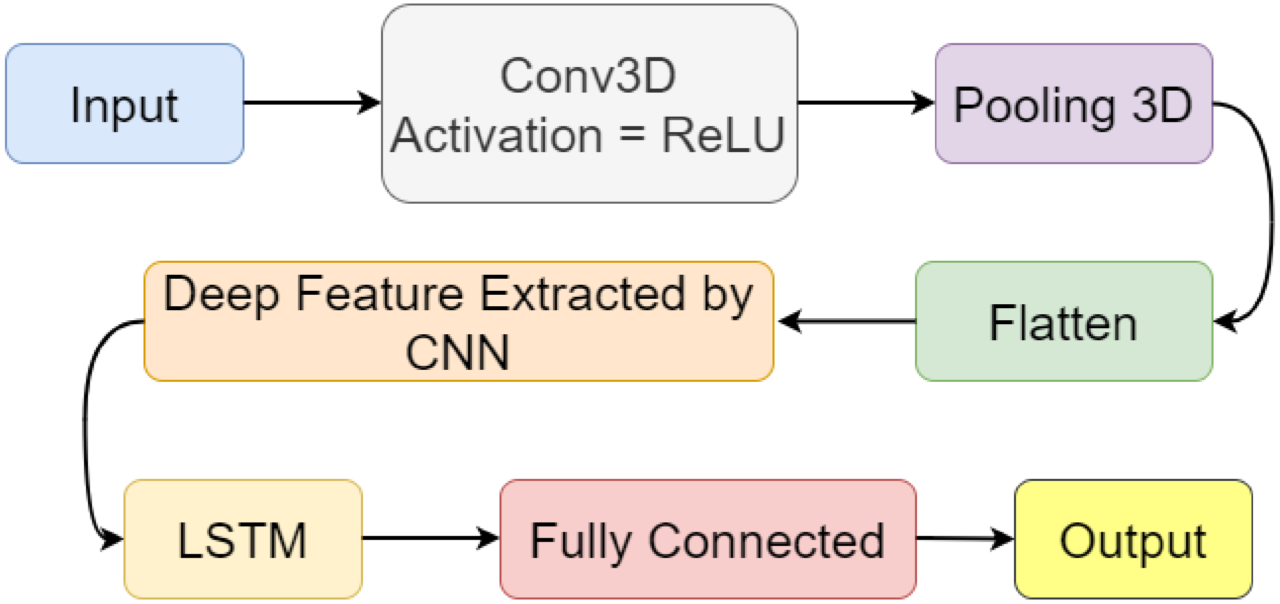
We may, however, be dealing with a time series of images in our data.

Our system will require a more complex architecture in which CNN and LSTM networks are combined. The CNN-LSTM network is derived from the combination of these two networks.

Institute of Petroleum Engineering

University of Tehran

3. **CNN-LSTM model:** A CNN-LSTM model consists of two stages: in the first stage, the CNN is used to extract features from the multidimensional time series, and in the second stage, the LSTM is implemented to predict the extracted features.



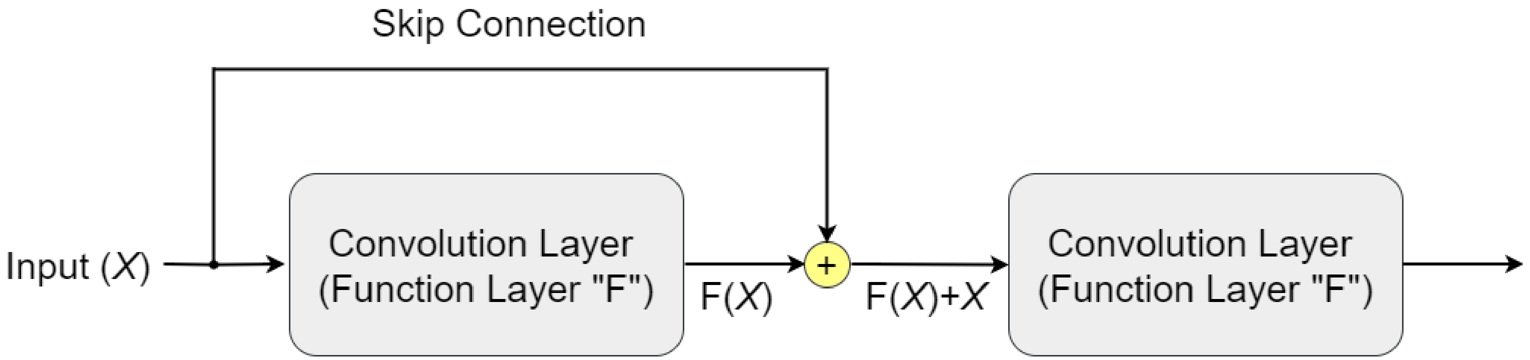

Institute of Petroleum Engineering

University of Tehran

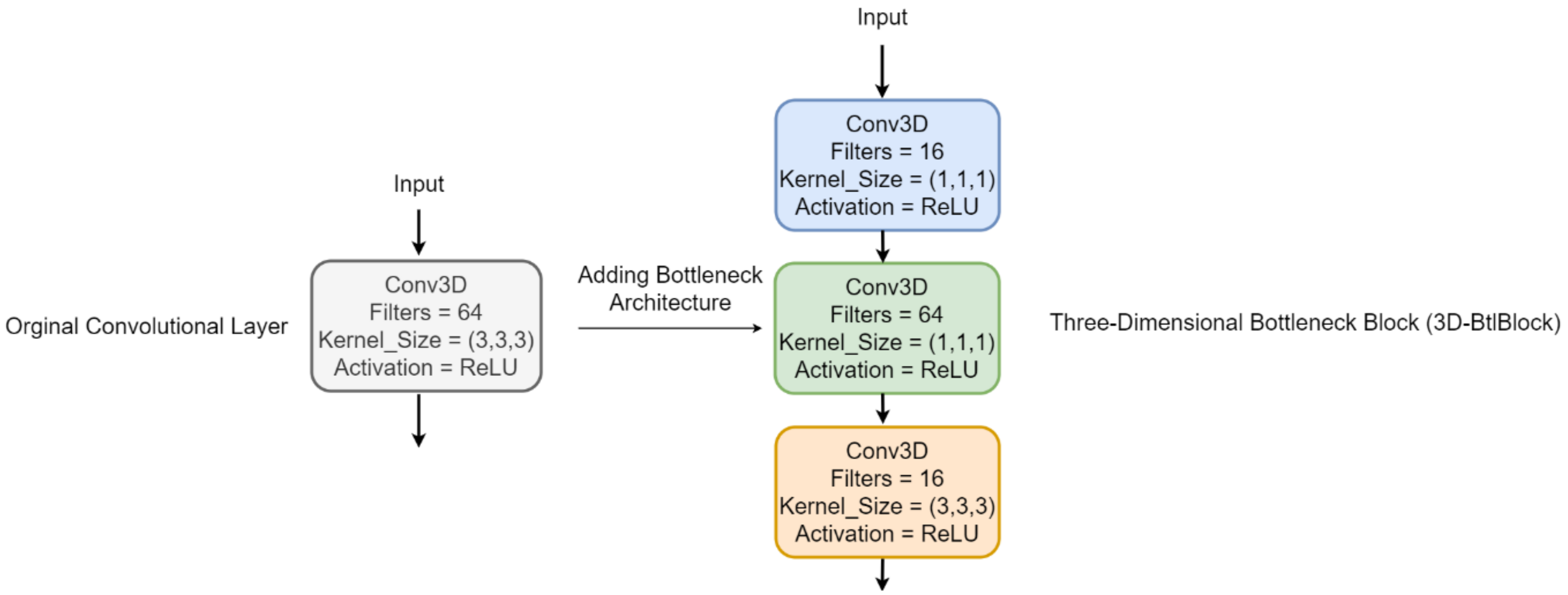
Institute of Petroleum Engineering

1. The first modification is **the elimination of the subsampling layer**, aiming at the prevention of information loss.

2. The second modification is **adding the residual architecture**.



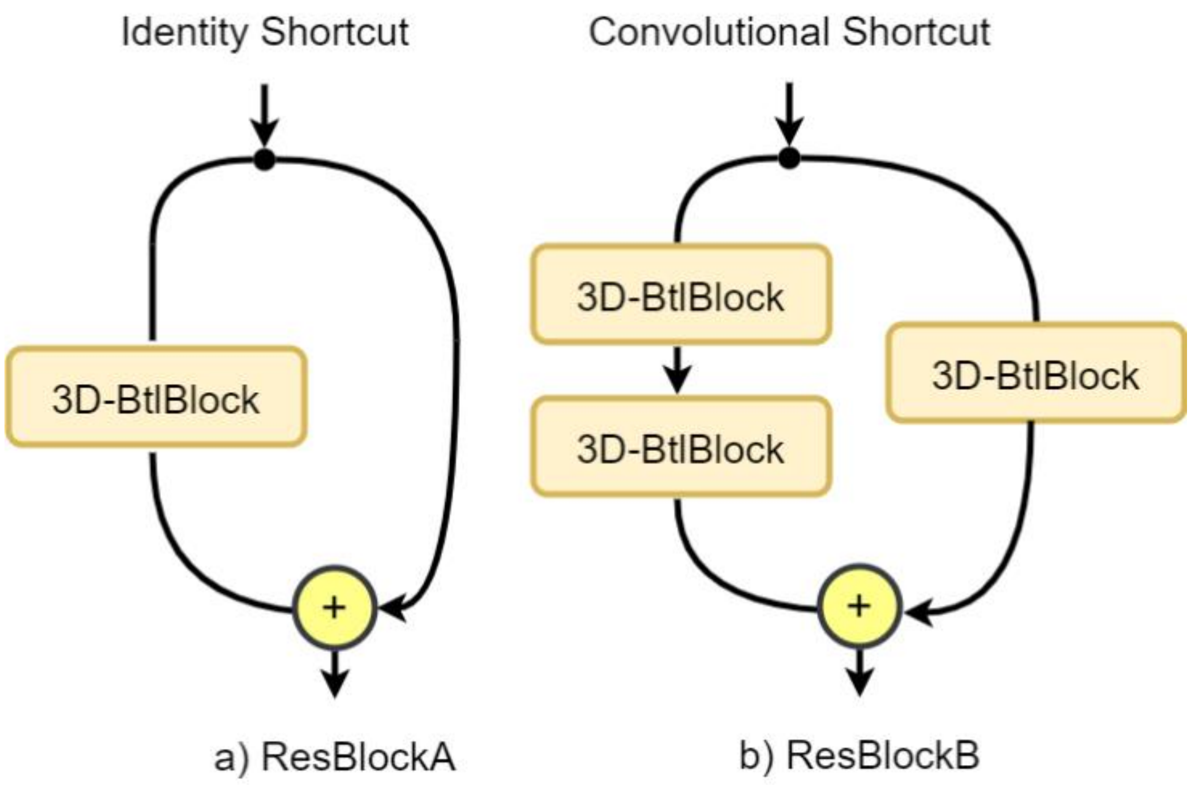
3. The third modification is utilizing **deeper bottleneck architecture**. In this study, adding bottleneck architecture decreased the number of trainable parameters from 4,156,273 to 1,065,217 which means 74.37% fewer parameters.



University of Tehran

Institute of Petroleum Engineering

- The applied modifications to a CNN structure discussed above are, in fact, based on the use of residual blocks, which are made of bottleneck blocks (3D-BtlBlock) inside a deep CNN architecture.

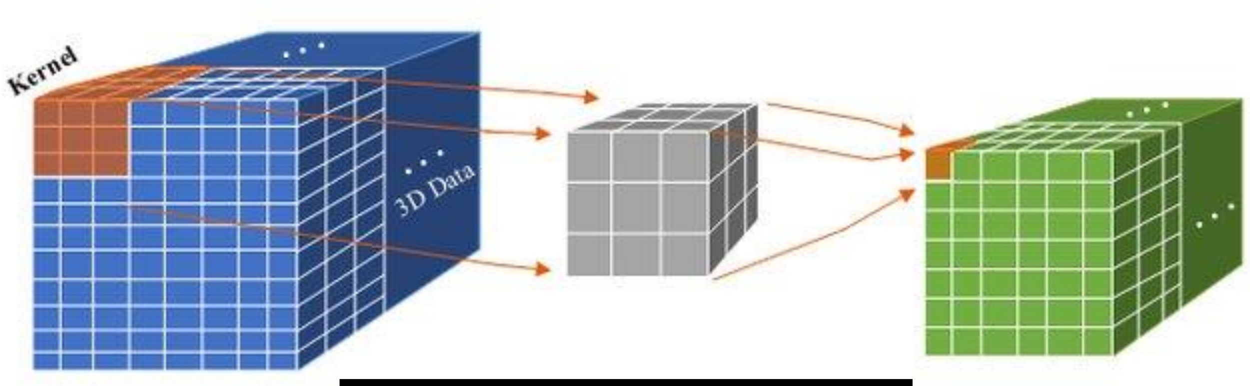


University of Tehran

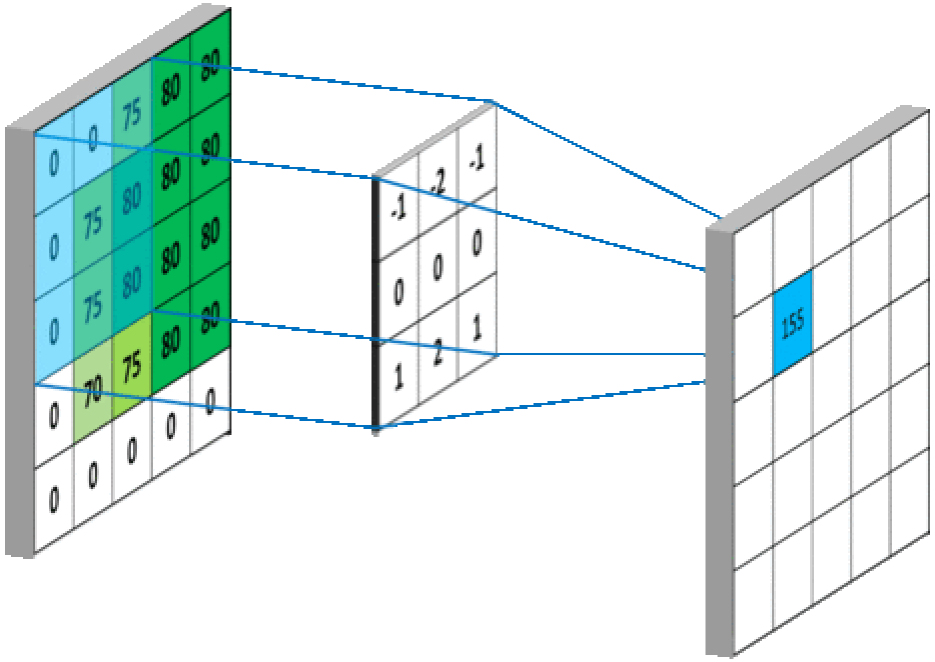
Institute of Petroleum Engineering

## 2. Results and Statistics of the proposed Residual 3D-CNN LSTM

❖ Production data can be calculated **independently** at the first layer of the traditional ANN because neurons within the same layer are not **interconnected**.




Kernel moving in 3D-CNN



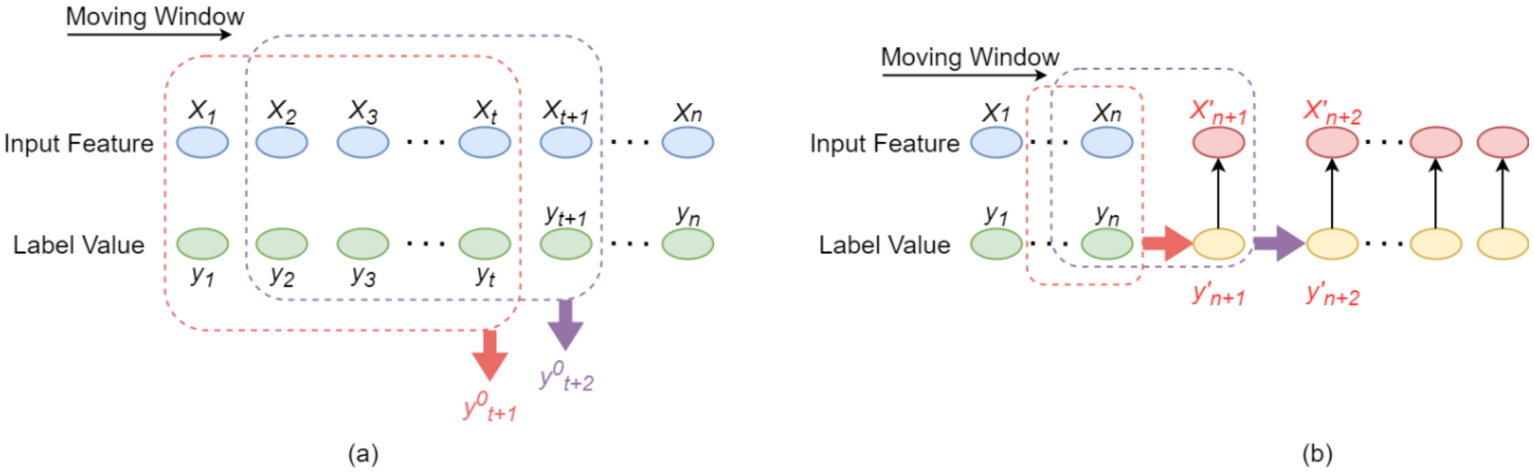

Kernel moving in 2D-CNN

University of Tehran

Institute of Petroleum Engineering

## Stage (3): the implementation of (training) and evaluation of intelligent models

**1. Predictive strategy:**




**2. Forecasting process:**

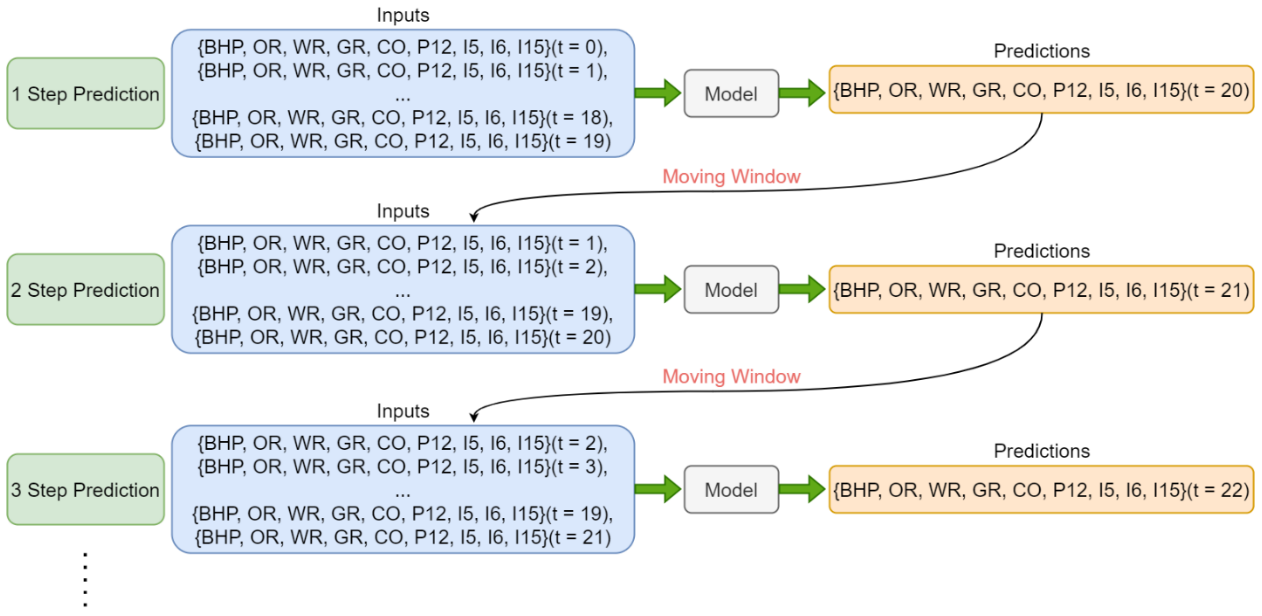
❖ To forecast the long-term production time series, we used the following two data-driven forecasting techniques:

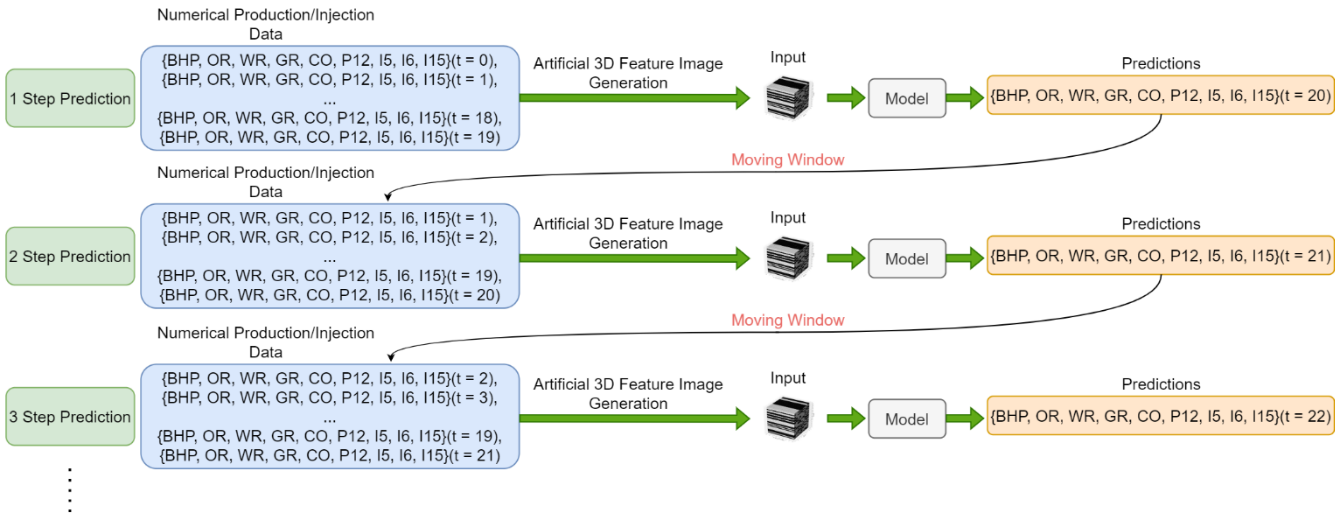
- a conventional deep LSTM model which takes numerical production/injection data as inputs, and
- a modified deep 3D-CNN LSTM (called Residual 3D-CNN LSTM) which takes artificial 3D feature images as inputs.

University of Tehran

Institute of Petroleum Engineering

## 2. Forecasting process:



University of Tehran

Institute of Petroleum Engineering

## 3. Hyper-parameters setting and training process:

| Hyper-parameter | Abbreviation | Type | Bounds or Values |
|---|---|---|---|
| **Optimizer** | optimizer | Choice | Values: $['adam', 'rms', 'sgd']$ |
| **Learning rate** | learning_rate | Range | Bounds: $[0.0001, 0.5]$ |
| **Activation function** | activation | Choice | Values: $['tanh', 'sigmoid', 'relu']$ |
| **Batch size** | batch_size | Choice | Values: $[8, 16, 32, 64, 128, 256]$ |
| **Number of neurons (Conv3D Layers)** | num_neurons_per_layer_1 | Choice | Values: $[8, 16, 32, 64, 128, 256]$ |
| **Number of neurons (LSTM Layers)** | num_neurons_per_layer_2 | Range | Bounds: $[1, 500]$ |
| **The width (number of time steps) of the input window** | t | Choice | Values: $[5, 10, 15, 20, 25, 30]$ |

University of Tehran

Institute of Petroleum Engineering

## 3. Hyper-parameters setting and training process:

**Algorithm 1: Training conventional deep LSTM**

Input:

| | |
|---|---|
| $X$ | Time series data in the training set (numerical production/injection data). |

Output:

| | |
|---|---|
| $w_L$ | The set of weights in LSTM. |
| $b_L$ | The set of biases in an LSTM. |
| $a^*$ | Other parameters of the trained model. |

```
Initialize parameters w_L, b_L, a in conventional deep LSTM;
for All x ∈ X do;
    Update w_L, a in conventional deep LSTM based on X;
    Min E_r = Σ_{t=1}^{m} (h_t − y_t)^2
    where h_t is the output and y_t is the target;
    Back-propagation;
    w_L ← w_L, b_L ← b_L, a* ← a;
end for
```

**Algorithm 2: Training Residual 3D-CNN LSTM**

Input:

| | |
|---|---|
| $X$ | Time series data in the training set (artificial 3D feature images). |

Output:

| | |
|---|---|
| $w_C$ | The set of weights in CNN. |
| $b_C$ | The set of biases in a CNN. |
| $w_L$ | The set of weights in LSTM. |
| $b_L$ | The set of biases in an LSTM. |
| $a^*$ | Other parameters of the trained model. |

```
Initialize parameters w_C, w_L, b_C b_L, a in Residual 3D-CNN LSTM;
for All x ∈ X do;
    Update w_C, w_L, a in Residual 3D-CNN LSTM based on X;
    Min E_r = Σ_{t=1}^{m} (H_t − y_t)^2
    where H_t is the output and y_t is the target;
    Back-propagation;
    w_C ← w_C, b_C ← b_C, w_L ← w_L, b_L ← b_L, a* ← a;
end for
```

University of Tehran

Institute of Petroleum Engineering

## 4. Evaluation criteria:

| Model evaluation parameters | Mathematical expression |
|---|---|
| Mean Square Error ($MSE$) | $\mathrm{MSE} = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2$ |
| Mean Absolute Error ($MAE$) | $MAE = \frac{1}{m}\sum_{i=1}^{m}\lvert y_i - \hat{y}_i\rvert$ |
| Mean Absolute Percentage Error ($MAPE$) | $MAPE = \frac{1}{m}\sum_{i=1}^{m}\left\lvert\frac{y_i - \hat{y}_i}{y_i}\right\rvert$ |

Note: where $m$ is the total number of data, $\hat{y}_i$ is the predicted value, and $y_i$ is the true value.

University of Tehran

# Results and Discussion

Institute of Petroleum Engineering

University of Tehran

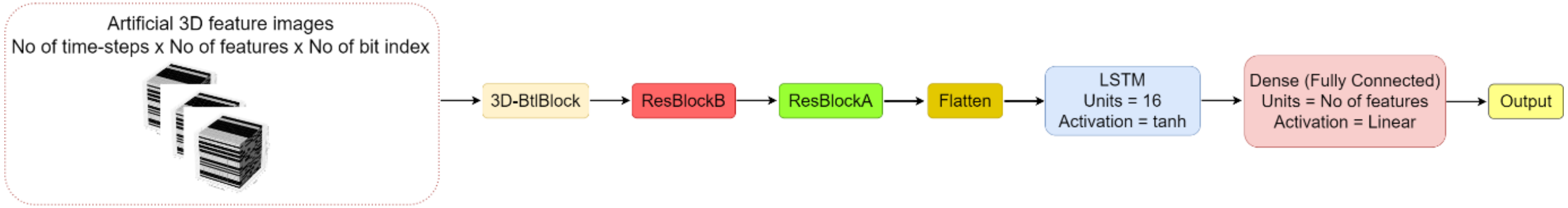
# 1. The architecture of the proposed data-driven forecasting technique

| Well | Optimizer | Learning rate | Activation functions | Batch size | Allocated number of epochs to train the model | The width (number of time steps) of the input window | Number of hidden layers | Number of neurons |
|---|---|---|---|---|---|---|---|---|
| NA1A | Adam | 0.00133 | Conv3D Layers: ReLU<br>LSTM Layers: Tanh<br>Dense Layer (Output layer): Linear | 64 | 500 | 20 | Conv3D Layers 15<br>LSTM Layers: 1 | Conv3D Layers: 16 & 64<br>LSTM Layer: 16<br>Dense Layer (Output layer): Number of input features (9) |
| NA3D | Adam | 0.00027 | Conv3D Layers: ReLU<br>LSTM Layers: Tanh<br>Dense Layer (Output layer): Linear | 32 | 500 | 20 | Conv3D Layers 15<br>LSTM Layers: 1 | Conv3D Layers: 16 & 64<br>LSTM Layer: 16<br>Dense Layer (Output layer): Number of input features (10) |

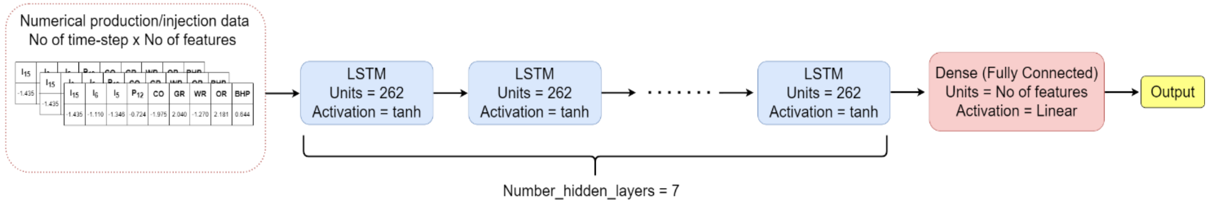
Institute of Petroleum Engineering

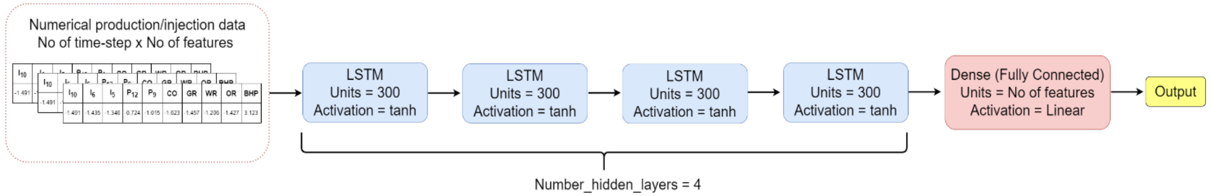
University of Tehran

# 1. The architecture of the proposed data-driven forecasting technique



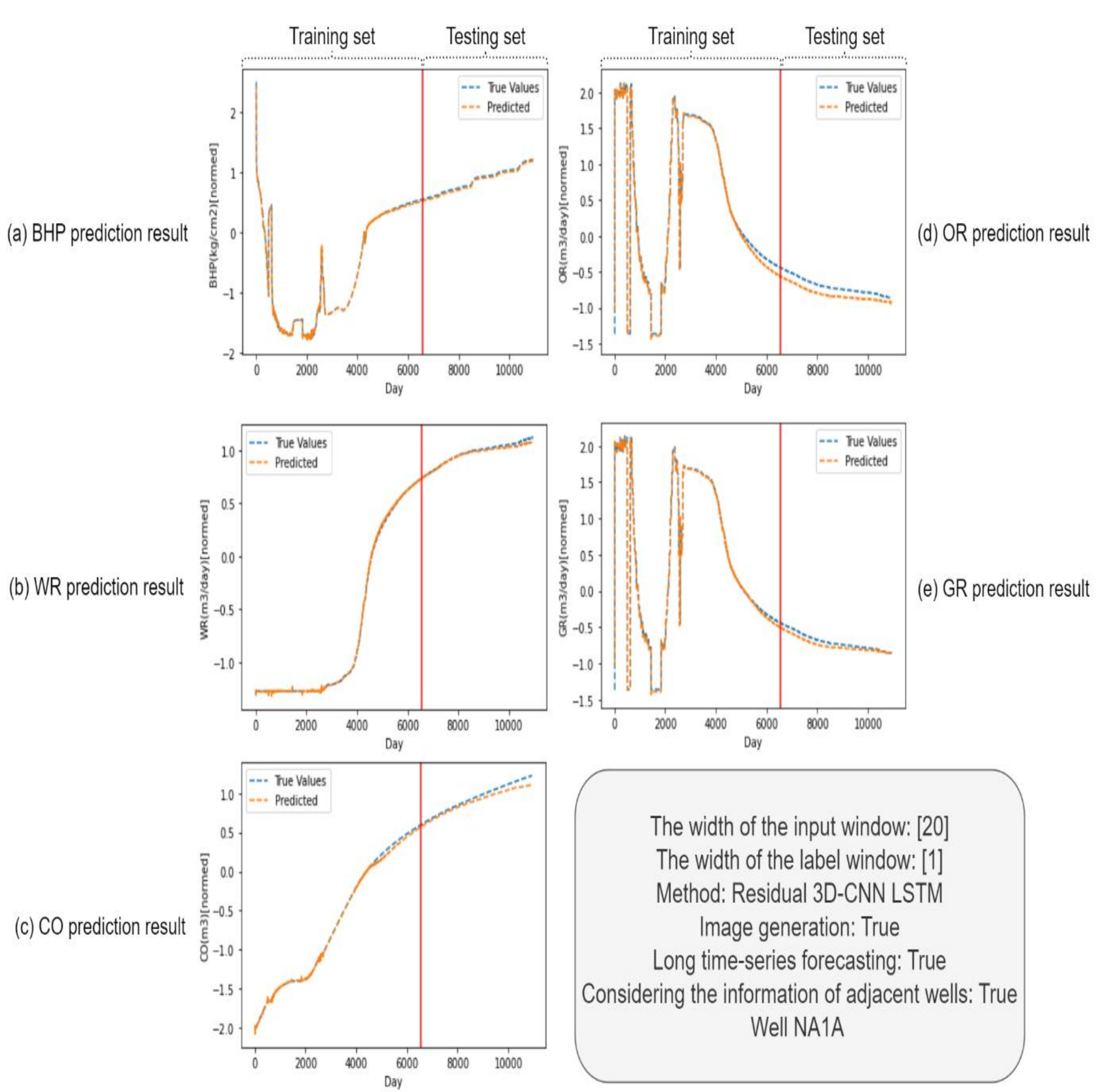

| Well | Optimizer | Learning rate | Activation functions | Batch size | Allocated number of epochs to train the model | The width (number of time steps) of the input window | Number of hidden layers | Number of neurons |
|---|---|---|---|---|---|---|---|---|
| NA1A | Nadam | 0.001624939 | LSTM Layers: Tanh Dense Layer (Output layer): Linear | 32 | 500 | 20 | LSTM Layers: 7 | LSTM Layers: 262 Dense Layer (Output layer): Number of input features (9) |
| NA3D | Nadam | 0.003059166 | LSTM Layers: Tanh Dense Layer (Output layer): Linear | 128 | 500 | 20 | LSTM Layers: 4 | LSTM Layers: 300 Dense Layer (Output layer): Number of input features (10) |

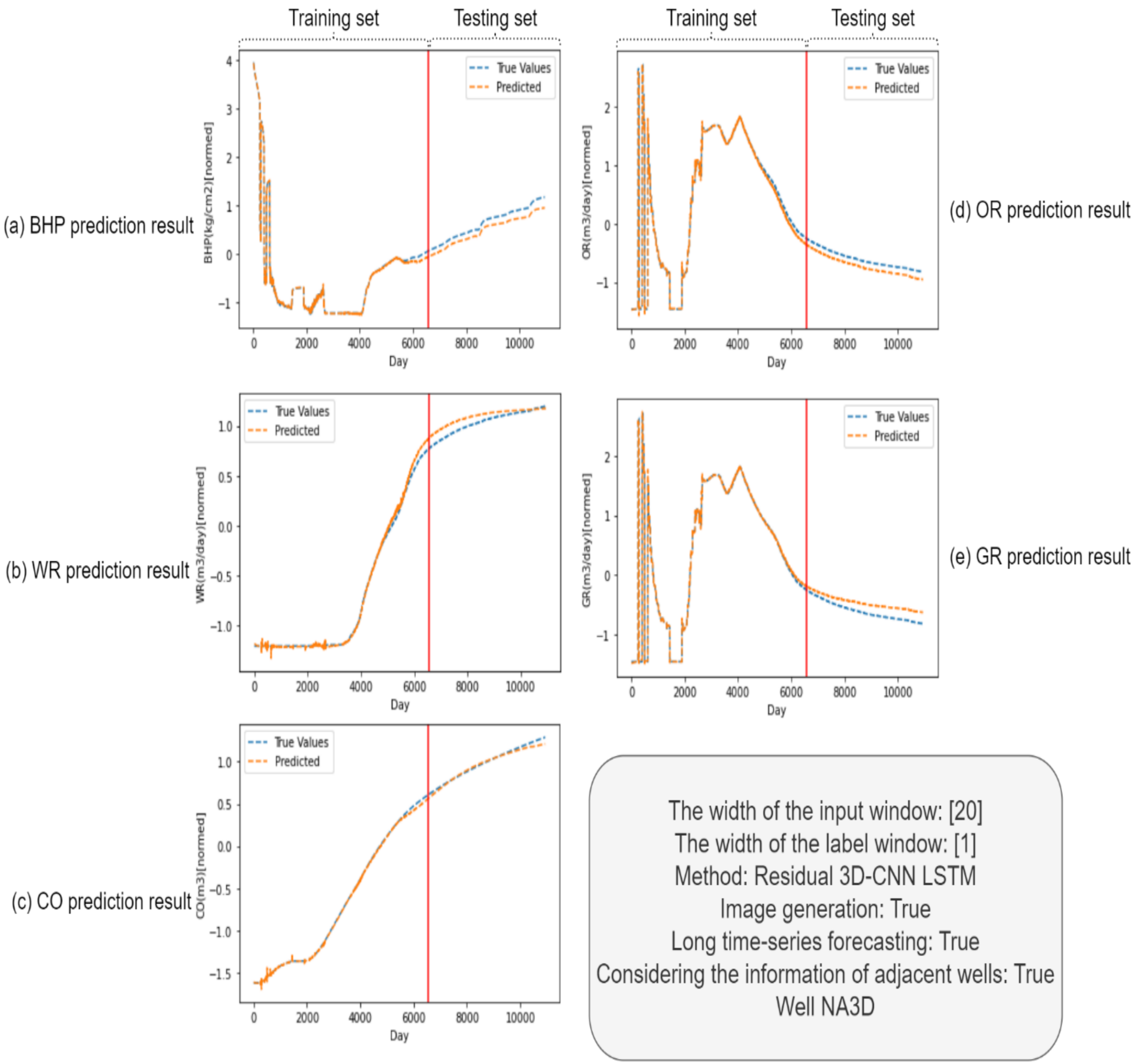


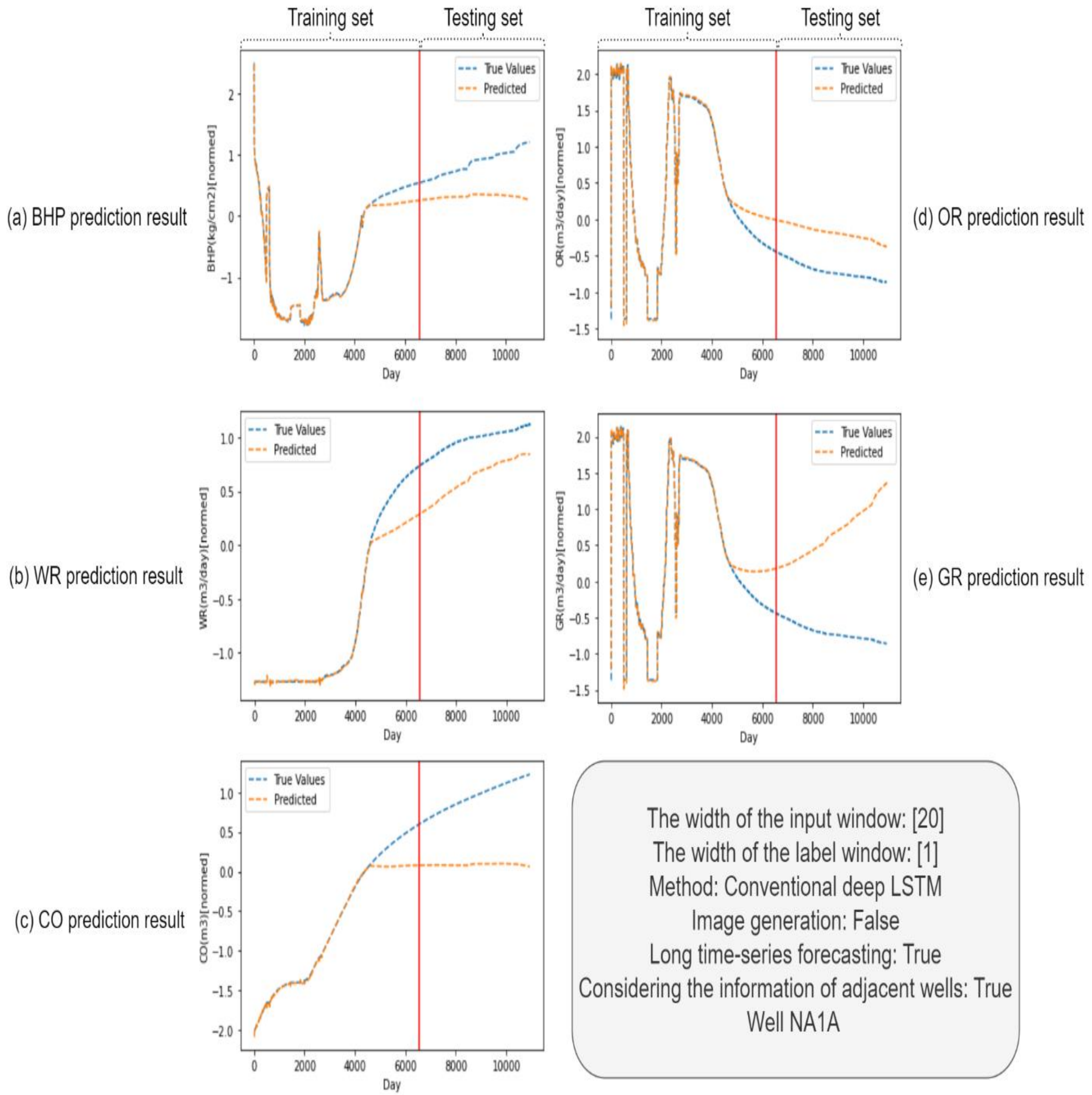

Institute of Petroleum Engineering

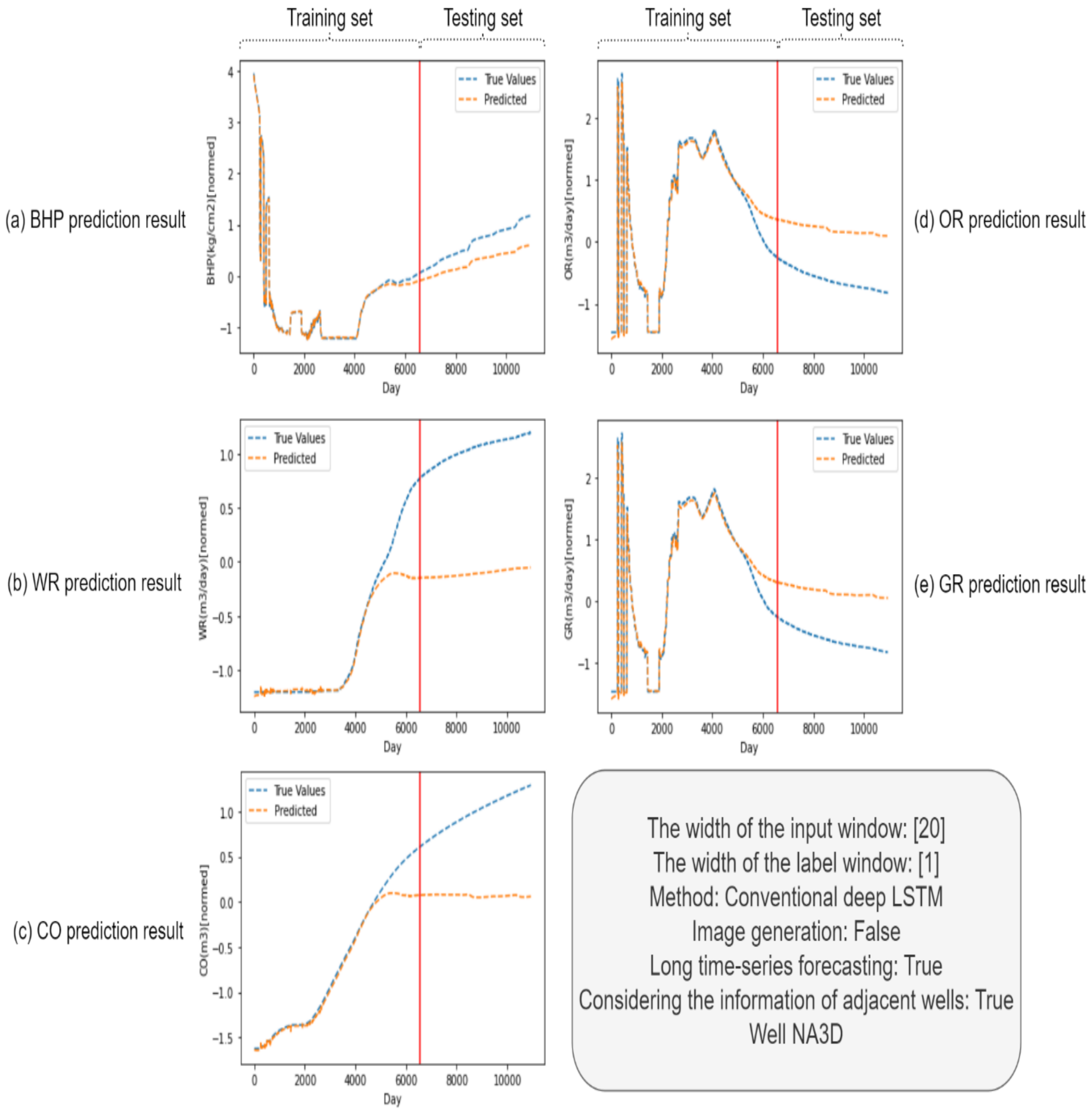
## 2. Results and Statistics of the proposed Residual 3D-CNN LSTM



## 2. Results and Statistics of the proposed Residual 3D-CNN LSTM



University of Tehran

## 2. Results and Statistics of the proposed Residual 3D-CNN LSTM

| Model evaluation parameters | Models | Res 3D-CNN LSTM model | | LSTM model | |
|---|---|---|---|---|---|
| | Datasets | Train | Test | Train | Test |
| MSE | BHP | 0.00069 | 0.00076 | 0.01355 | 0.32975 |
| MSE | OR | 0.00866 | 0.01161 | 0.03201 | 0.27706 |
| MSE | WR | 0.00280 | 0.07497 | 0.03553 | 0.12998 |
| MSE | GR | 0.00686 | 0.00261 | 0.04778 | 2.05802 |
| MSE | CO | 0.00073 | 0.00329 | 0.03304 | 0.76034 |

Well NA1A

Institute of Petroleum Engineering

University of Tehran

Institute of Petroleum Engineering

## 2. Results and Statistics of the proposed Residual 3D-CNN LSTM

| Model evaluation parameters | Models | Res 3D-CNN LSTM model | | LSTM model | |
|---|---|---|---|---|---|
| | Datasets | Train | Test | Train | Test |
| MSE | BHP | 0.00332 | 0.02325 | 0.00384 | 0.14944 |
| MSE | OR | 0.01295 | 0.01307 | 0.04532 | 0.67233 |
| MSE | WR | 0.00084 | 0.06209 | 0.08450 | 1.31154 |
| MSE | GR | 0.01140 | 0.02189 | 0.03980 | 0.58746 |
| MSE | CO | 0.00040 | 0.00092 | 0.02980 | 0.88208 |

Well NA3D

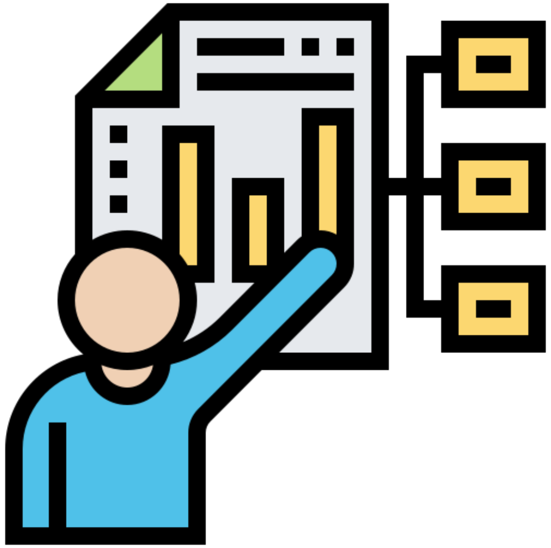
University of Tehran

Institute of Petroleum Engineering

# 3. Effect of residual and deeper bottleneck structures

| Method | Number of hidden layers | Number of trainable parameters |
|---|---|---|
| **Residual 3D-CNN LSTM** | Conv3D Layers:<br>15 | 1,065,217 |
| **Original 3D-CNN LSTM** | Conv3D Layers:<br>5 | 4,156,273 |

University of Tehran



# Conclusion

Institute of Petroleum Engineering

University of Tehran

Institute of Petroleum Engineering

The following four points summarize the contributions of this study:

- Using production and injection data, an innovative procedure is introduced to generate artificial 3D feature images to enhance the predictive performance of the proposed Residual 3D-CNN LSTM model.
- The proposed Residual 3D-CNN LSTM approach can simultaneously predict multiple features.
- By the use of the residual and deeper bottleneck structures, through the reduction of the number of trainable variables, our proposed network becomes deeper while there is no need for more computational complexity or additional parameters.
- In particular, for complex, long-sequence time series data, the proposed model shows superior performance to a conventional deep LSTM model.

University of Tehran

# Thank You

Thanks for your attention

Institute of Petroleum Engineering