Shahrood University of Technology

Journal of Mining and Environment (JME)

Journal homepage: www.jme.shahroodut.ac.ir

Iranian Society of Mining Engineering (IRSME)

Mitigating the Uncertainties of Geomechanical Models by Estimating the Shear Wave Slowness Using Highly Accurate Deep Neural Network Models

Mahdi Bajolvand[1*], Ahmad Ramezanzadeh[2], Amin Hekmatnejad[1], Mohammad Mehrad[2], Shadfar Davoodi[3], and Mohammad Teimouri[4]

1. Departamento de Ingeniería de Minería, Escuela de Ingeniería, Pontificia Universidad Católica de Chile, Santiago, Chile
2. Faculty of Mining, Petroleum and Geophysics Engineering, Shahrood University of Technology, Shahrood, Iran
3. School of Earth Sciences & Engineering, Tomsk Polytechnic University, Lenin Avenue, Tomsk, Russia
4. Iranian Central Oil Fields Company (ICOFC), Tehran, Iran

Article Info

Received 30 October 2024
Received in Revised form 30 October 2024
Accepted 17 March 2025
Published online 17 March 2025

DOI: 10.22044/jme.2025.15294.2932

Keywords

Shear wave slowness
Machine learning
Deep learning
Petrophysical logs
Analytical models

Abstract

Shear Wave Slowness Log (DTSM) is one of the most important petrophysical logs applicable for studying reservoirs, especially geomechanical studying of the oil and gas fields. However, lack of this parameter in wellbore logging can import great sources of uncertainty into geomechanical studies. This study aims to provide solutions for decreasing the uncertainty of geomechanical models with estimation of the DTSM log using the high accurate deep machine learning models. The main idea is using data from offset fields for extending the range of training data and improving the estimation ability and generalizability of machine learning models. For this purpose, petrophysical data from 8 wells of 4 Iranian oil fields were collected. In the first stage, data preprocessing was performed for reducing the effects of wrong data, missing value, noises, and outliers. Then, machine learning (regression learning-based and deep neural network-based) and analytical models implemented for estimating DTSM. The results indicated that the Gated Recurrent Unit (GRU) model with the values of 1.9 and 2.14 for RMSE and 0.99 for R-square had the most exact answers, for training and test data, respectively. Meanwhile, evaluation of the accuracy of the models on the validation well data indicated that GRU model with the values of 2.43 and 0.93 had been the most accurate model for RMSE and R-square, respectively. Accordingly, using a multi-field comprehensive data bank and applying machine learning methods are strongly recommended to estimate the DTSM, for the cases where limited offset data is available.

1. Introduction

Shear Wave Slowness Log (DTSM) is one of the key elements, which is specifically important in geo-mechanical studies of hydrocarbon reservoirs. In addition to its high importance, this log is collected in a limited number of hydrocarbon well fields, especially in reservoirs section, due to the exorbitant expenses of this measurement [1, 2, 3]. Shear wear slowness log (or in the form of shear wave velocity) is used to calculate rock strength and further for other geo-mechanical parameters. Accordingly, in a field where only a few numbers of wells and in a limited part of the reservoirs, this log exists, geo-mechanical estimations and calculations at other wells lacking this log and therefore developing three-dimensional models of the reservoirs are faced with high uncertainty. To face this challenge, several analytical equations have been developed during the last decades due to the high correlation between the shear wave slowness log and compression wave slowness log, and some analytical correlations have been developed between these two logs. Models developed by Pickett (1963), Castagna et al. (1993), and Brocher (2005) can be pointed out as

Corresponding author: mbajolvand@UC.cl (M. Bajolvand)

the commonly applied models in previous literature. On the other hand, given the importance of this parameter, extensive studies have been conducted during the last decades to study and investigate the relation between shear wave velocity and other petro-physical logs such as shear wave (V_p), Gamma Ray (GR), neutron porosity (NPHI), and bulk density (RHOB) using machine learning methods [2,3,7].

One of the commonly applied and simplest methods which has attracted much attention is to use single and multivariate regressions [1,7-12]. Meanwhile, the support vector regression method is also applied in different studies, and it has acceptable results compared to other intelligent methods [13,14,15]. However, a support vector machine algorithm with least squares regular and hybrid regression has been introduced as a powerful algorithm in this regard [2,3,7,9]. One of the most applied machine learning methods is the artificial neural network which has various applications in developing drilling penetration rate and torque [16,17,18]. One of the applications of this algorithm, which has been used both in the regular and hybrid forms, has been shear wave slowness (velocity) estimation [2,7,9,13,14,15,19,20]. Using compatible regular and hybrid fuzzy inference systems [18], and adaptive neuro-fuzzy inference system (ANFIS) with optimization algorithms have attracted the attention of the researchers for the estimation of the shear wave slowness from other petro-physical logs [14,15].

Meanwhile, the Multi-layer Extreme Learning Method (MELM) was compared for the first time as a simple form or in combination with other optimization algorithms with the performance of LSSVM and CNN models in a study by Mehrad et al. (2022) and Rajabi et al. (2022), which indicated that this model has good potential to be applied in this issue. In recent years, using deep learning networks has a special position in data-based studies due to its high strength in extracting complex relations dominated among the parameters. The application of LSTM and CNN deep learning networks has been assessed to estimate the shear wave slowness, which has more successful performance in these studies compared to other machine learning models [3,23-26]. However, as the application of these models is novel, more assessment in different studies and based on different data banks is needed.

In the most conducted studies, the dominant strategy has been the assessment of machine learning algorithms in the estimation capability of the shear wave slowness or velocity from other petrophysical logs, which has been with data banks including single field or, ultimately, double field data. In the real situation, when a limited number of the well have a shear wave slowness log, considering the heterogeneity of the carbonate formations at the parts of a field, the challenge of not covering the range of training algorithms to estimate shear wave slowness at other wells is created. Therefore, the basic question can be proposed in this way; how to overcome the limitation of training machine learning algorithms with limited data from each field? In this study, as an innovative approach, a multi-field data bank has been used to develop a high accurate model with a wide range of petrophysical parameters to estimate shear wave slowness. Also, in machine learning models, for the firs time in literature of DTSM estimation, the capability of gated recurrent unit (GRU) model as a deep learning model has been evaluated and compared with wide range of most used regression learning-based (SVR, GPR, BGT), deep neural network-based (ANN, RNN, CNN, and LSTM), and analytical models.

2. Methodology

This study aims to provide an innovative solution to estimate the DTSM log using a data bank made up of petro-physical data of carbonate formations of several oil fields. For this purpose, as shown in Figure 1, below steps are conducted for doing this study:

- **Step 1**: data collecting including 23069 data points from 8 drilled well in 4 Iran's southwest oil fields including South Azadegan (2 wells), Ahvaz (3 wells), Marun (2 wells), and Abteimour (1 well) oil fields.
- **Step 2**: Performing data preprocessing including data cleaning (rang check, missing value detection, noise reduction, and outlier elimination) and data preparation (normalization, partitioning, and feature selection).
- **Step 3**: Developing DTSM estimator models using machine learning algorithms based on the modeling data (training and testing).
- **Step 4**: Validation of analytical and developed machine learning models of DTSM estimation at the unseen data (validation data).

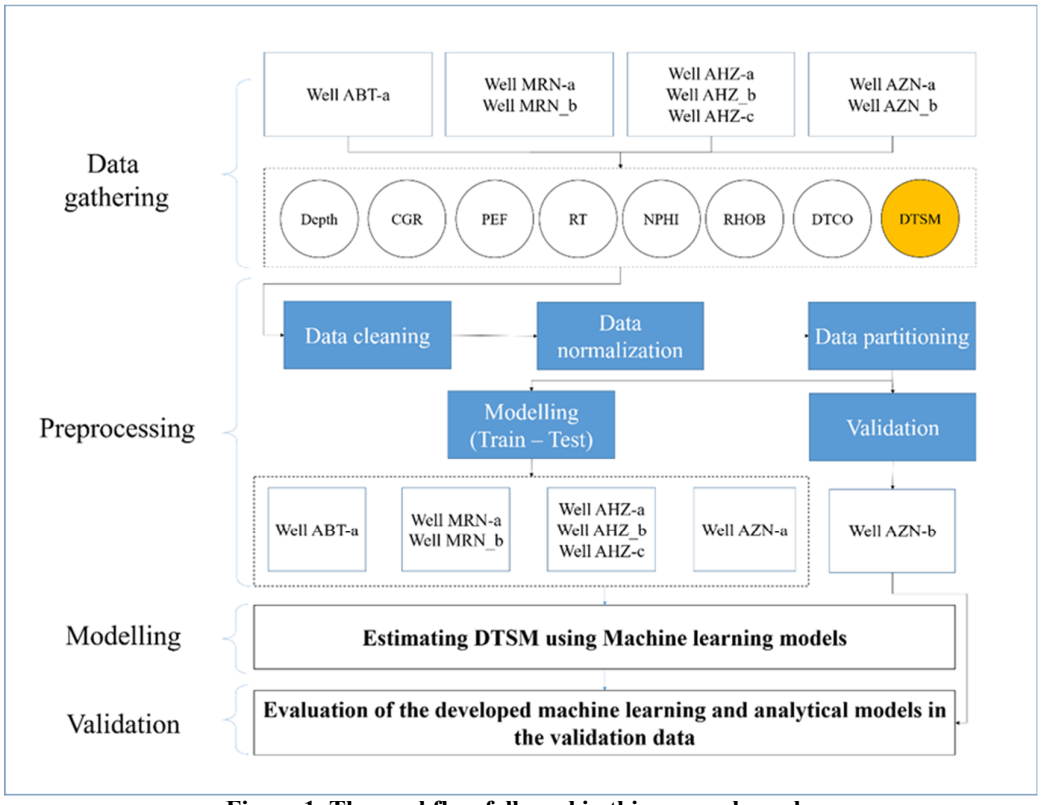


Figure 1. The workflow followed in this research work.

2.1. Data description

In this study, petro-physical information from 8 drilled well in 4 old fields southwest of Iran was collected to cover a broad range of data to train machine learning models in DTSM estimation. Wells were named respectively as AZNa, and AZNb at Azadegan oil field, AHZa, AHZb, and AHZc at Ahvaz oil field, MRNa and MRNb at Marun oil field, ABTa at Abteimur oil field. Data collected includes full set petro-physical logs, including Gamma ray (CGR), Resistivity (RT), Photo electric (PEF), Neutron porosity (NPHI), Density (RHOB), Compressive wave slowness (DTCO), and Shear wave slowness (DTSM). In Figure 2, the profile of petro-physical logs changes is indicated in terms of depth at Well AZNb studied zone. The profile of petro-physical logs, as well as statistical indicators extracted from total information of all studied wells, are provided in **Appendix A**.

2.2. Data pre-processing

The pre-processing operation, including data clearance and preparation, is one of the main stages in the studies and modeling based on machine learning. In this study, the pre-processing data stage has been conducted to decrease the effect of adverse factors on the final results of estimator models, as well as prepare a data bank with the required features to develop intelligent models. This process is further explained.

2.2.1. Data cleaning

In this work, we used petro-physical logs. The main problems of these data include wrong data, missing values, and noise and outlier existence. Meanwhile, the existence of at least 5% noise in real data is an inevitable problem [27,28]. Data having noise leads to inappropriate function of the machine learning due to extraction of wrong rules from the data, and therefore it causes problems in the generalizability of these models to predict new data [29,30]. In addition to noise, outlier data often generates a certain situation in the data that can considerably affect the results of intelligent models. Outliers are data whose amounts are not justifiable compared to other data. An appropriate strategy for these kinds of data can be their removal or replacement, depending on the level of sensitivity and complexity of the problem. However, generally, the management of these data after their detection has a high priority [31]. Accordingly, despite mentioned influential factors on the quality of the applied data, clearance is necessary at the first stage.

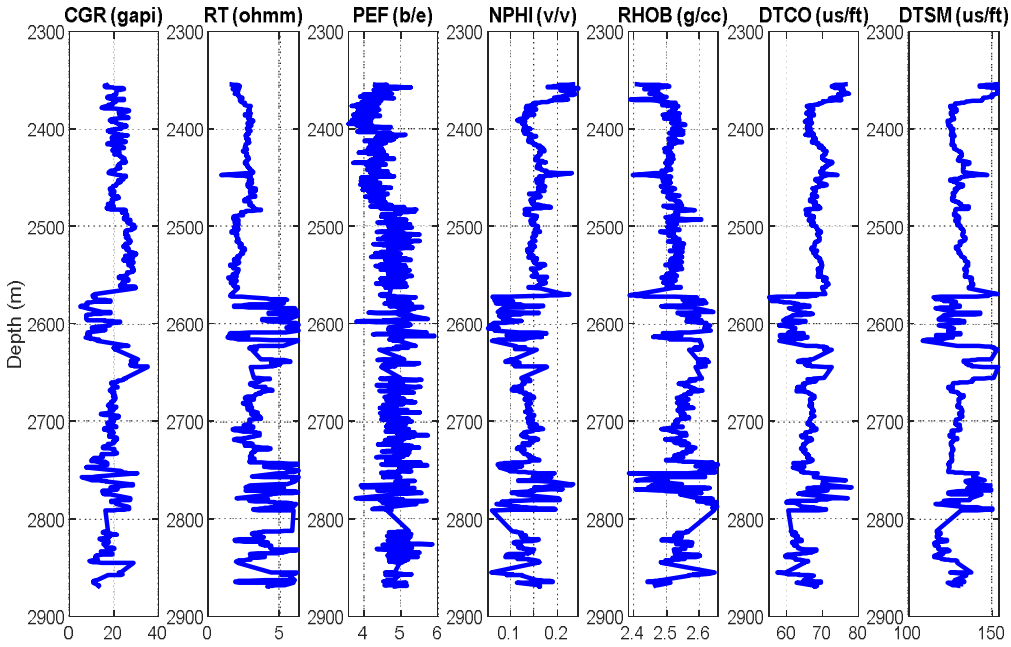


Figure 2. Profile of petrophysical logs in studied interval Well AZN-b.

2.2.2. Data preparation

One of the basic requirements for modeling stages with machine learning algorithms after data clearance is to create an appropriate database. For this purpose, normalization, partitioning, and superior feature selection operations were conducted in this stage on petro-physical data cleaned during the data preparation process. Further, each stage is explained in detail.

2.2.2.1. Data normalization

Data normalization is one of the actions to prepare data, which is very important for the machine learning step. In this process, all parameters are mapped using **Equation 1** at the range of [-1, 1] to eliminate the data scaling effect. In this way, the manner of changes and binary and multilateral associations of the parameters becomes possible.

$$X_n{}^i = 2\left(\frac{X_i - X_{min}}{X_{max} - X_{min}}\right) - 1 \qquad (1)$$

In this equation, $X_n{}^i$ is the normalized amount of ith, X_i is the amount of ith parameter, X_{min} and X_{max} are maximum and minimum amounts of parameter X at the whole data, respectively.

2.2.2.2. Data partitioning

The main scenario of this study is to estimate the DTSM log in a target field using measured information at the same field in addition to information obtained from a given log at the surrounding fields with identical formations. Therefore, as is indicated in the flowchart of Figure 1, the data partitioning strategy after normalization of the whole data is data partitioning in two modeling sections (for training and testing the models) and verification. Meanwhile, to avoid the effect of random data selection on the results of the training and testing of intelligent models, the k-fold cross-validation method was used in all models.

2.2.2.3. Feature selection

Feature selection or decreasing problem dimensions is one of the stages that usually is applied in developing intelligent models based on machine learning. Although there is not any restriction for the number of inputs in machine learning methods, it is always specified in several studies that the accuracy of the model is not a function of the number of the inputs, and the errors of estimator models do not decrease necessarily with an increase in the number of inputs. Rather for an arrangement of parameters with the most influence, the lowest model error is obtained, and with the addition of other parameters, no significant change has occurred in the accuracy of the model. Generally, two groups of methods, including Wrapper and Filter, are used to select and identify these parameters [3,18,32-36]. In most of these studies, it is proposed that the group of

Wrapper methods is used for the recognition of the superior features. Moreover, we should keep in mind that the role of co-linearity in selecting influential factors is very strong. Therefore, investigating the co-linearity of data using cross-plot graphs and the amount of correlation coefficient through the NSGA-II method, which is the Wrapper method, has been used to select superior features in developing DTSM estimator models.

2.3. Estimating DTSM using machine learning models

In order to development of high accurate DTSM model, different machine learning models have been used including regression learning-based models (Bootstrap aggregating (BGT), Support Vector Regression (SVR), and Gaussian Processing Regression (GPR)) and deep neural network-based models (Artificial neural network (ANN), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), Long Short Term Memory (LSTM), and Gradient Recurrent Unit (GRU).

2.3.1. Regression learning-based models

2.3.1.1. Bootstrap aggregating

Bootstrap aggregating or bagging tree (BGT) is one of machine learning model based on aggregating learning ensemble models in which aggregative intelligence is used, and its results are better than the best results of a given model because it works based on the aggregation of the comments and results of the models. In these algorithms, there are two strategies to use aggregated intelligence. In the first strategy, some models are trained for a data set, and results obtained from different predictors will be voted on. In the second strategy, some identical predictors with different data sets are trained with sampling using bootstrap, and its results are usually aggregated [36]. In Figure 3, modeling with bagging is provided. As it can be observed, in this method, unlike boosting, the accuracy of the second predictor is not dependent at all on the accuracy of the first predictor; likewise, up to the end and they are completely independent of each other. An important point, in this case, is the selection of the number of decision trees in each predictor and leaf, which needs optimization. In this study, this kind of bagging is used.

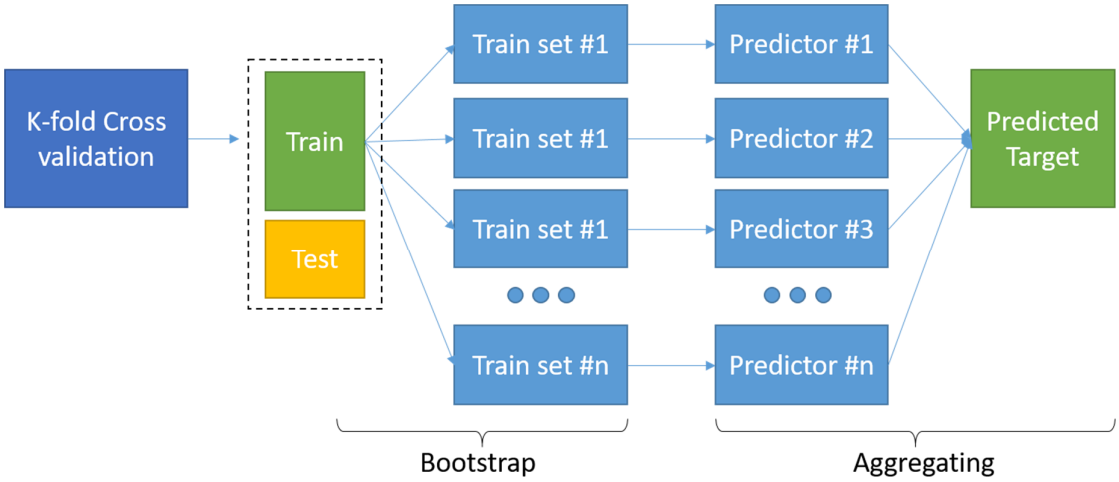


Figure 3. The structure of bootstrap aggregating (Bagging tree) model.

2.3.1.2. Support vector regression

Support Vector Regression (SVR) is one of the supervised methods of machine learning which is used to solve regression problems [37]. This method has good performance for the management of high-dimensional data to decrease the model complexity and the prediction error. Then this algorithm has been applied broadly as a tool to understand the effects of various influential parameters on the target parameter. This method solves the problem using different Kernel functions, which include the linear, RBF, Gaussian, and Polynomial models. The theory of this method is explained in detail by Awad and Kanna (2015) and Zhang and O'Donnell (2020).

2.3.1.3. Gaussian processing regression

Gaussian Processing Regression (GPR) is one of the most powerful machine learning algorithms, which, unlike many common machine learning

models, rely on a few parameters for prediction. As GPR is non-parametric, it can be efficiently applied to solve a broad range of supervised learning problems even when little data is available. Gaussian processing regression can be considered as kernelized Bayesian linear regression, in which, kernel parameterization is determined by selecting kernel function as well as data used for the prediction. Different kernel parameters are used in this model, which includes Squared Exponential, Radial Basis Function, Rational Quadratic, and Matern kernel function. Kuss (2006) provided an exact description of these algorithms.

2.3.2. Deep neural network-based models

2.3.2.1. Artificial Neural Network

Artificial neural network is another widely used supervised machine learning method that has been significantly used in fields related to the oil, gas and geothermal wells drilling, especially in the estimating the rate of penetration (ROP) [18,36]. In general, the types of artificial neural networks can be divided in Radial Basis Function (RBF), and Multi-Layer Perceptron (MLP). The structure of a MLP neural network includes an input layer, a number of hidden layers, and an output layer [41,42]. The objective function in this model is the model error, which is minimized using a learning algorithm during the feed forward back propagation (FFBP) process. According to the complexity of the problem, the number of hidden layers and the number of neurons in each layer increases. Accordingly, the MLP networks with more than one hidden layer can be called a deep MLP network. However, achieving the optimal number of layers and the number of neurons in each layer requires testing different models by trial and error on the problem dataset [42].

2.3.2.2. Convolutional neural network

The Convolutional Neural Network (CNN) model is one of the developed structures of the deep learning network [43]. Recently, this model has been used successfully in different studies on the topic of estimating shear wave velocity and estimation of drilling rate of penetration [3,22,33]. CNN model is considered as a deep feed-forward network structure, which has strong ability compared to the interconnected layer networks. Due to the good performance of CNN, it is used broadly in the issues such as image classification, object detection, velocity detection, sonic detection, vehicle detection, facial expressions detection, and other issues [44]. The general structure of a CNN is made up of an input layer, several parallel filters, a pooling layer, and a dense layer. Indolia et al. (2018) provided a complete explanation of how this model works. In a CNN, in addition to the adjustment of the weights and biases that are performed automatically and during the feed-forward back propagation process, parameters related to the network should be adjusted too. Therefore, one of the disadvantages of this method is the existence of a large number of adjustable parameters.

2.3.2.3. Recurrent Neural Network model

The Recurrent Neural Network (RNN) is a special type of artificial neural network that allows the continuation of information related to past knowledge using a special type of loop architecture. These types of networks are used in many fields for data with sequences, such as predicting the next word of a sentence [45]. In a RNN, unlike the traditional feed forward neural network, there are feedback connections that allow the RNN to model the effects of previous parts of the sequence on the next part of the sequence, which is a very important feature in modeling sequences [46].

2.3.2.4 Long Short Term Memory Model

The concept of Long short-Term Memory (LSTM) was introduced in 1997 [47]. LSTM is basically a type of RNN architecture that is commonly used in various applications and products such as speech recognition systems. A typical LSTM network has something called a memory cell. The memory cell can retain some information about the sequence, which allows it to detect important features at the beginning of the sequence that may affect later parts of the sequence, rather than calculating the output based only on the previous time step. The main components of LSTM are its gates. There are three gates in LSTM including the input gate, forgetting gate and output gate. The input gate controls the entry of new information into the cell. The forgetting gate controls the content of the memory, that is, the forgetting gate decides whether we want to forget some information in order to store new information. The output gate controls the time of using the information in the output of the cell [45].

2.3.2.5. Gated recurrent unit model

Gated recurrent Units (GRUs) are a gating mechanism in recurrent neural networks, introduced in 2014 [48]. GRU model is a simplified

and newer version of RNN and LSTM, which offers an improvement over the other two. Just like LSTM, GRU uses gates to control the flow of information. A GRU model composed of two gates and one candidate-state network, namely: reset gate, update gate, and candidate-state [49].These gates relatively new, as compared to LSTM. This is the reason they offer some improvement over LSTM and have simpler architecture. The update gate used by the GRU is equivalent to the forget and input gates in the LSTM model combined as a single network. It is used to determine what information to remove or add. The reset gate is used to determine how much information from the previous state to forget. In contrast to the LSTM, there is no cell state in the GRU network. In other words, the cell state can be seen as the previous hidden state. The network parameters of the GRU are less than those in LSTM and hence the network requires less training time to learn about dependencies among the time-step observations or sequence data [49].

2.4. Verification of the developed models

At the final stage, the DTSM is estimated in the validation well to assess the generalizability of the developed intelligent models based on a multi-field data bank. The accuracy of the trained model based on the extensive range of the data in DTSM log estimation in an unobserved well is an indication of the high generalizability of this model. Meanwhile, in this stage, the results of applying several empirical models of DTSM estimation have been also compared with the results of the machine learning models.

3. Results and Discussion

3.1. Data preprocessing analysis

3.1.1. Data cleaning

Due to the clearance of the petro-physical data, firstly, depth ranges having wrong amounts and missing values (in every parameter) are detected and deleted from the data. Then the operation of removing noise from the data using a one-dimensional median filtering smoothing algorithm was conducted. The one-dimensional median filtering method is a well-known method in processing the image against "Random," "Gaussian," and "Salt and Pepper" noises. This method is provided by Gonzalez and Wood (2008) in detail. Recently this method has been applied successfully in studies about artificial intelligence and using petro-physical and drilling logs [18,32,33]. The comparison of row and noise removed petro-physical logs at the studied zone of Well AZN-b has been provided in Figure 4 (the plot of other wells has been provided in Appendix B). As can be seen in this figure, using a one-dimensional median filter well preserved the general pattern and trend of information. Accordingly, the PEF log has been most influenced by noise. It is while RHOB log has not been much affected by the noise. It is worth mentioning that DTSM, DTCO, and NPHI logs have sudden picks, which are resulted from the outlier. Detecting and management of the outliers after applying a smoothing filter is very important. Different methods have been provided by researchers for this purpose. Tukey's method is one of the commonly applied methods for detecting outliers from petro-physical logs [18,36]. Therefore, at the last stage of this phase, outlier data were detected and removed using Turkey's method, and gap filling with nearest value method were used. The DTSM log range as a sample log after applying one-dimensional filtering for the different studied well zones has been provided in Figure 5. As it can be seen in Figure 5a, the DTSM log at well AZN-a, well MRN-b, and AZN-b have outliers in some zones. Sudden picks in Figure 4 indicated this fact too. Therefore, at this stage, detected outliers were removed, and its results can be observed in Figure 5b. Final range of cleaned logs is provided separately for each studied well in **Appendix B**.

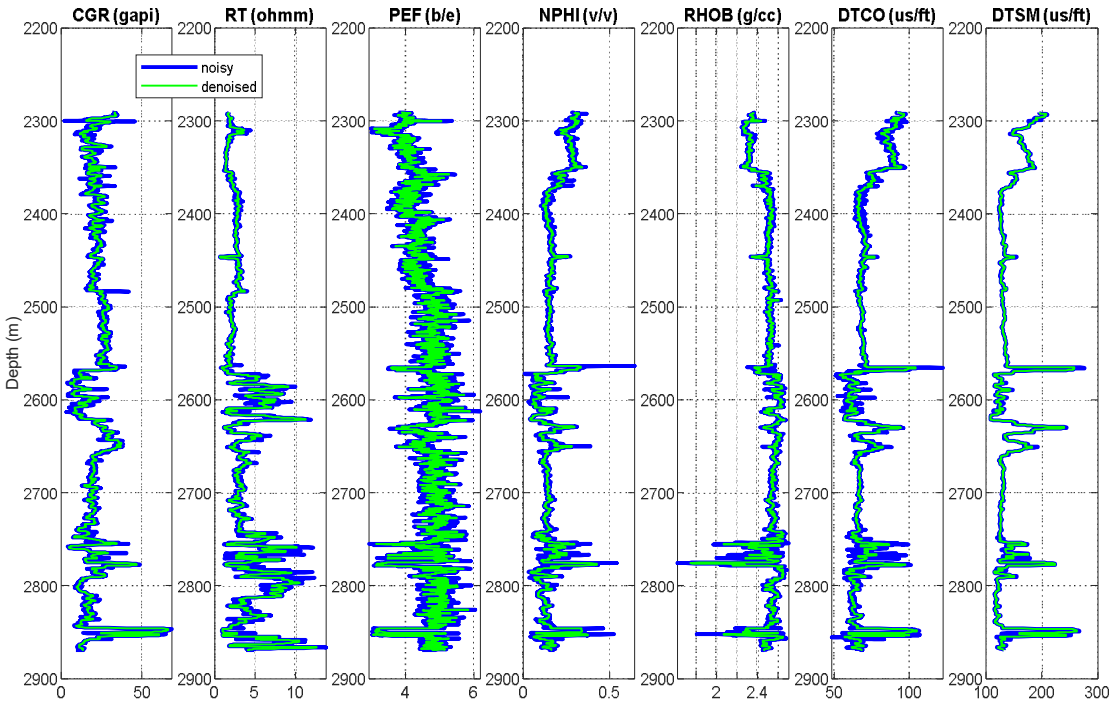
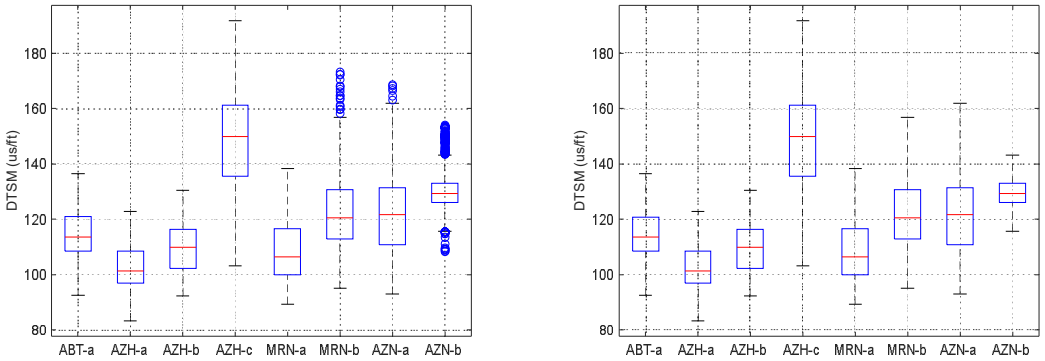


Figure 4. The comparison of row and de-noised petrophysical logs in studied interval of well AZN-b.



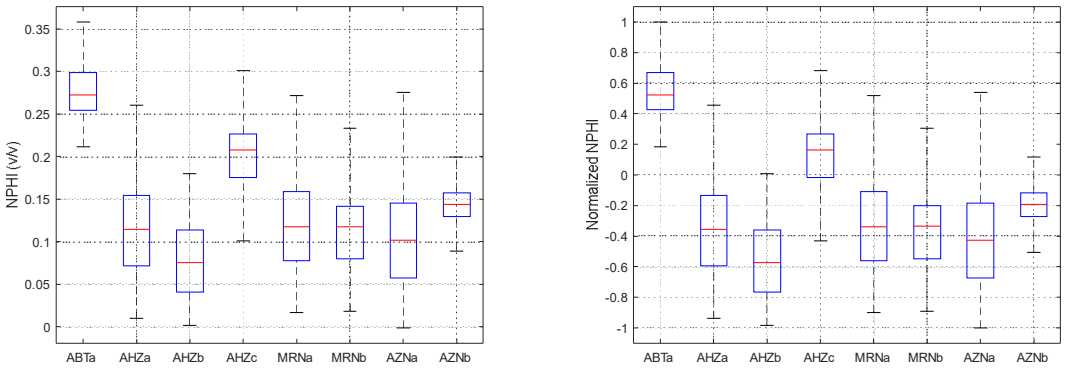
(a) **(b)**

Figure 5. DTSM log quality in studied wells; a) after denoising; b) after outlier elimination and gap filling.

3.1.2. Data preparation

3.1.2.1. Data normalization

All information after clearance has been normalized using Eq. 1. At this stage, as can be seen in Figure 6, the range of each log based on its maximum and minimum value at [-1,1] is mapped. For example, the real range for the NPHI log after clearance shown at different wells has been shown in Figure 6a. The range of this log with normalization mapping between -1 and 1 has been indicated in Figure 6b. Accordingly, the highest amount of NPHI log is at the range of [0.2118, 0.3583] (see Figure 6a), which has been mapped at the range of [0.186, 1] after normalization (see Figure 6b) as the maximum data value in this log is at this range.



(a) **(b)**

Figure 6. The box plots of DTSM in the studied wells; a) true scale, b) normalized scale.

3.1.2.2. Data partitioning

The main idea of this study for estimating shear wave slowness log is formed at the wells of Azadegan field. It was while at the given wells DTSM log data were mapped only in two wells of AZN-a and AZN-b. Given to broad range of amount changes of petro-physical parameters at different sites of the field and the limited range of petro-physical logs data of the wells in which DTSM log has been collected, the great challenge is generated to use machine learning models from the data training viewpoint. As a solution for increasing the training range for machine learning models, a databank composed of information related to several wells having DTSM logs from the fields surrounding the Azadegan field has been created. The considered strategy for partitioning data to modeling and validation data is in such a way that the information of the wells of the surrounding fields, along with the information of the AZN-a well, was considered as modeling data (see Figure 1). The range of petrophysical logs in Azadegan field and Modelling dataset is presented in Table 1. Accordingly, the data range of AZN-a and AZN-b wells in all logs is more restricted than modeling wells. Therefore, if the accuracy of the model is high on the total data of training, testing and validation, this model can be used to estimate the DTSM log in other wells of the Azadegan field within the trained range.

Table 1. The range of petrophysical logs in Azadegan field and modelling dataset.

Dataset		Depth	CGR	RT	PEF	NPHI	RHOB	DTCO	DTSM
Azadegan field	Min	2350	4.86	0.069	3.64	0.001	2.22	50.72	92.96
	Max	3545	34.21	5.86	5.74	0.27	2.76	86.63	161.92
Modelling data	Min	2350	1.93	0.069	1.81	0.001	2.14	46.56	81.18
	Max	4046	94.73	157.63	5.74	0.36	2.88	99.99	191.81

On the other hand, in modeling data partitioning to training and testing for the development of each model, it should be noted that the effect of random selection and partitioning of data should be removed. For this purpose, the accuracy of each model should be assessed with different training and test dataset. For this purpose k-fold, cross-validation method with k=5 has been used in this study at the stage of intelligent model development. With this method, a dataset is divided into five partitions, and in 5 iterations, one partition (20%) is introduced as test data and four other partitions (80%) as training data. Therefore, each intelligent model based on Figure 7 is developed with five different distributions of the training and test data, and the accuracy of each model has been stated based on five steps average.

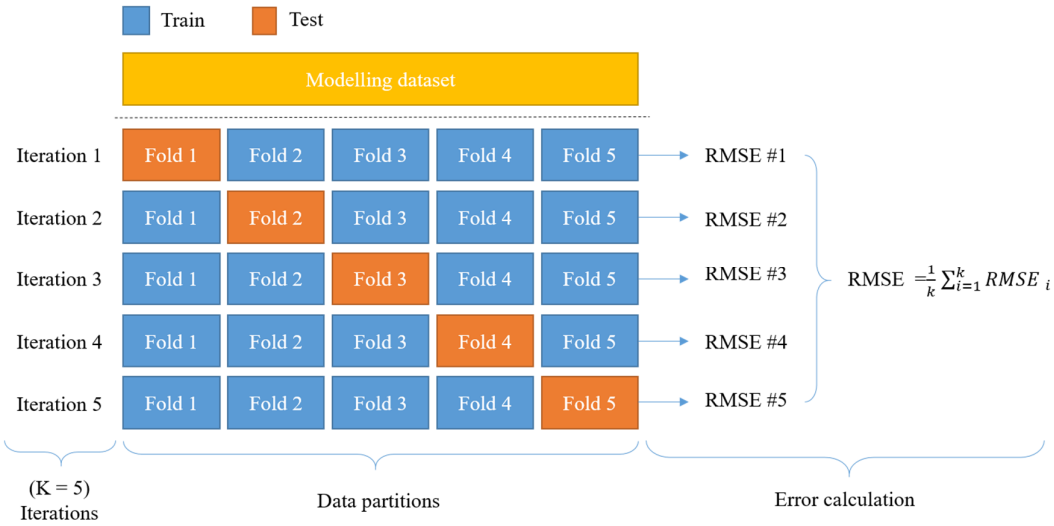


Figure 7. k-fold (k = 5) cross-validation approach implemented in this research work.

3.1.2.3. Feature selection

In the stage of feature selection, at first, co-linearity of data is assessed. The existence of co-linearity among data causes using data that actually do not have any significant role in increasing model accuracy, and their association with the target parameter is equivalent to the association of one or two other parameters with the target. In Figure 8, cross-charts of different petro-physical logs of the modeling dataset have been provided. Accordingly, there is no strong linear association between petro-physical logs in which way that one can be calculated based on the others with a linear relationship. Therefore, all petro-physical parameters to select superior features are entered as the input of the NSGA-II algorithm. Multi-purpose optimization algorithm Non-dominated Sorting Genetic Algorithm II (NSGA-II) integrated with a multi-layer perceptron neural network has been used as a wrapper feature selection method [13]. A complete explanation of the performance of this method has been provided in several studies [3,13,36,52,53]. For this purpose, the amount of 100, 0.74, 0.05, and 50 have been used respectively for population, cross-over, number of iterations, and mutation in this study. Meanwhile, given the high volume of computations and the necessity to study the decreasing trend of the model's error, a three-layered neural network with neuronal arrangement [5, 3, 3] has been used respectively in the first, second, and third layers.

The results of RMSE error decrease and R square increase with different parameter selection in Figure 9, and its numerical values, as well as applied parameters in combination, have been provided in Table 2. Accordingly, with an increase in the number of the input parameters to 5, RMSE decreasing trend and R square increase is notable. Nevertheless, with an increase in parameters to more than 5, the accuracy of the model does not increase significantly. Therefore, PEF, RT, Depth, DTCO, and NPHI parameters were applied to develop intelligent models of DTSM estimation. In Figure 10, the correlation coefficient (R) between selected parameters and DTSM parameters has been indicated. Accordingly, among selected parameters, the highest correlation coefficient is between DTCO-DTSM with the amount of 0.93, and the lowest correlation coefficient is between PEF-DTSM with the amount of 0.33.

Table 2. RMSE and R square values for identifying the optimal combination of features to include in modeling the DTSM (the best finding is in bold).

Number of Selected Parameters	Selected parameters	RMSE	R^2
1	Depth	3.89	0.28
2	Depth – DTCO	3.45	0.45
3	Depth – NPHI - DTCO	3.1	0.58
4	Depth – PEF – NPHI – DTCO	2.73	0.67
5	**Depth – RT-PEF – NPHI – DTCO**	**2.51**	**0.73**
6	Depth – CGR – RT –PEF- NPHI - DTCO	2.41	0.75
7	Depth – CGR – RT –PEF- NPHI –RHOB- DTCO	2.39	0.77

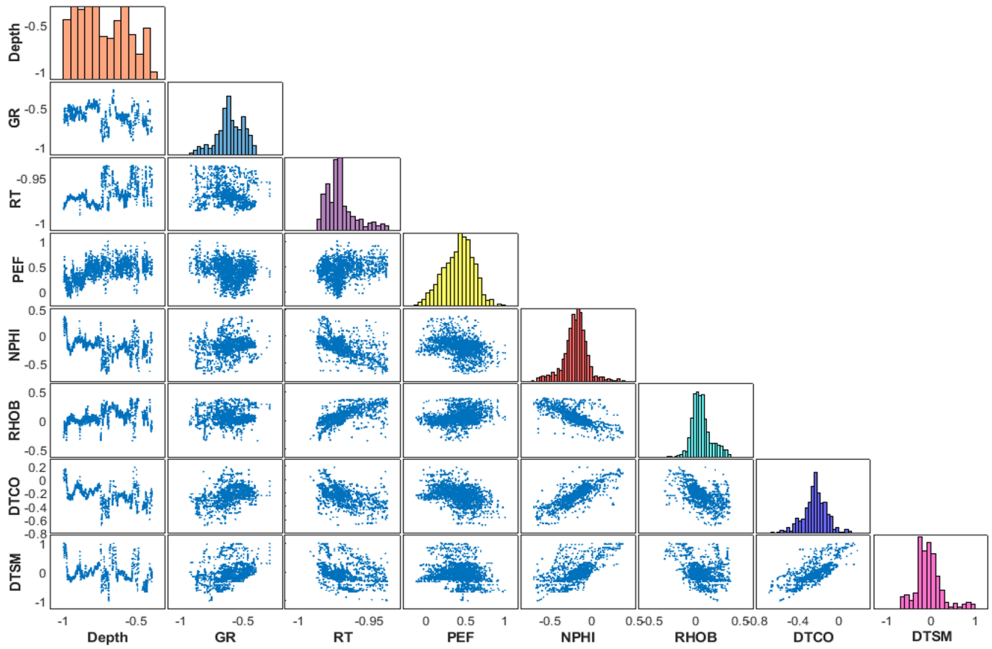


Figure 8. The cross plots of petrophysical logs data from all modelling wells for co-linearity analysis (data are in normalized form between [-1,1]).

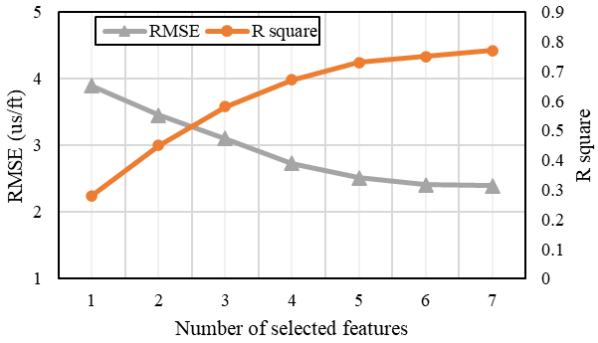


Figure 9. The RMSE and R square of DTSM estimation model during feature selection using NSGA-II coupled with ANN-MLP.

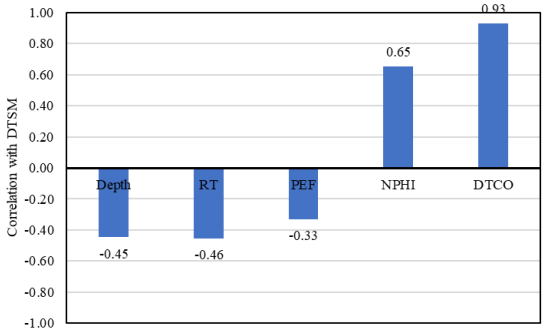


Figure 10. The correlation coefficient between selected features and DTSM.

3.2. DTSM estimation results

3.2.1. Regression learning-based models

3.2.1.1. Performance of BGT model

In order to develop the Bagging tree model, accessing optimized amounts of hyper parameters of this model, including the number of sampling sets (Bootstrap), which is known as "leaf size," as well as the number of the trees in each predictor, needs to solve an optimization problem. For this purpose, during a sensitivity analysis process and studying different leaf sizes and trees, it was specified that using five training datasets and five predictors with 20 trees have been taken the best conclusion. Figure 11 indicates the results of this comparison. Accordingly, with an increase in the number of leaf sizes, the accuracy of the model did not always increase the least error belonged to leaf size of five. It is while increasing decision trees up to 20 at each predictor always leads to a decrease in model error, and whatever the number of trees passes from 20, the decreasing trend of the errors remains constant. Then according to the studied dataset in this study, the best model structure for the Bagging tree is to have a leaf size of five and 20 decision trees in each predictor. Meanwhile, the results of the Bagging tree estimation model with the optimized structure on training and test data are provided in Figure 12. Accordingly, the performance of the BGT model on the test data has been close to training data, which is demonstrated the relatively good generalizability.

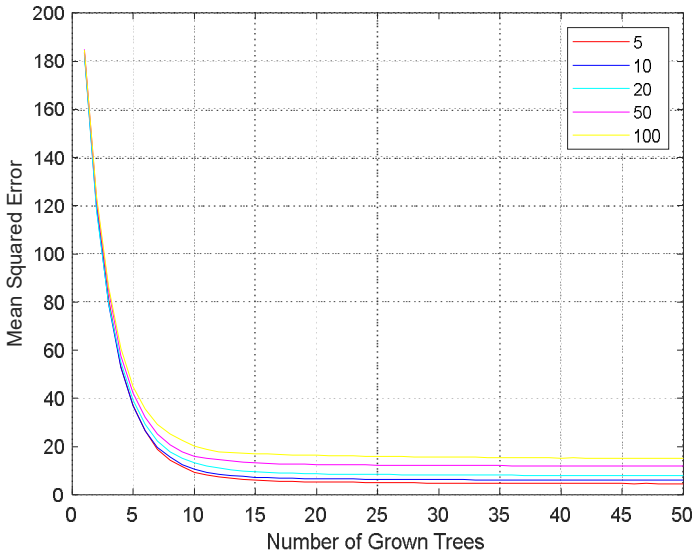
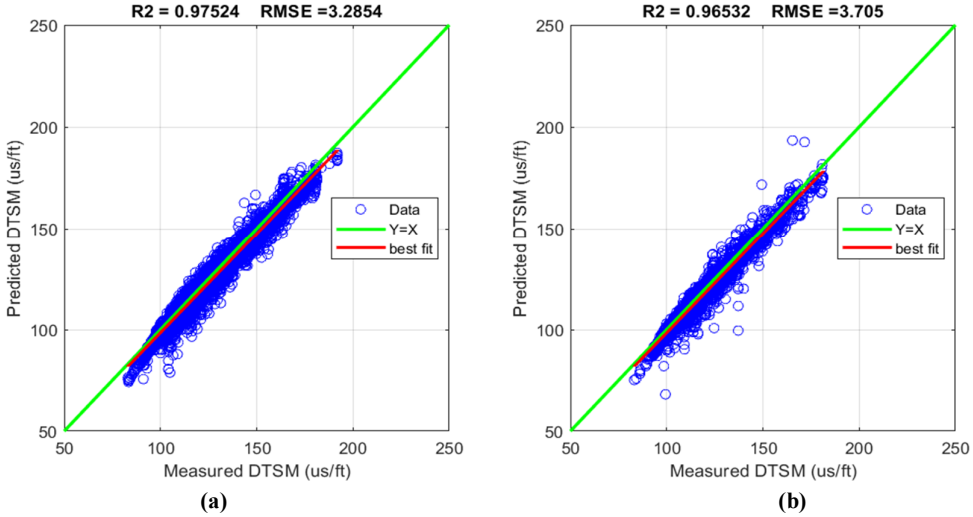


Figure 11. Mean square error with different leaf size (5, 10, 20, 50 and 100) and number of tree (1 to 50) during optimization of bagging tree structure.



(a) **(b)**

Figure 12. The cross-plots of bagging tree model results; a) training data, b) testing data.

3.2.1.2. Performance of the SVR model

In this study, different kernel functions including Gaussian, polynomial, RBF, and linear have been used for DTSM estimation with Support Vector Regression (SVR). The results of the model's accuracy on the training and test data and by considering 5-fold cross-validation for different kernel functions have been provided in Table 3. Accordingly, the best result has been obtained among the models with different kernel functions for the model with Gaussian kernel functions. The results of the DTSM estimation using the SVR model with Gaussian kernel functions on training and test data have been provided in Figure 13.

Table 3. The error evaluation indices of SVR model in estimating DTSM with different kernel functions using modelling dataset (the model with the best result is bolded).

Kernel function	RMSE		R-square		AAPD	
	Train	Test	Train	Test	Train	Test
Linear	4.95	5.23	0.93	0.92	3.5	3.25
RBF	4.68	5.02	0.95	0.91	3.2	3.43
Gaussian	**4.32**	**4.87**	**0.96**	**0.94**	**2.74**	**2.95**
Polynomial	4.54	4.98	0.95	0.92	2.95	3.23

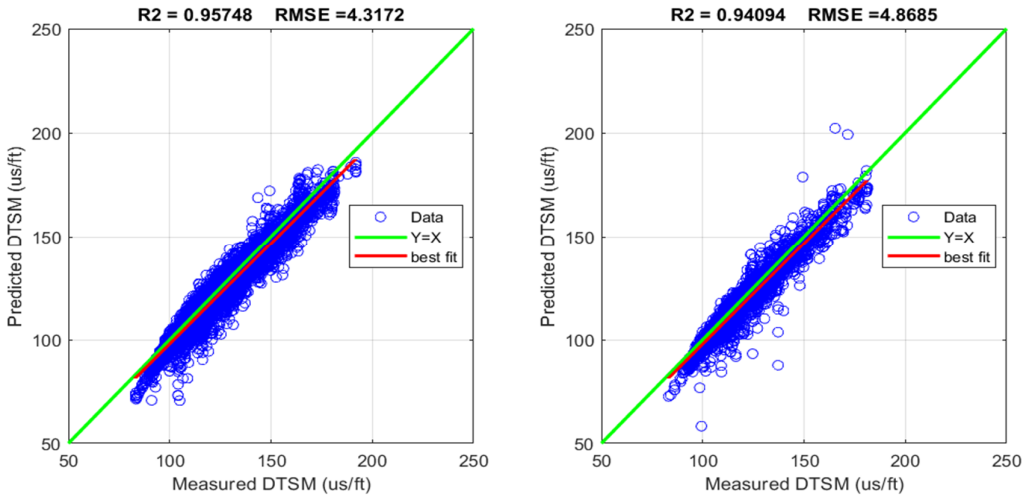


Figure 13. The cross plots of SVR model results; a) training data, b) testing data.

3.2.1.3. Performance of GPR model

In this study, different kernel functions including rational quadratic, squared exponential, RBF, and Matern have been used for DTSM estimation with using Gaussian Processing Regression (GPR) model. The results of the model's accuracy on the training and test data and by considering 5-fold cross-validation for different kernel functions have been provided in Table 4. Accordingly, the best result has been obtained among the models with different kernel functions for the model with Matern kernel functions. The results of the DTSM estimation using the GPR model with Matern kernel functions on training and test data have been provided in Figure 14.

Table 4. The error evaluation indices of GPR model in estimating DTSM with different kernel functions using modelling dataset (the model with the best result is bolded).

Kernel function	RMSE		R-square		AAPD	
	Train	Test	Train	Test	Train	Test
Squared exponential	5.21s	5.13	0.95	0.93	2.86	3.21
RBF	4.89	5.02	0.96	0.93	2.72	3.02
Matern	**4.1**	**4.62**	**0.96**	**0.95**	**2.6**	**2.8**
Rational Quadratic	4.52	4.89	0.95	0.94	2.7	2.95

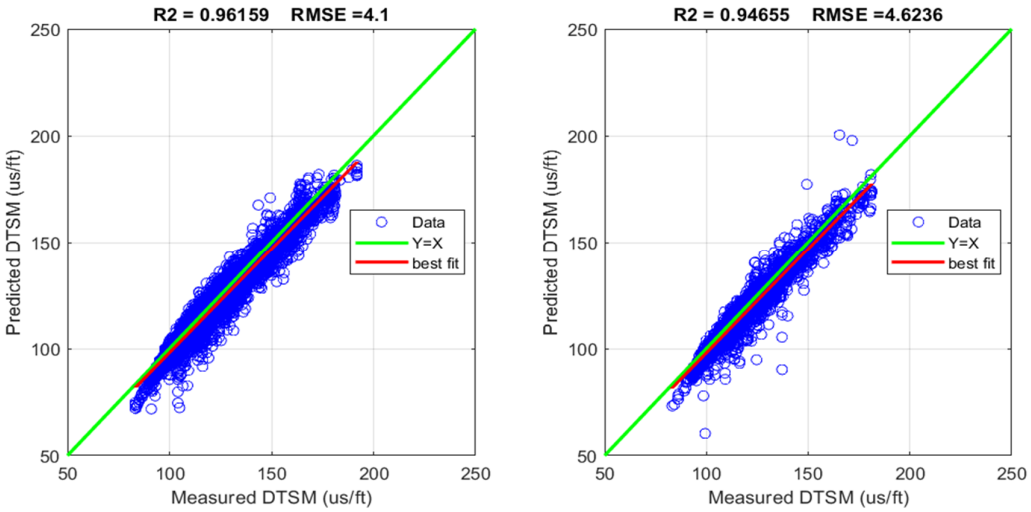
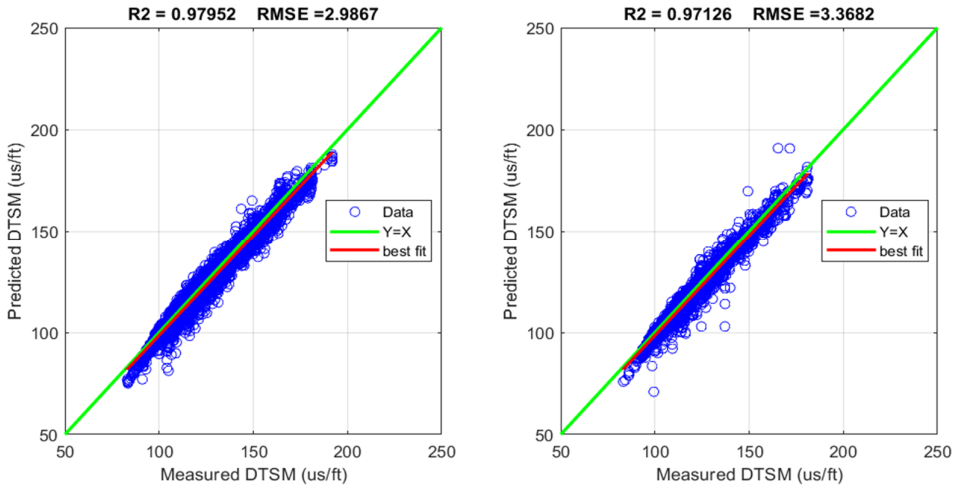


Figure 14. The cross plots of GPR model results; a) training data, b) testing data.

3.2.2. Deep neural network-based models

3.2.2.1. Performance of the ANN model

For developing the ANN model, accessing its optimized structure of multi-layer perceptron network including the number of hidden layer and the number of neurons in each layers is very important. For this purpose, an optimized structure including the three hidden layer network with 14, 12, and 9 neurons in first, second and third layer respectively, obtained during an error and trial process. The results of ANN model estimation with the optimized structure on the training and test data are provided in Figure 15. Accordingly, the ANN-MLP model is well fitted on the training data with values of 2.99 and 0.98 for RMSE and R-square, respectively. Checking the accuracy of the model on the testing data showed that this model performed relatively well on the test data with 3.37 and 0.97 for RMSE and R-square, respectively. In addition, AAPD errors are 1.9 and 2.04 percent for training and test data, respectively. Generally, the performance of the model on the test data has been close to training data, which is indicative of relatively good generalizability.



(a) **(b)**

Figure 15. The cross plots of ANN model results; a) training data, b) testing data.

3.2.2.2. Performance of CNN model

For developing the CNN model, accessing its optimized structure is very important. For this purpose, an optimized structure fits the data structure of this study according to the features provided in Table 5 obtained during an error and trial process. The results of CNN model estimation with the optimized structure on the training and test data are provided in Figure 16. Accordingly, the CNN model is well fitted on the training data with values of 2.72 and 0.98 for RMSE and R-square, respectively. Checking the accuracy of the model on the testing data showed that this model performed relatively well on the test data with 3.06 and 0.98 for RMSE and R-square. In addition, AAPD errors are 1.72 and 1.85 percent for training and test data, respectively. Therefore, the performance of the model on the testing and training data indicate the good generalizability.

Table 5. The structure of CNN employed for DTSM estimation.

Main layer	Number	Type/id	Properties	Value/type
Input layer	**1**	**Sequential**	**-**	**-**
Filter layer	4	Conv1D	Kernel size	3
			Filter number	200
			Padding	same
			Activation	relu
Dropout layer	1	Drop1	-	0.5
Pooling layer	1	Maxpooling1D	-	-
Flatten layer	1	Flatten 1	-	-
Fully connected layer	1	Dense 1	-	100
Fully connected layer	1	Dense 2	-	1

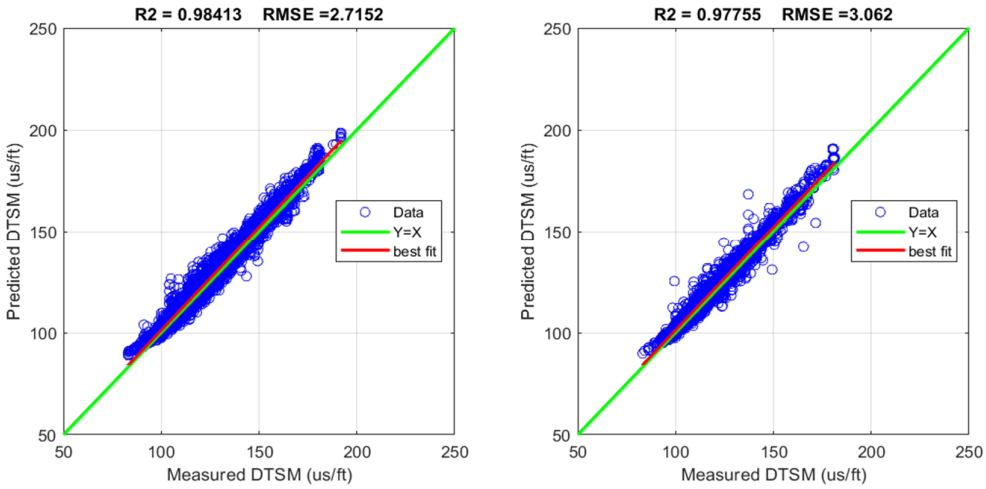


Figure 16. The cross plots of CNN model results; a) training data, b) testing data.

3.2.2.3. Performance of the RNN model

In this research work, a recurrent neural network with the architecture according to Table 6 has been used. The results of DTSM estimation on training and testing data with RNN network are shown in Figure 17. Accordingly, the RNN model is well fitted on the training data with values of 2.96 and 0.98 for RMSE and R-square. Also, the AAPD value for training data was 1.88 %. Checking the accuracy of the model on the test data showed that this model performed relatively well on the testing data with 3.34 and 0.97 for RMSE and R-square and 2.02 % for AAPD. However, the difference between the error values for the testing and training data has shown the relatively high generalizability of this model.

Table 6. The structure of RNN employed for DTSM estimation.

Main layer	Number	Type/id	Properties	Value/type
Input layer	**1**	**Sequential**	**-**	**-**
Lstm layer	1	LSTM	Hidden units number	200
			Optimizer	adam
			Output mode	last
Dropout layer	1	Drop 1	-	0.25
Fully connected layer	1	-	-	-
Regression layer	1	-	-	1

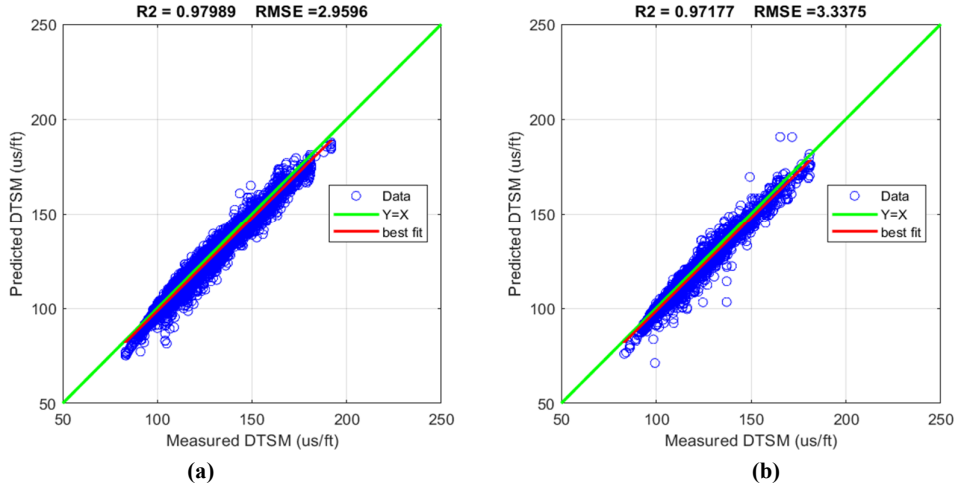


Figure 17. The cross-plots of RNN model results; a) training data, b) testing data.

3.2.2.4. Performance of the LSTM model

In this research work, in addition to simple RNN, an LSTM network with the architecture according to Table 7 has been used. The results of DTSM estimation on training and testing data with LSTM network are shown in Figure 18. Based on this, the LSTM model with values of 2.31 and 0.99 for RMSE and R-square, respectively. Also, the AAPD value for the training data in this model was 1.46 %. This is while checking the accuracy of the model on the testing data showed that this model with 2.6 and 0.98 for RMSE and R-square and 1.58% for AAPD has a good performance compared to RNN, ANN, CNN and other regression learning-based models on the testing data.

Table 7. The structure of LSTM employed for DTSM estimation.

Main layer	Number	Type/id	Properties	Value/type
Input layer	**1**	**Sequential**	**-**	**-**
Lstm layer	1	LSTM	Hidden units number	200
			Optimizer	adam
Dropout layer	1	Drop1	-	0.5
Relu layer	1	Relu 1	-	-
Fully connected layer	1	Dense1	-	100
Regression layer	1	-	-	1

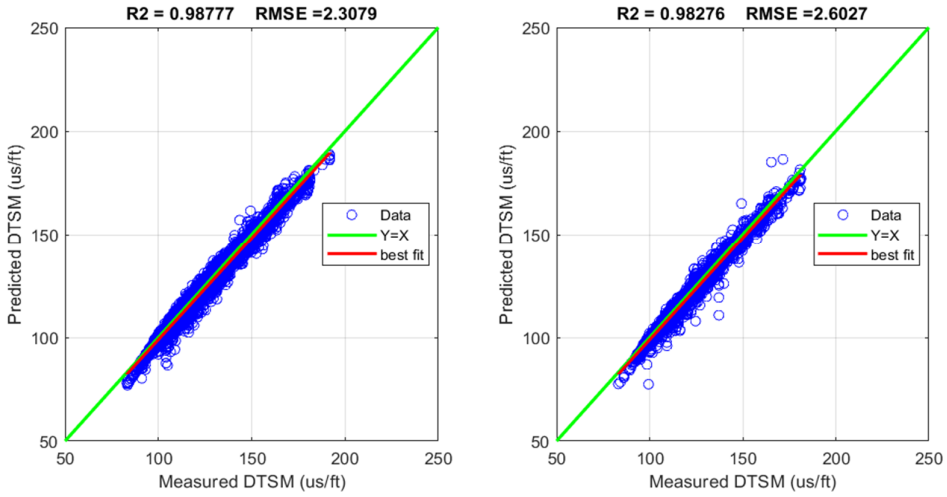


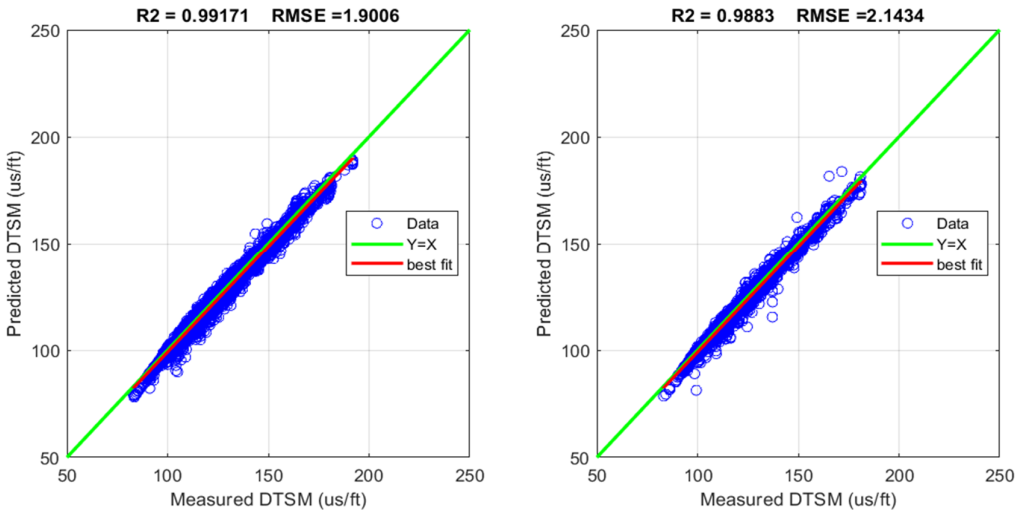
Figure 18. The cross-plots of LSTM model results; a) training data, b) testing data.

3.2.2.5. Performance of the GRU model

In this research work, for the first time in the literature of DTSM or shear wave velocity estimation, a gated recurrent unit with the architecture according to Table 8 has been used. The results of DTSM estimation on training and testing data with GRU network are shown in Figure 19. Accordingly, the GRU model with values of 1.9, 0.99 and 1.21 for RMSE, R-square, and AAPD, respectively. It is indicated good fitting on the training data. However, evaluating the accuracy of the model on the testing data showed that this model with 2.14, 0.99 and 1.3 for RMSE, R-square, and AAPD, respectively has a good performance compared to other machine learning models on the testing data.

Table 8. The structure of GRU employed for DTSM estimation.

Main layer	Number	Type/id	Properties	Value/type
Input layer	**1**	**Sequential**	**-**	**-**
gru layer	1	GRU	Hidden units number	200
			Reset Gate Mode	After multiplication
			State Activation Function	tanh
Fully connected layer	1	Dense1	-	100
Regression layer	1	-	-	1



(a) (b)

Figure 19. The cross plots of GRU model results; a) training data, b) testing data.

3.3. Validation

3.3.1. Analysis of machine learning models

At the last stage of workflow, data of Well AZN-b has been used for verification of the machine learning models in DTSM estimation. Cross plots and profile comparing the measured and predicted DTSM with different regression learning-based models are provided in Figure 20 and Figure 21, respectively**.** As can be seen, BGT model has the best performance among regression learning-based models in validation data. Also, the cross plots and profile comparing the measured and predicted DTSM with different deep neural network-based models are provided in Figure 22 and Figure 23, respectively**.** As can be seen, GRU model has the best performance among all machine learning models in validation data.

The values of RMSE, R-square, and AAPD for all models on modeling (training and testing) and validation dataset are provided in Table 9. Accordingly, CNN, LSTM, and GRU models had results close to DTSM estimation at Well AZN-b. Nevertheless, the GRU model obtained the best answers with the amounts of 2.43, 0.93, and 1.30 for RMSE, R-square, and AAPD, respectively.

Table 9. The performance of machine learning models in DTSM estimation on training, testing, and validation dataset.

	Dataset	SVR	GPR	BGT	ANN	CNN	RNN	LSTM	GRU
RMSE	Training	4.32	4.10	3.29	2.99	2.72	2.96	2.31	1.90
	Testing	4.87	4.62	3.70	3.37	3.06	3.34	2.60	2.14
	Validation	**4.55**	**4.24**	**4.08**	**3.65**	**2.81**	**3.46**	**2.82**	**2.43**
R-square	Training	0.96	0.96	0.98	0.98	0.98	0.98	0.99	0.99
	Testing	0.94	0.95	0.97	0.97	0.98	0.97	0.98	0.99
	Validation	**0.78**	**0.81**	**0.83**	**0.85**	**0.92**	**0.87**	**0.91**	**0.93**
AAPD	Training	2.74	2.60	2.09	1.90	1.72	1.88	1.46	1.21
	Testing	2.95	2.80	2.24	2.04	1.85	2.02	1.58	1.30
	Validation	**2.44**	**2.25**	**2.21**	**1.95**	**1.75**	**1.91**	**1.54**	**1.30**

3.3.2. Analysis of analytical models

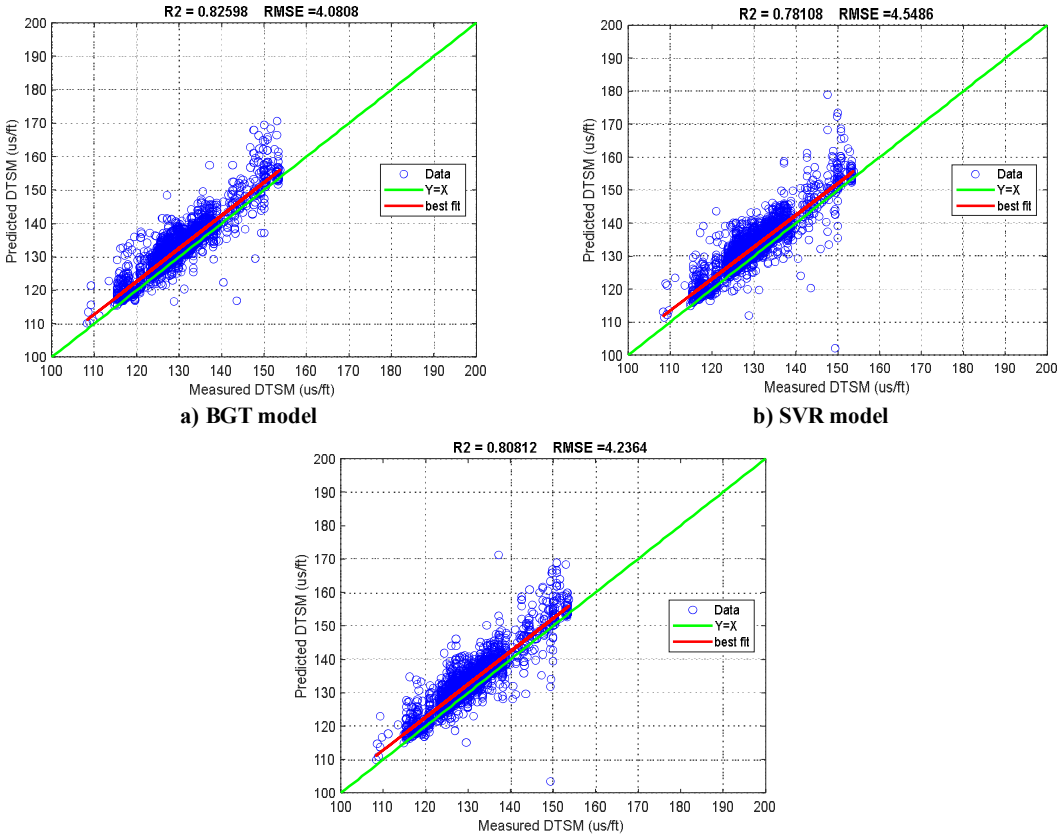
Meanwhile, four models provided by Pickett (1963), Castagna et al. (1993), Brocher (2005), and Anemangely et al. (2019) were applied to compare machine learning models with the results of commonly applied analytical equations of the shear wave (velocity) slowness estimation. Considering **Equation 2** and **Equation 3,** DTSM values can be obtained using equations provided in Table 10.

$$V_s(km/s) = \frac{1}{DTSM} \times 304.8 \quad (2)$$

$$DTSM(\mu s/ft) = \frac{1}{V_s} \times 304.8 \quad (3)$$

Table 10. Analytical relations between V_s and V_p.

Model Equations	Model name	Equation number
$V_s = \frac{V_P}{1.9}$	Pickett	(5)
$V_s = -0.05509\,V_P^2 + 1.0168\,V_P - 2.3057$	Castagna	(6)
$V_s = 0.7858 - 1.2344\,V_P + 0.7949\,V_P^2 - 0.1238\,V_P^3 + 0.0064\,V_P^4$	Brocher	(7)
$V_s = 0.6079 + 0.4207\,V_P$	Anemangely	(8)



a) BGT model **b) SVR model**

c) GPR model

Figure 20. The cross plots of regression learning-based models in DTSM estimation for validation data (well AZN-b).

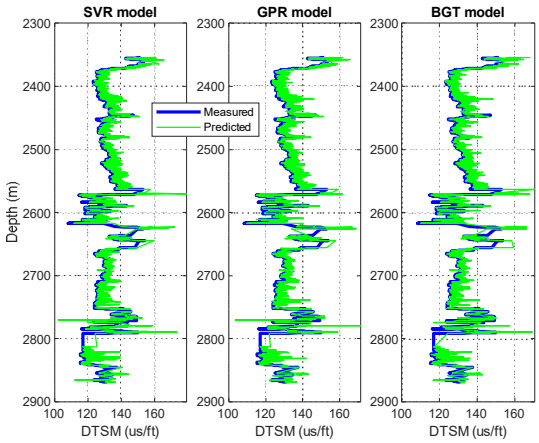


Figure 21. The comparisson between measured and predicted DTSM using regression learning-based models along the Welll AZN-b.

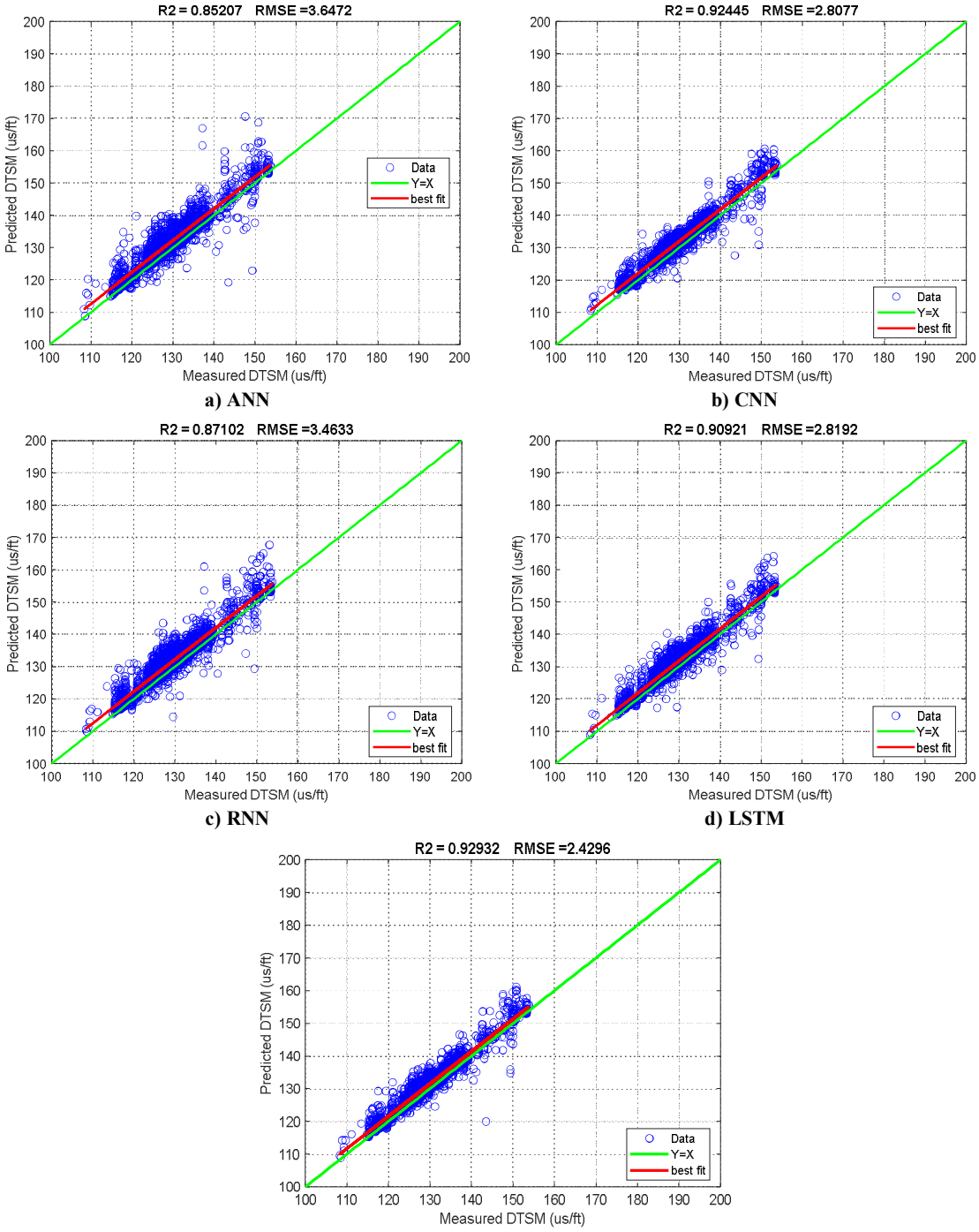


Figure 22. The cross plots of deep neural network-based models in DTSM estimation for validation data (Welll AZN-b).

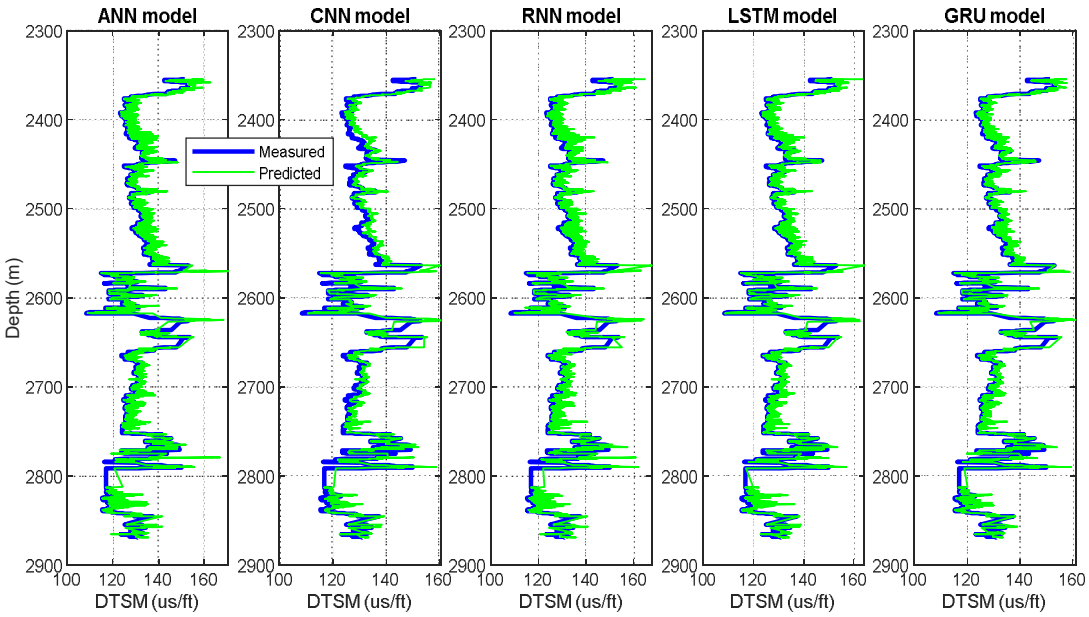
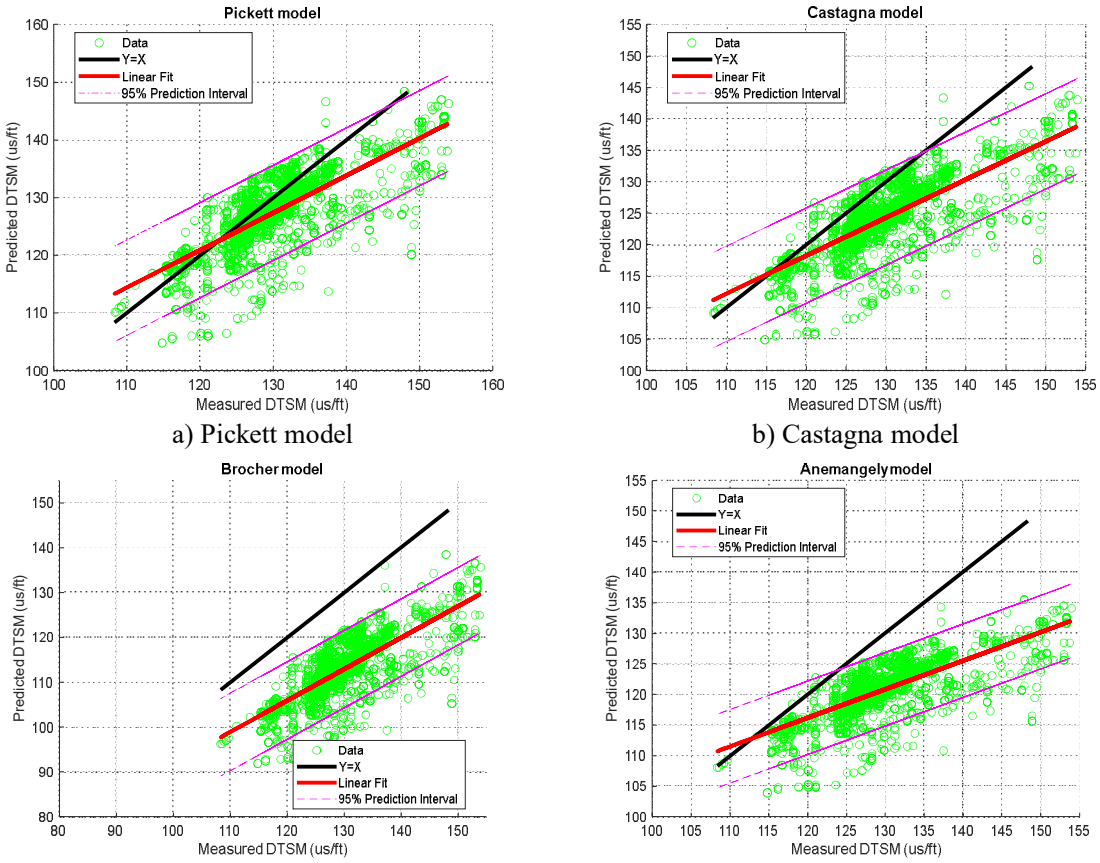


Figure 23. The comparisson between measured and predicted DTSM using Deep neural network-based models along the Welll AZN-b.

The cross-plot graphs comparing predicted values using analytical models and measured DTSM of well AZN-b are provided in Figure 24. As can be seen in this figure, among different models, only the Pickett model could provide relatively good performance. Predicted and measured DTSM of Well AZN-b for different analytical models are provided in Figure 25. Accordingly, it should be pointed out that predicted amounts through all analytical models had weaker performance than machine learning models. It is while that in the Brocher model, significant differences can be seen between the measured and predicted values. The values of RMSE, R square, and AAPD for all models in validation data are provided in Table 11. Accordingly, Pickett and Castagna's models had results close to DTSM estimation at Well AZN-b. Nevertheless, the Pickett model provided the best answer with the values of 5.5, 0.55, and 2.87 for RMSE, R square, and AAPD, respectively. These results indicate clearly to how using these models can increases error and uncertainty in geomechanical models and other applications of shear wave slowness. However, due to the low range of training data, these equations are used in some fields, and the results of this section provide a good idea and view of the possible amount of error in the DTSM estimation with these methods. However, results obtained from machine learning models, which are developed based on a multi-filed data bank, indicated that we could assure in the studies that the DTSM can be estimated in the broader range of the collected data and this estimation is acceptably exact. Therefore, a decrease in uncertainty of models such as geomechanical models would be imaginable using this method.

Table 11. The error evaluation indices of analytical models in estimating DTSM using validation dataset.

Model	RMSE	R square	AAPD
Pickett	5.50	0.55	**2.87**
Castagna	7.37	0.56	**4.40**
Brocher	17.72	0.57	**13.08**
Anemangely	10.33	0.55	**6.90**



a) Pickett model b) Castagna model

c) Brocher model d) Anemangely model

Figure 24. The cross-plots of analytical models in DTSM estimation for validation data (Welll AZN-b).

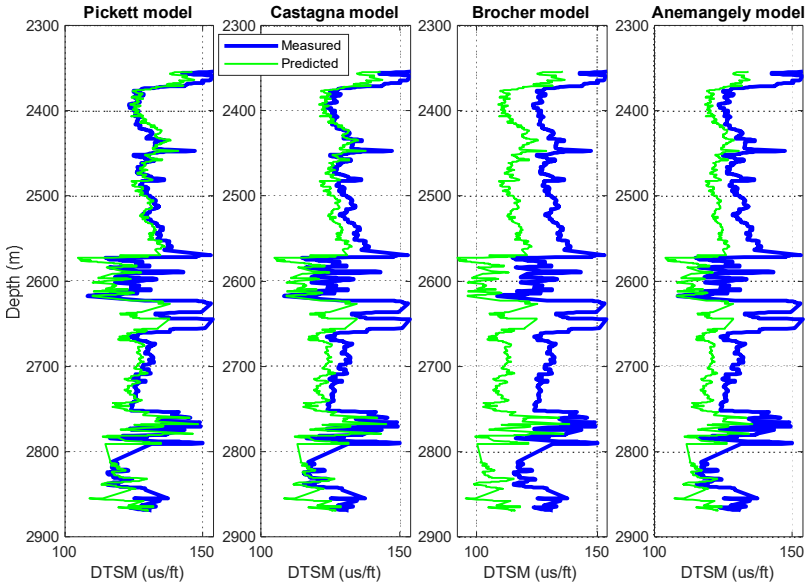


Figure 25. The comparisson between measured and predicted DTSM using analytical models along the Welll AZN-b.

4. Conclusions

In this study, in order to provide a practical solution for mitigating the uncertainties in geomechanical models, a multi-field data bank is used to increase the range of training data for the data-driven method in the shear wave slowness

(DTSM) estimation. For this purpose, the data of 8 drilled wells from 4 oil fields located southwest of Iran was collected. After preprocessing operation, including data clearance and data preparation, the dataset was divided into two sets of modeling and verification data at this stage. After conducting a superior feature selection stage and developing the different machine learning models including regression learning-based (BGT, GPR, SVR), and deep neural network-based (ANN, CNN, RNN, LSTM, and GRU) were developed and the accuracy of each model on the training, testing, and validation data was assessed. Generally, we can conclude the following results from this study:

- Depth, Resistivity (RT), Photoelectric (PEF), Neutron porosity (NPHI), and Compressive wave slowness (DTCO) parameters were identified as the most effective ones in shear wave slowness (DTSM) estimation of the feature selection stage.
- The results of assessing the accuracy of machine learning models on the modeling and verification data indicated that generally, using a multi-field data bank can be an effective idea to develop comprehensive models (with a broad range of training data) for the DTSM log estimation in the wells of the fields of where limited information has collected.
- Studying the accuracy of the machine learning models on the modelling dataset indicated that the GRU model had the most exact answers with the values of 1.9 and 2.14 for RMSE and 0.99 and 0.99 for R-square, for training and testing data, respectively.
- Studying the accuracy of the machine learning and analytical models on the validation dataset indicated that GRU had been the most accurate model with the values of 2.43 and 0.93 for RMSE and R-square, respectively. It is while the best results of commonly applied analytical models have very lower accuracy compared to machine learning models on the validation data with the values of 5.5 and 0.55 for RMSE and R-square, respectively.
- The comparison of the results obtained from the DTSM estimation on validation data using the analytical models and machine learning models indicated that the accuracy and generalizability of the machine learning models are significantly different from other commonly used analytical models. Accordingly, using this method can lead to a decrease in uncertainty of the geo-mechanical models and general models that use shear wave slowness log.

References

[1]. Dalvand, M., & Falahat, R. (2021). A new rock physics model to estimate shear velocity log, *Journal of Petroleum Science and Engineering*, *196*, 107697.

[2]. Anemangely, M., Ramezanzadeh, A., Amiri, H., & Hoseinpour, S. A. (2019). Machine learning technique for the prediction of shear wave velocity using petrophysical logs. *Journal of Petroleum Science and Engineering*, *174*, 306-327.

[3]. Mehrad, M., Ramezanzadeh, A., Bajolvand, M., & Hajsaeedi, M. R., (2022). Estimating shear wave velocity in carbonate reservoirs from petrophysical logs using intelligent algorithms. *Journal of Petroleum Science and Engineering, 212,* 110254.

[4]. Pickett, G.R., (1963). Acoustic Character Logs and Their Applications in Formation Evaluation. *Journal of Petroleum Technology, 15*, 659-667.

[5]. Castagna, J. P., Batzle, M. L., Kan, T. K., & Backus, M. M. (1993). Rock physics—The link between rock properties and AVO response. *Offset-dependent reflectivity—Theory and practice of AVO analysis: SEG*, *8*, 135-171.

[6]. Brocher, T. M. (2005). Empirical relations between elastic wavespeeds and density in the Earth's crust. *Bulletin of the seismological Society of America*, *95*(6), 2081-2092.

[7]. Miah, M. I. (2021). Improved prediction of shear wave velocity for clastic sedimentary rocks using hybrid model with core data. *Journal of Rock Mechanics and Geotechnical Engineering, 13(6)*, 1466-1477.

[8]. Khatibi, S., & Aghajanpour, A. (2020). Machine learning: A useful tool in geomechanical studies, a case study from an offshore gas field. *Energies, 13(14)*, 3528.

[9]. Oloruntobi, O., Onalo, D., Adedigba, S., James, L., Chunduru, R., & Butt, S. (2019). Data-driven shear wave velocity prediction model for siliciclastic rocks. *Journal of Petroleum Science and Engineering*, *183*, 106293.

[10]. Oloruntobi, O., & Butt, S. (2020). The shear-wave velocity prediction for sedimentary rocks. *Journal of Natural Gas Science and Engineering*, *76*, 103084.

[11]. Olayiwola, T., & Sanuade, O. A. (2021). A data-driven approach to predict compressional and shear wave velocities in reservoir rocks. *Petroleum*, *7*(2), 199-208.

[12]. Azadpour, M., Saberi, M. R., Javaherian, A., & Shabani, M. (2020). Rock physics model-based prediction of shear wave velocity utilizing machine learning technique for a carbonate reservoir, southwest Iran. *Journal of Petroleum Science and Engineering*, *195*, 107864.

[13]. Anemangely, M., Ramezanzadeh, A., & Tokhmechi, B. (2017). Shear wave travel time

estimation from petrophysical logs using ANFIS-PSO algorithm: A case study from Ab-Teymour Oilfield. *Journal of Natural Gas Science and Engineering*, *38*, 373-387.

[14]. Maleki, S., Moradzadeh, A., Riabi, R. G., Gholami, R., & Sadeghzadeh, F. (2014). Prediction of shear wave velocity using empirical correlations and artificial intelligence methods. *NRIAG Journal of Astronomy and Geophysics*, *3*(1), 70-81.

[15]. Bagheripour, P., Gholami, A., Asoodeh, M., & Vaezzadeh-Asadi, M. (2015). Support vector regression based determination of shear wave velocity. *Journal of Petroleum Science and Engineering*, *125*, 95-99.

[16]. Sabah, M., Talebkeikhah, M., Wood, D. A., Khosravanian, R., Anemangely, M., & Younesi, A. (2019). A machine learning approach to predict drilling rate using petrophysical and mud logging data. *Earth Science Informatics*, *12*, 319-339.

[17]. Mehrad, M., Bajolvand, M., Ramezanzadeh, A., & Neycharan, J. G. (2020). Developing a new rigorous drilling rate prediction model using a machine learning technique. *Journal of Petroleum Science and Engineering*, *192*, 107338.

[18]. Bajolvand, M., Ramezanzadeh, A., Mehrad, M., & Roohi, A. (2022). Optimization of controllable drilling parameters using a novel geomechanics-based workflow. *Journal of Petroleum Science and Engineering*, *218*, 111004.

[19]. Wang, P., & Peng, S. (2019). On a new method of estimating shear wave velocity from conventional well logs. *Journal of Petroleum Science and Engineering*, *180*, 105-123.

[20]. Akhundi, H., Ghafoori, M., & Lashkaripour, G. R. (2014). Prediction of shear wave velocity using artificial neural network technique, multiple regression and petrophysical data: A case study in Asmari reservoir (SW Iran). *Open Journal of Geology*, *2014*.

[21]. Rezaee, M. R., Ilkhchi, A. K., & Barabadi, gracias parA. (2007). Prediction of shear wave velocity from petrophysical data utilizing intelligent systems: An example from a sandstone reservoir of Carnarvon Basin, Australia. *Journal of Petroleum Science and Engineering*, *55*(3-4), 201-212.

[22]. Rajabi, M., Hazbeh, O., Davoodi, S., Wood, D. A., Tehrani, P. S., Ghorbani, H., ... & Radwan, A. E. (2023). Predicting shear wave velocity from conventional well logs with deep and hybrid machine learning algorithms. *Journal of Petroleum Exploration and Production Technology*, *13*(1), 19-42.

[23]. Wang, J., Cao, J., & Yuan, S. (2020). Shear wave velocity prediction based on adaptive particle swarm optimization optimized recurrent neural network. *Journal of Petroleum Science and Engineering*, *194*, 107466.

[24]. Jeong, J., Park, E., Emelyanova, I., Pervukhina, M., Esteban, L., & Yun, S. T. (2021). Application of conditional generative model for sonic log estimation considering measurement uncertainty. *Journal of Petroleum Science and Engineering*, *196*, 108028.

[25]. Zhang, D., Yuntian, C. H. E. N., & Jin, M. E. N. G. (2018). Synthetic well logs generation via recurrent neural networks. *Petroleum Exploration and Development*, *45*(4), 629-639.

[26]. Pham, N., & Naeini, E. Z. (2019, June). Missing well log prediction using deep recurrent neural networks. In *81st EAGE Conference and Exhibition 2019* (Vol. 2019, No. 1, pp. 1-5). European Association of Geoscientists & Engineers.

[27]. Maletic, J. I., & Marcus, A. (2005). Data cleansing. *Data mining and knowledge discovery handbook*, 21-36.

[28]. Wu, X. (1996). *Knowledge acquisition from databases*. Ablex Publishing Corp.

[29]. Garćia, L. P. F., de Carvalho, A. C., & Lorena, A. C. (2013). Noisy data set identification. In *Hybrid Artificial Intelligent Systems: 8th International Conference, HAIS 2013, Salamanca, Spain, September 11-13, 2013. Proceedings 8* (pp. 629-638). Springer Berlin Heidelberg.

[30]. Lorena, A. C., & de Carvalho, A. C. (2004). Evaluation of noise reduction techniques in the splice junction recognition problem. *Genetics and Molecular Biology*, *27*, 665-672.

[31]. Tunkiel, A. T., Sui, D., & Wiktorski, T. (2022). Impact of data pre-processing techniques on recurrent neural network performance in context of real-time drilling logs in an automated prediction framework. *Journal of Petroleum Science and Engineering*, *208*, 109760.

[32]. Matinkia, M., Sheykhinasab, A., Shojaei, S., Vojdani Tazeh Kand, A., Elmi, A., Bajolvand, M., & Mehrad, M. (2022). Developing a new model for drilling rate of penetration prediction using convolutional neural network. *Arabian Journal for Science and Engineering*, *47*(9), 11953-11985.

[33]. Matinkia, M., Amraeiniya, A., Behboud, M. M., Mehrad, M., Bajolvand, M., Gandomgoun, M. H., & Gandomgoun, M. (2022). A novel approach to pore pressure modeling based on conventional well logs using convolutional neural network. *Journal of Petroleum Science and Engineering*, *211*, 110156.

[34]. Sabah, M., Mehrad, M., Ashrafi, S. B., Wood, D. A., & Fathi, S. (2021). Hybrid machine learning algorithms to enhance lost-circulation prediction and management in the Marun oil field. *Journal of Petroleum Science and Engineering*, *198*, 108125.

[35]. Elkatatny, S. (2019). Development of a new rate of penetration model using self-adaptive differential

evolution-artificial neural network. *Arabian Journal of Geosciences*, *12*, 1-10.

[36]. Breiman, L. (1996). Bagging predictors. *Machine learning*, *24*, 123-140.

[37]. Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992, July). A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory* (pp. 144-152).

[38]. Awad, M., & Khanna, R. (2015). Support vector regression in efficient learning machines (pp. 67–80). *Apress, Berkeley, CA*.

[39]. Zhang, F., & O'Donnell, L. J. (2020). Support vector regression. In *Machine learning* (pp. 123-140). Academic Press.

[40]. Kuss, M. (2006). *Gaussian process models for robust regression, classification, and reinforcement learning* (Doctoral dissertation, echnische Universität Darmstadt Darmstadt, Germany).

[41]. Demuth, H. B., Beale, M. H., De Jess, O., & Hagan, M. T. (2014). *Neural network design*. Martin Hagan.

[42]. Haykin, S., & Network, N. (2004). A comprehensive foundation. *Neural networks*, *2*(2004), 41.

[43]. Nebauer, C. (1998). Evaluation of convolutional neural networks for visual recognition. *IEEE transactions on neural networks*, *9*(4), 685-696.

[44]. Indolia, S., Goswami, A. K., Mishra, S. P., & Asopa, P. (2018). Conceptual understanding of convolutional neural network-a deep learning approach. *Procedia computer science*, *132*, 679-688.

[45]. Subasi, A. (2020). *Practical machine learning for data analysis using python.* Academic Press.

[46]. Mandic, D. P., & Chambers, J. (2001). *Recurrent neural networks for prediction: learning algorithms, architectures and stability*. John Wiley & Sons, Inc.

[47]. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780.

[48]. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

[49]. Hamad, Z., & Abdulrahman, I. (2022). Deep learning-based load forecasting considering data reshaping using MATLAB\Simulink. *International Journal of Energy and Environmental Engineering*, *13*(2), 853-869.

[50]. Gonzalez, R. C., & Woods, R. E. (2008). Digital image processing: Pearson prentice hall. *Upper Saddle River, NJ*, *1*(376-376), 97.

[51]. Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. A. M. T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation*, *6*(2), 182-197.

[52]. Souier, M., Dahane, M., & Maliki, F. (2019). An NSGA-II-based multiobjective approach for real-time routing selection in a flexible manufacturing system under uncertainty and reliability constraints. *The International Journal of Advanced Manufacturing Technology*, *100*, 2813-2829.

Nomenclature

ANN:	Artificial Neural Network
BGT:	Bagging tree
CNN:	Convolutional Neural Network
DTCO:	Compressional wave slowness (μs/ft)
DTSM:	Shear wave slowness (μs/ft)
GR:	Gamma ray log (GAPI)
GPR:	Gaussian Processing Regression
GRU:	Gradient Recurrent Unit
LSTM:	Long Short-Term Memory
NSGA-II:	Non Dominated Sorting Genetic Algorithm
NPHI:	Neutron Porosity log (v/v)
PEF:	Photoelectric (B/e)
RHOB:	Density (gr/cc)
RNN:	Recurrent Neural Network
RMSE:	Root Means of Square Error
RT:	Resistivity (Ohmm)
SD:	Standard Deviation
SVR:	Support Vector Machine
V_p:	Compressional Wave velocity (km/sec)
V_s:	Shear wave velocity (km/sec)

Appendices

Appendix A: Dataset Descriptions and Analysis

- Data visualization

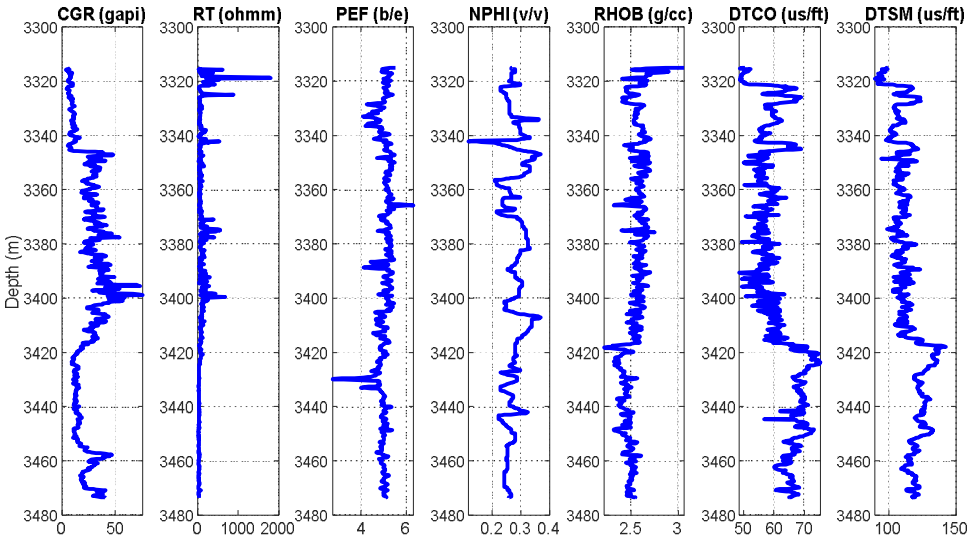


Figure A1. Profile of petrophysical logs in studied interval of Well ABT-a.

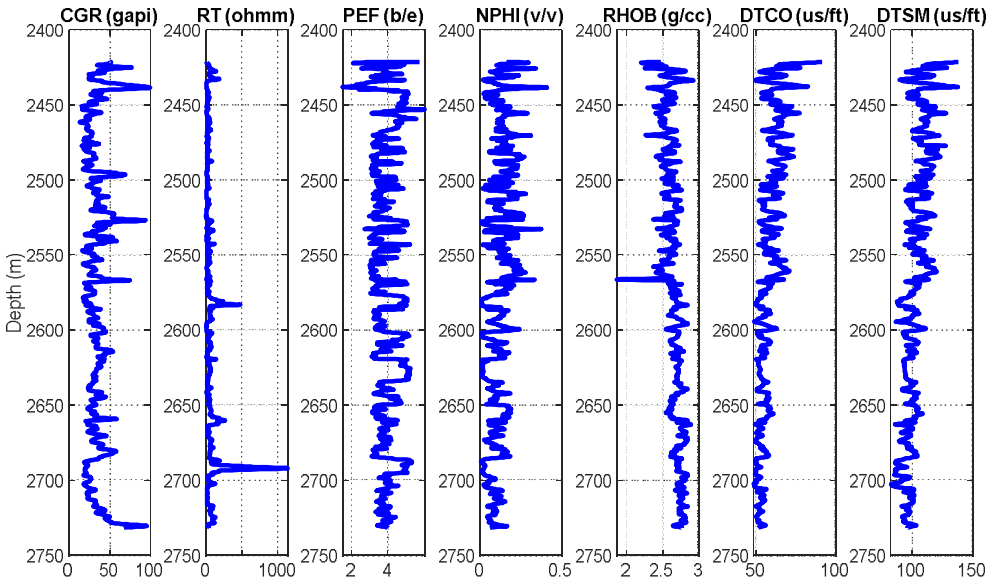


Figure A2. Profile of petrophysical logs in studied interval of Well AHZ-a.

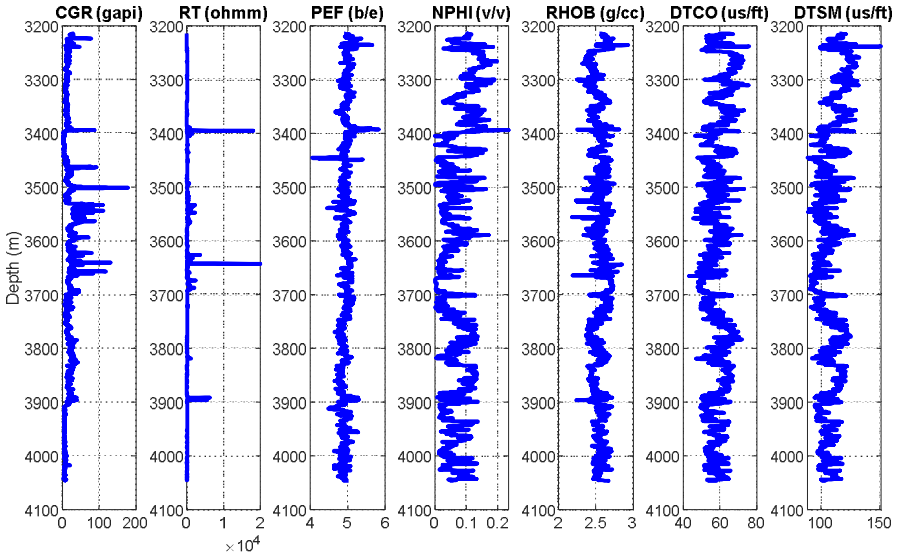


Figure A3. Profile of petrophysical logs in studied interval of Well AHZ-b.

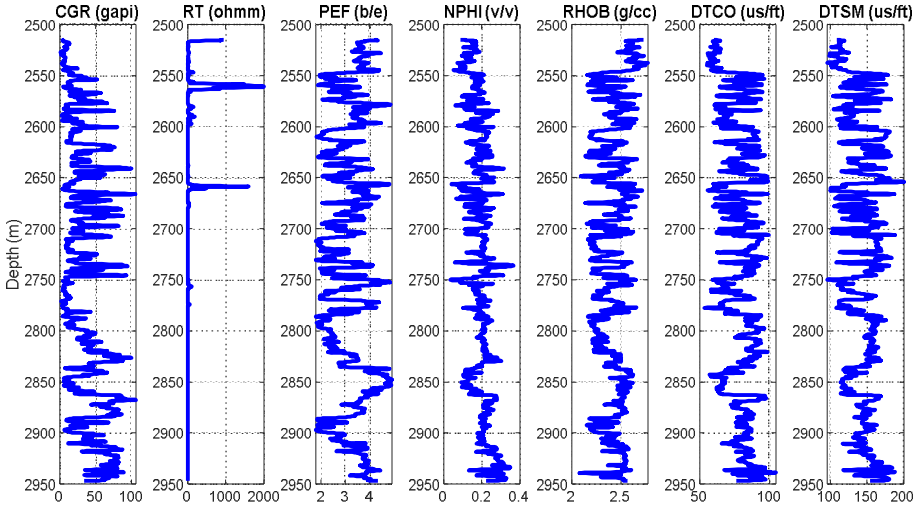


Figure A4. Profile of petrophysical logs in studied interval of Well AHZ-c.

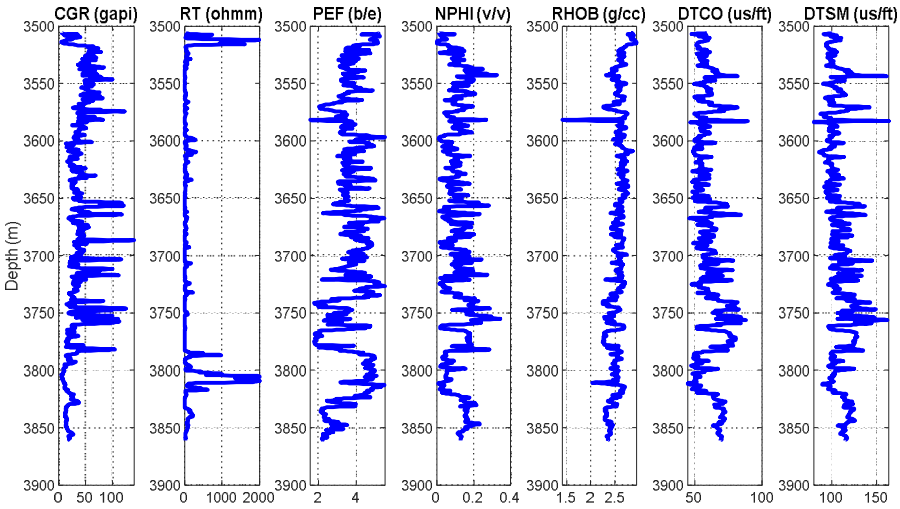


Figure A5. Profile of petrophysical logs in studied interval of Well MRN-a.

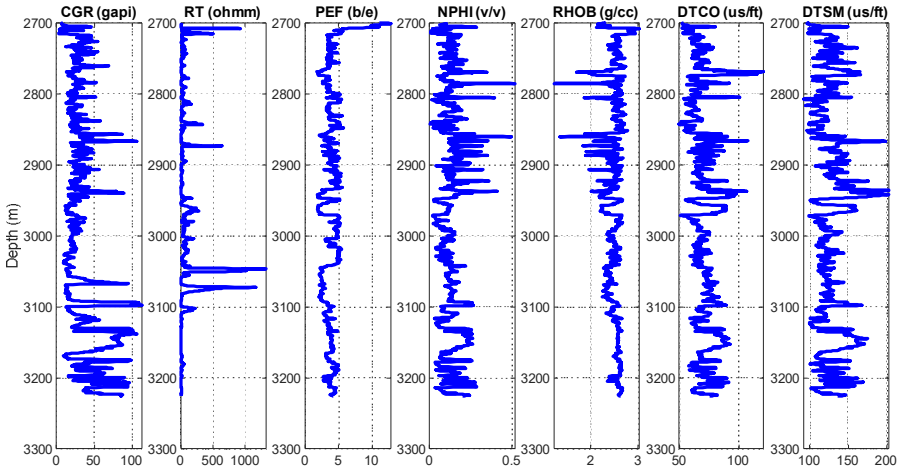


Figure A6. Profile of petrophysical logs in studied interval of well MRN-b.

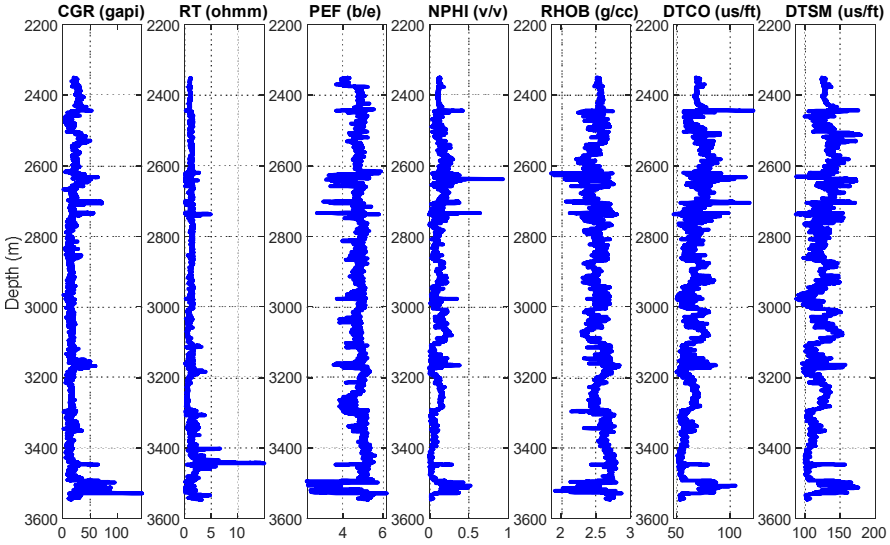


Figure A7. Profile of petrophysical logs in studied interval of Well AZN-a.

- **Statistical indices**

Table A1. Statistical indices of petrophysical logs collected from studied wells.

Well Name	Statistical indices	Depth	GR	RT	PEF	NPHI	RHOB	DTCO	DTSM
		m	gapi	ohmm	b/e	v/v	g/cc	us/ft	us/ft
ABT-a	Min	3315.16	5.38	1.21	4.38	0.21	2.25	49.26	92.58
	Mean	3397.94	22.57	59.46	5.03	0.28	2.53	62.03	115.23
	Median	3401.57	22.59	50.74	5.04	0.27	2.54	60.54	113.67
	Max	3473.65	55.47	163.47	5.70	0.36	2.78	74.77	136.53
	Mode	-	18.44	93.74	5.13	0.28	2.45	68.53	106.54
	Std. dev.	45.53	10.54	33.43	0.22	0.03	0.09	5.71	8.78
	Kurtosis	-1.25	-1.05	0.22	0.21	-0.44	-0.36	-1.01	-0.20
	Skewness	-0.05	0.21	0.95	-0.5	0.43	-0.28	0.29	0.45
	Count	923	923	923	923	923	923	923	923
AHZ-a	Min	2422.69	16.07	5.25	2.73	0.01	2.36	48.83	83.19
	Mean	2575.85	32.34	39.82	3.90	0.12	2.65	57.33	102.50
	Median	2573.96	31.36	35.07	3.75	0.11	2.66	56.10	101.24
	Max	2728.91	60.96	108.65	5.50	0.26	2.88	71.95	122.84
	Mode	-	30.15	34.88	4.02	0.12	2.62	57.83	95.98
	Std. dev.	87.57	8.96	22.74	0.60	0.06	0.10	5.49	7.72
	Kurtosis	-1.17	0.23	0.27	-0.7	-0.59	-0.48	-0.53	-0.47
	Skewness	0.03	0.70	0.83	0.64	0.24	-0.29	0.64	0.28
	Count	1839	1839	1839	1839	1839	1839	1839	1839
AHZ-b	Min	3214.27	1.91	1.54	4.61	0.00	2.31	46.57	92.25
	Mean	3640.57	14.58	58.96	4.92	0.08	2.54	59.60	109.80
	Median	3666.67	13.38	49.42	4.92	0.08	2.54	59.74	109.92
	Max	4046.37	42.68	199.65	5.25	0.18	2.72	73.04	130.39
	Mode	-	20.98	7.54	4.96	0.08	2.48	67.27	99.38
	Std. dev.	254.43	7.76	41.63	0.10	0.04	0.08	5.26	8.63
	Kurtosis	-1.36	0.11	0.97	-0.1	-1.09	-0.78	-0.98	-0.95
	Skewness	-0.09	0.75	1.14	0.21	0.17	0.04	0.05	0.19
	Count	4582	4582	4582	4582	4582	4582	4582	4582
AHZ-c	Min	2523.13	3.19	0.18	1.85	0.08	2.15	57.26	103.16
	Mean	2764.72	38.64	1.63	3.05	0.20	2.41	81.02	148.10
	Median	2776.96	34.16	1.43	3.01	0.21	2.43	83.32	150.00
	Max	2946.96	94.73	7.13	4.85	0.33	2.73	100.00	191.81
	Mode	-	32.70	0.76	2.16	0.17	2.53	67.93	144.46
	Std. dev.	111.89	24.21	1.34	0.78	0.05	0.13	9.45	18.30
	Kurtosis	-1.08	-1.02	2.13	-0.9	0.03	-1.15	-0.69	-0.63
	Skewness	-0.19	0.43	1.43	0.25	-0.03	-0.19	-0.48	-0.32
	Count	2208	2208	2208	2208	2208	2208	2208	2208
AZN-a	Min	2350.01	4.87	0.07	4.17	0.00	2.22	50.72	92.96
	Mean	2949.26	17.13	1.23	4.83	0.10	2.54	64.85	121.98
	Median	2953.51	15.99	1.28	4.84	0.10	2.55	64.27	121.79
	Max	3545.43	32.35	2.43	5.44	0.28	2.77	89.65	168.59
	Mode	-	19.90	1.37	4.84	0.08	2.55	66.43	115.83
	Std. dev.	317.88	5.29	0.41	0.21	0.06	0.11	7.71	13.21
	Kurtosis	-1.10	0.43	-0.14	0.67	-0.62	-0.67	-0.65	-0.61
	Skewness	-0.05	0.93	-0.01	-0.4	0.30	-0.24	0.34	0.27
	Count	6649	6649	6649	6649	6649	6649	6649	6649
AZN-b	Min	2354.22	5.21	0.91	3.60	0.05	2.39	55.14	108.34
	Mean	2584.81	20.45	3.05	4.70	0.14	2.54	67.01	129.97
	Median	2560.87	20.34	2.86	4.73	0.14	2.53	67.12	129.41
	Max	2869.64	35.32	6.35	5.90	0.24	2.66	78.10	153.97
	Mode	-	21.72	3.15	4.93	0.14	2.52	75.30	125.15
	Std. dev.	144.67	5.25	1.07	0.39	0.03	0.04	3.25	7.10
	Kurtosis	-1.15	0.00	1.00	-0.2	1.74	0.83	1.11	1.82
	Skewness	0.22	-0.41	1.16	-0.2	-0.07	0.37	-0.27	0.82
	Count	2643	2643	2643	2643	2643	2643	2643	2643
MRN-a	Min	3509.06	8.68	0.41	1.81	0.02	2.23	49.37	89.23
	Mean	3675.00	37.01	26.14	3.59	0.12	2.56	60.87	108.55
	Median	3671.82	34.57	18.32	3.54	0.12	2.59	59.13	106.35
	Max	3860.95	89.62	92.92	5.25	0.29	2.86	80.18	138.40
	Mode	-	60.90	8.70	3.13	0.11	2.84	55.58	104.15
	Std. dev.	98.26	16.98	22.17	0.80	0.05	0.13	7.49	10.46
	Kurtosis	-1.05	-0.03	0.42	-0.4	-0.50	-0.31	-0.38	-0.64
	Skewness	0.18	0.72	1.09	-0.2	0.08	-0.61	0.68	0.51
	Count	1859	1859	1859	1859	1859	1859	1859	1859
MRN-b	Min	2710.22	8.40	1.78	1.95	0.02	2.18	52.60	95.08
	Mean	2946.58	28.50	23.57	3.87	0.11	2.53	67.06	121.98
	Median	2926.55	26.57	16.73	3.81	0.12	2.54	66.26	120.47
	Max	3222.74	65.61	75.27	5.43	0.28	2.88	92.61	173.19
	Mode	-	23.62	32.25	2.47	0.05	2.61	65.93	124.10
	Std. dev.	148.81	10.72	18.93	0.79	0.05	0.10	6.55	13.04
	Kurtosis	-1.14	0.57	-0.27	-0.7	-0.08	0.26	0.53	0.19
	Skewness	0.21	0.82	0.89	-0.3	0.17	-0.40	0.54	0.53
	Count	2364	2364	2364	2364	2364	2364	2364	2364

Appendix B: Data Preparation

- **Data de-noising results (cleaned vs. measured logs).**

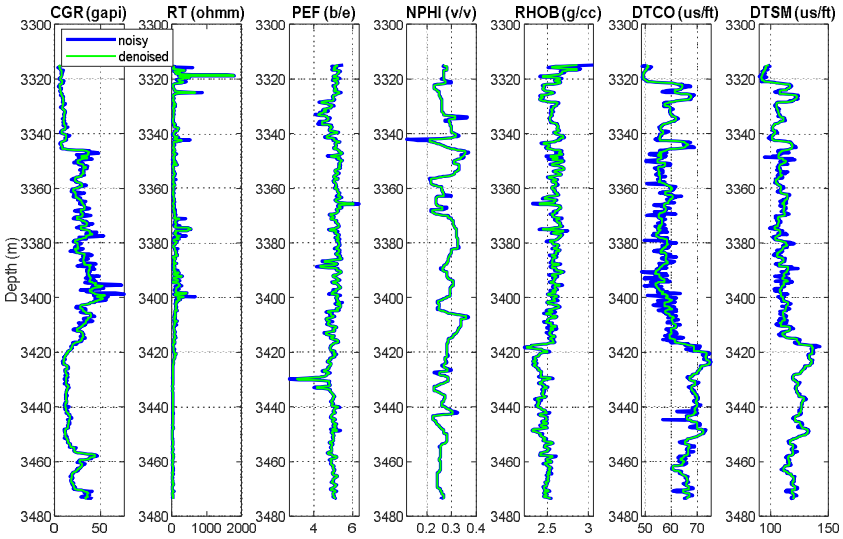


Figure B1. The comparison of row and de-noised petrophysical logs in studied interval of Well ABTa.

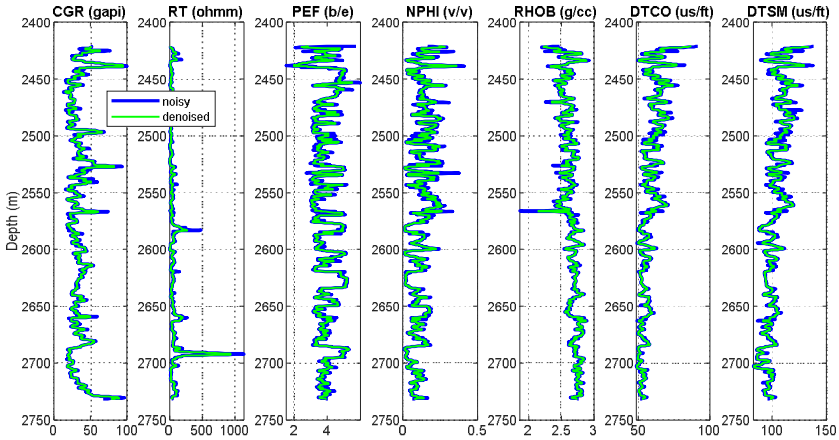


Figure B2. The comparison of row and de-noised petrophysical logs in studied interval of Well AHZa.

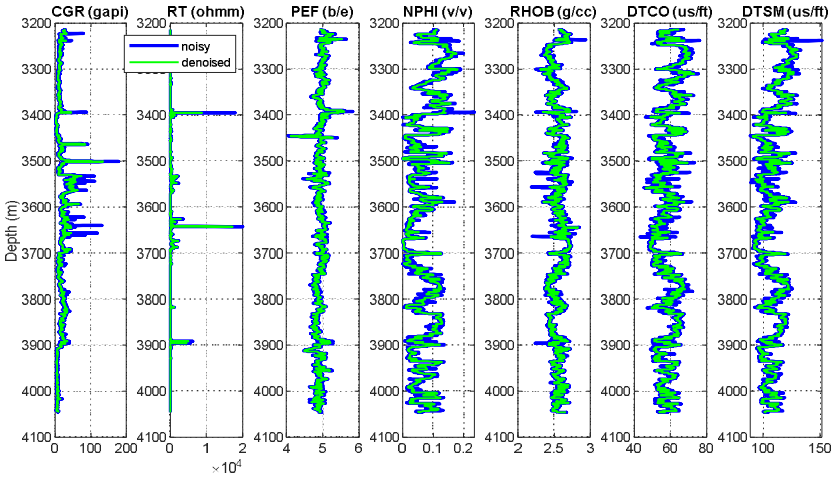


Figure B3. The comparison of row and de-noised petrophysical logs in studied interval of Well AHZ-b.

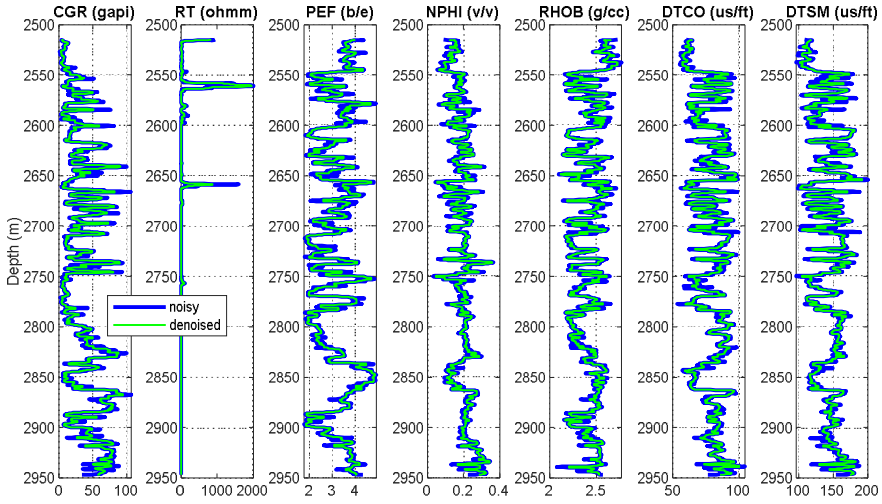


Figure B4. The comparison of row and de-noised petrophysical logs in studied interval of Well AHZ-c.

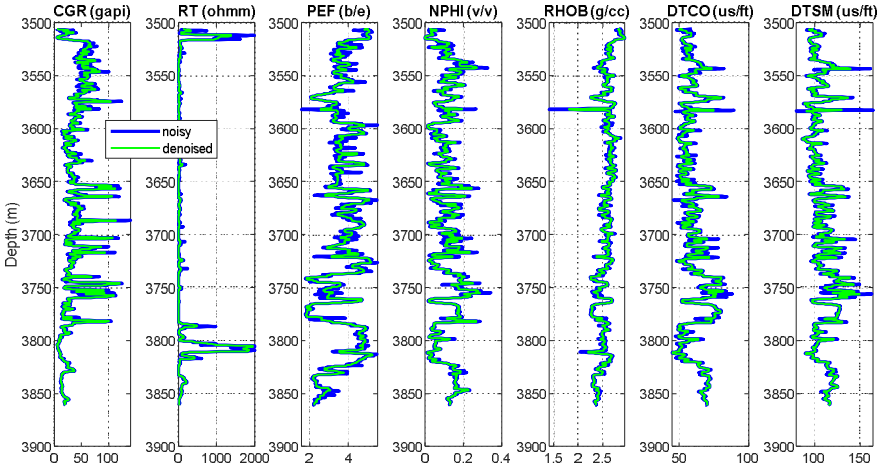


Figure B5. The comparison of row and de-noised petrophysical logs in studied interval of Well MRN-a.

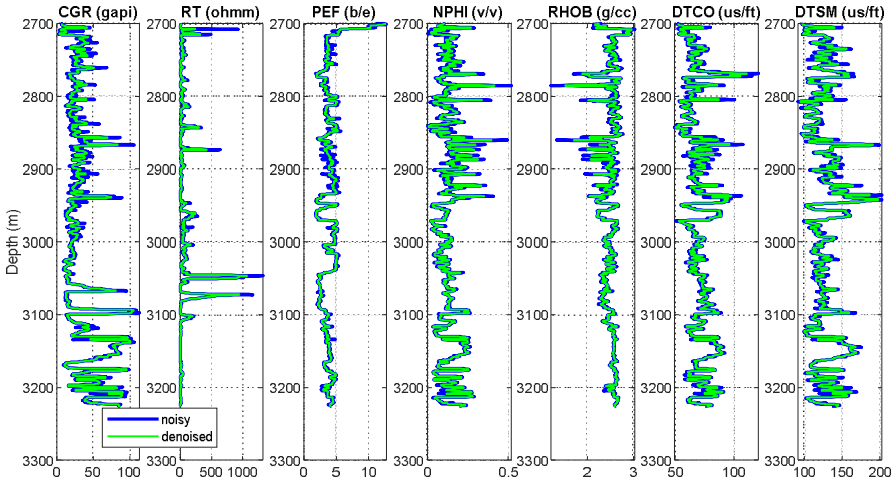


Figure B6. The comparison of row and de-noised petrophysical logs in studied interval of Well MRN-b.

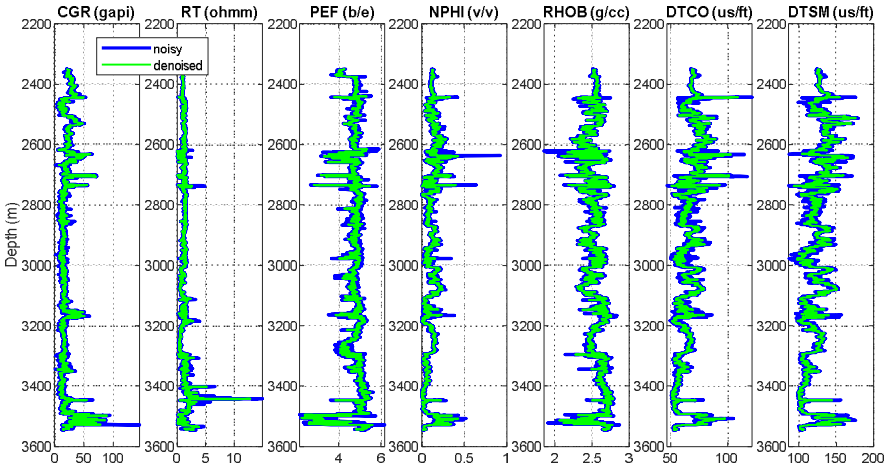
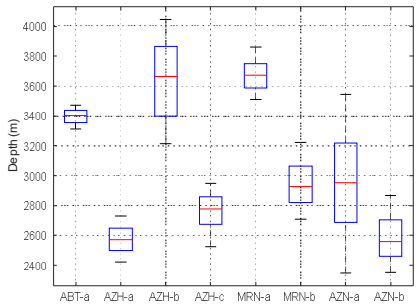
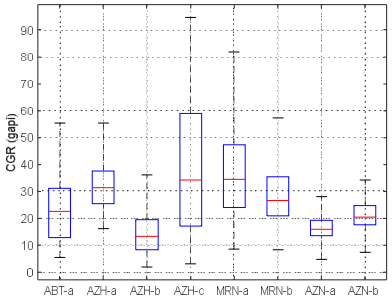


Figure B7. The comparison of row and de-noised petrophysical logs in studied interval of Well AZN-a.

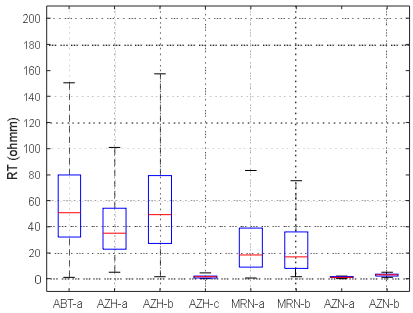
- **Range of cleaned data after outlier detection and elimination.**



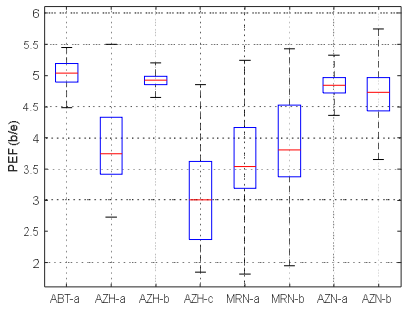
a) Depth



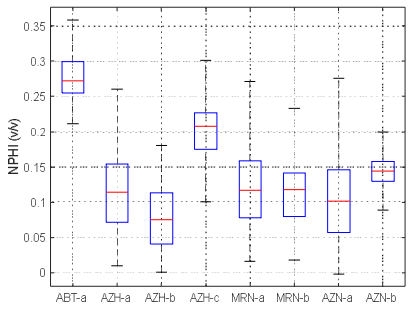
b) CGR



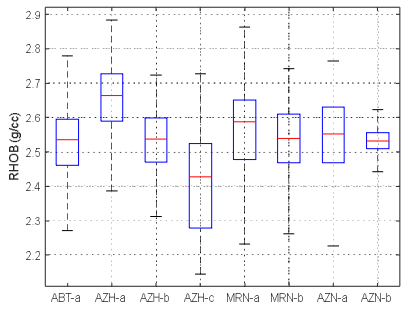
c) RT



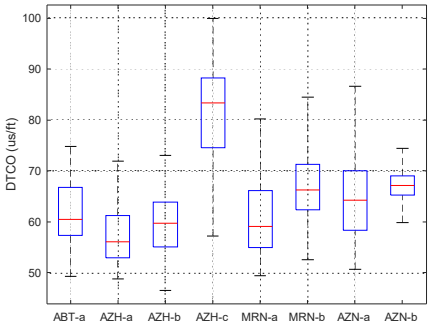
d) PEF



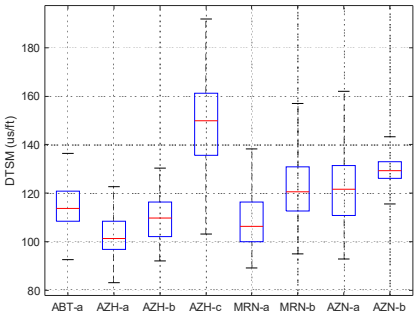
e) NPHI



f) RHOB



g) DTCO



h) DTSM

Figure B8. Range of cleaned data after outlier detection and elimination.

Appendix C: Error and Performance Calculation Equations

Equation (C1) was used to calculate Percent Deviation (PD) or Relative Error (RE) for each data point (i) in the dataset (containing n data points) based on the measured Parameters ($P_{measured}$) and predicted P ($P_{predicted}$).

$$PD_i = \frac{P_{measured} - P_{predicted}}{P_{measured}} \times 100 \quad \text{(C1)}$$

Once PD is known for each point in the dataset, one can calculate average percent deviation (APD) using Equation (C2).

$$APD = \frac{\sum_{i=1}^{n} PD_i}{n} \quad \text{(C2)}$$

Average absolute percent deviation ($AAPD$) is given through Equation (C3).

$$AAPD = \frac{\sum_{i=1}^{n} |PD_i|}{n} \quad \text{(C3)}$$

Standard deviation (SD) of error can be computed from mean error (Er_{mean}) and error terms at individual data points (Er) through Equation (C4).

$$SD = \sqrt{\frac{\sum_{i=1}^{n} (Er_i - Er_{mean})^2}{n-1}} \quad \text{(C4)}$$

RMSE of each model can then be evaluated using Equation (C5).

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} \left(P_{measured_i} - P_{predicted_i}\right)^2} \quad \text{(C5)}$$

For each prediction, the COD (R^2) is calculated through Equation (C6).

$$R^2 = 1 - \frac{\sum_{i=1}^{n} \left(P_{measured_i} - P_{predicted_i}\right)^2}{\sum_{i=1}^{n} \left(P_{predicted_i} - \frac{\sum_{i=1}^{n} P_{measured}}{n}\right)^2} \quad \text{(C6)}$$

کاهش عدم قطعیت مدل‌های ژئومکانیکی با تخمین کندی موج برشی با استفاده از مدل‌های شبکه عصبی عمیق

مهدی باجولوند[1*]، احمد رمضان زاده[2]، امین حکمت نژاد[1]، محمد مهراد[2] شادفر داودی[3] و محمد تیموری[4]

۱- بخش مهندسی معدن دانشکده مهندسی، دانشگاه کاتولیک پاپی شیلی، سانتیاگو، شیلی

۲- دانشکده مهندسی معدن، نفت و ژئوفیزیک، دانشگاه صنعتی شاهرود، شاهرود، ایران

۳- دانشکده مهندسی و علوم زمین، دانشگاه پلی تکنیک تامسک، تامسک، روسیه

۴- شرکت نفت مناطق مرکزی ایران، تهران، ایران

ارسال ۲۰۲۴/۱۰/۳۰، پذیرش ۲۰۲۵/۰۳/۱۷

* نویسنده مسئول مکاتبات: mbajolvand@uc.cl

چکیده:

لاگ کندی موج برشی (DTSM) یکی از مهم‌ترین لاگ‌های پتروفیزیکی است که در مطالعات مخازن، به‌ویژه در مطالعات ژئومکانیکی میادین نفت و گاز، کاربرد دارد. بااین‌حال، عدم وجود این پارامتر در لاگ‌گیری چاه می‌تواند منابع بزرگی از عدم قطعیت را در مطالعات ژئومکانیکی ایجاد کند. این مطالعه با هدف ارائه راهکارهایی برای کاهش عدم قطعیت در مدل‌های ژئومکانیکی از طریق تخمین لاگ کندی موج برشی با استفاده از مدل‌های دقیق یادگیری عمیق ماشین انجام شده است. ایده اصلی این تحقیق، استفاده از داده‌های میادین مجاور برای گسترش دامنه داده‌های آموزشی و بهبود توانایی تخمین و قابلیت تعمیم مدل‌های یادگیری ماشین است. بدین منظور، داده‌های پتروفیزیکی از ۸ چاه در ۴ میدان نفتی ایران جمع‌آوری شد. در مرحله نخست، پیش‌پردازش داده‌ها برای کاهش اثرات داده‌های نادرست، مقادیر ازدست‌رفته، نویز و داده‌های پرت انجام شد. سپس، مدل‌های یادگیری ماشین (مبتنی بر یادگیری رگرسیونی و شبکه‌های عصبی عمیق) و مدل‌های تحلیلی برای تخمین کندی موج برشی پیاده‌سازی شدند. نتایج نشان داد که مدل شبکه عصبی عمیق واحد بازگشتی دروازه‌ای (GRU) با مقادیر ۱.۹ و ۲.۱۴ برای مجذور میانگین مربع خطاها (RMSE) و ۰.۹۹ برای ضریب تعیین (R^2) به ترتیب در داده‌های آموزش و آزمون، دقیق‌ترین نتایج را ارائه داده است. همچنین، ارزیابی دقت مدل‌ها بر روی داده‌های چاه اعتبارسنجی نشان داد که مدل GRU با مقادیر ۲.۴۳ برای RMSE و ۰.۹۳ برای R^2 دقیق‌ترین مدل بوده است. بر این اساس، استفاده از یک بانک داده جامع چند میـــدانی و به‌کارگیری روش‌های یادگیری ماشین برای تخمین کندی موج برشی در شرایطی که داده‌های محدودی از میادین مجاور در دسترس است، به‌شدت توصیه می‌شود.

کلمات کلیدی: کندی موج برشی، لاگ پتروفیزیکی، یادگیری ماشین، یادگیری عمیق، مدل تحلیلی