



OPEN

Decoupling and predicting natural gas deviation factor using machine learning methods

Shaoyang Geng¹, Shuo Zhai¹, Jianwen Ye², Yajie Gao³, Hao Luo³, Chengyong Li¹,
Xianshan Liu¹✉ & Shudong Liu¹✉

Accurately predicting the deviation factor (Z-factor) of natural gas is crucial for the estimation of natural gas reserves, evaluation of gas reservoir recovery, and assessment of natural gas transport in pipelines. Traditional machine learning algorithms, such as Support Vector Machine (SVM), Extreme Gradient Boosting (XGBoost), Light Gradient Boosting Machine (LightGBM), Artificial Neural Network (ANN) and Bidirectional Long Short-Term Memory Neural Networks (BiLSTM), often lack accuracy and robustness in various situations due to their inability to generalize across different gas components and temperature-pressure conditions. To address this limitation, we propose a novel and efficient machine learning framework for predicting natural gas Z-factor. Our approach first utilizes a signal decomposition algorithm like Variational Mode Decomposition (VMD), Empirical Fourier Decomposition (EFD) and Ensemble Empirical Mode Decomposition (EEMD) to decouple the Z-factor into multiple components. Subsequently, traditional machine learning algorithms is employed to predict each decomposed Z-factor component, where combination of SVM and VMD achieved the best performance. Decoupling the Z-factors firstly and then predicting the decoupled components can significantly improve prediction accuracy of all traditional machine learning algorithms. We thoroughly evaluate the impact of the decoupling method and the number of decomposed components on the model's performance. Compared to traditional machine learning models without decomposition, our framework achieves an average correlation coefficient exceeding 0.99 and an average mean absolute percentage error below 0.83% on 10 datasets with different natural gas components, high temperatures, and pressures. These results indicate that hybrid model effectively learns the patterns of Z-factor variations and can be applied to the prediction of natural gas Z-factors under various conditions. This study significantly advances methodologies for predicting natural gas properties, offering a unified and robust solution for precise estimations, thereby benefiting the natural gas industry in resource estimation and reservoir management.

Keywords Machine learning, Natural gas, Deviation factor, Z-factor, Decomposition

The deviation factor stands out as one of the most critical thermodynamic properties of natural gas, serving as the foundation for computing various other properties such as natural gas density, volume coefficient, and heat capacity¹. Essentially, the natural gas Z-factor quantifies the ratio of the real natural gas volume to its theoretical volume. This parameter plays a pivotal role in correcting measurements of gas volume, as it encapsulates the non-ideality of natural gases. Its significance reverberates throughout the entire natural gas process, spanning from extraction to transportation². Accurately predicting the Z-factor of natural gas is paramount for ensuring the precision and consistency of gas measurement and metering. This level of accuracy carries significant implications across multiple domains, including the estimation of gas in place, determination of gas reservoir recovery, assessment of pipeline transport capacity, and comprehension of fluid dynamics behavior within pipelines^{3–5}.

The value of the Z-factor depends on factors such as the components, temperature, and pressure of natural gas. Generally, there are two methods to obtain the Z-factor. One involves experimental studies through constant volume depletion (CVD) tests, and the other relies on empirical or semi-empirical mathematical correlations for estimation^{6,7}. CVD experiments are considered the most accurate means of obtaining the Z-factor, requiring fluid samples from gas well to simulate reservoir behavior at different pressure depletion stages⁸. Despite laboratory

¹Chengdu University of Technology, College of Energy, Chengdu 610059, China. ²Sinopec Southwest Oil and Gas Company, Chengdu 611930, China. ³PetroChina Southwest Oil and Gasfield Company, Chengdu 610051, China. ✉email: liuxianshan@stu.cdut.edu.cn; lsddz2013@163.com

tests being theoretically more accurate than mathematical relationships in estimating the Z-factor, there are two significant drawbacks. Firstly, it requires obtaining well stream for experiments, which is impractical throughout the entire life cycle of natural gas production for CVD experiments. Additionally, for a gas reservoir, the impact of the natural gas accumulation process results in significant variations in gas components at different locations⁹. Sampling and testing for each well, especially in gas fields with potentially hundreds of wells, is impractical. Secondly, experimental acquisition of the Z-factor requires expensive laboratory facilities and complex experimental procedures, making it challenging to estimate the cost and time investment for non-specialists¹⁰. Consequently, laboratory testing is not suitable for routine industry analysis.

The second commonly used method to obtain the Z-factor is through empirical or semi-empirical mathematical approaches. Mathematical approaches are typically developed based on a large dataset of Z-factors to establish correlations applicable under specific conditions¹¹. Each Z-factor correlations developed in previous research has its own advantages and limitations in estimating the accuracy of the Z-factor. The applicability of these correlations largely depends on the gas components, including the content of non-hydrocarbons such as nitrogen, carbon dioxide, and heavy hydrocarbons¹². The content of these components varies based on region, pressure, and temperature range. None of these correlations can guarantee universal applicability across all components and under any temperature and pressure conditions¹³. Especially when the content of CO₂ and H₂S is high, most correlations failed to estimate Z-factors, leading to significant errors in predicting the Z-factor^{14,15}. Therefore, determining the optimal Z-factor correlation for different gas reservoirs in various regions remains a challenging task. Additionally, developing a suitable correlation is quite difficult. The mentioned developed Z-factor correlations typically involve over 10 undetermined parameters. The complexity of correlations is staggering, making the development of an accurate correlation challenging for engineers. Moreover, correlations based on the equation of state are often implicit, requiring expensive iterative computational costs¹⁶.

The rapid development of data-driven methods in recent years has provided a straightforward and swift path for the development of Z-factor prediction models. The correlation of natural gas Z-factors, and even other properties such as viscosity and density, is essentially a nonlinear function with pressure and temperature. Machine learning models demonstrate powerful capabilities in regression and prediction of nonlinear functions^{17–19}. Nevertheless, conventional machine learning algorithms like SVM, XGBoost and LightGBM have poor generalization ability in predicting Z-factor and are less commonly used. Therefore, researchers have developed numerous improved models based on ANNs to characterize the properties of natural gas²⁰. Chamkalani et al. introduced particle swarm optimization and genetic algorithms to optimize neural networks, thus developing an ANN model for predicting the Z-factor²¹. Tariq & Mahmoud utilized ANNs to linearly present an improved Z-factor empirical correlation, addressing the issue of correlations inaccurately predicting the Z-factor at high gas reservoir pressures¹⁶. Tariq et al. developed a data-driven model to estimate the natural gas density²². Orodu et al. compared conventional methods and machine learning methods in predicting the natural gas density at high temperatures and pressures²³. Faraji et al. used ANN to develop a model predicting the Z-factor of two-phase condensate gas⁹. Numerous ANN based models have been developed to address the issue of conventional correlations failing to accurately predict Z-factors under abnormal conditions^{24,25}. On the other hand, unlike conventional machine learning algorithms, the architecture of ANNs is arbitrary, and its design varies significantly across different datasets²⁶. ANNs do not have a unified architecture to predict the Z-factor under various conditions. Thus, researchers have selected a large number of other algorithms to predict the Z-factors under specific conditions. Gaganis et al. developed a hybrid model based on Truncated Regularized Kernel Ridge Regression algorithm in conjunction with a simple linear-quadratic interpolation scheme to estimate the Z-factor in the presence of impurities²⁷. Maalouf et al. applied the kernel ridge regression in the form of the recently developed truncated regularized kernel ridge regression algorithm to estimate Z-factor²⁸. Abdolhossein et al. proposed a second-order polynomial method based on the group method of data handling²⁹. However, due to the complexity of the Z-factor itself, these improved models are only applicable under specific conditions. The previously proposed algorithms including SVM, XGBoost, LightGBM ANNs and other algorithms mentioned above cannot be generalized to different gas components, temperature and pressure conditions, and therefore often lack accuracy and robustness in various prediction situations.

This paper aims to develop a straightforward and efficient machine learning framework capable of swiftly predicting Z-factors under diverse temperature and pressure conditions, as well as complex natural gas compositions. Given that natural gas comprises a mixture of various gas components, each exerting a collective influence on its thermodynamic properties, the Z-factor serves as a comprehensive representation of their combined impact. Drawing inspiration from this concept, we employ a signal decomposition algorithm to decoupled the Z-factor into several components. Subsequently, utilizing pseudo pressure, pseudo temperature, temperature, and pressure as independent variables, we apply conventional machine learning methods to predict these decoupled variables separately, before reassembling and summing the Z-factor. Remarkably, this model achieves exceptionally high accuracy without relying on complex network structures or intricate parameter tuning, rendering it easily accessible to engineers for practical applications in gas reservoir development and production.

Methodology

Basic dataset

The foundational Z-factor data for developing the machine learning models were sourced from gas wells of the Jurassic Shaximiao Formation gas reservoir in the Sichuan Basin, China. The Shaximiao Formation is a thick red formation developed in the Middle Jurassic of the Sichuan Basin, with a thickness ranging from 1000 to 2000 m. It primarily consists of dark purple-red mudstone, interspersed with multiple sets of sandstone. The reservoir pressure ranges from 19.1 to 26.5 MPa, and the pressure coefficient varies between 0.845 and 1.155. The reservoir temperature ranges from 323.86 to 356.75 K with an average of 338.23 K. Natural gas migrates from

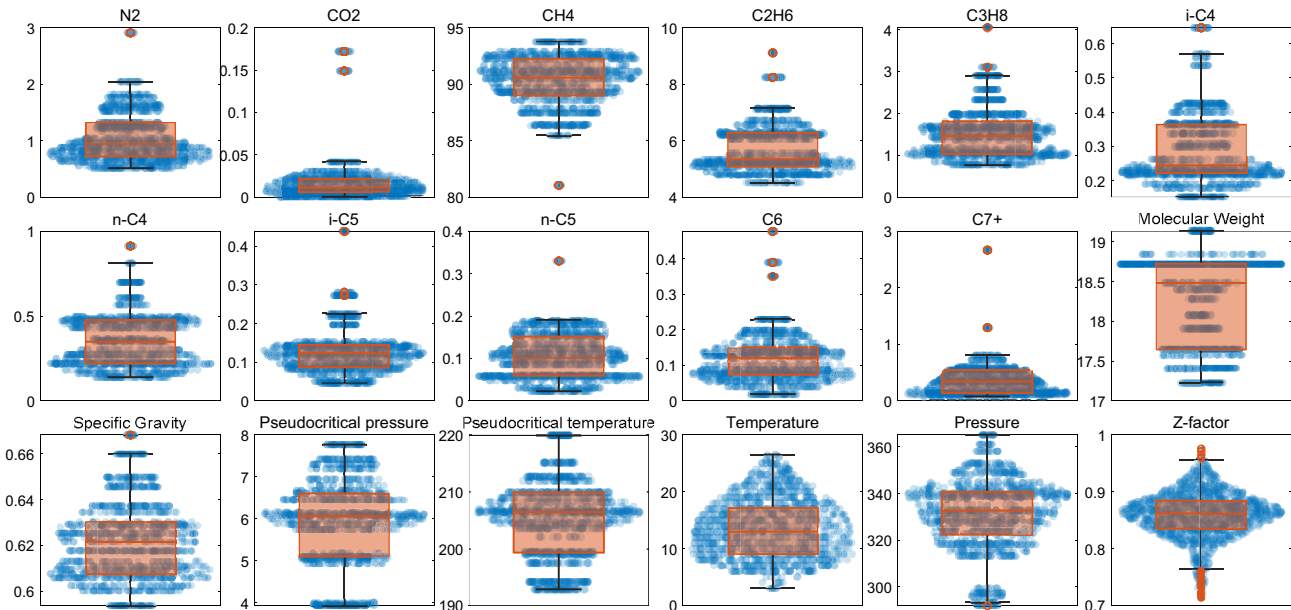


Figure 1. Boxplot of components and properties of natural gas.

Components	Min value (mol%)	Max value (mol%)	Average value (mol%)	Standard deviation	Skewness	Kurtosis
N ₂	0.239	2.91	1.077	0.49	1.284	4.865
CO ₂	0	0.172	0.081	0.043	2.748	9.198
CH ₄	93.728	73.391	89.84	2.306	− 0.861	4.18
C ₂ H ₆	4.512	10.4	5.955	0.971	1.118	4.063
C ₃ H ₈	0.772	7.253	1.668	0.621	1.25	4.916
i-C ₄	0.152	2.283	0.344	0.121	1.218	3.917
n-C ₄	0.0518	2.788	0.397	0.167	0.859	3.552
i-C ₅	0.045	1.23	0.149	0.066	1.841	8.241
n-C ₅	0.022	0.779	0.113	0.056	0.913	4.755
C ₆	0.017	0.574	0.12	0.084	1.895	7.675
C ₇ ⁺	0.00562	1.293	0.312	0.338	3.818	25.07
Molecular weight	17.222	23.721	18.383	0.599	− 0.357	1.543
Specific gravity	0.594	0.818	0.627	0.017	0.538	2.674
Pseudocritical pressure (MPa)	2.685	11.917	5.497	1.045	− 0.341	2.454
Pseudocritical temperature (K)	180.339	272.3	202.557	6.722	− 0.04	2.634
Temperature (K)	284.5	377.5	335.542	5.153	0.289	2.265
Pressure (MPa)	1.6	30.35	15.7	15.245	− 0.486	3.284
Z-factor	0.676	1.009	0.876	0.041	− 0.546	3.917

Table 1. The mole fraction and gas properties of natural gases used for modelling.

the underlying source rocks Xujiahe Formation through large faults³⁰. Influenced by the gas accumulation, trapping conditions and gas migration, there are significant differences in the components of natural gas at different locations. Among the sampled gas wells, 88% contain wet gas, while 12% contain condensate gas. We collected a total of 6914 sets of Z-factors with different components at different temperature and pressure. All Z-factors were obtained through CVD experiments.

The statistical summary of natural gas components and properties in the dataset is presented in Table 1 and Fig. 1. The hydrocarbon composition of natural gas is predominantly methane, with methane content ranging from 73.391 to 93.728%. Ethane content varies from 4.512 to 10.4%. The dryness coefficient ranges from 0.858 to 0.944, indicating predominantly wet gas³⁰. Additionally, the natural gas contains around 1% nitrogen, approximately 0.08% carbon dioxide, and around 0.3% C₇⁺ components.

Machine learning models

In this section, we briefly introduced the main principles of machine learning algorithms used in this study. This article primarily employs five regression algorithms: SVM, XGBoost, LightGBM, ANN and BiLSTM. The

solutions and implementations of the models can be found in most machine learning library such as Pytorch, Keras, TensorFlow, and MATLAB³¹. However, all the models in this paper are implemented under the MATLAB 2023a library (<https://www.mathworks.com>)³².

SVM

SVM is a supervised learning algorithm that includes Support Vector Regression (SVR) for regression. It demonstrates good robustness and adaptability in addressing non-linear regression problems. In regression problems, the goal is to fit the data and find a function that minimizes the error between predicted values and actual values. The loss is zero only when the actual value $f(x)$ and the predicted value y are exactly the same. In the case of SVR used for regression problems, the objective is to find a hyperplane that minimizes the distance between the training data points and this hyperplane within a certain tolerance ϵ . In other words, the loss is only calculated when $|f(x) - y| > \epsilon$. It is equivalent to constructing a band with a width of 2ϵ centered around $f(x) = \omega \cdot x + b$, as illustrated in Fig. 2a. If the training samples fall within the interior of this band, the prediction is considered correct, resulting in no loss³³.

SVR finds the optimal hyperplane by minimizing a loss function with a tolerance threshold and regularization term. Its optimization objective can be formulated as a minimization problem with a penalty term. Typically, a loss function with L1 or L2 regularization is employed to control the model's complexity³⁴. The minimized objective function can be expressed as:

$$\text{SVM_Loss} = \min_{\omega, b} \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^n L_{\epsilon}(y_i, \omega \cdot x_i + b) \quad (1)$$

where ω is the normal vector of the hyperplane; i is the data index; n is the number of training samples; C is the regularization parameter; L_{ϵ} is the loss function; y_i is the true label; b is the margin of the “band”, which serves as the penalty term and is typically a penalty for model complexity.

XGBoost

XGBoost is an ensemble learning method based on the gradient boosting framework. Its fundamental learner is a decision tree, and by combining multiple base learners, each iteration iteratively corrects the errors of the previous round's model, thus constructing a robust model. The difference between XGBoost and random forest lies in how the models are trained. In each step, XGBoost adds a tree based on the previous step, and the newly added tree is designed to address the deficiencies of the preceding one. Its basic structure is illustrated in Fig. 2b. Due to XGBoost incorporating the complexity of tree models into the regularization term, it excels in preventing overfitting and enhancing generalization performance as well³⁵.

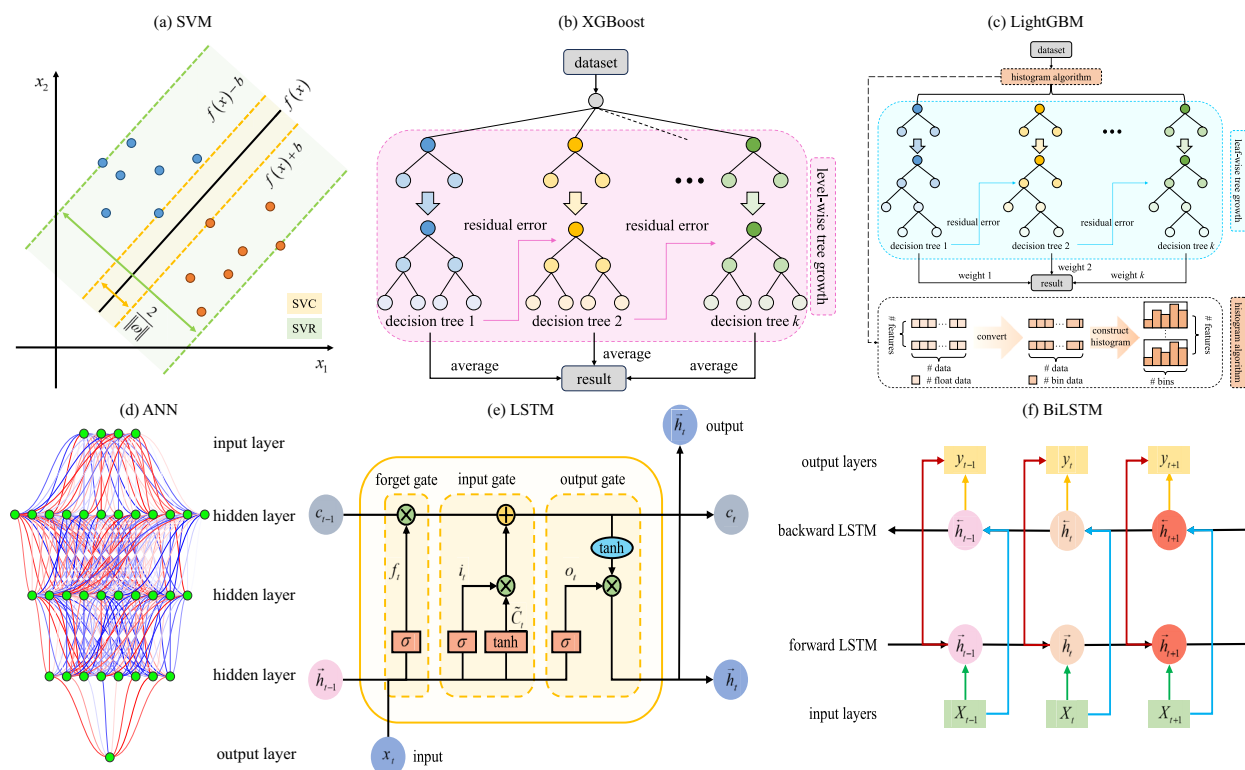


Figure 2. Structure of models used in this study.

XGBoost trains the model by minimizing the sum of weighted residuals. The loss function includes terms for prediction errors and regularization. The objective function of XGBoost is to minimize the loss function, and its general form is typically expressed as:

$$XGBoost_Loss = \sum_{i=1}^n L(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k) \quad (2)$$

where n is the number of training samples; \hat{y}_i is the predicted value; $L(y_i, \hat{y}_i)$ is typically the Mean Squared Error (MSE) loss function; $\Omega(f_k)$ is the regularization term for decision tree f_k , used to control model complexity and prevent overfitting.

LightGBM

LightGBM is similar to XGBoost, both being machine learning models based on the gradient boosting mechanism, as illustrated in Fig. 2. While XGBoost can precisely find split points, it incurs significant computation and memory overhead and is prone to overfitting. LightGBM utilizes a learning method optimized with histograms, adopts a leaf-wise growth strategy, and supports feature parallelism, reducing the number of candidate split points and memory usage, enhancing computational speed, and effectively preventing overfitting³⁶.

During the training process, the LightGBM algorithm differs from traditional Gradient Boosting Decision Trees (GBDT) algorithms by not requiring multiple passes through the entire dataset. Instead, it utilizes residuals, using the previous round of training as the input for the subsequent round. LightGBM primarily splits decision trees through histograms, employing an intelligent leaf-wise growth strategy based on depth constraints to compute, dividing feature values into discrete intervals and then constructing histograms. LightGBM employs a leaf-wise growth strategy with depth limitations, replacing the level-wise growth strategy typically used in GBDT algorithms. In comparison to XGBoost level-wise growth strategy, LightGBM leaf-wise growth strategy reduces the computational cost of finding the optimal split points, lowering the overall computational burden. This efficiency makes LightGBM more memory-efficient and mitigates the risk of overfitting³⁷.

Feature parallelism can effectively handle high-dimensional data, with each processing unit focusing on only a subset of features. LightGBM follows a parallel computing strategy, dividing the dataset into several nodes, where different nodes process features in parallel, and then model parameters are updated by merging the results from each node. Additionally, LightGBM supports cross-validation to evaluate model performance and provides early stopping functionality, which halts training when the performance on the validation set no longer improves to prevent overfitting. In practical use, LightGBM iteratively optimizes the tree structure and requires careful adjustment of hyperparameters based on the dataset and learning objectives to achieve optimal performance.

ANN

ANNs are suitable for handling complex nonlinear regression problems and can be adapted to specific tasks by adjusting the network structure and parameters. An artificial neural network consists of three main parts: the input layer, the hidden layers, and the output layer⁵. Each layer's neurons are connected by weights. The neurons in the input layer are used to receive input data from external sources. The hidden layers, located between the input and output layers, can consist of one or more layers. Each hidden layer is composed of multiple neurons, which are connected by weights. The main role of the hidden layers is to extract features. The output layer is the final layer of the network and is used to produce the final output variable in regression tasks. ANNs utilize multiple hidden layers and activation functions to capture the nonlinear relationships in the data, and they prevent overfitting through standardization and validation sets. This study employs a neural network with three hidden layers: the first layer has 12 neurons, the second layer has 10 neurons, and the third layer has 8 neurons, as shown in Fig. 2d. All hidden layers use the ReLU activation function, which is very effective in handling nonlinear problems and can help mitigate the vanishing gradient problem. The ReLU activation function is defined as follows:

$$ReLU(x) = \max(0, x) \quad (3)$$

BiLSTM

LSTM and BiLSTM are effective in capturing long-term dependencies in sequential data. Originally designed for time series prediction problems, both are frequently utilized in regression problems due to their powerful feature extraction capabilities. Derived from recurrent neural networks (RNNs), LSTM introduces improvements to the structure of the hidden layer on the traditional RNN architecture of input, hidden, and output layers. It achieves this by introducing gating mechanisms to control the information flow, selectively retaining or discarding information through these gates. This ultimately alleviates the issues of gradient vanishing or exploding caused by the derivative multiplication, effectively enhancing the memory capacity of deep learning networks³⁸. The basic architecture of an LSTM single hidden layer includes forget gate, input gate, and output gate, as described in Fig. 2e.

BiLSTM is an improvement of LSTM, and its hidden layer is composed of both forward LSTM and backward LSTM, considering both past and future information of the data. Its structure is shown in Fig. 2f, and the BiLSTM memory network has two-directional transmission layers. The forward layer follows the forward training time series, the backward layer follows the backward training time series, and both the forward and backward layers are connected to the output layer³⁹.

LSTM controls the selection or forgetting of data information through the forget gate, deciding which effective information in the training data is used for training. The amount of information to be stored in the memory

unit is determined by the input gate. The input gate is used to control the amount of current input data x_t flowing into the memory unit. The output gate controls the influence of the memory unit c_t on the current output value h_t^{40} . Their formulas are:

$$f_m = \sigma(W_m[h_{t-1}, x_t] + b_m) \quad (4)$$

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \quad (5)$$

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (6)$$

$$h_t = o_t \tanh(c_t) \quad (7)$$

where f_m is the state of the forget gate, the closer its value is to 0, the more forgotten information; σ is the Sigmoid function; W and b are the weight matrix and bias term of the forget gate, input gate, and output gate; h_{t-1} is the result of the previous output; x_t is the information input at the current time t ; i_t is the state of the input gate, the higher its value, the higher the importance of new information, and unimportant information will be deleted. o_t is the state of the output gate, the closer its value is to 0, the more external state h_t can obtain more information from the memory unit c_t .

BiLSTM consists of two LSTM networks in different directions. Input information is passed to the two-directional LSTM networks, and the input sample signal is output from the forward LSTM layer as h_t and from the backward LSTM layer as \bar{h}_t . The output y_t is jointly determined by the outputs of the LSTM from both directions, and its update formula is as follows,

$$\begin{cases} \vec{h}_t = \text{LSTM}(x_t, \vec{h}_{t-1}) \\ \overleftarrow{h}_t = \text{LSTM}(x_t, \overleftarrow{h}_{t-1}) \end{cases} \quad (8)$$

$$y_t = \vec{W} \vec{h}_t + \overleftarrow{W} \overleftarrow{h}_t + b_y \quad (9)$$

where \vec{W} and \overleftarrow{W} are the weight matrix from the forward and backward LSTM layer to the output layer, respectively; b_y represents the bias matrix of the output layer.

Decoupling methods

The method of decoupling Z-factors mainly employs signal decomposition algorithms. The primary purpose of signal decomposition algorithms is to decompose complex signals into different components for a better understanding of the structure, nature, and characteristics of the data. Utilizing signal decomposition algorithms to decouple Z-factors helps extract useful information from the Z-factors, remove noise, reveal patterns hidden in the data, and provide a foundation for further analysis and processing. In the process of decoupling Z-factors, we mainly use three signal decomposition methods: VMD⁴¹, EFD⁴², and EEMD⁴³. In subsequent studies, we compared the performance of the three hybrid models VMD+SVM, EFD+SVM and EEMD+SVM on the testing set, VMD was selected due to its best performance. Therefore, the usage process and principle of VMD is mainly introduced. The other two algorithms with poor performance are used for comparison, which will not be described in detail. The workflows of other signal decomposition methods are similar, and more detailed explanations about all decomposition methods can be found in the references cited above.

The specific implementation steps of using VMD to decouple Z-factors are as follows: (1) Establish a variational model. Hilbert transform is used to decompose the original data $f(t)$ into K components. Each component has a finite bandwidth around the center frequency. Then, data is shifted to baseband and mix the center frequency to estimate the signal bandwidth, ensuring that the sum of the estimated bandwidths of each mode is minimized, and the sum of all components is equal to the original signal. The constructed variational function model is as follows:

$$\begin{cases} \min_{\{u_k\}, \{\omega_k\}} \left\{ \sum_{i=1}^K \left\| \partial_t \left[\left(\delta(t) + \frac{j}{\pi t} \right) * u_k(t) \right] e^{-j\omega_k t} \right\|_2^2 \right\} \\ \text{s.t. } \sum_{i=1}^K u_k = f(t) \end{cases} \quad (10)$$

where u_k and ω_k are, the i -th modal component and central frequency after decomposition, respectively; ∂_t is the partial derivative with respect to time t ; $\delta(t)$ is the Dirac function; j is the imaginary unit; $u_k(t)$ is the modal function.

(2) Solve the optimal solution of the variational model. Lagrangian multiplier operator $\lambda(t)$, Lagrangian parameter λ , and quadratic penalty parameter α are introduced for solving. Equation (11) is transformed into the augmented Lagrangian function:

$$L(\{u_k\}, \{\omega_k\}, \lambda) = \alpha \sum_{i=1}^K \left\| \partial_t \left[\left(\delta(t) + \frac{j}{\pi t} \right) * u_k(t) \right] e^{-j\omega_k t} \right\|_2^2 + \left\| f(t) - \sum_{i=1}^K u_k(t) \right\|_2^2 + \left\langle \lambda(t), f(t) - \sum_{i=1}^K u_k(t) \right\rangle \quad (11)$$

where α is the penalty parameter; $\lambda(t)$ is the Lagrange multiplier.

(3) Alternate direction method of multipliers is used to solve u_k^{n+1} , ω_k^{n+1} and λ^{n+1} .

$$u_k^{n+1}(\omega) = \frac{f(\omega) - \sum_{i \neq k} u_i(\omega) + \frac{\lambda(\omega)}{2}}{1 + 2\alpha(\omega - \omega_k)^2} \quad (12)$$

$$\omega_k^{n+1} = \frac{\int_0^\infty \omega |u_k(\omega)|^2 d\omega}{\int_0^\infty |u_k(\omega)|^2 d\omega} \quad (13)$$

$$\lambda^{n+1}(\omega) = \lambda^n(\omega) + \tau \left(f(\omega) - \sum_k u_k^{n+1}(\omega) \right) \quad (14)$$

where n is the iterations; τ is the time step of the dual ascent.

(4) Repeat procedure in step (3) until solution satisfies the convergence condition. Then K components with limited bandwidth are obtained,

$$\sum_i^K \left(\|u_k^{(n+1)} - u_k^n\|_2^2 \right) / \|u_k^n\|^2 < \varepsilon \quad (15)$$

where ε is the convergence condition.

For the Z-factor with K modal components, the original Z-factor is obtained by summing up each component.

Snow ablation optimizer

When decoupling the Z-factor into multiple components using signal decomposition algorithms, it is necessary to set an appropriate number of decompositions K . At the same time, influenced by the decomposition algorithm, a large value of α may cause the loss of frequency band information, while a small value may lead to information redundancy. Therefore, it is necessary to determine the optimal parameter combination $[K, \alpha]$. There are two methods to determine the number of Z-factor decompositions. One is the manual parameter tuning method, where different K values are tried from small to large. This method is random and time-consuming^{44,45}. Another method is to use metaheuristic optimization algorithms for automatic optimization. The optimal values of the $[K, \alpha]$ combination were selected metaheuristics in this study. In the outer layer of the entire machine learning model, metaheuristic optimization algorithms, such as genetic algorithms, particle swarm optimization algorithms, etc., can be used as model parameter optimization algorithms⁴⁶. Here, we use the Snow Ablation Optimizer (SAO) as the method for optimizing model parameters. The SAO algorithm is a heuristic optimization algorithm that simulates the process of snow melting to solve optimization problems. This algorithm achieves the search process by adaptively adjusting temperature and melting speed and gradually finding the optimal solution through iterative optimization. The advantages of the SAO algorithm over other optimization algorithms lie in its unique dual-population mechanism, efficient exploration and exploitation strategies, and flexible position update equations. These features make it exhibit better balance, search efficiency, and adaptability when dealing with complex optimization problems, especially in the case of multi-peak and high-dimensional problems⁴⁷. The performance of the hybrid model in this study is mainly affected by the number of decompositions K , and each K corresponds to an α . α has a little impact on the model performance and is therefore ignored during the optimization process. The goal of SAO optimization is to find K with the smallest RMSE within a defined iteration. The SAO optimization process takes K as the main optimization variable, and its search range is $[1, 20]$. The results section below discusses the impact of the decompositions number on model performance. Finally, it is found that when the number of decompositions reaches 11, the performance of all models hardly increases, so the K value is 11 in this study.

Evaluation metrics

Root Mean Square Error (RMSE) and the adjusted coefficient of determination (R^2) are primarily employed to evaluate the model performance in this paper. A smaller RMSE indicates more accurate predictions of Z-factors by the model. A higher R^2 signifies better model performance⁴⁸. Their formulas are as follows:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (Z_i - Z_{pred})^2} \quad (16)$$

$$r^2 = 1 - \frac{\sum_{i=1}^N (Z_i - Z_{pred})^2}{\sum_{i=1}^N (Z_i - \bar{Z}_{pred})^2} \quad (17)$$

$$R^2 = 1 - \frac{(1 - r^2)(N - 1)}{N - m - 1} \quad (18)$$

where N represents the number of Z-factor samples, Z_i represents the true Z-factor, Z_{pred} represents the predicted Z-factor, \bar{Z} represents the sample mean Z-factor, and m represents the number of features influencing the Z-factor.

Additionally, some models also use Mean Absolute Percentage Error (MAPE) for evaluation, where a lower MAPE indicates a more perfect model. The formula is expressed as follows:

$$MAPE = \frac{100\%}{N} \sum_{i=1}^N \left| \frac{Z_{pred} - Z_i}{Z_i} \right| \quad (19)$$

The hybrid model

The conventional Z-factor correlation predicts the Z-factor by establishing a mathematical expression between the Z-factor and the pseudoreduced pressure and temperature of natural gas. Pseudoreduced pressure and temperature are defined as:

$$P_{pr} = \frac{P}{P_{pc}} \quad (20)$$

$$T_{pr} = \frac{T}{T_{pc}} \quad (21)$$

where P_{pc} and T_{pc} are pseudocritical pressure and temperature of natural gas, respectively.

This approach simplifies the difficulty of obtaining the Z-factor correlation to some extent. However, it has certain drawbacks, namely, the P_{pr} and T_{pr} for different natural gases with different components may be equal at different pressures and temperatures. Yet, their Z-factors may not be equal. Therefore, the Z-factor correlation obtained in this way is only applicable to constant component natural gas. When the component of natural gas changes, a new correlation needs to be established to predict the Z-factor. Machine learning models, to some extent, overcome this difficulty. They can establish a connection between multiple factors and the Z-factor by inputting parameters that include the composition of natural gas, temperature, pressure, and all other factors influencing the Z-factor⁹. Previous machine learning models also used P_{pr} and T_{pr} as input variables to establish Z-factor prediction models^{7,22}. This approach also faces the problem of data overlap. Considering the difficulty of obtaining the gas composition for all wells, and P_{pc} and T_{pc} of natural gas already partially contains information about different gas components. Therefore, in this paper, we employ P_{pc} , T_{pc} , P , and T as input variables to predict the Z-factor. The overall architecture of the proposed Z-factor prediction model is shown in Fig. 3. The workflow of predicting the Z-factor has four main steps:

Setp 1: Decoupling the Z-factor into several components using decomposition algorithm. For the original undecoupled data set:

$$D_{\text{original}} = \{[p_r, T_r, p, T]_{\text{Input}}, [Z]_{\text{Output}}\} \quad (22)$$

After the Z-factor is decomposed by the VMD, EFD or EEMD algorithms, k Z-factor components are obtained. The decoupled components are combined with the original data to form k data sequences, which can be expressed as:

$$\begin{cases} D_1 = \{[p_r, T_r, p, T]_{\text{Input}}, [Z_1]_{\text{Output}}\} \\ D_2 = \{[p_r, T_r, p, T]_{\text{Input}}, [Z_2]_{\text{Output}}\} \\ \dots \\ D_k = \{[p_r, T_r, p, T]_{\text{Input}}, [Z_k]_{\text{Output}}\} \end{cases} \quad (23)$$

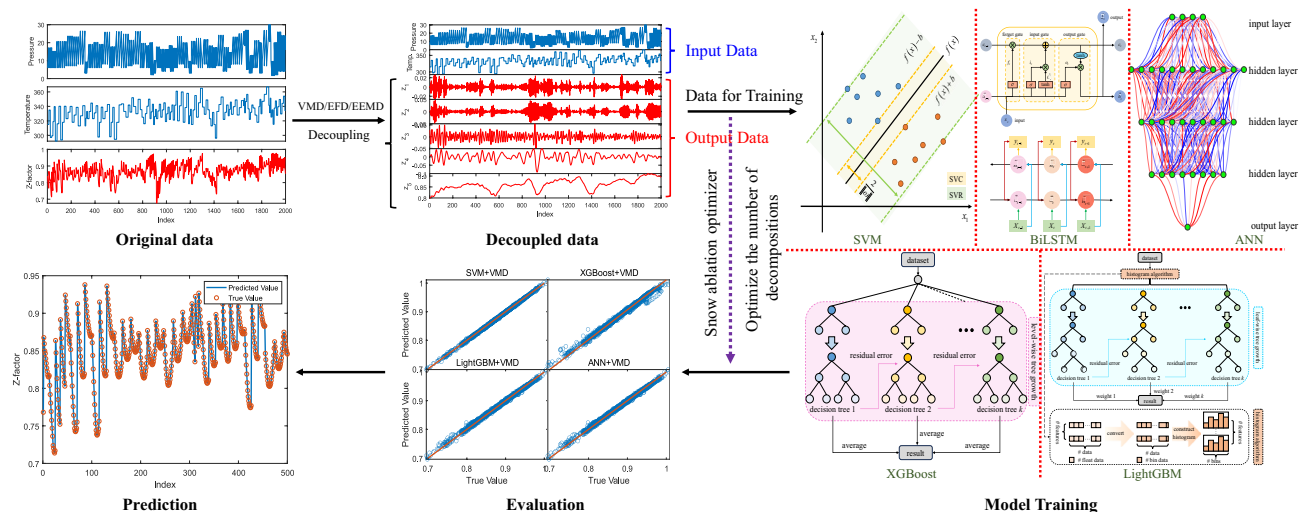


Figure 3. Proposed hybrid model for decoupling and predicting natural gas Z-factors.

Step 2: Training the hybrid model. Dividing the k data sequences into training sets and testing sets, machine learning algorithms (SVM, XGBoost, LightGBM, ANN and BiLSTM) are employed to train the models. For k data sequences, k models are trained respectively:

$$\text{Mdl}[i] = \text{train}\left([p_r, T_r, p, T]_{\text{Input}}, [Z_i]_{\text{Output}}\right), i = 1, 2, \dots, k \quad (24)$$

Step 3: Evaluating the performance of hybrid models using Eqs. (16) and (18). In this step, the hybrid model with the best performance and the optimal number of decompositions k are selected. When k changes, repeat steps 1 and 2 until the hybrid model reaches the optimal RMSE and R^2 within the defined iterations. Therefore, decompositions number k can be adjusted manually by user or automatically by SAO to achieve the optimal RMSE at this step.

Step 4: Predicting the Z-factor under various conditions using the well-trained hybrid model. After all the model is well-trained and achieves satisfactory performance in the testing set, the k well-trained models are used to predict k decoupled Z-factors:

$$Z[i] = \text{predict}\left(\text{Mdl}[i], [p_r, T_r, p, T]_{\text{Input for Prediction}}\right) \quad (25)$$

The final predicted Z-factor is obtained by adding k decoupled Z-factors:

$$Z = \sum_{i=1}^k Z[i] \quad (26)$$

The above statement briefly describes the entire prediction framework. It is easier to understand the prediction workflow by combining it with the codes. The source code of this study is available at <https://github.com/gengshaoyang/Decoupling-and-predicting-Z-factor.git>.

Results

Comparison with correlations

Initially, we compared the predictive performance of Z-factor correlations with the developed machine learning model using the base dataset. Three commonly employed correlations were selected, among which the correlation proposed by Papay can be directly calculated through T_{pr} and P_{pr} , while the DAK and Hall-Yarborough correlations necessitate iterative solutions^{49,50}. For machine learning, we utilized the SVM+VMD combination for prediction because this hybrid model performed best. The final results are illustrated in Fig. 4. Remarkably, the predictive accuracy of the machine learning model substantially surpassed that of correlations, yielding a correlation coefficient of 0.9982 and a MAPE of merely 0.2077%. This discrepancy arises because correlation coefficients are derived by fitting the Z-factor chart of Standing and Katz⁵¹, which is solely applicable to natural gas devoid of non-hydrocarbon components such as N_2 and CO_2 ⁵². However, the natural gas in our study predominantly comprises wet gas, partly containing condensate gas, and features varying levels of N_2 and CO_2 . Numerous studies have attested that correlations become obsolete for Z-factor prediction when significant changes in gas composition occur⁹. Moreover, the greater the disparity in gas composition, the larger the error in Z-factor prediction.

Performance on different machine learning models

Next, we conducted a comparison of the predictive performance of different machine learning models, divided into two parts: comparing different regression algorithms and evaluating the impact of utilizing the

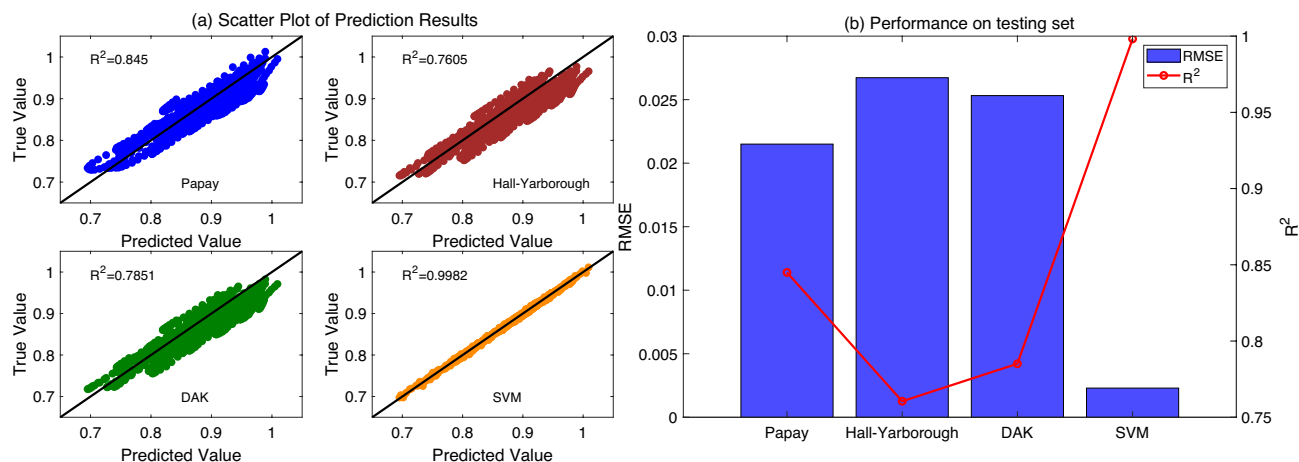


Figure 4. Prediction results comparison between traditional Z factor correlations and SVM+VMD machine learning model.

decomposition algorithm. Figure 5 displays scatter plots between predicted Z-factors and true Z-factors, as well as RMSE and R^2 values of the models. Initially, without utilizing the decomposition algorithm, ANN exhibited the best performance, while SVM showed the poorest performance. This explains why ANNs dominate the previous Z-factor prediction framework. However, upon introducing the VMD algorithm to decouple the Z-factor, the performance of all models significantly improved. Notably, after implementing VMD, the SVM model's performance notably enhanced, with MAPE decreasing from 0.65 to 0.24% and RMSE decreasing from 0.01195 to 0.002532, positioning it as the top-performing model in Z-factor prediction. This indicates the effectiveness of VMD in extracting relevant features or patterns from Z-factors that original machine learning models may not adequately capture. Integrating advanced machine learning models with VMD technology can substantially enhance the predictive performance of natural gas Z-factors. Additionally, upon scrutinizing the predictive results without Z-factor decoupling, it is evident that Z-factor predictions are generally accurate across most temperature and pressure levels, albeit with some deviations from true values. Upon investigation, it was found that these deviations primarily occurred in condensate gas data points. Since our dataset predominantly comprises wet gas (88%) and only a small portion of condensate gas (12%), the model predominantly learned about the variation patterns of wet gas Z-factors during training. Consequently, the model's generalization ability in predicting condensate gas Z-factors needs improvement. However, post-VMD implementation to decouple Z-factors, all models can more accurately capture variations in both wet gas and condensate gas Z-factors, significantly enhancing their generalization ability.

Uncovering the essence of decoupling

In the previous section, we observed a substantial enhancement in the performance of all models upon decoupling the Z-factors using VMD. The underlying mechanism driving this improvement is of significant interest. Understanding how decoupling Z-factors enhances model capability is crucial for improving model generalization, enhancing predictive performance, and extending the model's applicability to predicting Z-factors under diverse temperature, pressure, and gas component conditions. Given that natural gas consists of a mixture of several different gases, and the VMD algorithm decouples the Z-factors into multiple sub-components, it prompts the question of whether there exists a correlation between the decoupled Z-factor components and the natural gas components. To investigate this relationship, we sorted the natural gas components in ascending order based on their mole fractions and concurrently sorted the decoupled Z-factors from small to large (utilizing the example of decoupling the Z-factors into 11 components), as depicted in Fig. 6a and b. In the distribution of natural gas components, C_1 occupies the majority, while other hydrocarbons and non-hydrocarbon components constitute a smaller portion. Similarly, the decoupled Z-factors exhibit a comparable pattern, with 10 components representing a very small proportion, while the last one dominates the majority.

Following the application of the VMD algorithm to decouple the Z-factors and subsequent removal of maximum values from both the decoupled Z-factor components and the mole fractions of natural gas, we conducted regression analysis between these decoupled Z-factors and the mole fractions of natural gas, as depicted in Fig. 6c and d. Remarkably, we observed a highly positive correlation between the decoupled Z-factor components and the mole fractions of natural gas. This relationship is well-fitted with a logarithmic function, yielding correlation coefficients exceeding 0.8 for all cases. This observation substantiates the existence of an unknown

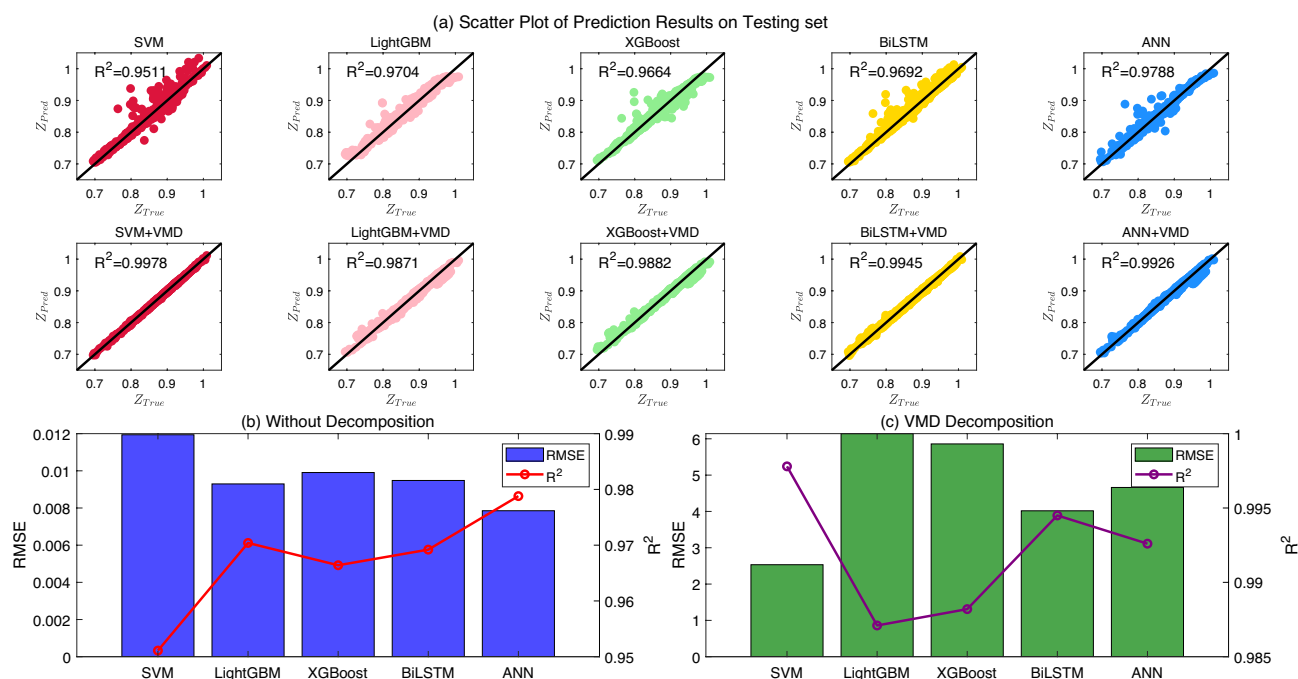


Figure 5. Prediction results comparison between different machine learning models.

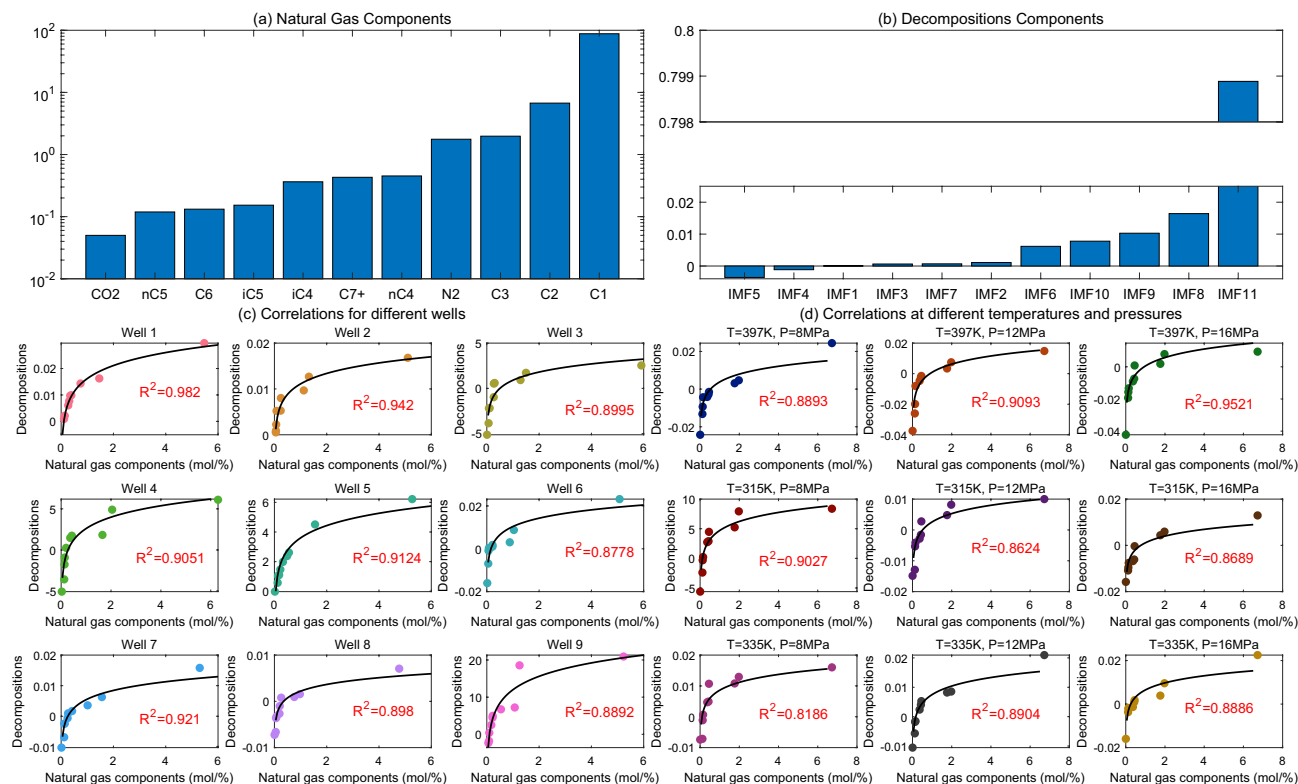


Figure 6. Correlations between gas components and decoupled decompositions. (a) Components of natural gas. (b) Compositions of Z-factor. (c) At different wells. (d) At different temperatures and pressures.

relationship between the decoupled variables and the mole fractions of natural gas. We posit that the decoupled Z-factor components signify the contribution of individual natural gas components to the overall Z-factors. The decomposition algorithm effectively segregates the influence of different gases on the Z-factors, with a higher molar fraction of a gas corresponding to a greater impact on the Z-factors. Moreover, certain decoupled variables exhibit negative values, indicative of CO₂ and heavy hydrocarbons exerting a negative influence on the Z-factors. Numerous studies have corroborated that as the content of CO₂ and heavy hydrocarbons increases, the Z-factors of natural gas tend to decrease^{14,53,54}. Following the decoupling of Z-factors, machine learning algorithms can more readily discern the variation patterns of Z-factors, thereby enhancing the performance of prediction models. In a word, the relationship between the decoupled Z-factor and gas composition underscores how the decomposition algorithm renders the initially chaotic Z-factor more orderly and predictable.

Effect of decomposition methods

We have demonstrated that using decomposition algorithms can significantly improve the ability of machine learning models to predict Z-factors. However, does this hold true for all decomposition algorithms? We compared the performance of three decomposition algorithms-EFD, EEMD, and VMD-as well as the performance without using any decomposition algorithm. All decomposition algorithms were used to decompose the Z-factors into 11 components. The scatter plots of different models on the training set and testing set are shown in Fig. 7. From the perspective of the prediction performance on the testing set, not all algorithms can effectively decouple Z-factors. Although all decomposition algorithms can eliminate the prediction anomalies, the EFD decomposition algorithm actually reduces the prediction results of Z-factors, which should have been predicted accurately. However, overall, using decomposition algorithms to decouple Z-factors first and then using machine learning algorithms ensures that the model has a relatively high lower limit of model performance, and the upper limit of model performance depends on the choice of the decoupling algorithm.

To explore the performance differences brought about by algorithms, we compared the differences of each component after decoupling Z-factors using three algorithms. We plotted the components of the 800 samples after decomposition along with their corresponding Z-factors into a heatmap, as shown in Fig. 8. The major decomposition components in the figure refer to the components with the highest content in Fig. 6a, which are relatively close to the Z-factors and are plotted together at the bottom of the figure. The remaining 10 minor decomposition components, which are relatively close in size, are plotted at the top of the figure. It is evident from the heatmap that the results obtained by the three decomposition algorithms are quite different. The minor components decomposed by EFD are different from each other, but for Z-factors at similar pressures, most components cannot completely separate them. In our tests, the model performance decreases with an increase of EFD decompositions number. The Z-factor components are complex, and the decoupling effect of EFD is relatively poor⁵⁵. The components obtained by EMD decomposition exhibit mode mixing, i.e., in different component components, there are signals with similar scales (such as components 2/3 and 4, where the differences

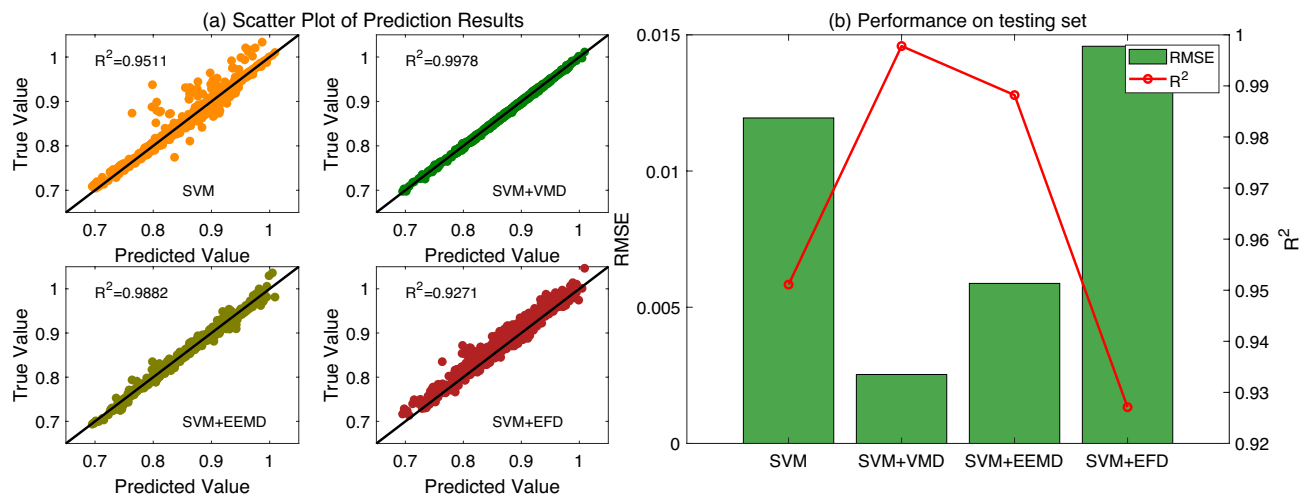


Figure 7. Prediction results comparison between different decomposition algorithms.

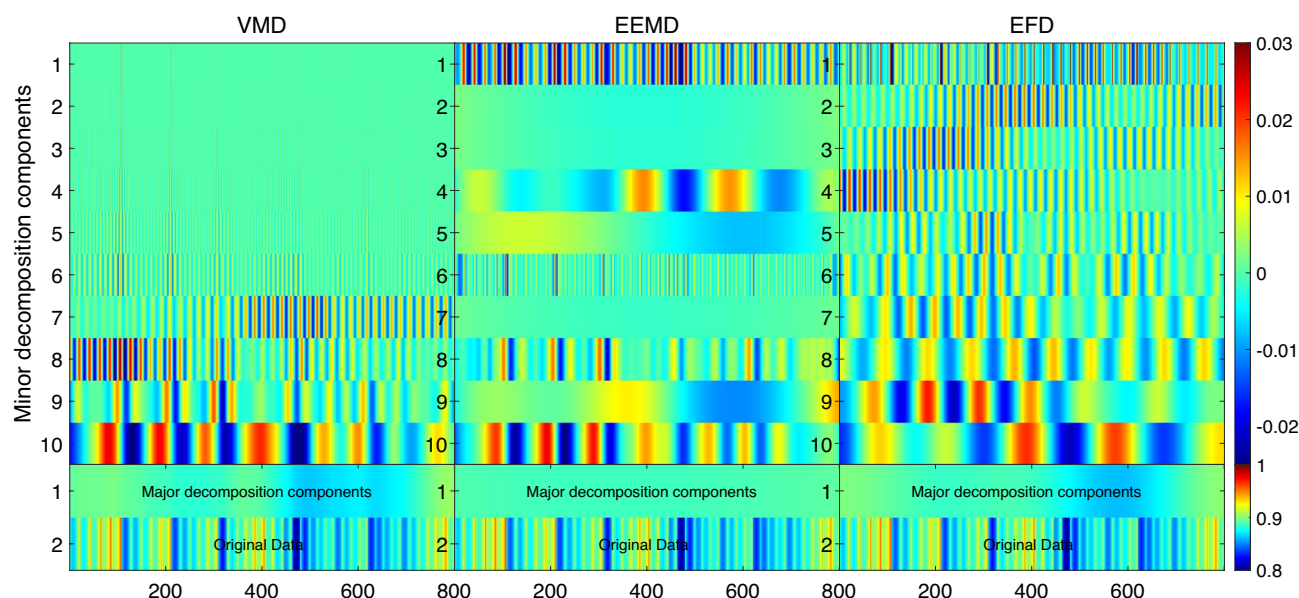


Figure 8. Heat map of the major components and minor components for different decomposition algorithms.

are hardly noticeable). VMD, with its adaptive update of the optimal center frequency and bandwidth for each component, can overcome the frequency mixing issues in EMD. For similar Z-factors, even for very small components, VMD can decompose them into several non-overlapping values (such as components 1/2 and 3), without the occurrence of frequency mixing⁵⁶.

To further compare the ability of different algorithms on decoupling Z-factors, we decomposed the Z-factors at different pressures at the same temperature and the same natural gas components, as shown in Fig. 9. The figure more accurately reflects the ability of different algorithms to decouple Z-factors. The VMD algorithm can accurately decouple Z-factors into components of different sizes at different pressures. Although EEMD and EFD can decouple Z-factors, the components decoupled by EEMD have extremely poor correlation with natural gas components. The components decoupled by EFD have a strong correlation with natural gas components but cannot distinguish Z-factors at different pressures.

Effect of decomposition number

Since natural gas is a mixture of various gases, the Z-factor is most affected by the gas components. The performance of decomposition algorithms is also influenced by the number of decomposition. Therefore, it is essential to study the impact of different decomposition numbers on the performance of the hybrid model. Based on the VMD decomposition algorithm, we trained machine learning models with decomposition numbers ranging from 0 to 14. Figure 10 depicts the heat maps of the major and minor decomposed components when the decomposition number changes from 6 to 11. When the decomposition number changes, the main decomposed components do not change much, and the overall trend of decomposition is consistent. With the increase in the

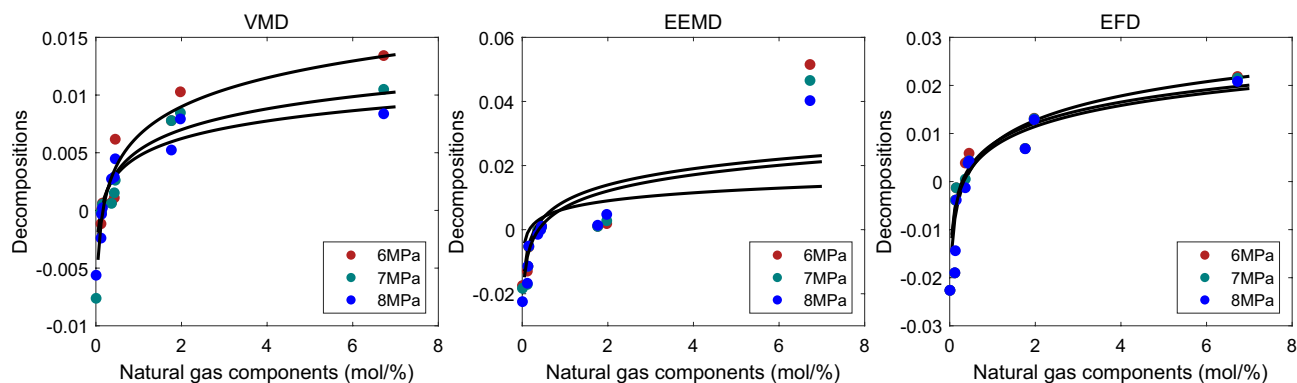


Figure 9. Decomposition components comparison for different decomposition algorithms under different pressures.

decomposition number, the additional components in the minor decomposed components are mostly small in value, and the overall trend of the other components remains unchanged, with only slight changes in numerical values. This indicates that regardless of how the decomposition algorithm changes, the main features of the Z-factors extracted by the decoupling algorithm remain unchanged, and with the increase in the decomposition number, more features are extracted by the decoupling algorithm.

We trained four regression algorithms under different decomposition numbers. Figure 11 shows the RMSE and R^2 of different models under various decomposition numbers. All models perform poorly when the decomposition number is small, especially when only one variable is decomposed, the model performance is even worse than not decomposing. When the decomposition number exceeds 6, the performance of the models generally improves significantly. When the decomposition number is 12, the model performance reaches its maximum, indicating that the decomposition algorithm has extracted all the features of the Z-factors. After that, increasing the decomposition number does not further improve the model performance. Among all the models, decoupling the Z-factors has the greatest impact on improving the performance of the SVM model.

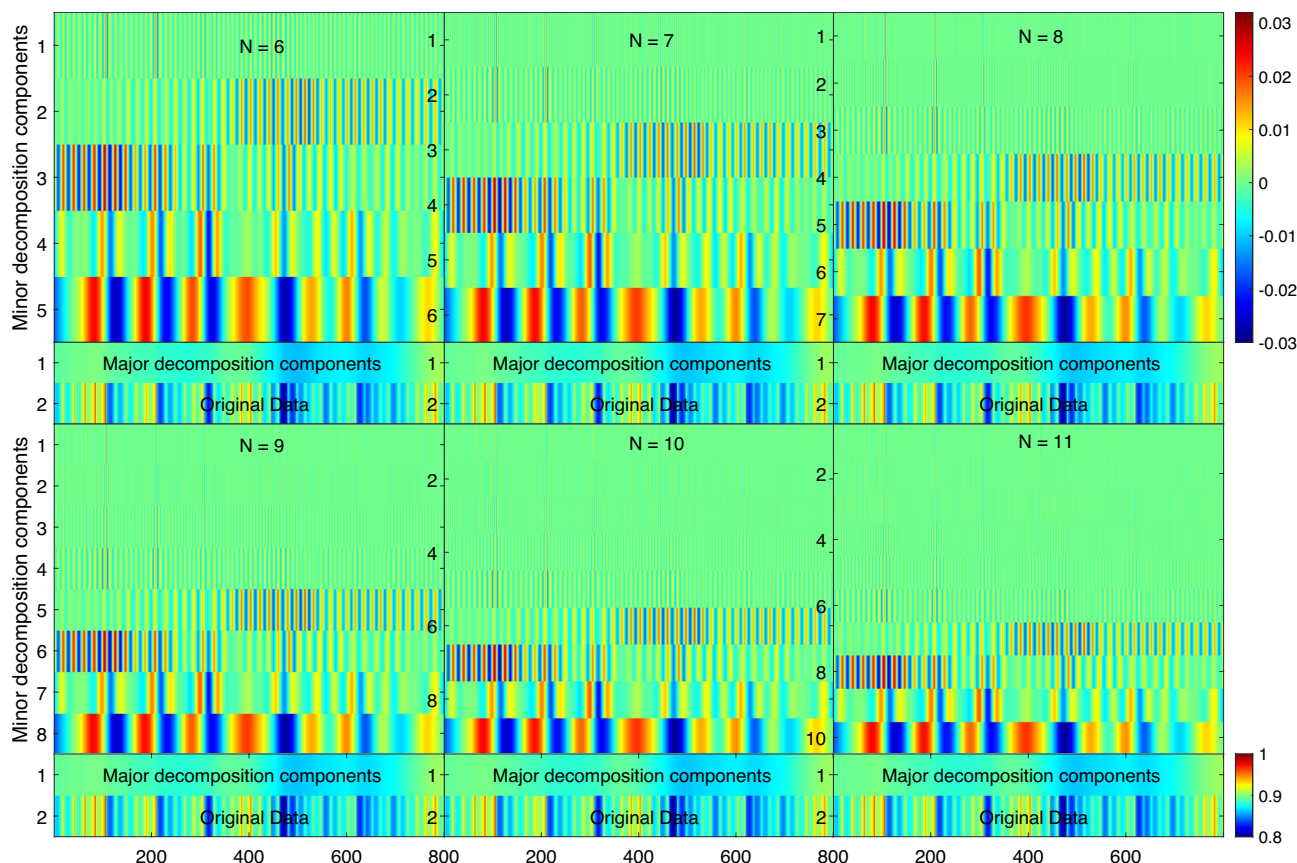


Figure 10. Heat map for the major components and minor components for different decomposition numbers.

Discussion

At this point, we have comprehended the reasons behind the performance improvement due to decoupling Z-factors and extensively discussed the impact of different decoupling methods and number on model performance. To demonstrate the superior performance of the proposed model, we employ multiple Z-factor datasets to validate our model. These datasets include Z-factors under different gas components and high temperature and pressure. Some of them involve natural gas components with higher contents of CO₂ or H₂S^{53,57}, some are related to condensate gas, and others are associated with Z-factors under high temperature and pressure⁵⁸. Some data are provided in the literature, while others are obtained by digitizing images of Z-factors recorded in the literature using OriginLab (a commercial plotting software). Furthermore, some literature only records T_{pr} and P_{pr} , and we predict Z-factors based on T_{pr} and P_{pr} . For data that provides natural gas components but lacks information on the T_{pc} and P_{pc} of natural gas, we calculate the T_{pc} and P_{pc} using the following formula⁵⁹:

$$T_{pc} = \sum_{i=1} y_i T_{ci} \quad (27)$$

$$P_{pc} = \sum_{i=1} y_i P_{ci} \quad (28)$$

For datasets with sizes smaller than 100 and containing similar natural gas components, we amalgamate them into corresponding datasets. Initially, all datasets are divided into an 8:2 ratio for separate training and prediction purposes. Subsequently, all data are merged for comprehensive evaluation. The final model evaluation results on the test sets of each dataset are presented in Table 2. The R^2 for all 10 datasets exceed 0.97, averaging 0.9918. Additionally, MAPE for all datasets generally remains below 2%, with an average of 0.83%. Datasets exhibiting lower model prediction accuracy are primarily attributed to the limited number of trainable sample points. Upon consolidating all data for training, the model achieves an impressive prediction accuracy of 0.9993, accompanied by an MAPE of merely 0.21%. The proposed model demonstrates satisfactory performance both on individual datasets and when trained on the entire dataset. Consequently, the model proposed in this paper is deemed applicable for predicting Z-factors in complex natural gas components under high pressure and high temperature conditions. Notably, on smaller datasets, the model's training and prediction time is less than one minute, while on datasets comprising more than 7,000 data points, the model requires less than 3 minutes. This indicates the model's remarkable accuracy and efficiency across varied dataset sizes.

Conclusion

In this study, we developed an innovative machine learning framework to predict the deviation factor (Z-factor) of natural gas with high accuracy. Using Z-factor data from 6914 wet gas samples from the Shaximiao Formation gas reservoir, our approach integrates signal decomposition algorithms with traditional machine learning

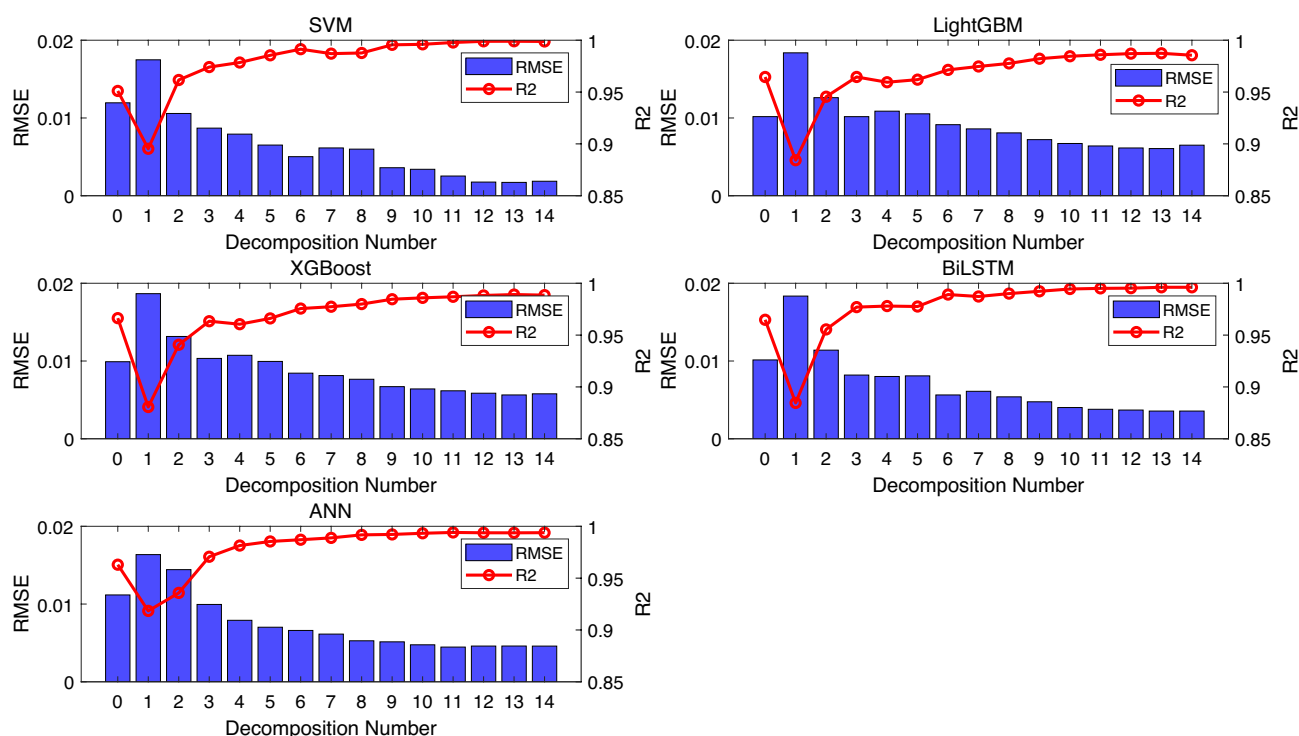


Figure 11. The effect of decomposition number on models performance.

Type of natural gas	Refs.	Gas component (mol%)				(MPa)	Average Z-factor	Dataset size	RMSE	R ²	MPAE (%)	CPU Time (s)
		H ₂ S	N ₂	CO ₂	CH ₄							
Wet gas	Ours.	0	1.08	0.021	89.85	2~30	0.876	6914	0.00253	0.9978	0.242	84.73
	51	0	0	0	80.58	/	0.916	1600	0.00261	0.9974	0.293	14.96
High CO ₂ Content	60	0	2.14	39.53	56.96	7~44	0.944	57	0.0146	0.9745	1.35	0.18
	53	0	7.32	21.4	68.93	3~59	1.054	28				
	57	0	6.44	31.86	59.44	3~60	1.032	133				
High H ₂ S Content	61	7.57	0	0	83.45	/	0.922	105	0.00639	0.9967	0.564	0.24
	/	4.38	0.63	7.28	87.36	12~80	1.08	207	0.0135	0.9909	1.06	0.21
Ultra-high pressure	6	0	0.63	0.94	93.6	10~114	1.266	70	0.0173	0.993	1.19	0.26
	62	0.36	0.88	1.44	97.17	6~100	1.283	140				
Condensate gas	63	0	0.32	11.81	76.79	/	0.893	165	0.0206	0.9802	1.25	0.18
	64	0	0.68	0.59	95.89	35~95	1.352	92	0.0192	0.992	1.13	0.13
	10	0	0.68	0.41	92.94	/	1.458	529	0.0161	0.9961	1.02	1.13
All dataset								7640	0.00367	0.9993	0.21	140.66

Table 2. Performance comparison of the proposed model on different datasets.

algorithms (SVM, XGBoost, LightGBM, ANN, BiLSTM), resulting in significant performance improvements under complex natural gas compositions, high-pressure, and high-temperature conditions. Key findings include:

1. Machine learning models surpass traditional correlations like DAK and Hall-Yarborough in predictive accuracy and ease of development for engineers.
2. The decoupled Z-factor components correlate strongly with natural gas components, enhancing predictive accuracy. VMD outperforms other decomposition methods in feature extraction.
3. Using the Variational Mode Decomposition algorithm to decouple Z-factors significantly improves performance of traditional machine learning algorithms, reducing the MAPE from 0.65% to 0.24% and the RMSE from 0.01195 to 0.002532.
4. The number of decoupled Z factors affects the prediction performance. When the number of decoupled Z factors is 11, the hybrid model of VMD and SVM achieves the best performance.

Data availability

The data and codes for this study is available at <https://github.com/gengshaoyang/Decoupling-and-predicting-Z-factor.git>.

Received: 15 January 2024; Accepted: 9 September 2024
Published online: 16 September 2024

References

1. Al-Fatlawi, O., Hossain, M. M. & Osborne, J. Determination of best possible correlation for gas compressibility factor to accurately predict the initial gas reserves in gas-hydrocarbon reservoirs. *Int. J. Hydrogen Energy* **42**, 25492–25508. <https://doi.org/10.1016/j.ijhydene.2017.08.030> (2017).

2. Heidaryan, E., Moghadasi, J. & Rahimi, M. New correlations to predict natural gas viscosity and compressibility factor. *J. Petrol. Sci. Eng.* **73**, 67–72. <https://doi.org/10.1016/j.petrol.2010.05.008> (2010).

3. de Almeida, J. C., Velásquez, J. A. & Barbieri, R. A methodology for calculating the natural gas compressibility factor for a distribution network. *Pet. Sci. Technol.* **32**, 2616–2624. <https://doi.org/10.1080/10916466.2012.755194> (2014).

4. Saghafi, H. & Arabloo, M. Development of genetic programming (gp) models for gas condensate compressibility factor determination below dew point pressure. *J. Petrol. Sci. Eng.* **171**, 890–904. <https://doi.org/10.1016/j.petrol.2018.08.020> (2018).

5. Zhai, S. *et al.* Prediction of gas production potential based on machine learning in shale gas field: A case study. *Energy Sourc. Part A: Recov. Utiliz. Environ. Effects* **44**, 6581–6601. <https://doi.org/10.1080/15567036.2022.2100521> (2022).

6. Liu, H. *et al.* Compressibility factor measurement and simulation of five high-temperature ultra-high-pressure dry and wet gases. *Fluid Phase Equilib.* **500**, 112256. <https://doi.org/10.1016/j.fluid.2019.112256> (2019).

7. Okoro, E. E., Ikeora, E., Sanni, S. E., Aimihke, V. J. & Ogali, O. I. Adoption of machine learning in estimating compressibility factor for natural gas mixtures under high temperature and pressure applications. *Flow Meas. Instrum.* **88**, 102257. <https://doi.org/10.1016/j.flowmeasinst.2022.102257> (2022).

8. Ahmed, T. Chapter 3—natural gas properties. In *Equations of State and PVT Analysis (Second Edition)* (ed. Ahmed, T.) 189–238 (Gulf Professional Publishing, addressBoston, 2016). <https://doi.org/10.1016/B978-0-12-801570-4.00003-9>.

9. Faraji, F., Ugwu, J. O. & Chong, P. L. Modelling two-phase z factor of gas condensate reservoirs: Application of artificial intelligence (ai). *J. Petrol. Sci. Eng.* **208**, 109787. <https://doi.org/10.1016/j.petrol.2021.109787> (2022).

10. Sun, C.-Y. *et al.* Experiments and modeling of volumetric properties and phase behavior for condensate gas under ultra-high-pressure conditions. *Ind. Eng. Chem. Res.* **51**, 6916–6925. <https://doi.org/10.1021/ie2025757> (2012).

11. Moiseeva, E. F. & Malyshev, V. L. Compressibility factor of natural gas determination by means of molecular dynamics simulations. *AIP Adv.* **9**, 055108. <https://doi.org/10.1063/1.5096618> (2019).

12. Faramawy, S., Zaki, T. & Sakr, A.-E. Natural gas origin, composition, and processing: A review. *J. Nat. Gas Sci. Eng.* **34**, 34–54. <https://doi.org/10.1016/j.jngse.2016.06.030> (2016).

13. Li, J. & Yu, B. Chapter one—gas properties, fundamental equations of state and phase relationships. In *Sustainable Natural Gas Reservoir and Production Engineering*, vol. 1 of *Series The Fundamentals and Sustainable Advances in Natural Gas Science and Eng* (eds. Wood, D. A. & Cai, J.) 1–28 (Gulf Professional Publishing, 2022). <https://doi.org/10.1016/B978-0-12-824495-1.00004-8>.
14. Jia, W., Li, Z., Liao, K. & Li, C. Using lee-kesler equation of state to compute the compressibility factor of co₂-content natural gas. *J. Nat. Gas Sci. Eng.* **34**, 650–656. <https://doi.org/10.1016/j.jngse.2016.07.032> (2016).
15. Li, Z., Jia, W. & Li, C. An improved pr equation of state for co₂-containing gas compressibility factor calculation. *J. Nat. Gas Sci. Eng.* **36**, 586–596. <https://doi.org/10.1016/j.jngse.2016.11.016> (2016).
16. Tariq, Z. & Mahmoud, M. New correlation for the gas deviation factor for high-temperature and high-pressure gas reservoirs using neural networks. *Energy Fuels* **33**, 2426–2436. <https://doi.org/10.1021/acs.energyfuels.9b00171> (2019).
17. He, X., Deng, R., Yang, J. & Geng, S. Adaptive material balance method for reserve evaluation: A combination of machine learning and reservoir engineering. *J. Energy Eng.* **148**, 04022018. [https://doi.org/10.1061/\(ASCE\)EY.1943-7897.0000830](https://doi.org/10.1061/(ASCE)EY.1943-7897.0000830) (2022).
18. Geng, S., Zhai, S. & Li, C. Swin transformer based transfer learning model for predicting porous media permeability from 2d images. *Comput. Geotech.* **168**, 106177. <https://doi.org/10.1016/j.compgeo.2024.106177> (2024).
19. Basha, S. M. & Rajput, D. S. An innovative topic-based customer complaints sentiment classification system. *Int. J. Business Innov. Res.* **20**, 375–391. <https://doi.org/10.1504/IJBIR.2019.102718> (2019).
20. Salem, A. M., Attia, M., Alsabaa, A., Abdelaal, A. & Tariq, Z. Machine learning approaches for compressibility factor prediction at high-and low-pressure ranges. *Arab. J. Sci. Eng.* **47**, 12193–12204. <https://doi.org/10.1007/s13369-022-06905-3> (2022).
21. Chamkalani, A., Mae'soumi, A. & Sameni, A. An intelligent approach for optimal prediction of gas deviation factor using particle swarm optimization and genetic algorithm. *J. Nat. Gas Sci. Eng.* **14**, 132–143. <https://doi.org/10.1016/j.jngse.2013.06.002> (2013).
22. Tariq, Z. *et al.* A data-driven machine learning approach to predict the natural gas density of pure and mixed hydrocarbons. *J. Energy Res. Technol.* **143**, 092801. <https://doi.org/10.1115/1.4051259> (2021).
23. Kale-Barbara-Orodu, G. K. E. & Orodu, O. D. Conventional and machine learning improved prediction of hydrocarbon density using volume-translation at high-pressure high-temperature conditions. *Energy Sourc. Part A Recov. Utiliz. Environ. Effects* **2021**, 1–14. <https://doi.org/10.1080/15567036.2021.1915433> (2021).
24. Wang, Y., Ye, J. & Wu, S. An accurate correlation for calculating natural gas compressibility factors under a wide range of pressure conditions. *Energy Rep.* **8**, 130–137. <https://doi.org/10.1016/j.egy.2021.11.029> (2022).
25. Xia, Y. *et al.* Improvement of gas compressibility factor and bottom-hole pressure calculation method for hthp reservoirs: A field case in junggar basin, china. *Energies* **15**, 145. <https://doi.org/10.3390/en1528705> (2022).
26. Vishnu, V. K. & Dharmendra, S. R. A review on the significance of machine learning for data analysis in big data. *Jordan. J. Comput. Inf. Technol.* **6**, 56 (2020).
27. Gaganis, V., Homouz, D., Maalouf, M., Khoury, N. & Polychronopoulou, K. An efficient method to predict compressibility factor of natural gas streams. *Energies* **12**, 20191. <https://doi.org/10.3390/en12132577> (2019).
28. Maalouf, M., Khoury, N., Homouz, D. & Polychronopoulou, K. Accurate prediction of gas compressibility factor using kernel ridge regression. In *2019 Fourth International Conference on Advances in Computational Tools for Engineering Applications (ACTEA)* 1–4 (2019). <https://doi.org/10.1109/ACTEA.2019.8851106>.
29. Hemmati-Sarapardeh, A. *et al.* Modeling natural gas compressibility factor using a hybrid group method of data handling. *Eng. Appl. Comput. Fluid Mech.* **14**, 27–37 (2020).
30. Zhang, B. *et al.* Compound gas accumulation mechanism and model of jurassic shaximiao formation multi-stage sandstone formations in jinjiu gas field of the sichuan basin. *Nat. Gas. Ind.* **42**, 51–61 (2022).
31. Erickson, B. J., Korfiatis, P., Akkus, Z., Kline, T. & Philbrick, K. Toolkits and libraries for deep learning. *J. Digit. Imaging* **30**, 400–405. <https://doi.org/10.1007/s10278-017-9965-6> (2017).
32. Inc., T. M. Matlab version: 9.14.0 (r2023a) (2023).
33. Chen, S., Ren, M. & Sun, W. Combining two-stage decomposition based machine learning methods for annual runoff forecasting. *J. Hydrol.* **603**, 126945. <https://doi.org/10.1016/j.jhydrol.2021.126945> (2021).
34. Chauhan, V. K., Dahiya, K. & Sharma, A. Problem formulations and solvers in linear svm: A review. *Artif. Intell. Rev.* **52**, 803–855. <https://doi.org/10.1007/s10462-018-9614-6> (2019).
35. Chen, T. & Guestrin, C. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* 785–794 (2016).
36. Ke, G. *et al.* Lightgbm: A highly efficient gradient boosting decision tree. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* 3149–3157 (2017).
37. Liu, Y., Zhu, R., Zhai, S., Li, N. & Li, C. Lithofacies identification of shale formation based on mineral content regression using lightgbm algorithm: A case study in the luzhou block, south sichuan basin, china. *Energy Sci. Eng.* **11**, 4256–4272. <https://doi.org/10.1002/ese3.1579> (2023).
38. Smagulova, K. & James, A. P. A survey on lstm memristive neural network architectures and applications. *Eur. Phys. J. Spec. Top.* **228**, 2313–2324. <https://doi.org/10.1140/epjst/e2019-900046-x> (2019).
39. Siami-Namini, S., Tavakoli, N. & Namin, A. S. The performance of lstm and bilstm in forecasting time series. In *2019 IEEE International Conference on Big Data (Big Data)* 3285–3292. <https://doi.org/10.1109/BigData47090.2019.9005997> (2019).
40. Sherstinsky, A. Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *Phys. D* **404**, 132306. <https://doi.org/10.1016/j.physd.2019.132306> (2020).
41. Dragomiretskiy, K. & Zosso, D. Variational mode decomposition. *IEEE Trans. Signal Process.* **62**, 531–544. <https://doi.org/10.1109/TSP.2013.2288675> (2014).
42. Zhou, W., Feng, Z., Xu, Y., Wang, X. & Lv, H. Empirical fourier decomposition: An accurate signal decomposition method for nonlinear and non-stationary time series analysis. *Mech. Syst. Signal Process.* **163**, 108155. <https://doi.org/10.1016/j.ymssp.2021.108155> (2022).
43. Wu, Z. & Huang, N. E. Ensemble empirical mode decomposition: A noise-assisted data analysis method. *Adv. Adapt. Data Anal.* **01**, 1–41. <https://doi.org/10.1142/S1793536909000047> (2009).
44. Yu, M. *et al.* A novel framework for ultra-short-term interval wind power prediction based on rf-woa-vmd and bigru optimized by the attention mechanism. *Energy* **269**, 126738. <https://doi.org/10.1016/j.energy.2023.126738> (2023).
45. Ouyang, J., Geng, S. & Zhai, S. An optimization model for monthly time-step drilling schedule under planned field production. *Heliyon* **10**, 569. <https://doi.org/10.1016/j.heliyon.2024.e28979> (2024).
46. Mirjalili, S. & Lewis, A. The whale optimization algorithm. *Adv. Eng. Softw.* **95**, 51–67. <https://doi.org/10.1016/j.advengsoft.2016.01.008> (2016).
47. Deng, L. & Liu, S. Snow ablation optimizer: A novel metaheuristic technique for numerical optimization and engineering design. *Expert Syst. Appl.* **225**, 120069. <https://doi.org/10.1016/j.eswa.2023.120069> (2023).
48. Zhai, S. *et al.* An improved convolutional neural network for predicting porous media permeability from rock thin sections. *Gas Sci. Eng.* **127**, 205365. <https://doi.org/10.1016/j.jgsce.2024.205365> (2024).
49. Dranchuk, P. & Abou-Kassem, H. Calculation of Z factors For natural gases using equations of state. *J. Can. Petrol. Technol.* **14**, 145. <https://doi.org/10.2118/75-03-03> (1975).
50. Hall, K. R. & Yarborough, L. A new equation of state for z-factor calculations. *Oil Gas J.* **71**, 82–92 (1973).
51. Standing, M. B. & Katz, D. L. Density of natural gases. *Trans. AIME* **146**, 140–149. <https://doi.org/10.2118/942140-G> (1942).

52. Rayes, D. G., Piper, L. D., McCain, J. W. D. & Poston, S. W. Two-phase compressibility factors for retrograde gases. *SPE Form. Eval.* **7**, 87–92. <https://doi.org/10.2118/20055-PA> (1992).
53. Lei, X., Dai, J., Chen, J., Han, X. & Lu, R. Study on influence of high co₂ content on gas deviation factor of natural gas. *J. Southwest Petrol. Univ.* **41**, 120–126 (2019).
54. Liu, H. *et al.* Study the high pressure effect on compressibility factors of high co₂ content natural gas. *J. Nat. Gas Sci. Eng.* **87**, 103759. <https://doi.org/10.1016/j.jngse.2020.103759> (2021).
55. Zheng, J., Cao, S., Pan, H. & Ni, Q. Spectral envelope-based adaptive empirical fourier decomposition method and its application to rolling bearing fault diagnosis. *ISA Trans.* **129**, 476–492. <https://doi.org/10.1016/j.isatra.2022.02.049> (2022).
56. Xu, C., Yang, J., Zhang, T., Li, K. & Zhang, K. Adaptive parameter selection variational mode decomposition based on a novel hybrid entropy and its applications in locomotive bearing diagnosis. *Measurement* **217**, 113110. <https://doi.org/10.1016/j.measurement.2023.113110> (2023).
57. Lu, R., Wang, W., Zhang, Q., Hu, L. & Chen, J. The expansion and application of deviation factor chart of super-high pressure and high co₂ gas reservoir. *J. Southwest Petrol. Univ.* **45**, 97 (2023).
58. Ghanem, A., Gouda, M. F., Alharthy, R. D. & Desouky, S. M. Predicting the compressibility factor of natural gas by using statistical modeling and neural network. *Energies* **15**, 456. <https://doi.org/10.3390/en15051807> (2022).
59. Wang, X. & Economides, M. Chapter 1—natural gas basics. In *Advanced Natural Gas Engineering* (eds. Wang, X. & Economides, M.) 1–34 (Gulf Publishing Company, 2009). <https://doi.org/10.1016/B978-1-933762-38-8.50008-3>.
60. Bian, X. & Du, Z. Experimental study on the phase behavior and fluid physical parameters of high co₂-content natural gas. *Xinjiang Petrol. Geol.* **31**, 63–65 (2013).
61. Satter, A. & Campbell, J. M. Non-ideal behavior of gases and their mixtures. *Soc. Petrol. Eng. J.* **3**, 333–347. <https://doi.org/10.2118/566-PA> (1963).
62. Deng, B. *et al.* Calculation method of deviation factor and early reserve prediction of shuangyushi ultra-deep gas reservoirs with high temperature and pressure. *Spec. Oil Gas Reserv.* **29**, 73 (2022).
63. Buxton, T. S. & Campbell, J. M. Compressibility Factors for Lean Natural Gas-Carbon Dioxide Mixtures at High Pressure. *Soc. Petrol. Eng. J.* **7**, 80–86. <https://doi.org/10.2118/1590-PA> (1967).
64. Liu, H. *et al.* Phase behavior and compressibility factor of two china gas condensate samples at pressures up to 95mpa. *Fluid Phase Equilib.* **337**, 363–369. <https://doi.org/10.1016/j.fluid.2012.10.011> (2013).

Author contributions

Writing - Original Draft and Conceptualization, S.G.; Methodology and Investigation, S.Z.; Data Curation, J.Y., Y.G and H.L.; Supervision and Writing - Review & Editing, C.L., X.L and S.L. All authors reviewed the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1038/s41598-024-72499-5>.

Correspondence and requests for materials should be addressed to X.L. or S.L.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2024