Current Science
CS 6 (1), 5898-5913 (2026)
Received 05 January 2026 | **Accepted 12 February** 2026 | **Online Available 25 February** 2026 **https://doi.org/10.52845/CS/2026-6-1-20**

Current Science

Current Science

OPEN ACCESS

ISSN (O) 2795-8639

Original Article

A Novel Method for Predicting the Fall Ratio of Kill Fluids Based on the Xgboost Model

Ruiwen Feng[1*], Yuxin Wang[2]

[1]School of Petroleum Engineering, China University of Petroleum (East China), Qingdao 266580, China

[2]Sinopec Zhongyuan Petroleum Engineering Ltd, Puyang 457001, China

***Corresponding Author: Ruiwen Feng**

Abstract:
With the advancement of oil and gas exploration into deep and ultra-deep formations, drilling operations face increasingly complex geological conditions, resulting in a higher risk of well blowouts. During a blowout, drilling fluid is rapidly expelled from the wellbore, making conventional well-killing methods ineffective. In such cases, the ability of the kill fluid to descend to the bottom of the well is critical for blowout control and operational safety.To accurately predict the fall ratio of kill fluid, this study proposes an intelligent prediction method based on the XGBoost model. A wellbore kill-fluid descent experimental system was constructed, and experiments were conducted under various gas–liquid velocity conditions. A dataset incorporating gas–liquid velocities, fluid physical properties, and rheological parameters was established and normalized for model training. Key hyperparameters of the XGBoost model were optimized to improve predictive performance.The results show that the proposed model achieves an MSE of 0.004, an RMSE of 0.066, and an MAE of 0.046, with both R^2 and EVS reaching 0.962. Compared with Random Forest, Gradient Boosted Decision Tree, Linear Regression, and Multi-Layer Perceptron models, the proposed method demonstrates superior accuracy and robustness. Prediction errors remain within 10% of experimental measurements, confirming the reliability of the model.Overall, this study provides an efficient and accurate approach for predicting the kill-fluid fall ratio, offering valuable support for well-killing operations.

Keywords: kill fluid; fall ratio; XGBoost; machine learning; well control

Introduction

With the rapid growth of the global economy, the demand for oil and natural gas resources has continued to rise across countries, driving exploration and development into deeper and ultra-deep formations (Dengfa et al. 2023). During deep and ultra-deep well drilling, operations are often confronted with complex geological conditions such as high temperature and high pressure, posing severe challenges to well control. In particular, when formation pressure exceeds the pressure within the wellbore during drilling, large volumes of formation gas may rapidly invade the wellbore, generating high-speed gas flow that carries drilling fluid to the surface. This results in the complete ejection of drilling fluid and may trigger catastrophic blowout events. To effectively suppress blowouts, it is essential to inject an appropriate kill fluid

into the wellbore when its pressure falls below the formation pressure, thereby reestablishing pressure balance between the wellbore and the formation—a process known as well-killing (Hossain and Al-Majed 2015). Evidently, the success of well-killing operations largely depends on whether the injected kill fluid can smoothly descend to the bottom of the well and form an effective hydrostatic column pressure. Therefore, accurately predicting the descent behavior of kill fluid within the wellbore is a critical step in achieving effective well control, ensuring drilling safety, and mitigating the risk of blowouts.

At present, whether the kill fluid can descend within the wellbore is typically as-sessed by comparing the gas velocity against a certain critical threshold. When the gas velocity exceeds this threshold, it is generally believed that the fluid will be fully carried out of the wellbore by the upward gas flow, potentially leading to a blowout. The determination of this critical velocity mainly relies on critical liquid-carrying flow theories. Researchers have proposed three primary theoretical approaches: droplet theory (Turner et al. 1969; Coleman et al. 1991; Nosseir et al. 2000; Belfroid et al. 2008; Zhou and Yuan 2010), liquid film theory (Barnea 1986; Luo et al. 2014; Shekhar et al. 2017), and the minimum pressure gradient theory (Zabaras et al. 1986; Yuan et al. 2013).

The droplet theory is established based on force balance analysis of individual droplets within the wellbore, where the drag force exerted by the gas equals the gravi-tational force on the droplet. Under this condition, the corresponding gas velocity is defined as the critical liquid-carrying velocity. Since droplet force depends on the droplet's area and the drag exerted by the gas, the key to this model lies in determining the drag coefficient and droplet geometry. The Turner model (Turner et al. 1969), as one of the earliest droplet-based mathematical models, investigated liquid loading in gas wells, assuming spherical droplets with fixed values for the Weber number (30) and drag coefficient (0.44). Turner also introduced a safety correction by increasing the drag coefficient by 20%, and validated the model using data from 106 field gas wells. Later, Coleman et al. (1991) argued that the uncorrected Turner model better matched data from 56 field wells. Nosseir et al. (2000) suggested that the drag coefficient is influenced by the Reynolds number, which varies with flow regime. Belforid et al. (2008) incorporated the influence of pipe inclination into the Turner model, while Zhou et al. (2010) modified the critical gas velocity formula by considering the effects of droplet concentration on coalescence and breakup.

The liquid film theory assumes that kill fluid descent is governed by the reversal of a liquid film along the pipe wall. Under low-to-medium liquid loading conditions, Barnea (Barnea 1986) proposed a liquid film model based on the theory of minimum interfacial shear stress. The critical gas velocity in this model is determined according to liquid film thickness. Luo et al. (2014) extended Barnea's model by considering the influence of pipe inclination on the distribution of maximum film thickness. Shekhar et al. (2017) further incorporated the effect of pipe inclination on the interfacial friction factor to develop an improved liquid film model.

The minimum pressure gradient theory is based on the total pressure drop char-acteristics of gas-liquid two-phase flow in the wellbore. The total pressure drop con-sists of three components: frictional loss, gravitational loss, and acceleration loss. At high gas velocities, the liquid film is thin, and frictional loss dominates, resulting in a positive correlation between total pressure drop and apparent gas velocity. At low gas velocities, the thickening of the liquid film makes gravitational loss the dominant fac-tor, and total pressure drop increases as gas velocity decreases—showing a negative correlation. Hence, a minimum pressure drop occurs at a

specific gas velocity, which corresponds to the critical liquid-carrying velocity. Zabaras et al. (1986) in their experi-mental study of annular-mist flow in vertical pipes, found that once the gas velocity exceeds this critical point, liquid is carried upward primarily as film and droplets. Yuan et al. (2013) confirmed this by conducting fluid accumulation experiments in in-clined pipes and observed that the onset of liquid loading coincides with the reversal of liquid film flow direction—also corresponding to the minimum pressure gradient point.

However, the prediction results from the three critical liquid-carrying flow theo-ries often diverge significantly under identical operating conditions. Furthermore, ex-isting theories generally assume a binary outcome: when the gas velocity is below the critical value, all kill fluid falls; when it is above, all fluid is carried upward. In reality, experimental evidence indicates that the falling behavior of kill fluid within the well-bore cannot be accurately described by a simplistic binary outcome (complete fall or complete upward transport) (Guan et al. 2021). Current research on quantitative prediction of kill fluid fall ratio remains limited. Lou et al. (2023) conducted wellbore experiments and developed a basic predictive model considering only gas and liquid velocities. Guan et al. (2024) further introduced multiple dimensionless parameters to fit experimental data and proposed a model for predicting kill fluid fall flux. However, model's accuracy still requires improvement. Limitations remain in terms of predictive precision, applicabil-ity, and robustness.

To address the challenge of accurately estimating the kill fluid fall ratio, this study conducts systematic experiments under various gas-liquid conditions and constructs a fall ratio dataset. Based on this, an artificial intelligence approach is introduced, inte-grating gas-liquid velocities, fluid physical properties, and rheological parameters. An XGBoost-based intelligent prediction model is developed. Comparative evaluations with other mainstream machine learning models and experimental validations demonstrate that the proposed model achieves superior prediction accuracy, stability, and generalization ability, offering reliable technical support for well control opera-tions in the field.

2 Experimental Systems and Methods

2.1 Experimental Conditions

In this study, a wellbore fall experiment for kill fluid was designed with the aim of obtaining fall ratio data, thereby providing a dataset for subsequent model training. Compressed air was used as the gas phase, while four different liquid phase materials were selected: laboratory tap water, 0.5 wt% CMC solution, 1.0 wt% CMC solution, and 1.5 wt% CMC solution. For ease of reference, these four liquids were designated as S1, S2, S3, and S4, respectively. All experiments were conducted under ambient temperature (25 °C) and atmospheric pressure (0.1 MPa). The physical and rheological parameters of the liquid phase materials were measured under the same conditions. The rheological properties of the CMC solutions were determined using a six-speed rotational viscometer, and their rheological curves are shown in Figure 1. The density and surface tension of the liquids were measured using a liquid densitometer and a surface tensiometer, respectively. Detailed parameters of the experimental liquids are listed in Table 1. The liquid viscosity reported herein refers to the apparent viscosity commonly used in field drilling fluid measurements. According to the API-recommended standard procedures for field testing of drilling fluids, the apparent viscosity, consistency index, and flow behavior index are calculated based on viscometer readings at 300 and 600 RPM (Hemphill et al. 1993; Recommended Practice. 2011). The corresponding formulas are as follows:

$$\mu_l = \frac{R_{600}}{2} \quad (1)$$

$$k = \frac{0.511 R_{600}}{1022^n} \quad (2)$$

$$n = 3.32 \log \frac{R_{600}}{R_{300}} \quad (3)$$

Where μ_l denotes the apparent viscosity; k represents the consistency index; n is the flow behavior index; R_{300} is the dial reading of the six-speed viscometer at 300 r/min; and R_{600} is the dial reading at 600 r/min.

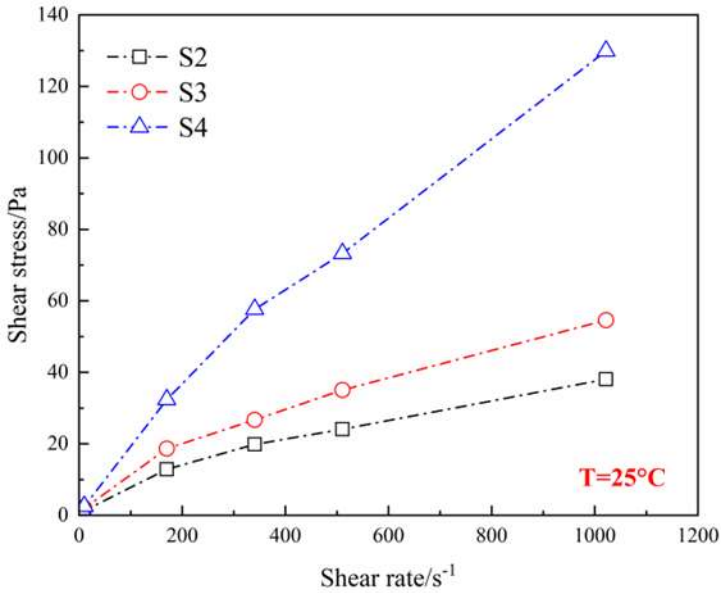


Fig.1 Rheological curves of CMC solutions at different concentrations

Table 1 Physical Parameters and Rheological parameters of experimental fluid

Name of solution	Apparent viscosity/ (mPa·s)	Density/ (kg/m^3)	Surface tension/ (N/m)	k/ (Pa·s^n)	n
S1	1	0.98	0.072	—	—
S2	37.27	1.01	0.066	0.21	0.77
S3	53.43	1.02	0.063	0.34	0.75
S4	127.14	1.04	0.059	0.37	0.85

2.2 Experimental Apparatus

The experimental apparatus, as shown in Figure 2, primarily consists of a vertical pipe, fluid and gas injection systems, and various measurement instruments. The vertical pipe is a 5.4-meter-long transparent acrylic tube with an inner diameter of 50 mm and a wall thickness of 3 mm. A liquid inlet is located at the top of the pipe, through which the liquid is injected from a supply tank via a pump. The liquid flows through an injection pipeline with an inner diameter of 30 mm and then enters a steel pipe of the same diameter, which extends vertically into the top section of the wellbore. The liquid flow rate is controlled by adjusting the pump speed. Gas is injected from the bottom of the pipe using an air compressor, with the gas flow rate regulated by a valve.

To prevent gas-liquid interaction near the top of the system from causing liquid to be ejected—thus compromising measurements—a transparent cylindrical glass shield is installed at the top of the vertical pipe. This cylinder has a height of 1.5 m, an inner diameter of 47 cm, and is inserted 1.2 m into the upper section of the pipe to serve as a protective barrier. The top of the cylinder includes a small opening vented directly to the atmosphere, serving as the gas outlet. A three-way valve is installed at the bottom of the cylinder to control the discharge of liquid. The upper valve is connected to the liquid measurement device, while the lower valve leads to a storage tank and serves as the return line for liquid circulation.

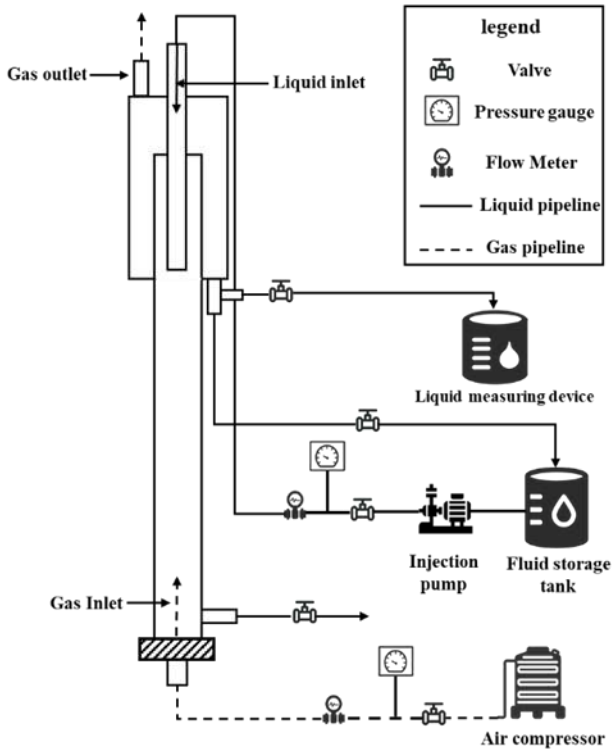


Fig.2 The Schematic diagram of experimental equipment

2.3 Experimental methods

After both gas and liquid flow rates stabilized within the pipe, real-time readings from the gas and liquid flowmeters were recorded. Simultaneously, the valve connected to the liquid measurement device was opened to collect and measure the total mass of liquid carried upward to the top of the pipe by the gas phase.

Two types of controlled-variable experimental strategies were employed:(1) Holding the liquid flow rate constant while varying the gas flow rate to investigate the fall behavior of kill fluid under different gas velocities;(2) Holding the gas flow rate constant while varying the liquid flow rate to examine fall characteristics under different liquid velocities. To minimize the impact of measurement error and enhance data reliability, multiple repeated tests were conducted under each combination of gas and liquid flow rates.

3 Model Theory and Construction

3.1 Method Overview

This study proposes a prediction method for the fall ratio of kill fluid based on the XGBoost algorithm. The overall methodological framework is illustrated in Figure 3. The specific implementation steps are as follows:

(1) First, a series of kill fluid fall experiments

were conducted using the self-designed wellbore experimental apparatus under various gas-liquid velocity combinations. Experimental data on the fall ratio of kill fluid were collected to construct a representative dataset.

(2) Next, gas-liquid velocity, fluid physical properties, and rheological parameters were selected as input features for the model. To eliminate the dimensional inconsistencies among features and enhance the stability of model training, all feature data were normalized using the Min-Max Normalization method. This also ensures a balanced contribution of each feature during the learning process.

(3) The normalized dataset was then split into a training set and a test set at a ratio of 7:3. These data were input into the XGBoost model for supervised training, resulting in an initial predictive model for the fall ratio of kill fluid.

(4) Finally, key hyperparameters such as learning rate and tree depth were fine-tuned to improve the model's fitting capability and generalization performance. The optimized model demonstrated strong prediction accuracy and provides technical support for rapid and reliable estimation of the fall ratio of kill fluid during field well-killing operations.

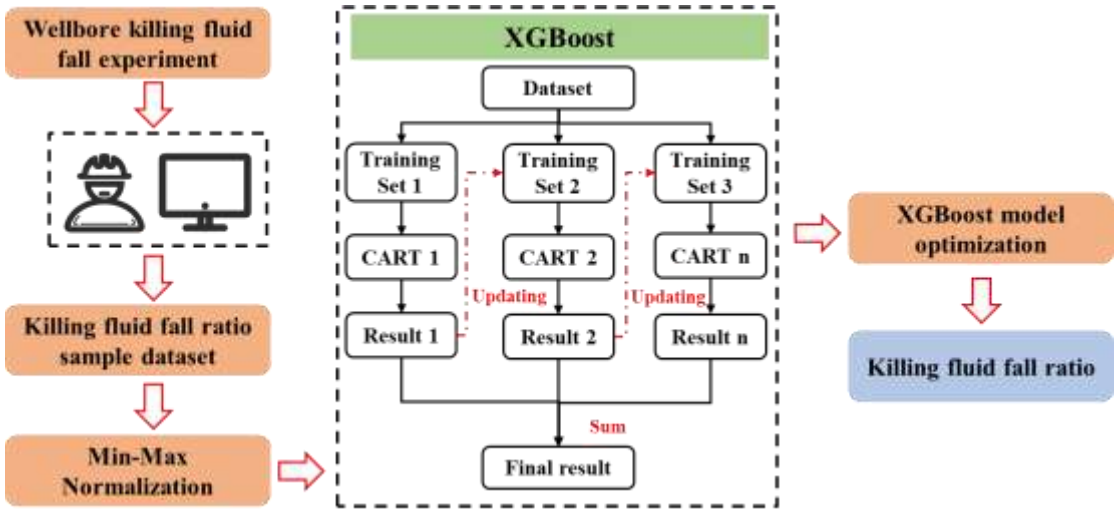


Fig.3 Framework of the XGBoost-Based Method for Predicting the Fall Ratio of Kill Fluids

3.2 XGBoost Model

XGBoost (eXtreme Gradient Boosting) is an ensemble learning method based on the gradient boosting framework. Its core structure consists of multiple CARTs (Classification and Regression Trees) (Chen et al. 2015; Dhaliwal et al. 2018; Zhang et al. 2022). During training, XGBoost constructs a series of regression trees sequentially, with each tree aiming to fit the residual errors—i.e., the difference between the actual target values and the accumulated predictions of all previously built trees. In the tree construction process, XGBoost evaluates the gain for all candidate features and split points, and selects the optimal split that maximizes the improvement of the objective function. The trees are built iteratively to generate an approximately optimal structure. Compared with traditional boosting methods, XGBoost incorporates second-order gradient information and regularization terms during tree growth, node splitting, and parameter updating. This enhances both convergence speed and generalization capability. The final prediction is obtained by aggregating the weighted outputs of all individual trees.

Assuming the training dataset is:

$$D = \{(x_i, y_i)\}_{i=1}^{n} \quad (4)$$

XGBoost approximates the target by sequentially adding base learners (regression trees). The model prediction at iteration t is expressed as:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i) \quad (5)$$

Where K denotes the number of regression trees, and f_k represents the k-th regression tree. The training objective consists of a loss function and a regularization term:

$$L = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \quad (6)$$

l(·) is the loss function, which measures the error between the predicted and true values. Ω(f) is the regularization term used to penalize model complexity, and it is defined as follows:

$$\Omega(f) = \gamma T + 0.5\lambda \sum_{j=1}^{T} w_j^2 \quad (7)$$

Where γ denotes the regularization coefficient for the number of leaf nodes, λ represents the L2 regularization coefficient for the leaf weights, and T is the number of leaf nodes in a single tree. To efficiently optimize the objective function, XGBoost performs a second-order Taylor expansion of the loss function around the current prediction:

$$l(y_i, \hat{y}_i^t) \approx l(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(x_i) + 0.5 h_i f_t^2(x_i) \quad (8)$$

Where g_i and h_i are the first- and second-order gradients, respectively:

$$\begin{cases} g_i = \dfrac{\partial l(y_i, \hat{y}_i^{(t-1)})}{\partial \hat{y}_i^{(t-1)}} \\ h_i = \dfrac{\partial^2 l(y_i, \hat{y}_i^{(t-1)})}{\partial (\hat{y}_i^{(t-1)})^2} \end{cases} \quad (9)$$

The optimal leaf weight in the regularized objective is:

$$w_j^* = -\frac{G_j}{H_j + \lambda} \quad (10)$$

Where G_j and H_j are the accumulated first- and second-order gradients for leaf node j:

$$\begin{cases} G_j = \sum_{i \in I_j} g_i \\ H_j = \sum_{i \in I_j} h_i \end{cases} \quad (11)$$

Where g_i denotes the first-order derivative of the loss function with respect to the current prediction

for the iii-th sample, and h_i denotes the second-order derivative for the *i*-th sample at the current prediction.

Substituting the optimal weights into the objective yields the optimal value of the regularized objective function:

$$L_{opt} = -0.5\sum_{j=1}^{T}\frac{G_j^2}{H_j+\lambda}+\gamma T \qquad (12)$$

During tree growth, for a candidate split that divides a node *I* into left and right subsets, the split gain is defined as:

$$G_{\text{ain}} = 0.5\left[\frac{G_L^2}{H_L+\lambda}+\frac{G_R^2}{H_R+\lambda}-\frac{(G_L+G_R)^2}{H_L+H_R+\lambda}\right]-\gamma \qquad (13)$$

To prevent overfitting and overly rapid convergence, XGBoost introduces a learning rate η after each iteration. The prediction update rule is given by:

$$\hat{y}_i^t = \hat{y}_i^{(t-1)} + \eta f_t(x_i) \qquad (14)$$

3.3 Dataset Construction

In this study, a dataset of kill fluid fall ratios was constructed based on systematically collected experimental data from the wellbore fall tests. During dataset construction, key physical and rheological parameters of the gas and liquid phases involved in the falling process were selected as input features. Specifically, eight input features were used: liquid velocity, gas velocity, liquid apparent viscosity, gas apparent viscosity, liquid density, gas density, surface tension of the liquid, and pipe diameter. The corresponding output parameter (label) is the fall ratio of kill fluid under different gas-liquid flow conditions.

3.4 Normalization Processing

Due to significant differences in the units and magnitudes of the input features, directly feeding raw data into the model could result in uneven contributions of features to the loss function, thereby affecting the model's convergence efficiency and prediction performance. To mitigate the adverse effects caused by such scale disparities and to enhance the stability and generalization capability of the model, Min-Max Normalization was employed to preprocess the original dataset. This normalization method applies a linear transformation to scale each feature value into a specified range (Patro and Sahu 2015), typically [0, 1]. The mathematical expression of Min-Max normalization is shown in Equation (15):

$$x_{\text{i}}^{*} = \frac{x_i - x_{i\min}}{x_{i\max} - x_{i\min}} \qquad (15)$$

Where x_i^* represents the normalized value, x_i is the original data value, x_{imax} is the maximum value, and x_{imin} is the minimum value.

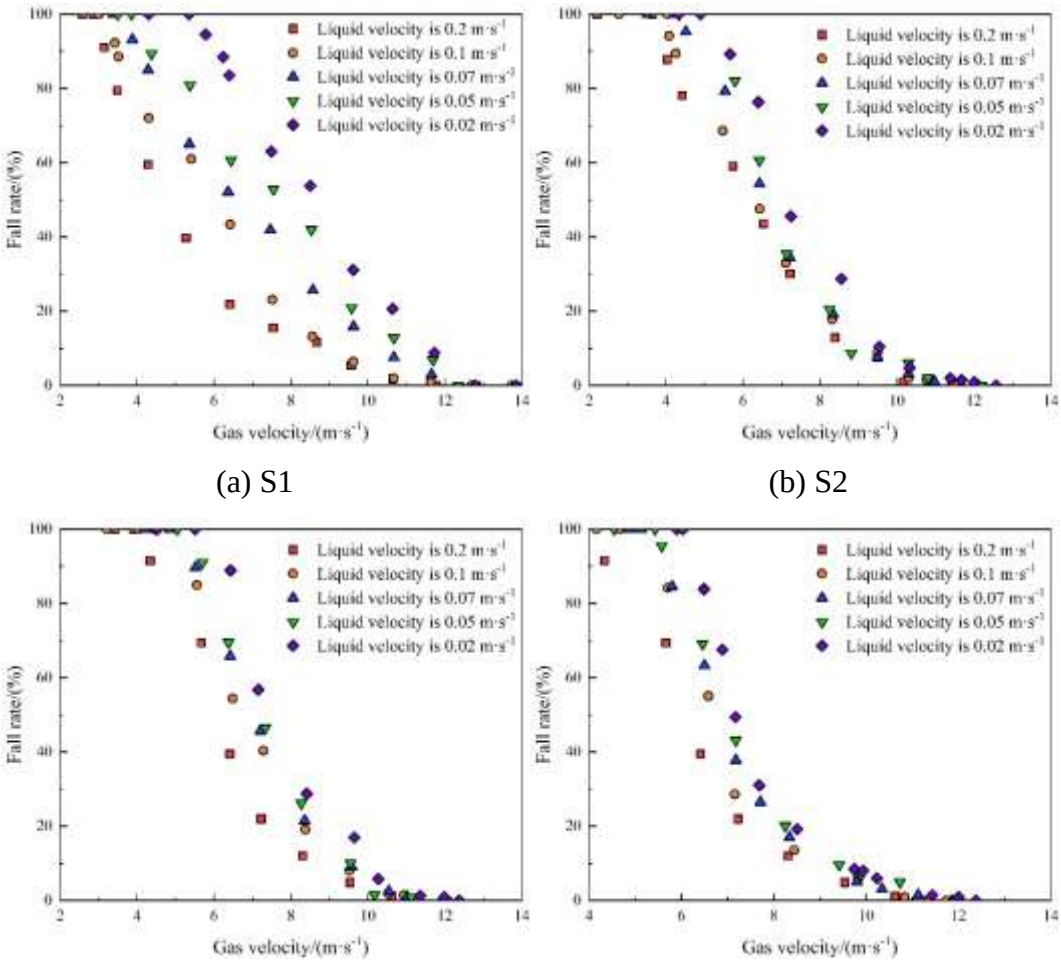
Normalization preprocessing not only improves consistency in feature sensitivity but also significantly enhances the convergence speed and stability of the XGBoost model during training. Specifically, unified feature scales facilitate more efficient gain calculations and optimal split decisions at each tree node, prevent features with larger numerical ranges from dominating the split

process, and reduce computational errors. These improvements collectively contribute to better model generalization and more accurate prediction performance (Chen and Guestrin 2016).

4 Experimental Results and Discussion

4.1 Fall Ratio of Kill Fluid

By conducting wellbore descent experiments for kill fluid under varying liquid velocities and gas-liquid conditions, the fall ratios were obtained. The experimental results are presented in Figure 4, where subfigures 4(a), 4(b), 4(c), and 4(d) correspond to the test cases S1, S2, S3, and S4, respectively. As shown in the figures, the fall ratio decreases as the gas velocity increases due to the upward transport of liquid caused by gas flow. When the gas velocity exceeds a certain critical threshold, the kill fluid begins to reverse its falling direction and is partially carried upward, resulting in a sharp decrease in fall ratio. In this transitional phase, the fall ratio becomes highly sensitive to gas velocity. As gas velocity continues to increase, most of the liquid is carried upward by the gas, and the fall ratio approaches zero, at which point the sensitivity of the fall ratio to gas velocity diminishes. Moreover, under the same gas velocity conditions, the fall ratio decreases inversely with increasing liquid velocity. This indicates that higher liquid velocities intensify the shear interaction between the gas and liquid phases, increasing the likelihood of the liquid being carried upward, and thereby reducing the effective fall ratio of the kill fluid.



(a) S1 (b) S2

(c) S3 (d) S4

Fig.4 Fall Ratio of Kill Fluids with Different Rheological Properties

4.2 Model Optimization and Evaluation

Model optimization and evaluation play a critical role in the machine learning modeling process. By appropriately adjusting hyperparameters and optimizing the model structure, the fitting capability, convergence speed, and ability of the XGBoost model to represent complex data distributions can be significantly improved. Iterative optimization on the training set helps reduce prediction errors and suppress overfitting, thereby enhancing the model's generalization performance on unseen data.

Model evaluation involves the quantitative assessment of prediction results through multiple performance metrics. These metrics comprehensively reflect the model's accuracy, stability, and robustness, providing a solid foundation for subsequent result interpretation and model selection.

4.2.1 Model Evaluation Metrics

To systematically evaluate the predictive performance of the XGBoost model, five commonly used evaluation metrics are adopted in this study: Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Coefficient of Determination (R^2), and Explained Variance Score (EVS) (Pedregosa et al. 2011; Zhou 2021). The definitions of these metrics are as follows:

(1) Mean Squared Error (MSE): MSE quantifies the average squared difference between predicted values and actual observations. It is defined as:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2 \qquad (16)$$

Where y_i denotes the true value, $\hat{y}_i$ denotes the predicted value, and n is the number of samples.

(2) Root Mean Squared Error (RMSE): RMSE is the square root of MSE and retains the same unit as the original data. It is calculated as:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2} \qquad (17)$$

(3) Mean Absolute Error (MAE): MAE represents the average absolute deviation between predicted values and true values, defined as:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \qquad (18)$$

(4) Coefficient of Determination (R^2): R^2 measures the proportion of variance in the target variable that is explained by the model. Its range is $(-\infty,1]$, and it is defined as:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{n}\left(y_i - \bar{y}\right)^2} \qquad (19)$$

Where $\bar{y}$ denotes the mean of the true values.

(5) Explained Variance Score (EVS): EVS evaluates the consistency between the variance of predicted values and that of the actual values. It is defined as:

$$EVS = 1 - \frac{Var(y - \hat{y})}{Var(y)} \quad (20)$$

These metrics provide a comprehensive evaluation of the model's fitting accuracy and generalization ability, serving as quantitative criteria for performance comparison and result analysis.

4.2.2 Model Optimization

During the construction of the XGBoost model, hyperparameter optimization plays a crucial role in enhancing prediction performance and improving the model's generalization capability. The learning rate is a key parameter that controls the impact of each newly added tree on the overall model during training. A smaller learning rate allows the model to gradually approach the optimal solution, thereby reducing the risk of overfitting; however, it may also increase the number of training iterations and computational cost. The maximum tree depth (max_depth) determines the complexity of features that a single tree can capture. A greater tree depth improves the model's ability to fit complex data patterns but increases the risk of overfitting, while a shallower depth helps maintain model simplicity and generalizability (Brownlee 2016). Therefore, appropriate tuning of the learning rate and tree depth is essential to achieving a balance between fitting accuracy and model complexity, enabling effective modeling and accurate prediction of high-dimensional feature data.

To improve model performance, this study investigates various hyperparameter combinations by selecting four values for tree depth (4, 5, 6, and 7) and four values for learning rate (0.01, 0.05, 0.1, and 0.2). The final training results under different hyperparameter settings are summarized in Table 2. The results indicate that the learning rate has a significant negative correlation with the model's convergence performance. Specifically, higher learning rates may initially accelerate convergence but often result in gradient instability, leading to higher prediction errors and reduced model stability. In contrast, lower learning rates facilitate smoother convergence during training and lead to lower prediction errors and more accurate outcomes.

After comprehensive comparison of the error metrics across all parameter combinations, the optimal configuration was determined to be a learning rate of 0.01 and tree depth of 6. Under this configuration, the model achieved its best performance, with a mean squared error (MSE) of 0.004, root mean squared error (RMSE) of 0.066, and mean absolute error (MAE) of 0.046. Additionally, the model obtained a coefficient of determination (R^2) and explained variance score (EVS) of 0.962, indicating excellent fitting and explanatory capabilities.

Table 2 Training Performance under Varying Hyperparameter Settings

Learning Rate	Max Depth	MSE	RMSE	MAE	R2	EVS
0.01	4	0.006	0.078	0.057	0.948	0.949
	5	0.005	0.072	0.051	0.951	0.952
	6	0.004	0.066	0.046	0.962	0.962
	7	0.005	0.069	0.048	0.956	0.954
0.05	4	0.008	0.099	0.075	0.926	0.926
	5	0.007	0.094	0.071	0.932	0.932
	6	0.007	0.086	0.061	0.941	0.941
	7	0.007	0.086	0.059	0.941	0.941
0.1	4	0.016	0.176	0.149	0.851	0.851
	5	0.015	0.156	0.133	0.869	0.869

	6	0.012	0.137	0.111	0.889	0.889
	7	0.012	0.137	0.109	0.888	0.887
0.2	4	0.026	0.175	0.149	0.769	0.772
	5	0.026	0.157	0.131	0.783	0.787
	6	0.024	0.155	0.128	0.791	0.792
	7	0.023	0.134	0.107	0.812	0.815

4.2.3 Evaluation of Different Regression Algorithms

To further assess the performance advantages of the proposed model, four commonly used regression algorithms—Random Forest (RF), Linear Regression (LR), Gradient Boosting Decision Tree (GBDT), and Multi-Layer Perceptron (MLP)—were selected for comparative analysis. All models were trained and tested on the same dataset to ensure fairness and consistency in the evaluation. While maintaining the overall model architecture unchanged, relevant input-layer hyperparameters for each model were appropriately adjusted to ensure optimal performance. The comparison results of model performance on the test set are presented in Table 3. The results indicate that the Linear Regression model exhibited the poorest performance, with a mean squared error (MSE) of 0.114, root mean squared error (RMSE) of 0.218, mean absolute error (MAE) of 0.175, coefficient of determination (R^2) of 0.808, and explained variance score (EVS) of 0.807. In contrast, the model proposed in this study achieved the best overall performance across all evaluation metrics, followed by the MLP model. Compared to the MLP model, the proposed model reduced MSE, RMSE, and MAE by 50%, 31.25%, and 43.90%, respectively, and improved R^2 and EVS by 3.33% and 2.45%. When compared to the LR model, the proposed model demonstrated a 19.06% improvement in R^2 and a 19.21% improvement in EVS, while reducing MSE, RMSE, and MAE by 96.49%, 69.72%, and 73.71%, respectively. These results highlight the significantly superior performance of the proposed model over the linear regression approach. In summary, the XGBoost-based fall ratio prediction model developed in this study exhibits clear advantages in both accuracy and generalization ability, offering a more precise and reliable tool for predicting the fall ratio of kill fluid during well-killing operations.

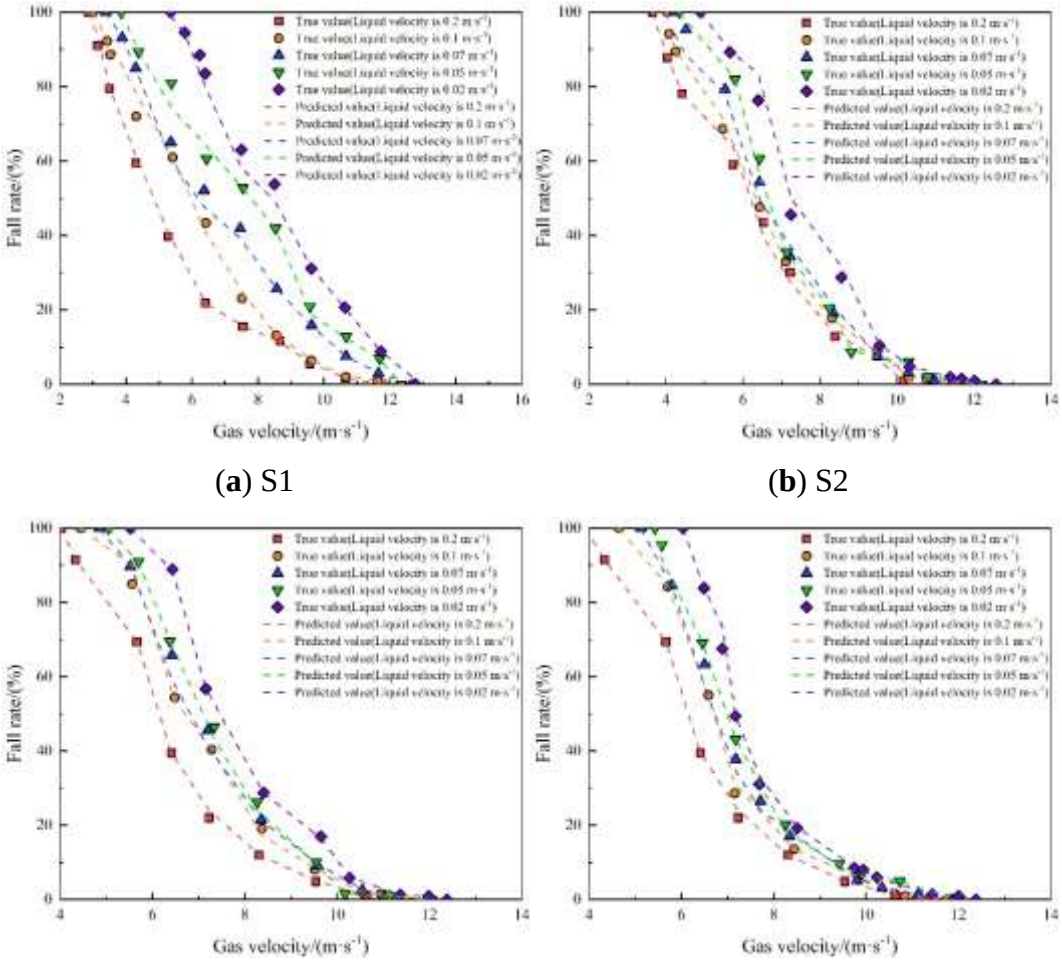
Table 3 Training Performance under Varying Hyperparameter Settings

Model	MSE	RMSE	MAE	R^2	EVS
XGBoost	0.004	0.066	0.046	0.962	0.962
RF	0.015	0.122	0.088	0.871	0.872
LR	0.114	0.218	0.175	0.808	0.807
GBDT	0.013	0.096	0.082	0.913	0.914

4.3 Model Validation and Consistency Analysis with Experimental Results

To further verify the predictive capability and generalization performance of the proposed model under actual well-killing conditions, the optimized XGBoost model was validated using the experimental dataset. Specifically, the fall ratios of the kill fluid measured under various gas-liquid velocity combinations were used as benchmarks to assess the consistency between model predictions and experimental observations. The model prediction results are shown in Figure 5. As illustrated in the figure, the predicted values closely match the experimental data, indicating that the model can accurately describe the fall

behavior of kill fluid in the wellbore under different gas-liquid flow conditions. To more intuitively evaluate the prediction accuracy of the XGBoost-based fall ratio model, an error plot comparing the predicted and measured fall ratios is presented in Figure 6, with a reference error margin of 10% indicated. It can be observed from Figure 6 that the prediction errors for all experimental samples are within 10%, demonstrating the model's high accuracy and stability. These results confirm that the proposed model can reliably predict the fall ratio of kill fluid in the wellbore, thereby offering robust technical support for subsequent well-killing operations.



(**a**) S1 (**b**) S2

(**c**) S3 (**d**) S4

Fig.5 Prediction Results of the XGBoost-Based Kill Fluid Fall Ratio Method on Experimental Data

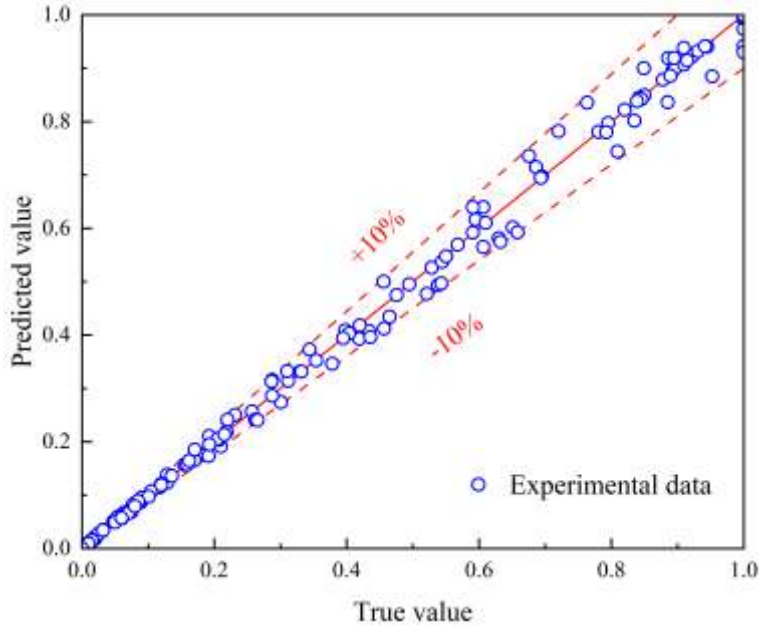


Fig.6 Prediction Error of the XGBoost-Based Kill Fluid Fall Ratio Method on Experimental Data

5 Conclusion

To address the challenge of accurately predicting the fall ratio of kill fluid during well-killing operations, this study designs and constructs a laboratory apparatus to investigate the vertical descent of kill fluid within the wellbore. A series of experiments are conducted under varying gas and liquid velocities to observe the fall behavior of kill fluid, thereby establishing a representative dataset of fall ratios under different operating conditions. Building upon this experimental foundation, an intelligent prediction method for the fall ratio of kill fluid is proposed based on the XGBoost algorithm, drawing from artificial intelligence theory. The key findings of this study are as follows:

(1) Experimental results reveal that the fall ratio decreases as the gas velocity increases. A critical gas velocity exists, beyond which the descent behavior of the kill fluid shifts, leading to partial upward movement. In this transitional stage, the fall ratio becomes highly sensitive to gas velocity. As gas velocity further increases, the motion of the kill fluid transitions into a completely upward flow, resulting in a fall ratio of zero and a decreased sensitivity to gas velocity. Furthermore, under the same gas velocity conditions, the fall ratio decreases inversely with increasing liquid velocity.

(2) During model development, key hyperparameters such as learning rate and tree depth are systematically optimized. The resulting XGBoost-based prediction model demonstrates strong performance across all evaluation metrics, achieving a mean squared error (MSE) of 0.004, root mean squared error (RMSE) of 0.066, mean absolute error (MAE) of 0.046, coefficient of determination (R^2) of 0.962, and explained variance score (EVS) of 0.962. Compared with four traditional regression models—Random Forest (RF), Linear Regression (LR), Gradient Boosting Decision Tree (GBDT), and Multi-Layer Perceptron (MLP)—the proposed model exhibits superior prediction accuracy, convergence speed, and generalization capability.

(3) Error analysis further validates the consistency between the model predictions and experimental measurements. Results indicate that the overall prediction error of the XGBoost-based fall ratio model remains within 10%, demonstrating the model's strong capability to accurately estimate the fall ratio of kill fluid under diverse wellbore

conditions. This provides a valuable technical reference for practical well-killing operations.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Author Contributions

CRediT: Ruiwen Feng: Conceptualization, Methodology, Software, Supervision, Writing – original draft; Yuxin Wang: Investigation, Data curation, Formal analysis Writing – review & editing.

Acknowledgment

The authors declare that they have no conflict of interest.

Data availability

Data will be made available on request.

Nomenclature

XGBoost	Extreme Gradient Boosting
MSE	Mean Squared Error
RMSE	Root mean squared error
MAE	Mean Absolute Error
EVS	Explained Variance Score
RF	Random Forest
LR	Linear Regression
GBDT	Gradient Boosting Decision Tree
MLP	Multi-Layer Perceptron

References

1. Barnea D. Transition from annular flow and from dispersed bubble flow—unified models for the whole range of pipe inclina-tions[J]. International journal of multiphase flow, 1986, 12(5): 733-744.
2. Belfroid S P C, Schiferli W, Alberts G J N, et al. Prediction onset and dynamic behaviour of liquid loading gas wells[C]//SPE Annual Technical Conference and Exhibition. SPE 2008: SPE-115567-MS.
3. Brownlee J. XGBoost With python: Gradient boosted trees with XGBoost and scikit-learn[M]. Machine Learning Mastery, 2016.
4. Chen T, Guestrin C. Xgboost: A scalable tree boosting system[C]//Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. 2016: 785-794.
5. Chen T, He T, Benesty M, et al. Xgboost: extreme gradient boosting[J]. R package version 0.4-2, 2015, 1(4): 1-4.
6. Coleman S B, Clay H B, McCurdy D G, et al. A new look at predicting gas-well load-up[J].Journal of petroleum technology, 1991, 43(03).
7. Dengfa H E, Chengzao J I A, Wenzhi Z, et al. Research progress and key issues of ultra-deep oil and gas exploration in China[J]. Petroleum Exploration and Development, 2023, 50(6): 1333-1344.
8. Dhaliwal S S, Nahid A A, Abbas R. Effective intrusion detection system using XGBoost[J]. Information, 2018, 9(7): 149.
9. Guan L, Li J, Yang H, et al. Research on the Falling Flux of Kill Fluid in High-Production Gas Wells[C]//International Conference on Computational & Experimental Engineering and Sciences. Cham: Springer Nature Switzerland, 2024: 779-796.
10. Guan L, Wang Z, Lou W, et al. Discrimination Model of Critical Gas Velocity in Partial Falling Area of Kill Fluid in Deepwater Blowout[C]//International Technical Symposium on Deepwater Oil and Gas Engineering. Singapore: Springer Singapore, 2021: 267-281.
11. Hemphill T, Campos W, Pilehvari A. Yield-power law model more accurately predicts

mud rheology[J]. Oil and Gas Journal;(United States), 1993, 91(34).
12. Hossain M E, Al-Majed A A. Fundamentals of sustainable drilling engineering[M]. John Wiley & Sons, 2015.
13. Lou W, Wang Z, Zhang J, et al. Flow regime evolution mechanism and falling flux prediction model for bypass injection of viscous liquid in vertical T-junction[J]. Chemical Engineering Journal, 2023, 476: 146601.
14. Luo S, Kelkar M, Pereyra E, et al. A new comprehensive model for predicting liquid loading in gas wells[J]. SPE Production & Operations, 2014, 29(04): 337-349.
15. Nosseir M A, Darwich T A, Sayyouh M H, et al. A New Approach for Accurate Prediction of Loading in Gas Wells Under Different Flowina Conditions[J]. SPE Production & Facilities, 2000, 15(04): 241-246.
16. Patro S, Sahu K K. Normalization: A preprocessing stage[J]. arXiv preprint arXiv:1503.06462, 2015.
17. Pedregosa F, Varoquaux G, Gramfort A, et al. Scikit-learn: Machine learning in Python[J]. the Journal of machine Learning research, 2011, 12: 2825-2830.
18. Recommended Practice, 1988. Standard Procedure for Field Testingdrilling Fluids In:Recommended Practice, twelfth ed., 13 B. API, Washington, USA, pp. 7–9 (RP13B). Chhabra R P, Richardson J F. Non-Newtonian flow and applied rheology: engi-neering applications[M]. Butterworth-Heinemann, 2011.
19. Shekhar S, Kelkar M, Hearn W J, et al. Improved prediction of liquid loading in gas wells[J]. SPE Production & Operations, 2017, 32(04): 539-550.
20. Turner R G, Hubbard M G, Dukler A E. Analysis and prediction of minimum flow rate for the continuous removal of liquids from gas wells[J]. Journal of Petroleum technology, 1969, 21(11): 1475-1482.
21. Yuan G., Pereyra E., Sarica C., et al. An experimental study on liquid loading of vertical and deviated gas wells[C]. SPE164516, 2013.
22. Zabaras G., Dukler A. E., Moalem - Maron D. Vertical upward cocurrent gas- liquid annular flow[J]. AIChE journal, 1986, 32(5): 829-843.
23. Zhang P, Jia Y, Shang Y. Research and application of XGBoost in imbalanced data[J]. International Journal of Distributed Sensor Networks, 2022, 18(6): 15501329221106935.
24. Zhou D, Yuan H. A new model for predicting gas-well liquid loading[J]. SPE Production & Operations, 2010, 25(02): 172-181.
25. Zhou Z H. Machine learning[M]. Springer nature, 2021.