Hindawi
Geofluids
Volume 2022, Article ID 9443955, 31 pages
https://doi.org/10.1155/2022/9443955

Research Article

Petrophysical Regression regarding Porosity, Permeability, and Water Saturation Driven by Logging-Based Ensemble and Transfer Learnings: A Case Study of Sandy-Mud Reservoirs

Shenghan Zhang,[1] Yufeng Gu,[2] Yinshan Gao,[3] Xinxing Wang,[3] Daoyong Zhang,[2] and Liming Zhou[2]

[1]*Sinopec Geophysical Research Institute, Nanjing Jiangsu 211103, China*
[2]*Strategic Research Center of Oil and Gas Resources, Ministry of Natural Resources, Beijing 100034, China*
[3]*Oil Production Plant 5, PetroChina Changqing Oilfield Company, Xi'an Shaanxi 710200, China*

Correspondence should be addressed to Yufeng Gu; aaaaa3377@126.com

Received 17 August 2022; Revised 13 September 2022; Accepted 21 September 2022; Published 5 October 2022

Academic Editor: Hailing Kong

From a general review, most petrophysical models applied for the conventional logging interpretation imply that porosity, permeability, or water saturation mathematically have a linear or nonlinear relationship with well logs, and then arguing the prediction of these three parameters actually is accessible under a regression of logging sequences. Based on this knowledge, ensemble learning technique, partially developed for fitting problems, can be regarded as a solution. Light gradient boosting machine (LightGBM) is proved as one representative of the state-of-the-art ensemble learning, thus adopted as a potential solver to predict three target reservoir characters. To guarantee the predicting quality of LightGBM, continuous restricted Boltzmann machine (CRBM) and Bayesian optimization (Bayes) are introduced as assistants to enhance the significance of input logs and the setting of employed hyperparameters. Thereby, a new hybrid predictor, named CRBM-Bayes-LightGBM, is proposed for the prediction task. To validate the working performance of the proposed predictor, the basic data derived from the member of Chang 8, Jiyuan Oilfield, Ordos Basin, Northern China, is collected to launch the corresponding experiments. Additionally, to highlight the validating effect, three sophisticated predictors, including k-nearest neighbors (KNN), support vector regression (SVR), and random forest (RF), are introduced as competitors to implement a contrast. Since ensemble learning models universally will cause an underfitting issue when dealing with a small-volumetric dataset, transfer learning in this circumstance will be employed as an aided technique for the core predictor to achieve a satisfactory prediction. Then, three experiments are purposefully designed for four validated predictors, and given a comprehensive analysis of the gained experimented results, two critical points are concluded: (1) compared to three competitors, LightGBM-cored predictor has capability to produce more reliable predicted results, and the reliability can be further improved under a usage of more learning samples; (2) transfer learning is really functional in completing a satisfactory prediction for a small-volumetric dataset and furthermore has access to perform better when serving for the proposed predictor. Consequently, CRBM-Bayes-LightGBM combined with transfer learning is solidly demonstrated by a stronger capability and an expected robustness on the prediction of porosity, permeability, and water saturation, which then clarify that the proposed predictor can be viewed as a preferential selection when geologists, geophysicists, or petrophysicists need to finalize a characterization of sandy-mud reservoirs.

1. Introduction

In the field of logging interpretation, petrophysical models are the common approach applied to address predictions of some reservoir parameters such as porosity, permeability, and water saturation, which sometimes will be unavailable or ineffective when lacking the support of some experimental data, e.g., resistivity of formation water, diameter of sand grains, and content of clay minerals [1–3]. Then, to make such parametric prediction more accessible, new solution

or computing mechanism must be introduced. From a general review of Table 1, the essence of most classic petrophysical models implies a truth that porosity, permeability, or water saturation mathematically presents a linear or nonlinear relationship with other reservoir characters, and more importantly, these characters can be directly measured by well logs or indirectly calculated by logging-based petrophysical models, which then argue that the prediction of three mentioned reservoir parameters can be completed under a regression of logging sequences [1–11]. Each computed parameter shown in the table can be comprehended via the help of the listed references. For example, since the shale will exist in disperse, structural, or laminated shape around the porous space of sandy-mud reservoirs, the real porosity actually should be determined from a rectification of the apparent porosity, and then the compaction factor raised by the shale is proposed to implement a simple rectification in the computing model. Based on the essence of the exampled petrophysical models, fitting techniques become the new key solution for the focused prediction task. Stepwise is a well-known multivariable regression and, given its capability on the resolving of collinearity of inputs, has been employed by many researchers to finalize the petrophysical characterization [12–15]. Although this technique is proved successful in the study, it is also questionable owing to two reasons: (1) as the reasonable fitting relationship between applied well logs and porosity, permeability, or water saturation is uncertain, the linear stepwise routinely used in the practical cases might not be a smart solution, and hence, the calculating results of stepwise is universally unqualified or unreliable; (2) sometimes, to gain a satisfactory fitting, stepwise will be complicated by adding some cross terms of inputs, which then will dramatically reduce its generalization or extremely cause an overfitting. Thereby, stepwise seems not to be a preferential selection.

Machine learning (ML) is partially developed for fitting problems and, compared to stepwise, can complete a regression in an implicit computing mechanism or without considering the specific input-output relationship, thus presenting a better generalization in the prediction [16–19]. Conventional neural network, or called two-layer network, is classic in the regression. Its computation imitates the operation of human brain by the connections among input, hidden, and output three layers and finalizes a convergence via back propagation algorithm. Ahmadi et al. [20], Ahmadi *et al.* [16], and Ahmadi et al. [21] have well validated the working performance of typical neural network in the logging-based fitting of porosity and permeability. However, its performance is much sensitive to the initialization and accordingly will easily trap in a local minimum. Besides, its convergence appears to be much slower under modern conditions and has a difficult trade-off between underfitting and overfitting in the training. Thereby, this predictor seems to be not sophisticated. K-nearest neighbors (KNN) and support vector regression (SVR) are two ML representatives in the fitting. KNN primarily utilizes several learning neighbors closer to the test sample to generate an approximate regression [22]. As such computation is simple and easily implemented, some researchers employ KNN to realize the data-driven petrophysical characterization and, finally according to the analysis of validated results, confirm the effectiveness of KNN on the prediction of reservoir parameters [23–25]. Since KNN is featured by a lazy learning which means all learning samples will be scanned to search out the required neighbors for each test sample, its prediction of a test dataset with a large volume will cause a serious time-consuming phenomenon, and then "KD-tree" or "Ball-tree," which will assist KNN to form a presearching path of neighbors, is commonly used in practical case [23, 24]. However, even employing such tree-based pretraining, KNN still will be low-efficient in the prediction, because to obtain a stable input-output mapping, a large-volumetric learning dataset is usually required, while training more learning samples inevitably will decelerate the speed of construction and query of "KD-tree" or "Ball-tree." Hence, the working performance of KNN is not desirable enough. Being different from KNN, SVR applies some significant learning samples that decide the computing effect as support vectors to execute a prediction. To find out the support vectors in a simpler way, raw data will be projected into a high-dimensional space via a kernel function [26]. Thus, by adopting a suitable kernel function, SVR is capable to produce an expected fitting, especially for the nonlinear regression [26]. Based on the super power of SVR shown in the fitting, Al-Anazi and Gates [27] and other researchers launched the SVR-based predictions for some reservoir parameters and through a comparison verified that SVR is a potential candidate in the petrophysical prediction [28, 29]. Nonetheless, as the support vectors produced by SVR are unexplainable, the practical meaning of them for each test sample becomes vague, and then a deeper analysis for the relationship between learning and test samples is inaccessible, which solidly indicates the major shortcoming of SVR in the regression.

If learning samples can be clustered and stored via a logistic searching path, the prediction for test samples will be explainable. Classification and regression tree (CART) clusters the learning data by leaf nodes and connects them through branches and then, in comparison with SVR, has the capability to explain the practical meaning of the used learning samples for each test point and thus presents as a more powerful solver for the fitting of reservoir parameters [30]. Aforementioned "KD-tree" and "Ball-tree" are good derived cases of CART, whereas a single CART still will be failure to fit a test sample if the achieved clusters are unqualified, or in other words, the samples within each cluster are too mathematically dissimilar to generate an acceptable fitting error. Therefore, ensemble learning (EL) is created, which will employ a series of CART to minimize the fitting remain of each test sample [30, 31]. Currently, EL generally can be divided into "bagging" and "boosting" two subcategories. Random forest (RF) is the representative of bagging-based EL, which first will randomly apply partial learning samples to establish a CART and subsequently complete a prediction, and at last, under a loop of such computing operation, the average of all gained fitting results will be regarded as the final predicted outcomes for test dataset [31]. As the result of each test sample is an average

Table 1: Classic petrophysical models employed to predict effective porosity, effective permeability, and water saturation of conventional petroleum-bearing reservoirs.

Reservoir parameter	Model	General expression	Variable
Porosity (ϕ, %)	Single-log model [4]	$\phi = (1/C_{sh})(x - x_{ma}/x_f - x_{ma})$	C_{sh}: compaction factor; x: value from acoustic, density, or neutron log; x_{ma}: logging value of matrix; x_f: logging value of fluid
	Wyllie-Rose model [5]	$\phi = 1 - (\Delta t_{ma}/\Delta t)^{1/zp}$	Δt_{ma}: acoustic logging value of matrix; Δt: value of acoustic log; zp: formation factor
	Density-neutron root-mean-square (RMS) model [1]	$\phi = (1/C_{sh})\sqrt{\left(\rho - \rho_{ma}/\rho_f - \rho_{ma}\right)^2 + \left(d - d_{ma}/d_f - d_{ma}\right)^2}$	C_{sh}: compaction factor; ρ, d: values of density and neutron logs; ρ_{ma}, d_{ma}: density and neutron logging values of matrix; ρ_f, d_f: density and neutron logging values of fluid
Permeability (K, mD)	Kozeny-Carman model [6, 7]	$K = a\phi^3/(1-\phi)^2 S^2$	a: empirical coefficient; ϕ: porosity; S: specific surface area of rock
	Krumbein-Monk model [8]	$K = a_1 g^{a_2 g_c}$	a_1, a_2: empirical coefficients; g: median grain diameter; g_c: standard deviation of g
	Timur model [2]	$K = a_1 \phi^{a_2}/S_{wi}^2$	a_1, a_2: empirical coefficients; ϕ: porosity; S_{wi}: irreducible water saturation
Water saturation (S_w, %)	Archie model [9]	$S_w = \sqrt[n]{(abR_w/\phi^m R_t)}$	a, b, m, n: empirical coefficients; R_w: resistivity of formation water; R_t: resistivity of true formation
	Waxman-Smits-based model [3]	$C_t = \alpha S_w^n + \beta S_w^q$	C_t: conductivity of true formation; α, n: coefficient and power of water saturation term; β, q: coefficient and power of shale term
	Poupon-Leveaux-based model [10]	$C_t = \alpha S_w^n + \gamma S_w^p + \beta S_w^q$	C_t: conductivity of true formation; α, n: coefficient and power of water saturation term; γ, p: coefficient and power of cross term; β, q: coefficient and power of shale term

estimation of many CARTs, the impact of underfitting or overfitting on the prediction, to a large extent, is reduced, hence RF displaying an ideal robust nature [31]. Ao et al. [32] and other researchers noticed this advantage of RF in the fitting and then utilized it to implement the application in the petrophysical characterization [33, 34]. Although the experiments launched by them manifest RF that is capable of completing a satisfactory fitting for porosity, permeability, or water saturation, this model is still an undesirable regression solver, because to measure the working performance of a fitting solver, robustness is only the secondary metric, and the primary pursue is a capability that will enable the predictor to gain a minimum error for the computing objective.

Gradient boosting decision tree (GBDT) is a classic boosting model, fundamentally defining the basic computing rule of boosting-based EL that the fitting errors will be progressively reduced to a minimum by a set of CARTs [30]. Specifically, test dataset first will be predicted by an average of all learning samples, and correspondingly the produced fitting error information will be used to establish the first CART, and next the remaining error determined by this CART for each test sample will be assembled to create the following CART, and finally given such computing loop, the fitting errors will be gradually reduced to a minimum [30]. GBDT manages to gain a minimum upon the fitting errors, thus exhibiting as a more suitable solver in the regression in comparison with RF. Nonetheless, the achieved experimental proofs indicate that GBDT generally is incapable to produce a perfect prediction and always causes a tremendous waste on the memory of training data; Chen and Guestrin [35] then provided some theoretical improvements in terms of loss function and data storage and eventually proposed a new model called extreme gradient boosting (XGBoost). This GBDT-based model can indeed obtain a wonderful score on the fitting precision, but since its computing speed will be exponentially decelerated when more learning samples are used in the training, it usually performs inefficiently in the process of big data [36]. Ke *et al.* [36] emphatically analyzed the construction of CART and purposefully designed several algorithms such as gradient-based one-side sampling (GOSS), exclusive feature bundling (EFB), and histogram to accelerate the establishment of each CART, consequently creating a XGBoost-based model named light gradient boosting machine (LightGBM). Based on some tests, it is proved that LightGBM can complete a prediction faster compared to XGBoost, and the computing performance is also acceptable and sometimes even better than that of XGBoost [36]. Therefore, LightGBM shows the greater potential for the fitting of reservoir parameters. Zhou *et al.* [37] employed LightGBM to predict permeability based on a feature selection of well logs, and in accordance with the analysis of the obtained results, ensured LightGBM is a "sharp tool' for the petrophysical prediction. Hadavimoghaddam et al. [38] studied an automatic regression for

water saturation, and through a comparison demonstrated, LightGBM is a better selection.

Although the strong capability of LightGBM for the fitting issue is solidly verified in practical cases, the prediction of a small-volumetric dataset which will arise an underfitting for LightGBM is never considered. Transfer learning is a conception of deep learning, specially addressing the training of a small-volumetric dataset [39]. If the handed samples have the similar characterization, the dataset with a smaller volume can be trained well by a ready-made network established by the rest larger-volumetric dataset, which is just the computing mechanism of transfer learning [39, 40]. Hence, with the integration of transfer learning, the generalization of LightGBM will be enhanced, especially for the process of a small-volumetric dataset. Additionally, to guarantee the predicting quality of LightGBM, two advanced techniques, continuous restricted Boltzmann machine (CRBM) and Bayesian optimization (Bayes), are introduced as assistants to improve the significance of inputs and the setting of employed hyperparameters [41, 42]. Accordingly, on the basis of transfer learning, a new hybrid EL-based model is proposed for the fitting of porosity, permeability, and water saturation, called CRBM-Bayes-LightGBM. In the following paragraphs, methodology, data validation, and discussion of experimental results for the proposed predictor will be described in details orderly.

2. Methodology

In this chapter, methodology of the proposed predictor will be described by several sections, including preprocessing of raw samples, dimensional reduction of input logs, modeling of LightGBM, optimization of hyperparameters, embedding of transfer learning, and performance measure of fitting. Given the understanding of each computing section, the computing flow, established on the basis of ensemble and transfer learnings, applied to regress three target reservoir characters, will be provided as a final section.

2.1. Preprocessing. As well as logs are measured by electronic apparatuses, the achieved logging sequences will inevitably be affected by noisy information. Then, to raise signal-to-noise ratio (SNR) of the raw inputs, noisy samples must be excluded. Since generally measuring value will exceed a normal varying limitation upon the impact of noise, a noisy point can actually be viewed as an outlier, and therefore, a detection of outliers becomes accessible to filter the basic dataset. Tukey's method is skilled in removing outliers and, as it only applies quartile information, is easily implemented and thus adopted to detect the noisy inputs [43]. Lower inner fence (LIF) and upper inner fence (UIF) will be employed by this method to form a normal varying limitation and then conduct a judgment for outliers. The equation set calculating two fences is given below [43]:

$$\begin{aligned} \mathrm{IQR} &= 1.5 \times (Q_3\text{-}Q_1), \\ \mathrm{LIF} &= Q_1\text{-IQR}, \\ \mathrm{UIF} &= Q_3 + \mathrm{IQR}, \end{aligned} \tag{1}$$

where Q_1 is the lower quartile, Q_3 is the upper quartile, and IQR means the inner quartile range.

For an input log, if values are larger or smaller than UIF or LIF, they will be determined as outliers and then excluded from the raw dataset [43]. Nonetheless, prior to modeling, the scale of each log also has to be considered. Since conventional logging sequences vary with different orders of magnitude, the contribution provided by the logs with small orders in the prediction will be dramatically reduced if all logs are directly applied during the modeling. Hence, normalization for well logs becomes essential.

Now, if the original input matrix is expressed by $\mathbf{A}_{m_1 n_1} = [\mathbf{X}^{\mathrm{ori}}, \mathbf{Y}^{\mathrm{ori}}]$, where m_1 is the number of input samples, n_1 is the number of columns of the input matrix, $\mathbf{X}^{\mathrm{ori}}$ is the original logging matrix, and $\mathbf{Y}^{\mathrm{ori}}$ stands for the core-measured vector of porosity, permeability, or water saturation, after a detection of outliers and a normalization, it can be rewritten by $\mathbf{A}_{mn_1} = [\mathbf{X}^{\mathrm{pre}}, \mathbf{Y}^{\mathrm{pre}}]$, where m is the new number of input samples and $\mathbf{X}^{\mathrm{pre}}$ and $\mathbf{Y}^{\mathrm{pre}}$ are the logging matrix and core-measured vector gained from preprocessing, respectively.

2.2. Dimensional Reduction of Well Logs. Although the preprocessing has enhanced the quality of the input logs, the amount of utilized logs—which is a crucial element impacting the predictor's computation speed—is never taken into account. Faster prediction will be made with fewer input variables, and training data must then be reduced in dimension [36]. LightGBM employs EFB to reduce the dimensionality of input data, whereas conventional logging sequences are generally not mutually exclusive or specifically they can take nonzero values simultaneously [36]. EFB hence presents unsuitably for the process of well logs. To find a powerful solver for the dimensional reduction, restricted Boltzmann machine (RBM), being well-known as a feature extractor from deep belief network (DBN), can be regarded as a potential candidate because it is feasible to extract more or less new variables from the raw dataset and meanwhile can ensure that the extracted information is beneficial in the prediction [44]. As original RBM is only functional for the binary information, CRBM, a RBM-based extractor specially applied to compute the data varying continuously, is adopted to realize a reduction on the dimensionality of the logging matrix [41]. The construction of CRBM is simple and only composed by a visible and a hidden layer. The mathematical expressions of two layers are written below [41, 45]:

$$\begin{aligned} P_{\boldsymbol{\theta}}(\mathbf{h}_j|\mathbf{V}) &= C_{sf}\left(\mathbf{V}^{\mathrm{T}}\mathbf{W} + C_\sigma C_{\mathrm{norm}}\right)(\mathrm{transiting}), \\ P_{\boldsymbol{\theta}}(\mathbf{v}_i|\mathbf{H}) &= C_{sf}(\mathbf{W}\mathbf{H} + C_\sigma C_{\mathrm{norm}})(\mathrm{reconstructing}), \\ C_{sf}(\mathbf{x}) &= C_{la} + \frac{(C_{ua} - C_{la})}{\left(1 + e^{-C_\mu \mathbf{x}}\right)}, \end{aligned} \tag{2}$$

where P represents the probabilistic activity function, $\boldsymbol{\theta}$ is the set of hyperparameters, $\mathbf{V}$ is the visible matrix, $\mathbf{H}$ is the hidden matrix, $\mathbf{v}_i$ is ith visible vector, $\mathbf{h}_j$ is jth hidden vector,

$\mathbf{W}$ is the weight matrix, C_σ is the noise variance, C_{norm} stands for the normal distribution used to generate noisy information, C_{sf} represents the sigmoid function, C_{la} is the lower limitation of the sigmoid, C_{ua} is the upper limitation of the sigmoid, and C_μ is the noise controller.

For a review of the above expressions, it is clear that the input data will be handled by the visible and then transited to the hidden via the weight matrix. To guarantee the transiting quality, the hidden matrix will be sent back to the visible as a reconstructed matrix, and then a check will be executed in which the transited data will be proved qualified if the error between the observation and reconstructed data is acceptable. If the transiting is unqualified, the data currently held by the visible will be trained by CRBM once again, and subsequently a second round checking the reconstructed data will be conducted. Upon a loop of transiting and reconstructing, an iterative training for CRBM is formed [41, 45]. Since the weight matrix and noise controller routinely are viewed as the hyperparameters, the training targets of CRBM contain $\mathbf{W}$ and C_μ [41, 45]. Hinton [46] proposed a faster training algorithm named contrastive divergence (CD) for RBM-based extractors and, demonstrated a satisfactory training that can be gained only after one iteration, then CD also known as CD-1. However, with an increasing on the size of input matrix, CD-1 will be exponentially slower [44]. Thereby, a mini batch technique should be embedded during the training. This technique will divide the raw inputs into several minibatches, and it is validated that the computing time cost of all minibatches is much less than that of an entire input matrix [47]. Accordingly, with the introduction of CD-1 and minibatch technique, the iteration for two target hyperparameters can be expressed mathematically as [41, 45–47]

$$\begin{aligned} w_{ij}^{ct} &= C_m w_{ij}^{ct-1} + \left(\frac{C_{lr}}{C_{\text{mini}}}\right) \times \sum_{z=1}^{Z} \left(P_{\boldsymbol{\theta}}\left(\mathbf{h}_j \middle| \mathbf{V}^{0z}\right)\mathbf{v}_i^{0z} - P_{\boldsymbol{\theta}}\left(\mathbf{h}_j \middle| \mathbf{V}^{1z}\right)\mathbf{v}_i^{1z}\right), \\ C_\mu^{ct} &= C_m C_\mu^{ct-1} + \left(\frac{C_{lr}}{C_{\text{mini}}}\right) \times \sum_{z=1}^{Z} \left(\frac{\left(P_{\boldsymbol{\theta}}\left(\mathbf{h}_j \middle| \mathbf{V}^{0z}\right)^2 - P_{\boldsymbol{\theta}}\left(\mathbf{h}_j \middle| \mathbf{V}^{1z}\right)^2\right)}{\left(C_\mu^{ct-1}\right)^2}\right), \end{aligned} \tag{3}$$

where w_{ij} is the element of the weight matrix in ith raw and j th column, C_m is the momentum coefficient, C_{lr} is the learning rate, C_{mini} is the size of a minibatch, Z is the number of minibatches, superscripts ct and $ct-1$, respectively, represent ctth and $ct-1$th epoch, and superscripts $0z$ and $1z$, respectively, stand for the original and the first reconstructed status of a mini batch.

The epoch ct means the iteration of CRBM. Since a training of all minibatches also will be iteratively completed, to make a discrimination, the iteration corresponding to the training of CRBM is named as epoch.

Given the application of CRBM, the input matrix $\mathbf{A}_{mn_1} = [\mathbf{X}^{\text{pre}}, \mathbf{Y}^{\text{pre}}]$ can be rewritten as $\mathbf{A}_{mn} = [\mathbf{X}, \mathbf{Y}]$, where n is the number of columns of the new input matrix and $\mathbf{X}$ and $\mathbf{Y}$ represent the CRBM-transformed logging matrix and core-measured vector, respectively.

2.3. Modeling. The procedure enters the modeling step when the input data have been prepared. In this level, the executor is the core predictor, LightGBM, and it mostly uses a strong learning machine to do the job. A strong learning tool is made up of several CARTs, or more technically, several week learning tools [36]. Then, according the computing theory of EL, the modeling implemented by LightGBM can be expressed as [36–38]

$$\begin{aligned} F_{lgb}(\mathbf{x}_i) = \arg\min_{\alpha} \sum_{i=1}^{m} L_{\text{loss}}(y_i, \alpha) + \sum_{l=1}^{L_{cn}} \sum_{j=1}^{J_l} \sum_{i \in Q_j} \\ \cdot \left[\frac{-L_{lr}\left(\partial L_{\text{loss}}\left(y_i, y'_{i,l-1}\right)/\partial y'_{i,l-1}\right)}{\partial^2 L_{\text{loss}}\left(y_i, y'_{i,l-1}\right)/\left(\partial y'_{i,l-1}\right)^2 + L_{r1}} \right. \\ \left. + L_{r2}\left(\frac{\partial L_{\text{loss}}\left(y_i, y'_{i,l-1}\right)/\partial y'_{i,l-1}}{\partial^2 L_{\text{loss}}\left(y_i, y'_{i,l-1}\right)/\left(\partial y'_{i,l-1}\right)^2}\right)^2 \right], \end{aligned} \tag{4}$$

where F_{lgb} represents the strong learning machine, $\mathbf{x}_i$ is ith input sample, L_{loss} stands for the loss function, y_i is the observation of ith input sample, α is a constant, L_{cn} is the number of week learning machines, J_l is the number of leaf nodes of lth CART, Q_i is the zone of jth leaf node, L_{lr} is the learning rate, $y'_{i,l-1}$ is the predicted value gained from $l-1$th CART, and L_{r1} and L_{r2} are the regularizations.

The type of L_{loss} commonly used is squared, and hence, a selection of α can be an average of all input samples. Since the equation given above is derived from XGBoost, which will be rather low-efficient when dealing with more input samples, LightGBM will also simultaneously conduct some algorithms including GOSS, EFB, histogram, and leaf-wise to accelerate the computation during the training [36]. GOSS will abandon the samples having smaller gradient contributions in the modeling of each week learning machine, then realize a gradual shrinking of the size of the input matrix, and accordingly lift the modeling efficiency. As aforementioned, EFB is inappropriate for the process of well logs and thus can be ignored. Histogram is applied to search the best split for a leaf node. Compared to the classic searching mode, since histogram demands that a test point for the best split can be selected within the histogram-based statistical results of all input samples, the trials used for the searching of the split will be much fewer, and then the time spending on the searching will be dramatically reduced. Thereby, by histogram, a CART will be rapidly established. Leaf-wise is a relative concept of level-wise, which only allows CART to generate one leaf node in each depth. Based on this rule, a CART will grow faster, but a deeper construction also will be obtained, which will possibly cause an overfitting. Thus, the depth of each CART universally will be restricted in the training of LightGBM. As GOSS, histogram, and leaf-wise algorithms are indispensable

for LightGBM, in the following validation, they will be used as a default setting and never mentioned.

2.4. Optimization. The core predictor applies many hyperparameters during the training, and then to guarantee the predicting quality, a parametric optimization is required. Bayes at present is hot in the field of EL owning to its high efficiency in the multiobjective optimization [42, 48, 49]. Compared to random search (RS), because of employing surrogate posterior information to determine an optimal solution, Bayes seems to be more reasonable, and compared to swarm intelligence (SI), it can utilize fewer trials to complete an optimization faster [42, 48, 49]. Therefore, Bayes is adopted as a more potential optimizer for LightGBM. This optimizer will employ a surrogate model to compute prior and posterior data, and a common choice for the surrogate model is the Gaussian process (GP). GP assumes the variation of each hyperparameter for the computing objective complies with a normal distribution, and then in this circumstance, the expression of each hyperparameter can be written as [42, 48, 49]

$$\beta_i^j \sim N\left(\mu_i^j, \sigma_i^j\right), \tag{5}$$

where β_i^j represents the jth status of ith hyperparameter, N stands for the normal distribution, and μ_i^j and σ_i^j are the mean and variance corresponding to β_i^j, respectively.

If more input information is available for GP, the variation of each hyperparameter will be more stable, and accordingly the optimal setting for the computing objective will be more easily searched out [42, 48, 49]. Then, for a hyperparameter, when an initial GP is formed, how to appropriately acquire the rest optimizing information becomes a key problem. Acquisition function is an answer, which will assist Bayes to find out the best iterative point of each hyperparameter from the current GP [42, 48–50]. Probability of improvement (PI), expected improvement (EI), and Gaussian process-upper confidence bound (GP-UCB) are the three classic acquisition functions, and upon the previous findings, it is argued that EI and GP-UCB are relatively more effective in acquiring the best solution for the next iteration of Bayes [48–50]. As EI is more complex by applying cumulative distribution function (CDF) and probability distribution function (PDF), GP-UCB then becomes a simpler as well as effective selection for Bayes [48–50]. The equation of GP-UCB is given below [48–50]:

$$\beta_i^{q+1} = \arg\max_j \left(\boldsymbol{\mu}_i^q + 1.96\boldsymbol{\sigma}_i^q\right), \tag{6}$$

where β_i^{q+1} is the $q+1$th status of ith hyperparameter, $\boldsymbol{\mu}_i^q$ is the GP-estimated vector composed by former q statuses of the mean of ith hyperparameter, $\boldsymbol{\sigma}_i^q$ is the GP-estimated vector composed by former q statuses of the variance of ith hyperparameter, and $j = 1, 2, \cdots, q$.

Given the usage of GP-UCB, Bayes will first apply the original input information as prior data to produce posterior data via GP, subsequently determine the best iterative values for hyperparameters and save them as the new posterior data for the next computing round, and finally, when the optimizing iteration is ceased, will figure out the best parametric setting [48–50].

2.5. Transfer Learning. LightGBM or broadly EL will cause an overfitting or an underfitting when dealing with a small-volumetric dataset and preferentially encounters the underfitting [37, 38]. Then, in practical case, there exists a new challenge for the prediction of LightGBM, which should be seriously considered. Transfer learning, a concept or a skill in the deep learning, could be a potential solver because it is particularly developed for the process of a small set of samples [39, 40]. According to the principle of transfer learning, a small-volumetric dataset with the characteristics comparable to another big set can be trained successfully if a ready-made predictor created by this large set is available [39, 40]. If a ready-made strong learning machine can be used in the training, LightGBM will then be applicable for a smaller number of samples by mimicking this computing process. Specifically, given an available strong learning machine trained by the 1st dataset, for the 2nd small-volumetric dataset featured similarly with the 1st dataset, its modeling and parametric optimization can be directly initialized on the basis of the strong learning machine, which then will enable LightGBM to be capable in creating a fast as well as effective training and meanwhile, to a large extent, to avoid an occurrence of the overfitting or underfitting during the modeling. Therefore, for the training and prediction of a small-volumetric dataset in the following experiment, an integration of the proposed predictor and transfer learning will be applied to provide an effective solution.

2.6. Performance Measure. The common metric applied to measure the fitting performance is mean squared error (MSE), while for porosity and water saturation, this metric would be too small to generate a better discrimination [8, 17–19]. Then, root-mean-square error (RMSE) is adopted to evaluate the fitting quality of these two reservoir characters. The equation computing RMSE is shown below:

$$\mathrm{RMSE} = \sqrt{\sum_{i=1}^{m}\left(\frac{\left(y_i - y_i'\right)^2}{m}\right)}, \tag{7}$$

where y_i' is the predicted value of ith sample.

Permeability normally varies with different orders of magnitude so that RMSE can be implemented based on the logarithmic values of the predicted permeability data [17–19]. There has an example that can make a better illustration for the advantage of the usage of logarithmic permeability in the performance measure. If the observation of a sample is 1 mD and there exist two predicted results which are 0.1 mD and 2 mD, the absolute fitting errors of them will be 0.9 mD and 1 mD, respectively, and then 0.1 mD will be considered as the better fitting result owing to its smaller fitting error. However, according to the theory of logging

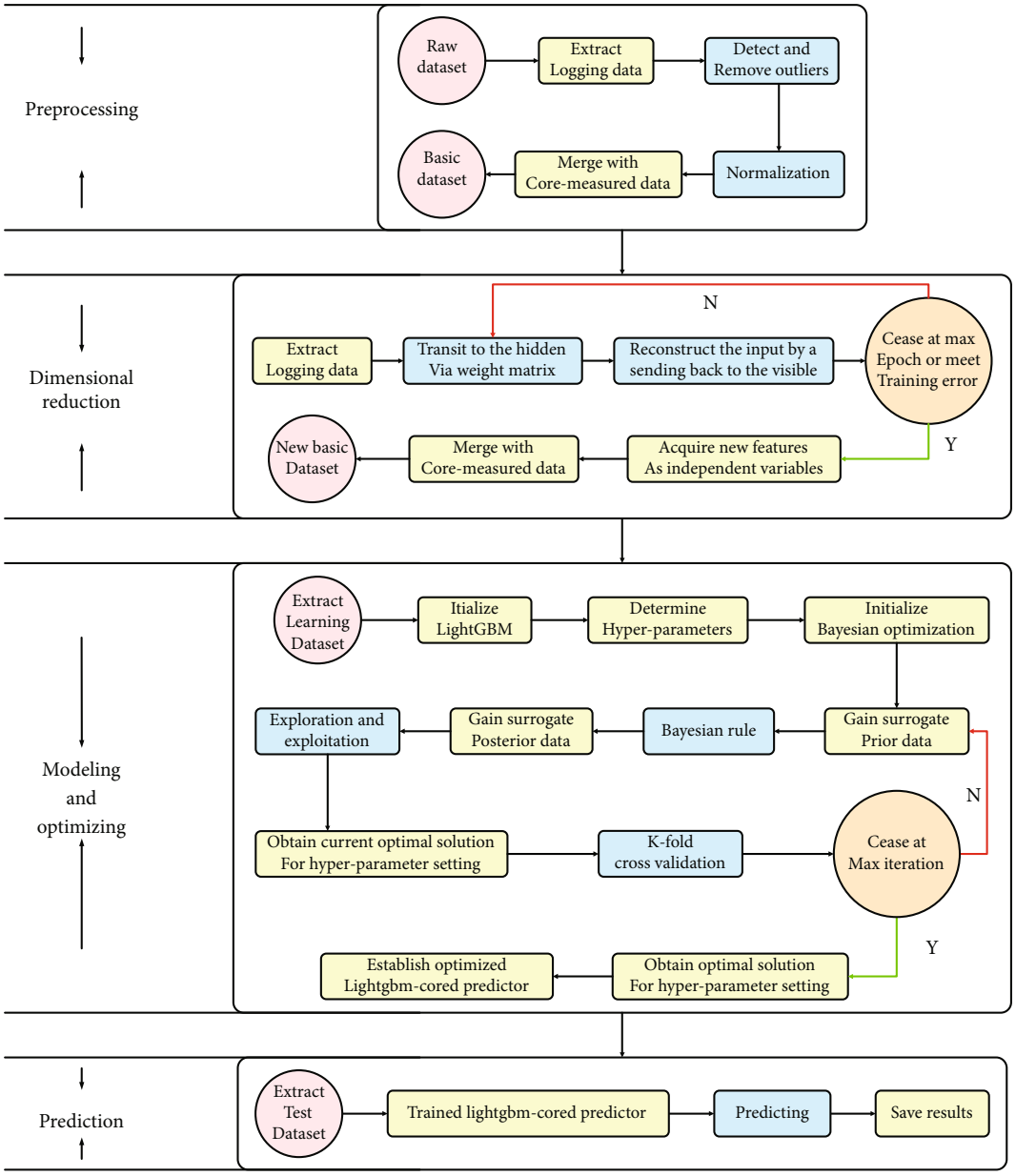


Figure 1: Computing flow of LightGBM-cored predictor for petrophysical regression of sandy-mud reservoirs. LightGBM = light gradient boosting machine.

interpretation which demonstrates that the permeability values in the same order of magnitude will be regarded more closely, 2 mD should be a more reasonable result [2, 8, 17, 18, 19]. Then, if the logarithmic values of 0.1 mD and 2 mD are applied, the fitting errors will be 1 mD and 0.3 mD, and consequently, 2 mD will be viewed as the better fitting outcome. Given this explanation, RMSE used to measure the permeability data can be written as

$$\mathrm{RMSE}_p = \sqrt{\sum_{i=1}^{m}\left(\frac{\left(\lg\left(y_i/y_i'\right)\right)^2}{m}\right)}, \tag{8}$$

where RMSE_p stands for the RMSE of permeability information.

2.7. Computing Flow. Based on the theoretical analysis above, a computing flow of the proposed predictor for the petrophysical regression and another case including transfer learning are designed and illustrated in Figures 1 and 2, respectively.

The workflow shown in Figure 1 overall contains four major steps: (1) preprocessing: upon the preparation of the raw dataset, the logging data first will be detected by Equation (1) to remove outliers and subsequently processed under a normalization. Finally, the achieved new logging matrix and the corresponding core-measured vector will be

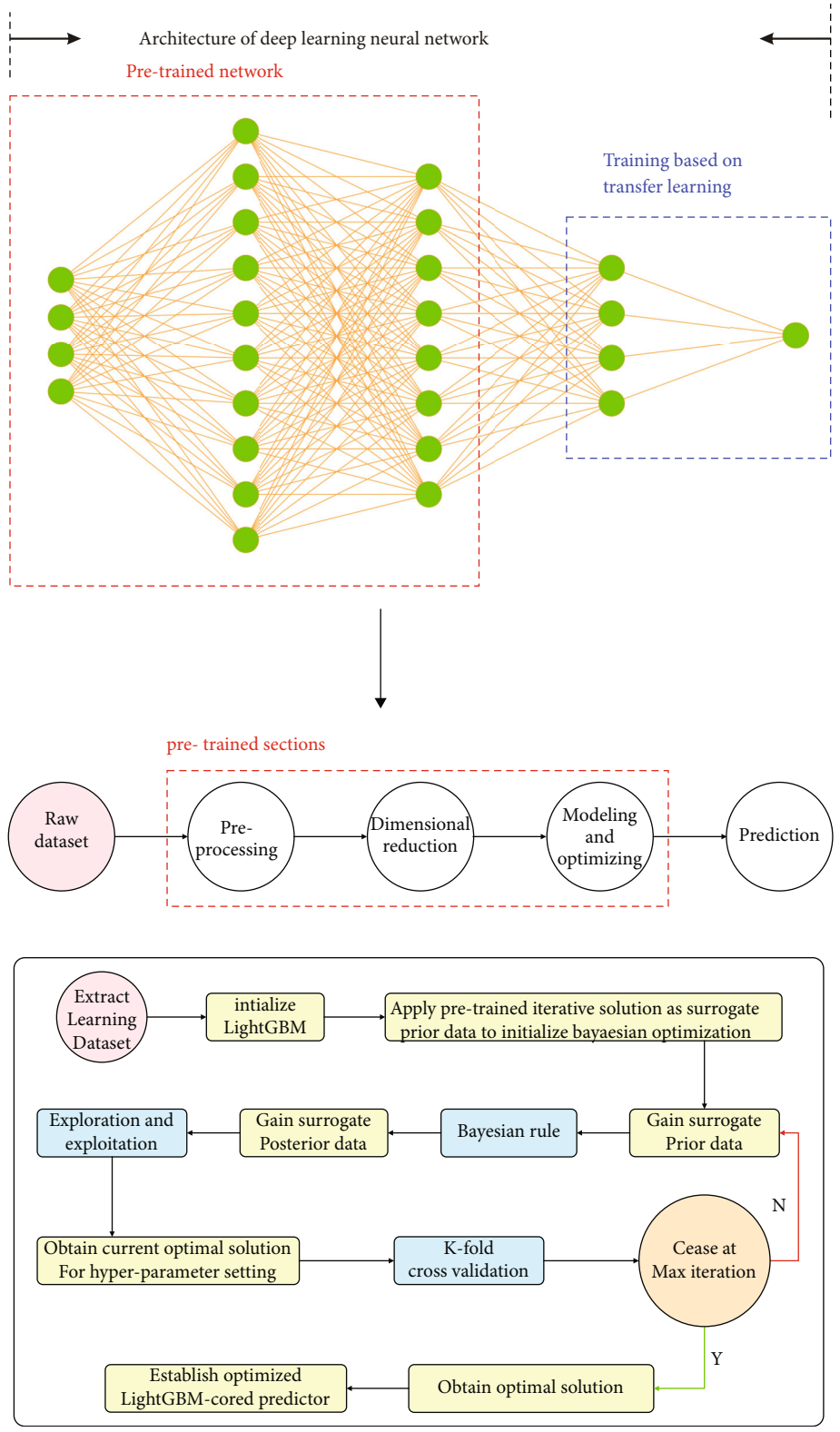
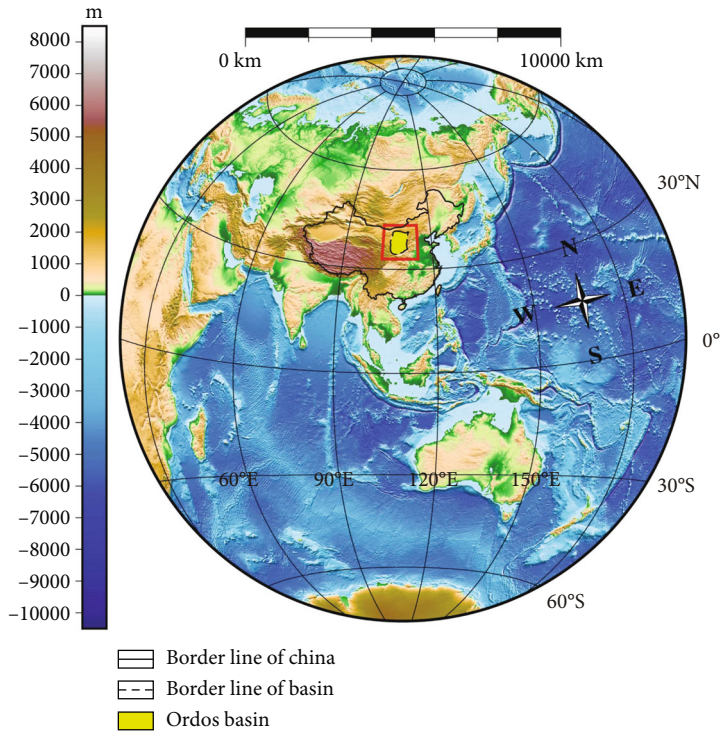
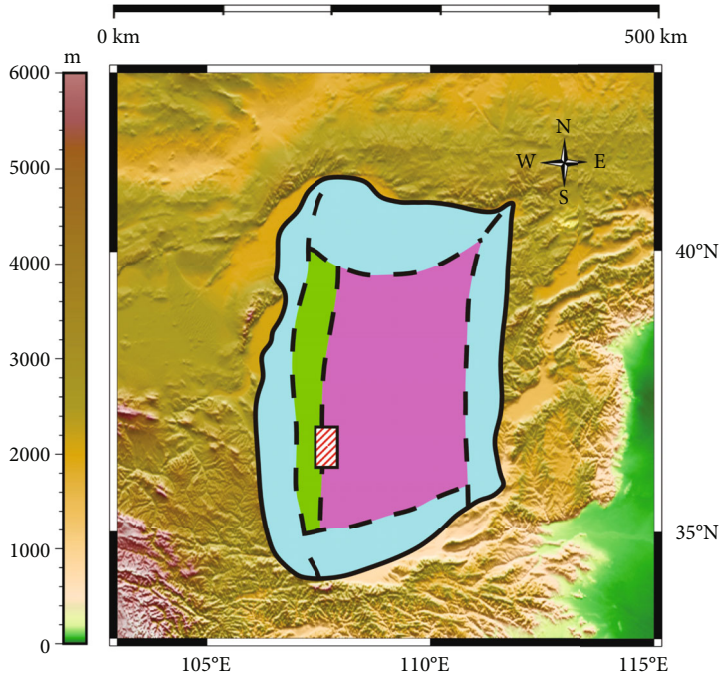


Figure 2: Computing flow of LightGBM-cored predictor for petrophysical regression of sandy-mud reservoirs based on transfer learning. LightGBM = light gradient boosting machine.

merged as the basic dataset. The core-measured vector can be composed by porosity, permeability, or water saturation data. (2) Dimensional reduction: this step will reduce the number of inputs and meanwhile enhance the significance of the inputs for the prediction. Input logs first will be loaded by the visible and transit to the hidden. To check the transiting quality, the gained hidden matrix will be sent back to the visible, and then a comparison between the observation and

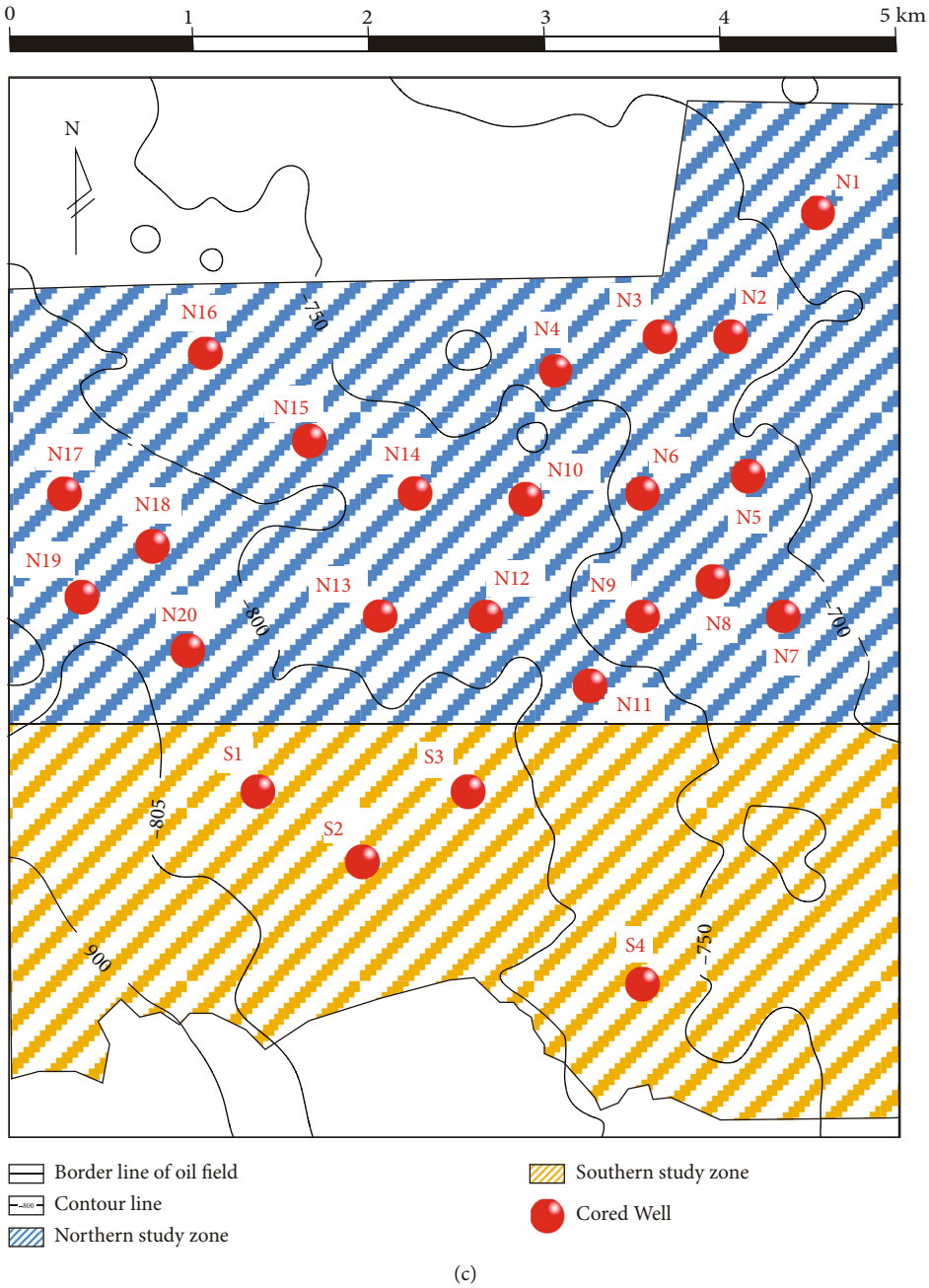


(a)



(b)

Figure 3: Continued.



(c)

Figure 3: Geographic locations of the Ordos Basin (a) and study zone (b) and distribution of cored wells within the study zone (c).

reconstructed data in the visible will be launched. If the reconstructed error is acceptable or the training is ceased at the max epoch, the output in the hidden extracted by CRBM will be regarded as new input variables, or the training will be continued. All operations will be implemented by Equations (2) and (3). Since less new variables are required, the dimensionality of input matrix actually is reduced. The matrix composed by new variables at last will be merged with core-measured vector, and a new basic dataset accordingly will be obtained. (3) Modeling and optimizing: to initialize LightGBM, the learning part of the new basic dataset will be taken. Bayes can then be used after determining all hyperparameters. In order to find the optimal iterative point, the optimizer first applies GP as a surrogate model, then calculates the posterior data using the Bayesian method, and finally makes a trade-off using Equation (6). To acquire a robust iterative result, a K-fold cross validation subsequently will be utilized during the optimization. When Bayes is ceased at the max iteration, the optimal parametric setting will be known, and accordingly, Equation (4) can be confirmed. (4) Prediction: under the usage of the established LightGBM, the test part of the new basic dataset can be predicted, and the estimation of the fitting results for porosity and water saturation can be finalized by Equation (7) and for permeability by Equation (8).

Table 2: Summary of quartile-based statistical information of applied well logs.

log[1]	AC (μs/m)	CNL (%)	DEN (g/cm^3)	GR (API)	SP (mV)	PE (b/e)	AT10 (Ω·m)	AT20 (Ω·m)	AT30 (Ω·m)	AT60 (Ω·m)	AT90 (Ω·m)
	212.80	22.05	2.62	92.90	84.32	3.35	19.07	17.48	16.64	15.98	15.63
	204.64	20.46	2.62	82.80	83.01	3.33	24.73	20.40	19.22	18.14	17.83
	202.41	18.37	2.61	78.19	81.60	3.31	25.58	18.86	17.49	16.36	16.08
Value	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	230.86	21.03	2.50	96.94	52.34	2.71	8.63	6.12	5.19	4.64	4.51
	232.66	21.58	2.50	91.91	51.94	2.71	8.95	6.29	5.30	4.72	4.59
	235.76	22.22	2.50	82.69	51.62	2.74	9.63	6.69	5.60	4.98	4.84
Max	330.12	76.56	2.68	150.23	88.26	4.22	73.08	41.19	38.88	39.86	40.45
Min	183.99	9.16	1.53	40.49	23.93	1.72	1.70	2.09	2.78	1.95	1.73
Q_1	223.98	19.82	2.41	72.60	43.39	2.74	7.58	6.29	5.07	4.00	3.74
Q_3	247.38	24.06	2.59	102.58	80.25	3.21	14.43	13.13	12.86	12.76	12.78
IQR[2]	35.10	6.37	0.28	44.97	55.28	0.71	10.28	10.26	11.69	13.14	13.56
LIF[3]	188.88	13.45	2.13	27.63	-11.89	2.04	-2.70	-3.97	-6.62	-9.14	-9.82
UIF[4]	282.48	30.43	2.88	147.55	135.53	3.92	24.71	23.39	24.55	25.90	26.35

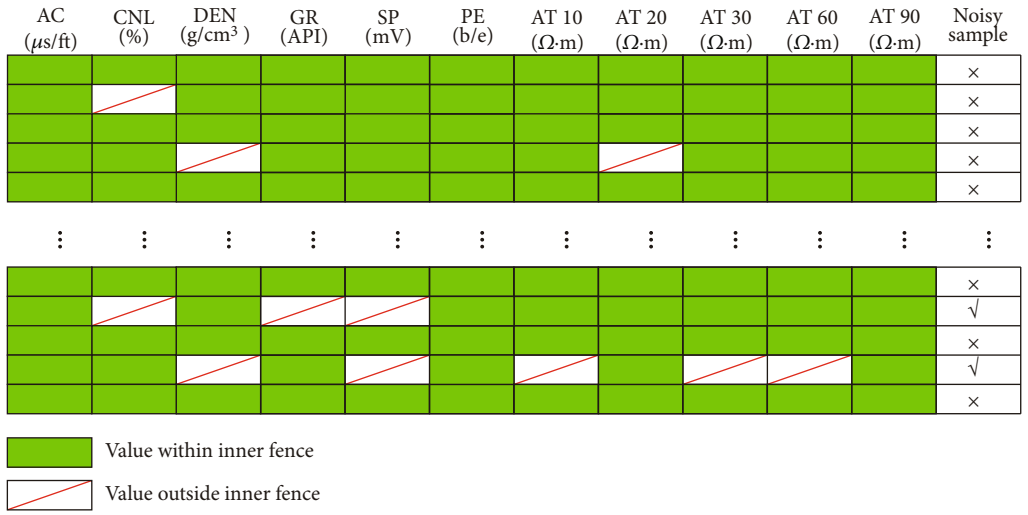


Figure 4: Judgment of noisy samples for logging samples. AC = acoustic log; CNL = compensated neutron log; DEN = density log; GR = gamma ray; SP = spontaneous potential; PE = photoelectric absorption cross-section index; AT10, AT20, AT30, AT60, and AT90 = resistivity of formation measured by array induction log at 10-inch, 20-inch, 30-inch, 60-inch, and 90-inch logging depth.

Figure 2 displays another workflow containing transfer learning. The upper part illustrates how transfer learning is applied in the training of a deep neuron network. The pretrained network established by a large set of samples can be used as a front-end engine to launch the training for a small set of samples, and then by imitating this computing mechanism, the achieved LightGBM-cored predictor can be regarded as a basis to execute a training for a small-volumetric dataset. Specifically, preprocessing, dimensional reduction, and modeling and optimizing obtained previously can be viewed as the pretrained sections, and then for the training of a small dataset, the modeling and optimizing can be initialized on the basis of the previous strong learning machine and parametric setting. To gain a better fitting for the small dataset, the previous strong learning machine can be enhanced by adding more week learning machines. Since the data of small set is featured similarly with that used by the pretrained predictor, the optimal setting theoretically will be not much different with the pretrained one and then can be fast searched out via transfer learning. Consequently, an effective LightGBM-based training for a small-volumetric dataset becomes accessible under the support of transfer learning. When the training is completed, the predictor produced from a small dataset can be employed to execute a qualified prediction.

3. Validation, Results, and Discussion

In this chapter, the predicting capability of LightGBM-cored predictor or the feasibility of the designed computing flows will be validated by the data collected from the study zone.

Table 3: Initial settings of employed approaches for dimensional reduction.

Approach	CRBM (continuous restricted Boltzmann machine)	PCA (principal component analysis)
Parametric setting	Size of visible layer $(C_v) = 11$ Size of hidden layer $(C_{\mathbf{h}}) = 6$ Learning rate $(C_{lr}) = 0.1$ Max epoch $(C_{EP}) = 100$ Training error $(C_e) = 0.001$ Size of mini batch $(C_{\text{mini}}) = 100$ Momentum coefficient $(C_m) = 0.9$ Noise variance $(C_\sigma) = 0.2$ Noise controller $(C_\mu) = 0.2$ Lower limitation of the sigmoid $(C_{la}) = 0$ Upper limitation of the sigmoid $(C_{ua}) = 1$	Solver $(S_{\text{PCA}}) = \text{SVD}^*$ Number of reserved variables $(N_{\text{PCA}}) = 6$

* SVD = singular value decomposition.

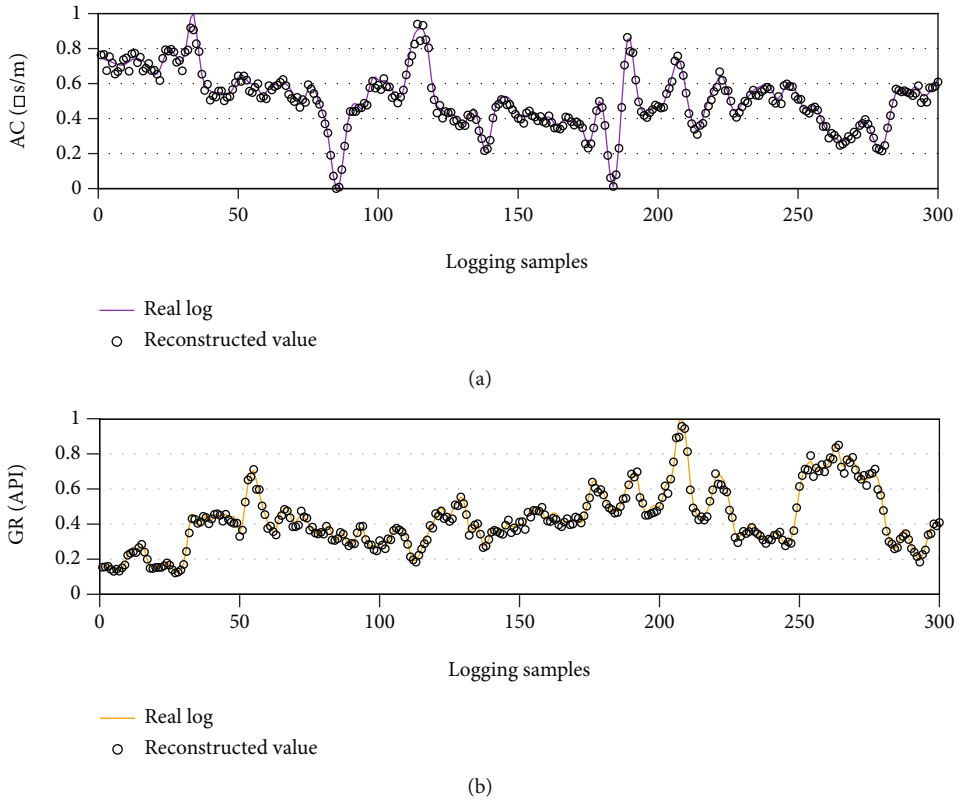


Figure 5: Reconstructions of AC (a) and GR (b) implemented by continuous restricted Boltzmann machine (CRBM). AC = acoustic log; GR = gamma ray.

Some experiments will be designed purposefully to obtain the results and then from different perspectives to reveal the working performance of the proposed predictor. Finally, given the achieved experimental results, a comprehensive discussion will be provided to argue the capability and generalization of CRBM-Bayes-LightGBM in the practical case.

3.1. Data Source and Experimental Design. The Ordos Basin, geographically located in the northern China as shown in Figure 3(a), is a giant petroleum-bearing basin, and given the previous findings, it is uncovered that there still exist a great amount of hydrocarbon resources and most of them are accumulated within the sandy-mud reservoirs [51, 52]. As a result, there is still considerable work to be done in the exploration of the Ordos Basin, and one key goal is to get a better understanding of reservoir classification. Porosity, permeability, and water saturation are the three important indicators, and the petrophysical condition of the reservoirs may well define the storage capabilities of oil and gas. Thus, the prediction of these three reservoir characteristics becomes more vital.

The study zone for the petrophysical prediction in this paper is in the Jiyuan Oilfield of the Ordos Basin, located between the Tianhuan Depression and Yishan Slope as

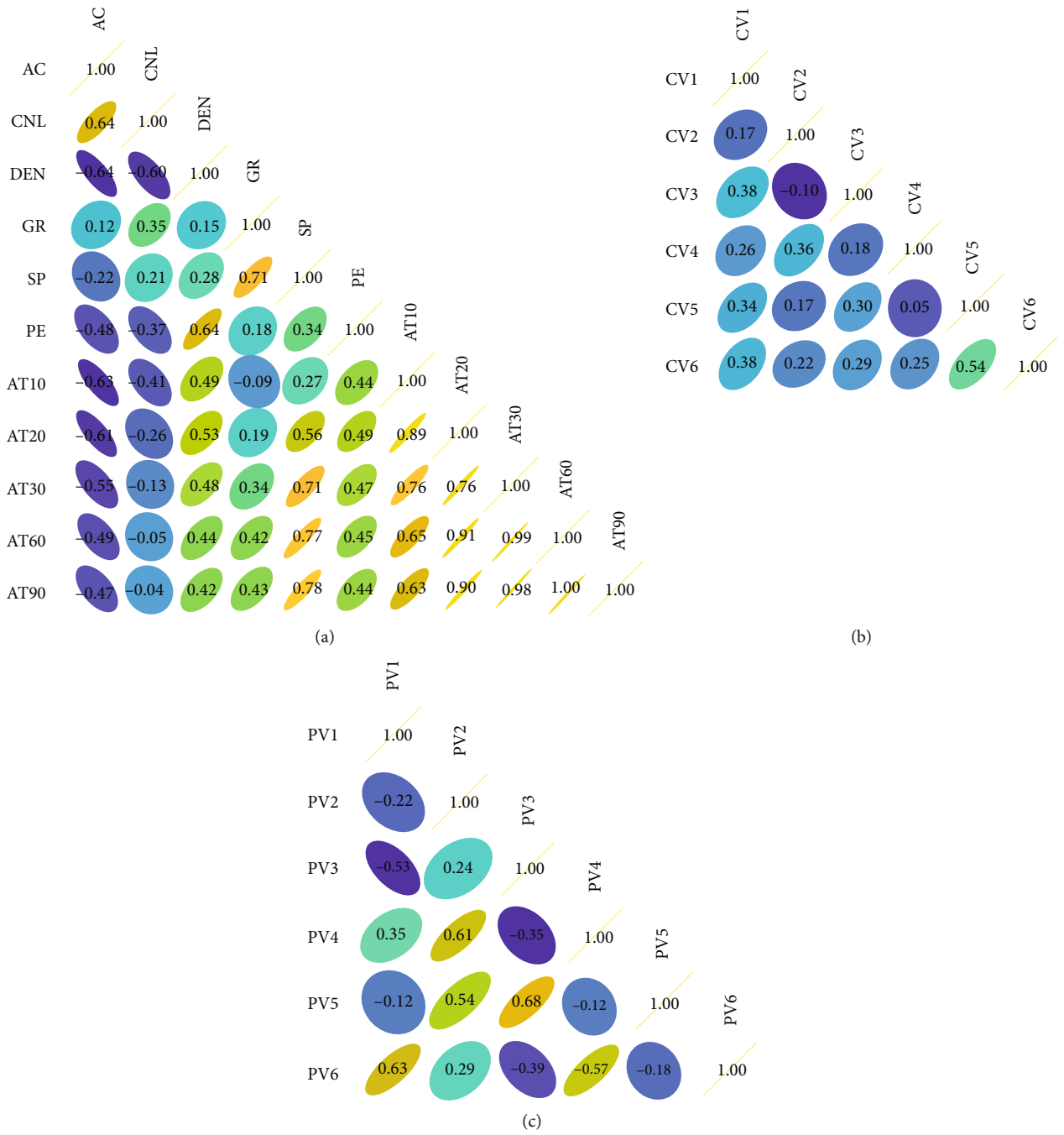


Figure 6: Correlations of applied well logs (a), variables extracted by continuous restricted Boltzmann machine (CRBM) (b), and variables gained by principal component analysis (PCA) (c). AC = acoustic log; CNL = compensated neutron log; DEN = density log; GR = gamma ray; SP = spontaneous potential; PE = photoelectric absorption cross-section index; AT10, AT20, AT30, AT60, and AT90 = resistivity of formation measured by array induction log at 10-inch, 20-inch, 30-inch, 60-inch, and 90-inch logging depth; CV1 to CV6 are variables extracted by CRBM; PV1 to PV6 are variables produced by PCA.

displayed in Figure 3(b) [51, 52]. 24 cored wells presented in Figure 3(c) are available to provide the predicting data. Routinely, when the basic computing materials are prepared, the petrophysical prediction can be directly implemented by the models listed in Table 1, but in this study, only well logs and some core-measured data of three target reservoir characters can be used, resulting in an ineffective of those models. As mentioned before, the essence of the petrophysical prediction is a logging-based regression, and then CRBM-Bayes-LightGBM is proposed as a potential fitting predictor. To validate the predicting capability of LightGBM-cored predictor, the handled materials from the reservoirs within the member of Chang 8 are collected and assembled as the raw dataset. The data from the northern subzone is provided by 20 wells as shown in Figure 3(c), and 3013 samples are assembled.

Table 4: Initial settings of light gradient boosting machine (LightGBM) predictor and optimizers, and variation ranges of hyper-parameters.

Initial setting of core predictor		Number of CARTs $(L_{cn}) = 100$[1] Learning rate $(L_{lr}) = 0.001$ Max depth of a CART $(L_{md}) = 5$ Max leafs of a CART $(L_{\ln}) = 8$ Max bins to split a node $(L_{mb}) = 8$[2] Min leafs in a node $(L_{ml}) = 2$[3] Min gain to split a node $(L_{mg}) = 0.001$[4] L1 regularization $(L_{r1}) = 0.001$ L2 regularization $(L_{r2}) = 0.001$
Initial setting of optimizer [5]	RS	Max iteration $(R_{mi}) = 50$ Next range for linearly increased variable $(R_{le}) = [x/5, 5x]$[6] Next range for exponentially increased variable $(R_{ep}) = [0.05lgx, 20lgx]$[7]
	PSO	Max iteration $(PS_{mi}) = 50$ Number of seeds $(PS_n) = 10$ Initial inertia weight $(PS_{iw}) = 0.9$ Final inertia weight $(PS_{fw}) = 0.4$ Elastic coefficients $(PS_{e1}, PS_{e2}) \in [0, 1]$[8] Acceleration coefficients $(PS_{a1}, PS_{a2}) = 2$
	Bayes	Max iteration $(B_{mi}) = 50$ Surrogate model $(B_m) =$ GP[9] Balance coefficient for acquisition function $(B_v) = 0.5$
Variation ranges of hyperparameters		$L_{cn} \in [100, 1500]$ $L_{lr} \in [0.001, 1]$ $L_{md} \in [5, 20]$ $L_{\ln} \in [8, 2048]$ $L_{mb} \in [8, 2048]$ $L_{ml} \in [2, 5]$ $L_{mg} \in [0, 2]$ $L_{r1} \in [0.001, 10]$ $L_{r2} \in [0.001, 10]$

[1]CART = classification and regression tree; [2]max bins employed by Histogram algorithm to split a leaf node; [3]min leafs required at a leaf node or there will have a cut for this leaf node; [4]min gain required to split a leaf node or the growth of this node will be ceased; [5]RS = random search; PSO = particle swarm optimization; Bayes = Bayesian optimization; [6]lower and upper limits set by a fifth and five times of target x, respectively; [7]lower and upper limits set by a twentieth and twenty times of log10 base of target x, respectively; [8]random values varying within [0,1]; [9]GP = Gaussian process.

For the southern part, only 4 wells offer the predicting materials, and the number of samples is just 280. The logging part of each sample is the same, composed of 11 well logs including acoustic log (AC, μs/m), compensated neutron log (CNL, %), density log (DEN, g/cm^3), gamma ray (GR, API), spontaneous potential (SP, mV), photoelectric absorption cross-section index (PE, b/e), and 5 array induction logs (AT10, AT20, AT30, AT60, and AT90, Ω·m). Since the set of samples offered by the southern subzone is much smaller and meanwhile the samples of two subzones are featured by the same logging sequences, the prediction of the southern subzone meets the computing rule of transfer learning and thereby will be executed in accordance with the workflow shown in Figure 2 [39, 40]. Accordingly, the pretrained predictor will be established by the data derived from the northern subzone, and the computing flow should comply with Figure 1.

Consequently, three experiments are designed purposefully to implement the data validation. The first experiment will verify the computing capability of the proposed predictor based on the application of the data of the northern subzone. The second one will testify whether the working performance can be improved when more learning samples are trained. In the last experiment, the data of the southern subzone will be predicted under a combination of ensemble and transfer learnings to demonstrate whether the workflow given by Figure 2 is applicable. The platform for the following computation is Spyder 3 (Python 3.7.6).

3.2. The First Experiment. Since the second experiment will apply more samples to conduct a test, 2513 samples are preserved in the first experiment, and the rest 500 samples are left to the next validation. According to the computing flow

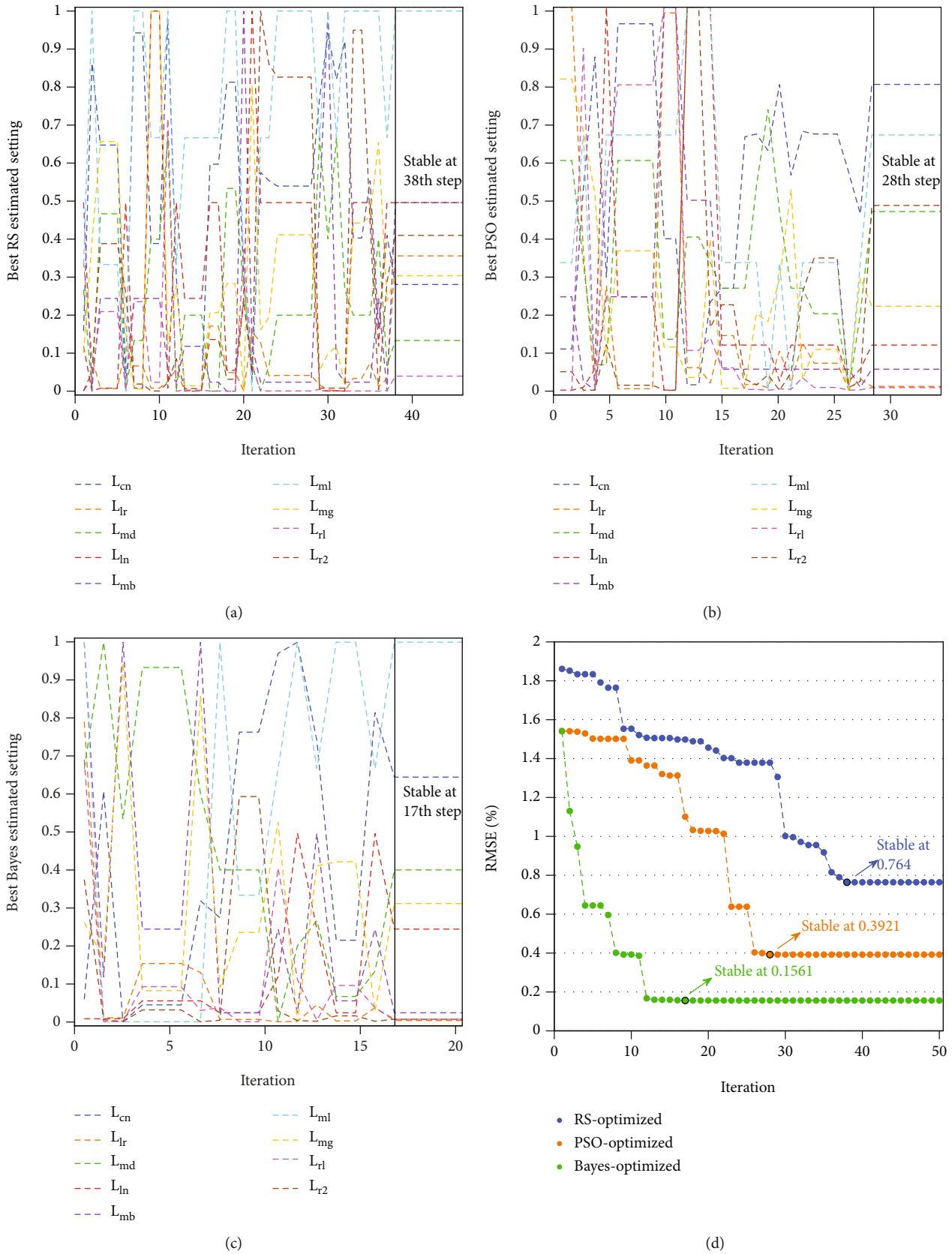


Figure 7: Variations of hyperparameters of light gradient boosting machine (LightGBM) implemented by RS (a), PSO (b), and Bayes (c), and downtrends of RMSE of porosity values generated by three applied optimizers during the whole iteration (d). RS = random research; PSO = particle swarm optimization; Bayes = Bayesian optimization; RMSE = root-mean-square error.

Table 5: Initial settings of competitive predictors and variation ranges of applied hyperparameters.

Core predictor [1]	Initial setting	Variation ranges of hyperparameters
KNN	Number of neighbors $(K_n) = 5$ Solver $(K_{al}) = [1, 2]^{2}$ Distance function $(K_{df}) = [1, 2]^{3}$	$K_n \in [5, 50]$ $K_{al} \in [1, 2]^{8}$ $K_{df} \in [1, 2]^{9}$
SVR	$S_v = 0.1^{4}$ Regularization $(S_C) = 0.01$ Smoothing factor $(S_\sigma) = 0.01^{5}$ Kernel function (S_{kf}) = RBF*	$S_v \in [0.1, 0.6]$ $S_C \in [0.01, 10]$ $S_\sigma \in [0.01, 1]$
RF	Number of CARTs $(R_{cn}) = 100^{6}$ Max depth of a CART $(R_{md}) = 5$ Min samples to split a node $(R_{ms}) = 5^{7}$ Min leafs in a node $(R_{ml}) = 2$	$R_{cn} \in [100, 1500]$ $R_{md} \in [5, 20]$ $R_{ms} \in [5, 20]$ $R_{ml} \in [2, 5]$

[1]KNN = k-nearest neighbors; SVR = supper vector regression; FR = random forest; [2]"1" means KD-tree, "2" means Ball-tree; [3]"1" means Manhattan distance, "2" means Euclidean distance; [4]an upper bound on the fraction of training errors and a lower bound of the fraction of support vectors; [5]control the window length of each probability density distribution; [6]CART = classification and regression tree; [7]min samples required to split a leaf node or the growth of this node will be ceased; [8]only 1 or 2 will be chosen during iteration; [9]only 1 or 2 will be chosen during iteration; * RBF = radial basis function, which is a non-hyperparameter.

shown in Figure 1, the first task is preprocessing. The detection of outliers will be executed by Equation (1). Table 2 displays a summary of quartile-based statistical information of all well logs. Prior to the removing of outliers, one problem should be considered that since each kind logging value will play as an outlier, a sample will contain one or more outliers and then how to conduct a judgment on the detection becomes a key. Here, a rule is defined that a sample containing three or more outliers will be viewed as a noisy sample and then must be excluded. Figure 4 provides a related illustration. Based on this rule, 18 noisy samples are detected by the computed values of LIF and UIF, and thus, the number of the used samples is 2495. Subsequently, all logs have to be normalized. The normalizing range is set by [0,1], which means the variation of each log will be restricted within an interval between 0 and 1.

Next task is the dimensional reduction, which will be implemented by CRBM. To make a contrast, principal component analysis (PCA), another classic approach used to reduce the dimensionality of inputs is introduced. Upon the previous findings, the empirical as well as useful initial settings of CRBM and PCA are given in Table 3 [41, 45, 53, 54]. As 11 well logs are employed, the size of visible layer is 11, and to create a fast prediction, a half reduction of input well logs is required, and thus, the size of the hidden is 6. To be fair, the number of reserved variables of PCA should be same and accordingly is assigned by 6. For the computing of CRBM, the transiting quality should be checked primarily. Figure 5 presents the reconstructing situation of two exampled logs, and the better matching in any subplot well demonstrates the transiting is qualified. After the working of CRBM and PCA, there remains a question that how to argue the variables extracted by which approach are more effective. Commonly, for the regression, the collinearity of independent variables will dramatically affect the reliability of fitting results, and then the input variables should be nonlinear as much as possible [12–15]. In this way, the correlation of variables will be a good illustration to rise an argument. Generally, if the correlation coefficient is larger than 0.5, the related two variables will be considered in a collinear relationship [8, 12–15]. Figure 6(a) shows the correlation of all well logs, and from the values it can be known that most logs are collinear, especially for the array induction logs. Then, the direct usage of all logs is unsuitable to launch a petrophysical regression. Figures 6(b) and 6(c) display the correlation of the extracted variables of CRBM and PCA, respectively. Given a counting, only one coefficient larger than 0.5 (CV5-CV6) is found in Figures 6(b), and 6 unexpected coefficients are discovered in Figure 6(c), clarifying that most variables produced by CRBM are nonlinear and therefore the output information from the CRBM computation is more beneficial for the following fitting. Through the dimensional reduction, the real basic dataset used for the prediction is achieved.

Now, the process comes into the modeling and optimizing. 2075 samples are chosen randomly to construct the learning dataset, and the rest is employed as a test dataset. A suggested initial setting of LightGBM is shown in Table 4 [36–38]. To stress the optimizing effect of Bayes, RS and particle swarm optimization (PSO) which is a representative of SI are adopted as competitive optimizers.

The settings of three optimizers are given in the middle part of Table 4 [42, 48–50, 55–59]. 5-fold cross-validation is employed to generalize the optimizing results according to the design of the computing flow. Then, in each cross validation, 415 learning samples will be predicted in the optimization. Figures 7(a)–7(c) illustrate the variations of all hyperparameters of LightGBM, respectively, implemented by RS, PSO, and Bayes. Each subplot indiscriminately shows a fierce variation, implying the optimal parametric setting is very different with the initial one and then emphasizing the significance of the optimization in the modeling. Figure 7(d) displays a measure of the optimizing results. Although every optimizer presents a downtrend on the RMSE evaluation,

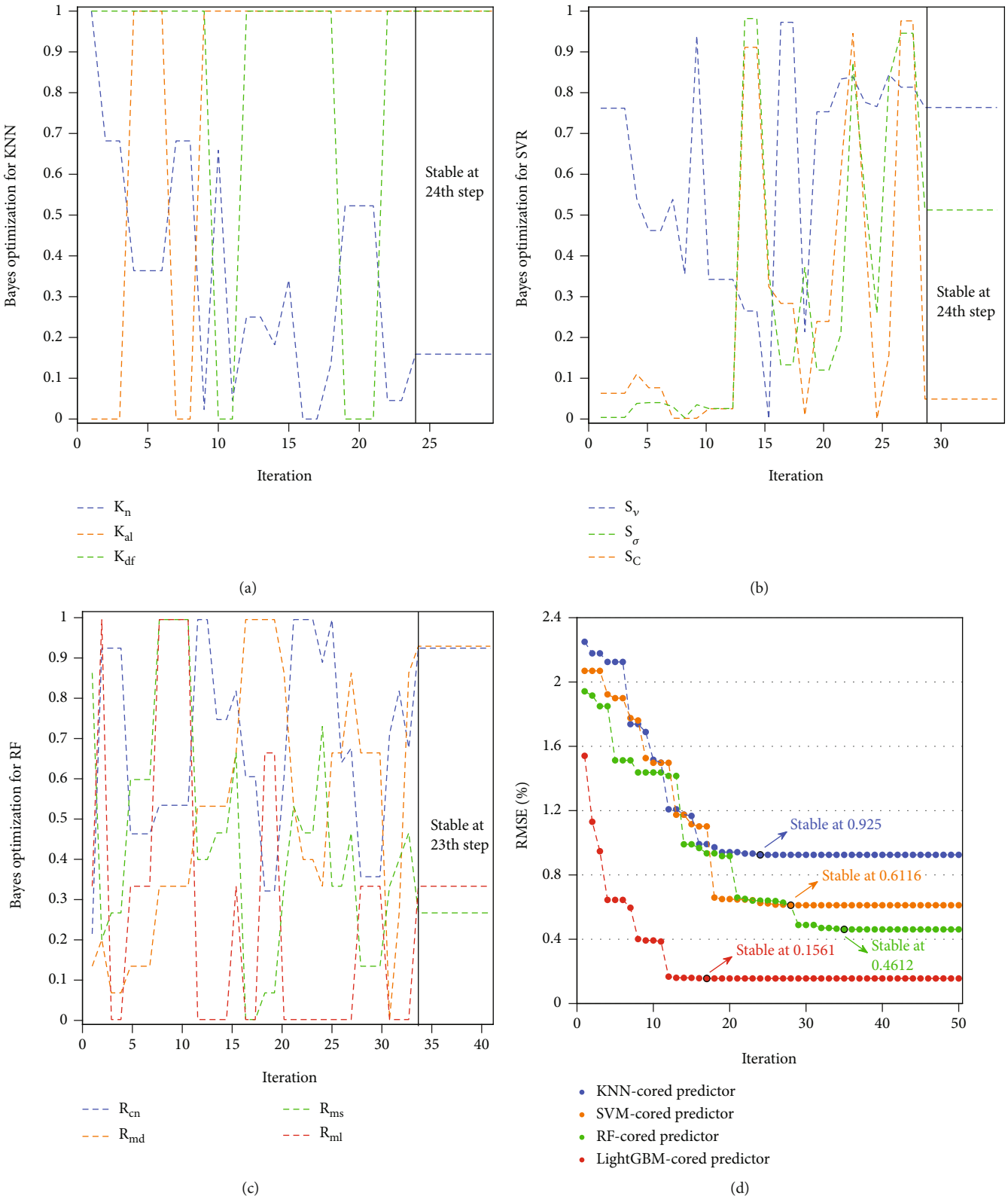


Figure 8: Variations of hyperparameters of KNN (a), SVR (b), and RF (c) implemented by Bayesian optimization and downtrends of RMSE of porosity values obtained by four validated predictors during the whole iteration (d). KNN = k-nearest neighbors; SVR = support vector regression; RF = random forest; LightGBM = light gradient boosting machine; RMSE = root-mean-square error.

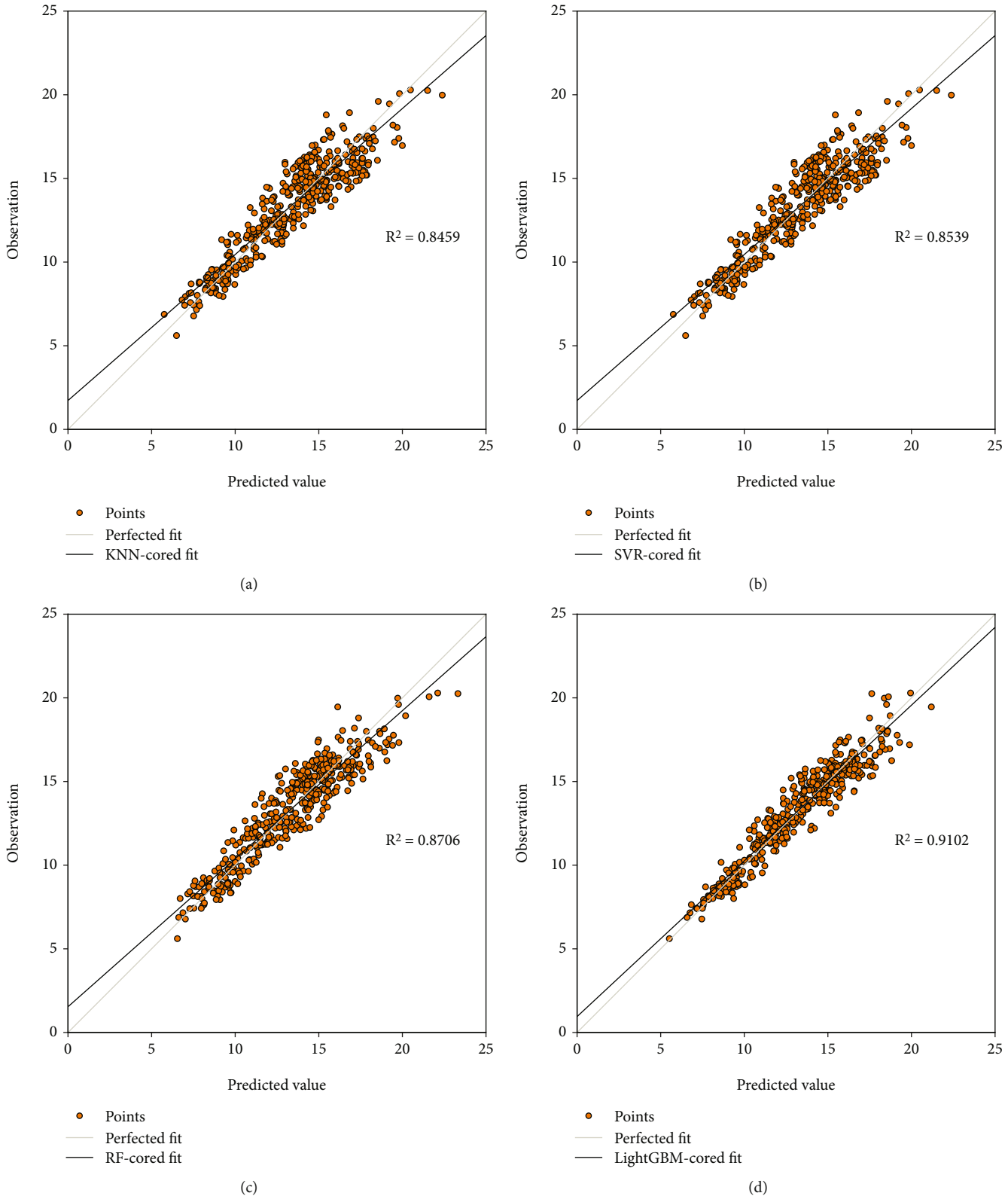


Figure 9: Fitness of porosity results provided by KNN-cored predictor (a), SVR-cored predictor (b), RF-cored predictor (c), and LightGBM-cored predictor (d) in the first experiment. KNN = k-nearest neighbors; SVR = support vector regression; RF = random forest; LightGBM = light gradient boosting machine; RMSE = root-mean-square error.

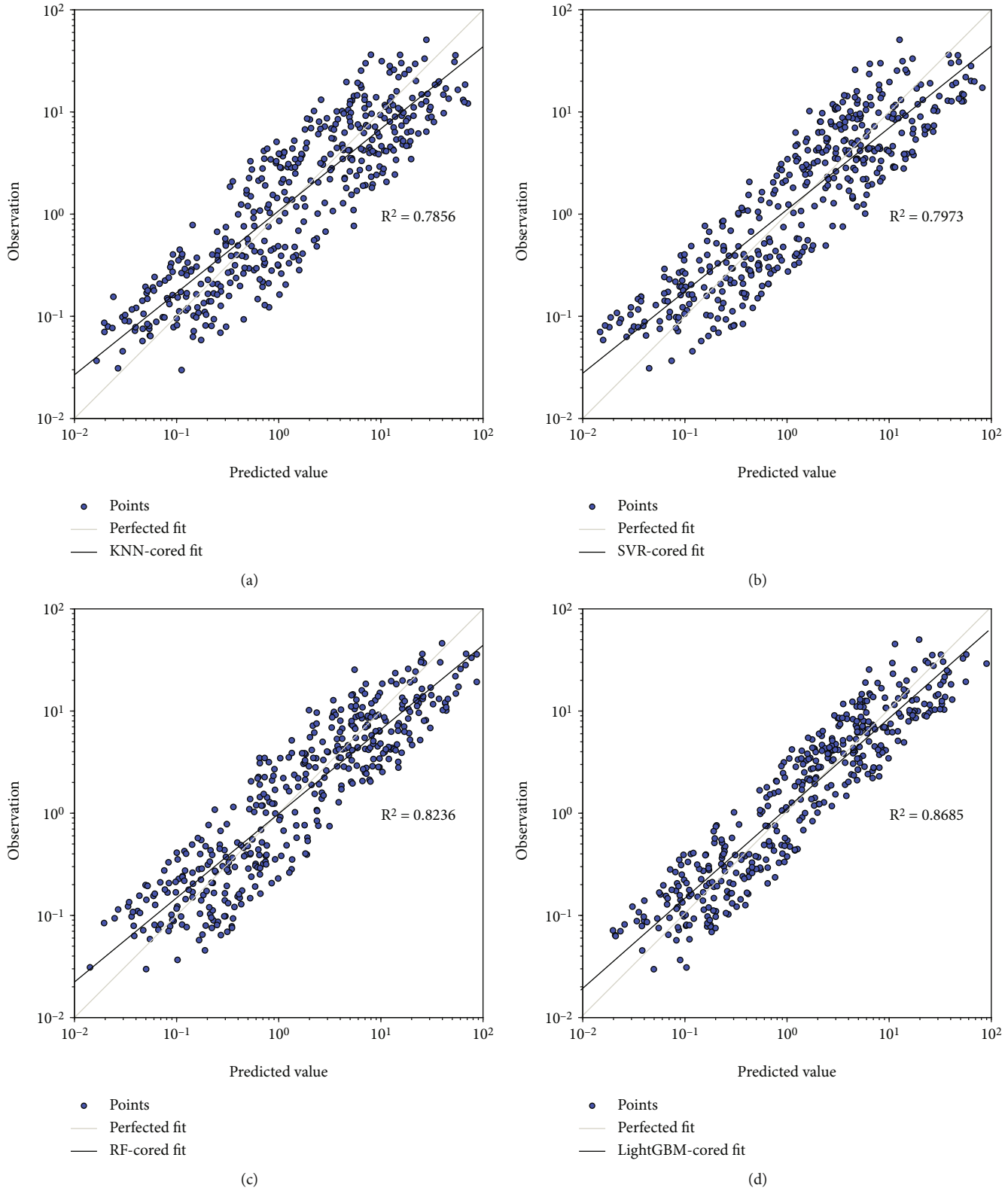


Figure 10: Fitness of permeability results provided by KNN-cored predictor (a), SVR-cored predictor (b), RF-cored predictor (c), and LightGBM-cored predictor (d) in the first experiment. KNN = k-nearest neighbors; SVR = support vector regression; RF = random forest; LightGBM = light gradient boosting machine; RMSE = root-mean-square error.

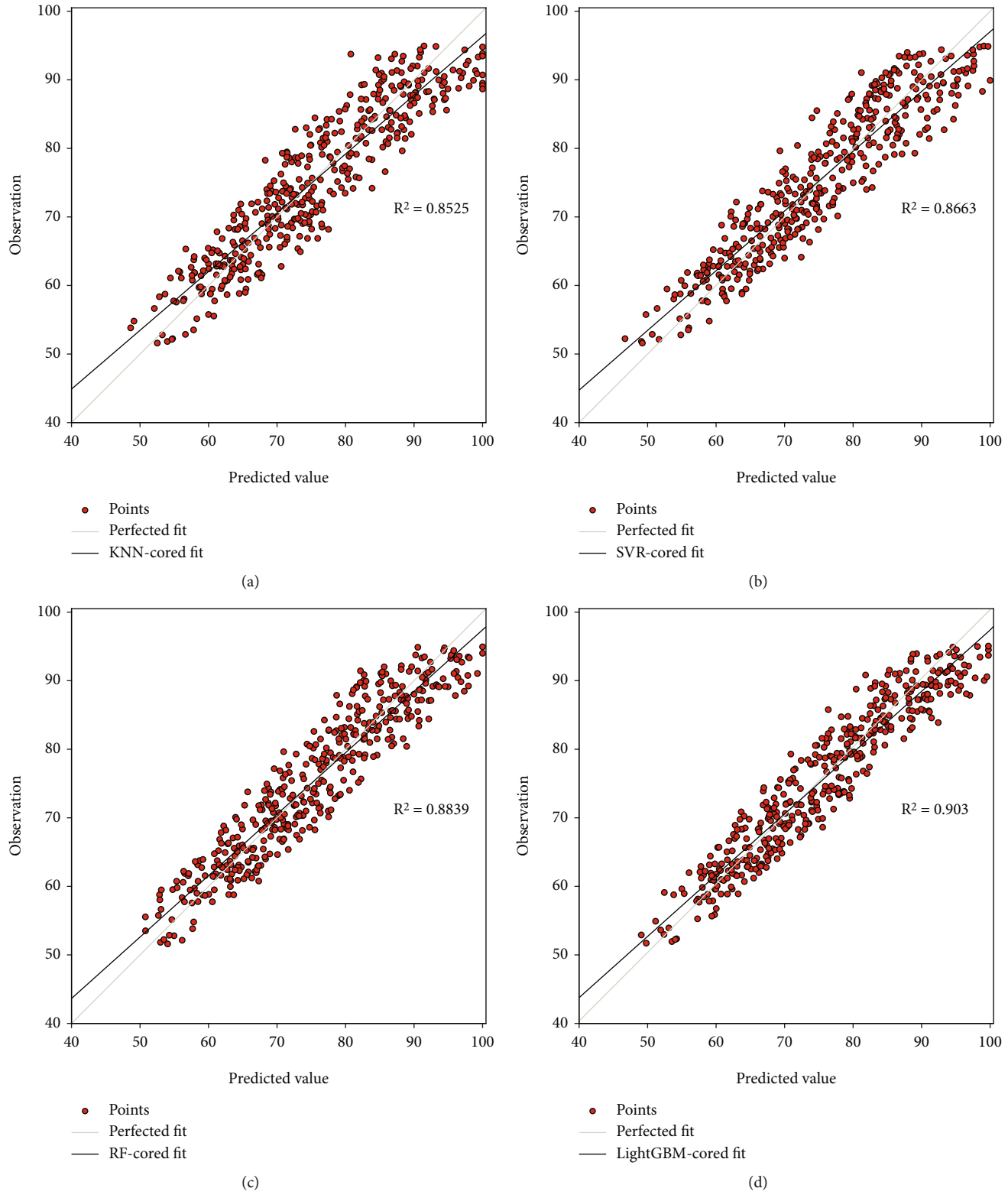


Figure 11: Fitness of water saturation results provided by KNN-cored predictor (a), SVR-cored predictor (b), RF-cored predictor (c), and LightGBM-cored predictor (d) in the first experiment. KNN = k-nearest neighbors; SVR = support vector regression; RF = random forest; LightGBM = light gradient boosting machine; RMSE = root-mean-square error.

Table 6: Measured information of porosity, permeability, and water saturation gained by four validated predictors in three experiments.

Model [1]	RMSE results for porosity (%) evaluated by Equation (7) [2]		
	1st experiment	2nd experiment	3rd experiment nontrans/trans [3]
KNN-cored predictor	1.2048	0.9882	0.8234/0.5543
SVR-cored predictor	1.1751	0.9215	0.7436/0.5168
RF-cored predictor	1.1024	0.8502	0.6466/0.4599
LightGBM-cored predictor	0.8947	0.7091	0.6096/0.3934
	RMSE results for permeability (mD) evaluated by Equation (8)		
	1st experiment	2nd experiment	3rd experiment nontrans/trans
KNN-cored predictor	0.4027	0.3466	0.1375/0.1015
SVR-cored predictor	0.3956	0.3440	0.1700/0.1189
RF-cored predictor	0.3660	0.3201	0.1499/0.1070
LightGBM-cored predictor	0.3024	0.2510	0.1115/0.0761
	RMSE results for water saturation (%) evaluated by Equation (7)		
	1st experiment	2nd experiment	3rd experiment nontrans/trans
KNN-cored predictor	4.5537	3.8576	6.5779/4.8992
SVR-cored predictor	4.3016	3.7810	6.1735/4.6108
RF-cored predictor	3.9348	3.4938	5.7748/4.3459
LightGBM-cored predictor	3.6630	2.8941	4.8863/3.2708

[1]KNN = k-nearest neighbors; SVR = supper vector regression; FR = random forest; LightGBM = light gradient boosting machine; [2]RMSE = root-mean-square error; [3]"nontrans" means normal prediction, and "trans" stands for the prediction implemented under the support of transfer learning.

the Bayes-optimized line gains the smallest value, and meanwhile, the iteration of Bayes is ceased most early, which then strongly argues that Bayes is higher-efficient in comparison with RS and PSO in the optimization. As a better optimizing effect is demonstrated by Bayes, the reasonability of the integration of CRBM and Bayes for LightGBM is proved.

After the optimization is completed, the construction of LightGBM or the expression of Equation (4) can be confirmed. To highlight the predicting performance, three sophisticated fitting models are introduced as competitors, including KNN, SVR, and RF. Since these competitive solvers also employ hyperparameters to implement the modeling, CRBM and Bayes will also be applied as assistants, and then the real names of three competitors are CRBM-Bayes-KNN, CRBM-Bayes-SVR, and CRBM-Bayes-RF, respectively. The initial settings empirically used for three competitive predictors are displayed in Table 5 [23–25, 27–29, 32–34]. Figures 8(a)–8(c) record the Bayes-optimized variations of the hyperparameters of KNN, SVR, and RF, respectively. Similarly, for any subplot, the frequent changing of each hyperparameter manifests that the initial setting is far from the optimized status, once again underlining the essential application of the optimizer in the ML-based prediction. The RMSE estimation for the training of four validated predictors is presented in Figure 8(d). Obviously, the proposed predictor still holds the smallest RMSE and gains this score at the earliest iteration, indicating that LightGBM-cored predictor comparatively has the higher efficiency in the prediction and also implying that this predictor could have greater potential to produce the reliable results in the practical prediction.

When all predictors are trained, the test dataset composed by 420 samples will be predicted in the final stage of the data process. Figures 9–11 exhibit the fitting between the observations and the predicted results of porosity, permeability, and water saturation, respectively. Figure 9 here is exampled. If the predicted values are closer to the observations, a larger R^2 will be gained [12–15]. Hence, through a comparison, LightGBM-cored predictor becomes the winner owning to its largest R^2. For other two figures, the R^2 information also points out the proposed predictor achieves a victory. Moreover, Table 6 summarizing the experimental information presents that no matter in what kind of petrophysical regression, the smallest RMSE value is always held by LightGBM-cored predictor. Overall, given the better working performance or the more reliable experimental results, CRBM-Bayes-LightGBM is proved more capable in the regression of porosity, permeability, and water saturation.

3.3. Second Experiment. Generally, for a ML-based predictor, training more samples will reinforce the input-output mapping, and then a better prediction will be obtained [17–19]. Therefore, in this experiment, all learning samples will be used, and the aim is to validate whether the application of a large set of learning samples can raise an enhancement on the predicting capability of the proposed predictor. The number of learning samples has now reached up to 2575. All training conditions for four validated predictors will follow the previous ones. Figures 12–14 illustrate the fitting of porosity, permeability, and water saturation, respectively. Similarly, Figure 12 is selected as an instance. For each subplot, since the fitting line produced in this experiment is closer to the perfect fit in comparison with the previous one, a larger R^2 is obtained, and thus, one thing is verified that training more learning samples is indeed effective to

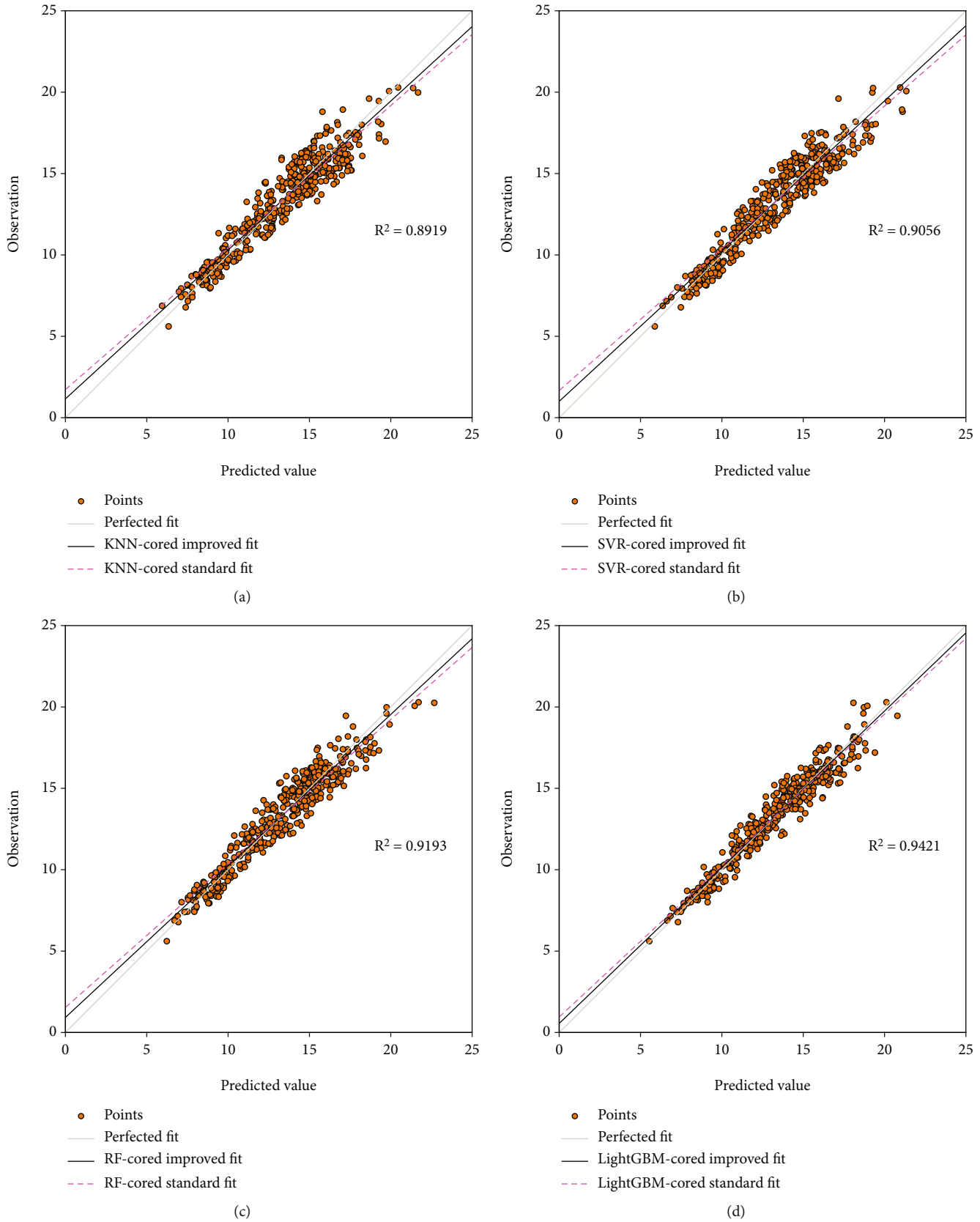


Figure 12: Fitness of porosity results provided by KNN-cored predictor (a), SVR-cored predictor (b), RF-cored predictor (c), and LightGBM-cored predictor (d) in the second experiment. "standard fit" means the fitting is gained in the first experiment; KNN = k-nearest neighbors; SVR = support vector regression; RF = random forest; LightGBM = light gradient boosting machine; RMSE = root-mean-square error.

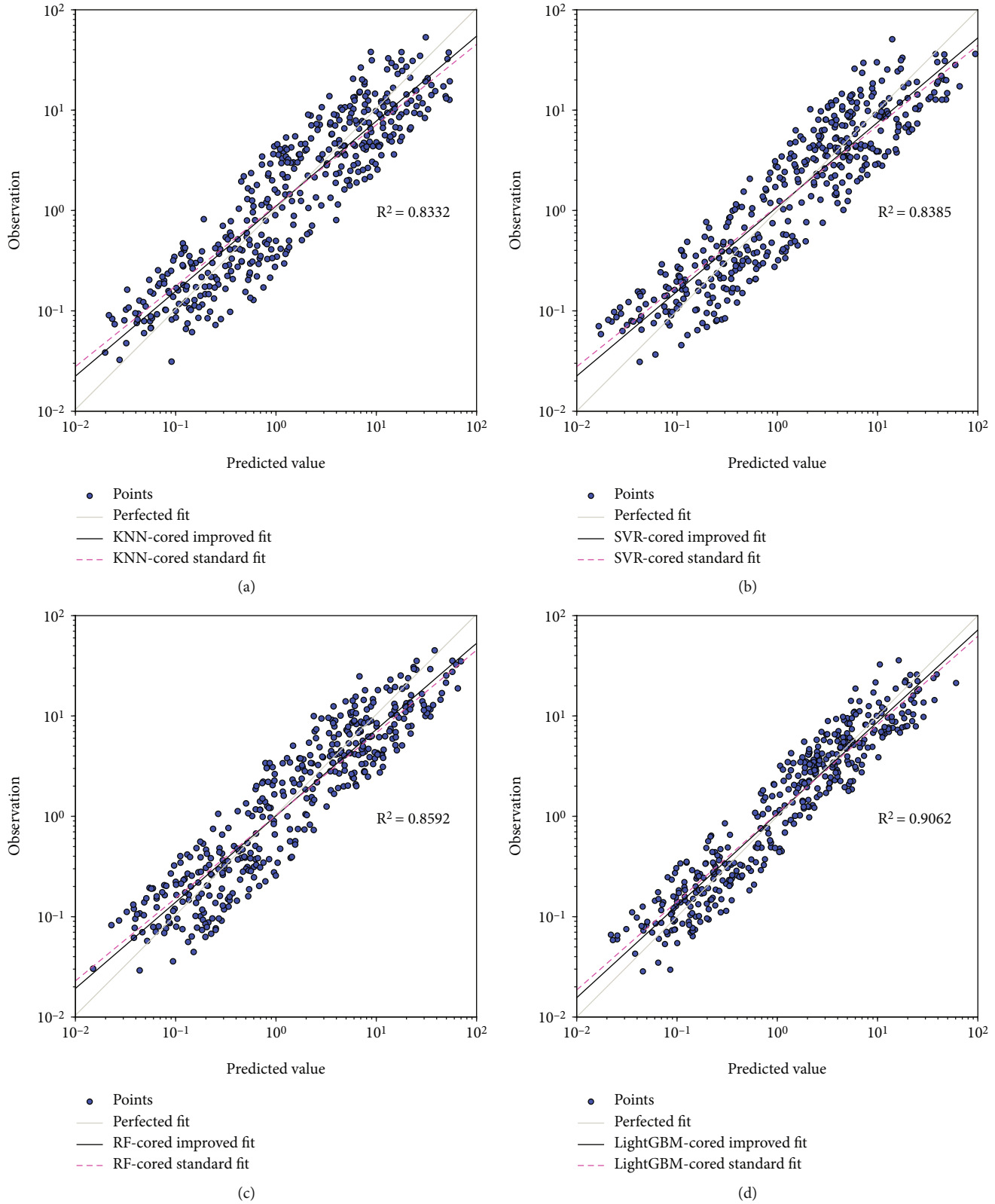


Figure 13: Fitness of permeability results provided by KNN-cored predictor (a), SVR-cored predictor (b), RF-cored predictor (c), and LightGBM-cored predictor (d) in the second experiment. "standard fit" means the fitting is gained in the first experiment; KNN = k-nearest neighbors; SVR = support vector regression; RF = random forest; LightGBM = light gradient boosting machine; RMSE = root-mean-square error.

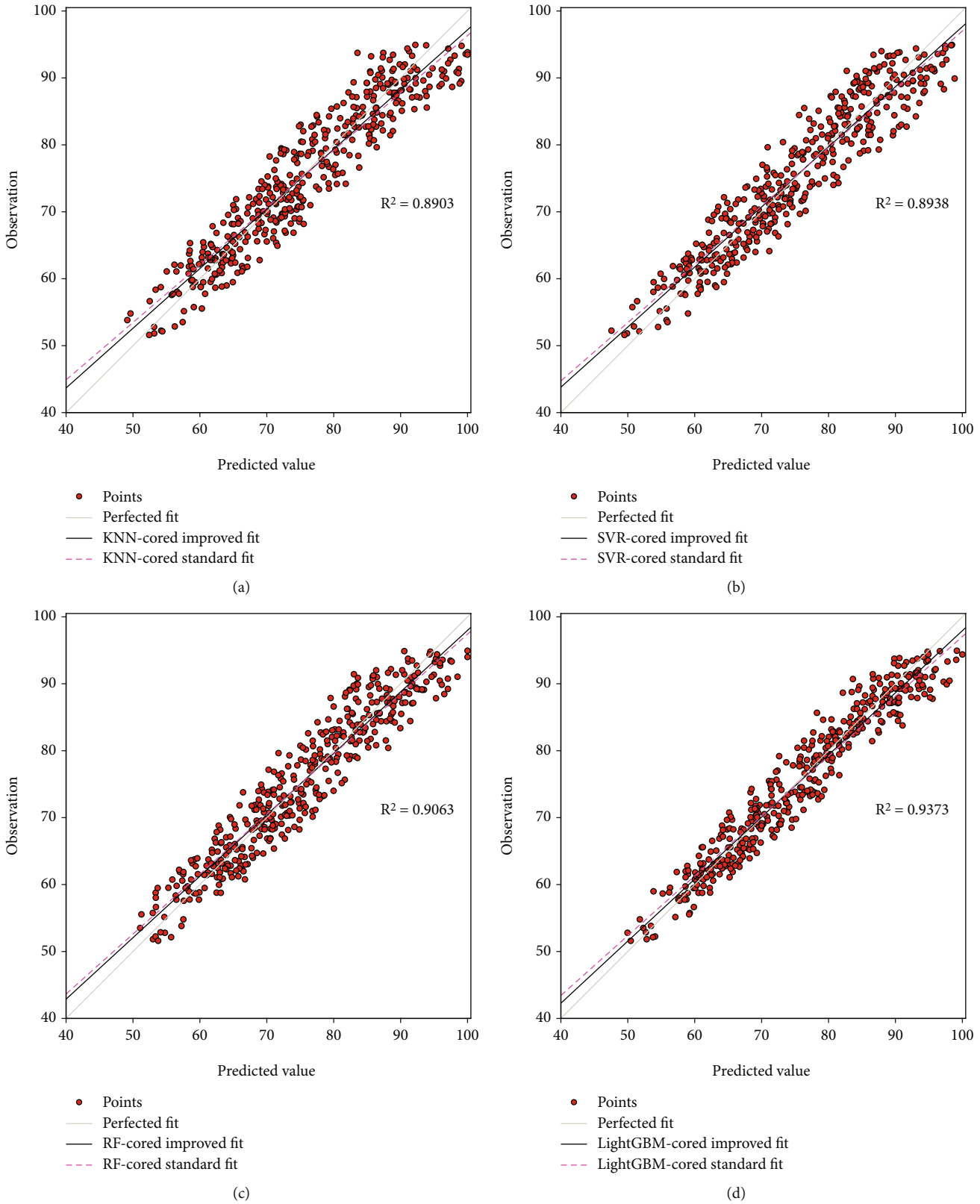


Figure 14: Fitness of water saturation results provided by KNN-cored predictor (a), SVR-cored predictor (b), RF-cored predictor (c), and LightGBM-cored predictor (d) in the second experiment. "standard fit" means the fitting is gained in the first experiment; KNN = k-nearest neighbors; SVR = support vector regression; RF = random forest; LightGBM = light gradient boosting machine; RMSE = root-mean-square error.

improve the working performance of any competitive predictor. Figures 13 and 14 and the RMSE information shown in Table 6 also manifest the same conclusion, which then solidly demonstrate the benefit of the application of more learning samples in a prediction. Besides, under a contrast, a better score in terms of R^2 or RMSE estimation is yet generated by LightGBM-cored predictor, arguing once again that the proposed predictor can be viewed as a preferential selection for the petrophysical regression.

3.4. Third Experiment. The data derived from the southern subzone will be trained and predicted in this experiment. As aforementioned, this dataset is much small, only containing 280 samples, and then its operation needing the support of transfer learning. The workflow will reference Figure 2. Based on the detection of outliers, 2 noisy samples are labeled, and thus, the number of the used samples is 278. 75 samples are chosen in a random pattern to assemble the test dataset, and the rest ones compose the learning dataset. The pretrained section for each validated predictor follows the one obtained in the 2nd experiment. Other training conditions yet employ the settings given by the 1st experiment. When the modeling work is completed, the prediction can be implemented. Figures 15–17 exhibit the fitting of three target reservoir characters. For the regression case of porosity as shown in Figure 15, two kinds of information are revealed: (1) without the usage of transfer learning, the fitting marked by pink color presents much unreliably in any subplot; (2) after equipping with transfer learning, the predictor becomes capable to yield a better fit and a higher R^2. The content shown in Figures 16 and 17 is similar, which then indicate that the training of a small-volumetric dataset is accessible for any competitor to produce a qualified petrophysical regression under the application of transfer learning. The RMSE information given by Table 6 is another proof for this indication. Furthermore, since the proposed predictor still performs better due to its higher R^2 in the related figures and lower RMSE values in Table 6, a fact is fully evidenced that no matter what kind of dataset there uses, LightGBM-cored predictor always can gain more satisfactory predicted results and then present with a better generalization and stronger robustness. Consequently, the proposed predictor acquires a complete victory in three experiments, acting as a more intelligent solver for the petrophysical regression.

3.5. Discussion. In the 1st experiment, the selection of approach in the dimensional reduction and optimization is demonstrated comparatively. To create a fast prediction, less input variables are required, and then a reduction should be implemented on the dimensionality of inputs. Since the task is a regression, the independent variables should be nonlinear as much as they possibly can to avoid the occurrence of collinearity, and then the correlation of input variables becomes an accessible indicator to measure the quality of the dimensional reduction. Figures 6(b) and 6(c) indicate that compared to the output of PCA, the variables extracted by CRBM are more nonlinear as only one pair of collinear variables is created, which then testifies that the reduction executed by CRBM is more beneficial for the following regression. For the demonstration of the optimizing section, Figure 7(d) evidences the superiority of Bayes because this optimizer can gain a lower RMSE estimation and simultaneously figure out this score at an earlier iteration in comparison with RS and PSO. Therefore, given the comparative analysis of the experimental results, the integration of CRBM and Bayes for the core predictor is proved both reasonable and effective in the petrophysical regression.

A simple as well as practical approach to enhance the predicting capability of ML-based models is to apply more learning samples during the training. The theoretical explanation is that with the usage of more learning samples, the input-output mapping established in the training stage will be reinforced and given that the model will become capable to produce a more satisfactory prediction. Then, in the 2nd experiment, a larger set of learning samples is employed to complete a petrophysical fitting. Through the observation of Figures 12–14 and a comparative analysis among the values in Table 6, one thing can be confirmed that the results generated from the 2nd experiment are more qualified than the previous ones. Thereby, the operation of training more learning sample is demonstrated applicably to raise an improvement on the working performance of any validated predictor. Accordingly, in the petrophysical regression, if the fitting results are unsatisfactory, training a larger set of samples will be a smart alternative.

Sometimes, in the petroleum exploration, fewer logging materials will be available, and then when only a small-volumetric dataset can be applied for the fitting of porosity, permeability, or water saturation, ML-based models extremely will encounter an underfitting. Given the computing theory of transfer learning, the underfitting caused by a smaller set of samples can be well addressed by a pretrained model constructed by a larger set of samples, but a precondition is that all samples used should be featured similarly. Since fewer cored wells within the southern subzone as shown in Figure 3(c) are available while the compositions of the handled logging sequences for two subzones are same, the prediction for the southern subzone actually meets the computing mechanism of transfer learning. Hence, the third experiment is designed to verify whether an expected petrophysical regression can be gained for the southern subzone under the support of transfer learning. According to the workflow displayed in Figure 2, several validations are conducted, and given the results both shown in Figures 15–17 and recorded in Table 6, a fact is strongly argued that the regression of a small-volumetric dataset for any predictor will be very unqualified without the application of transfer learning, whereas by taking advantage of the pretrained information gained on the basis of transfer learning, any predictor established from the training of a small set of samples will become capable to produce the expected fitting results. Then, the content exhibited in Figure 2 is proved feasible, which can be employed in the practical case.

Last but by no means least, upon a comprehensive analysis of the information both illustrated in Figures 9–17 and given in Table 6, it is discovered that the larger R^2 and smaller RMSE are always held by LightGBM-cored predictor, which then solidly demonstrate that no matter what kind of dataset there uses or what kind of prediction there has, the proposed prediction always can be regarded as a

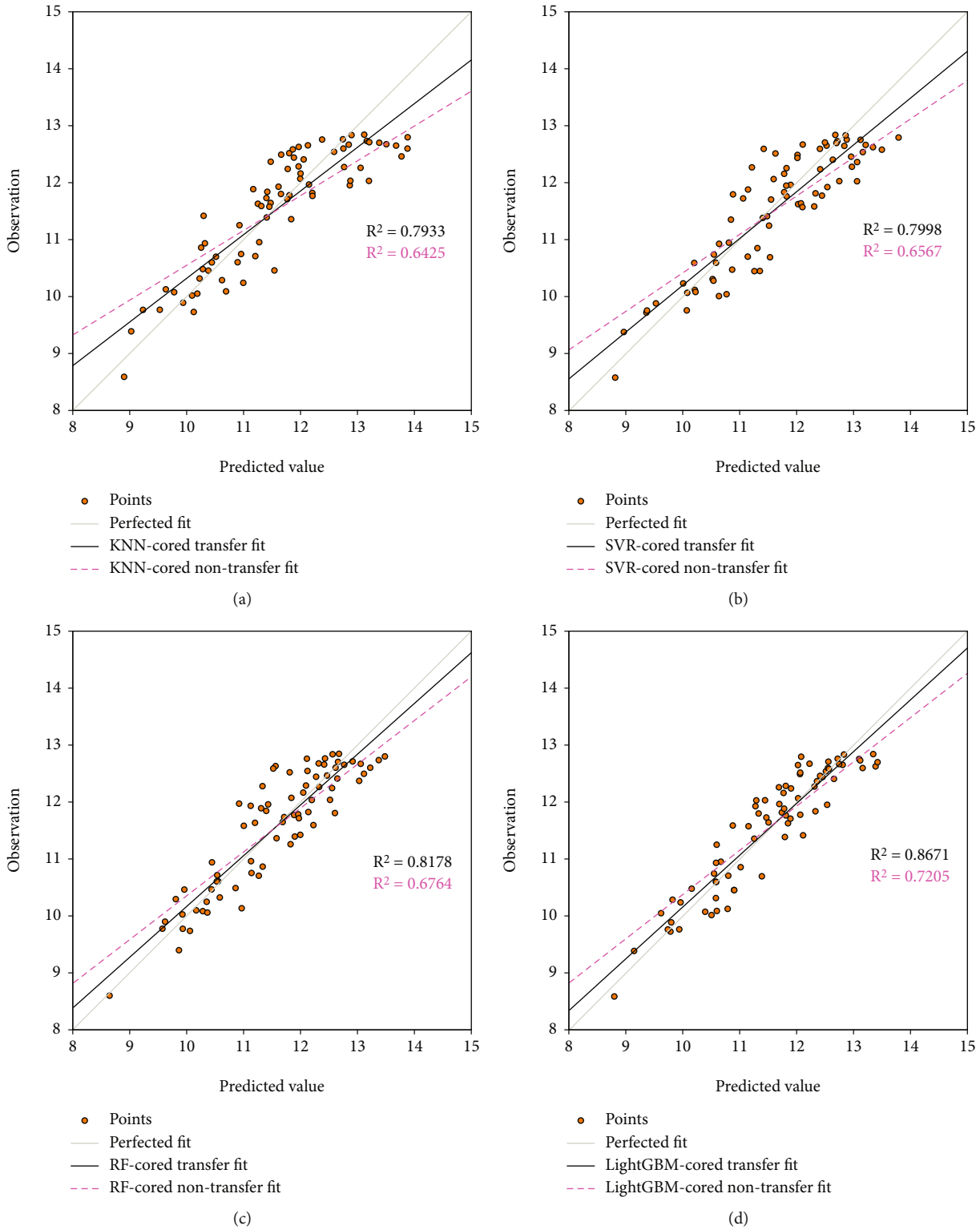


Figure 15: Fitness of porosity results provided by KNN-cored predictor (a), SVR-cored predictor (b), RF-cored predictor (c), and LightGBM-cored predictor (d) in the third experiment. KNN = k-nearest neighbors; SVR = support vector regression; RF = random forest; LightGBM = light gradient boosting machine; RMSE = root-mean-square error.

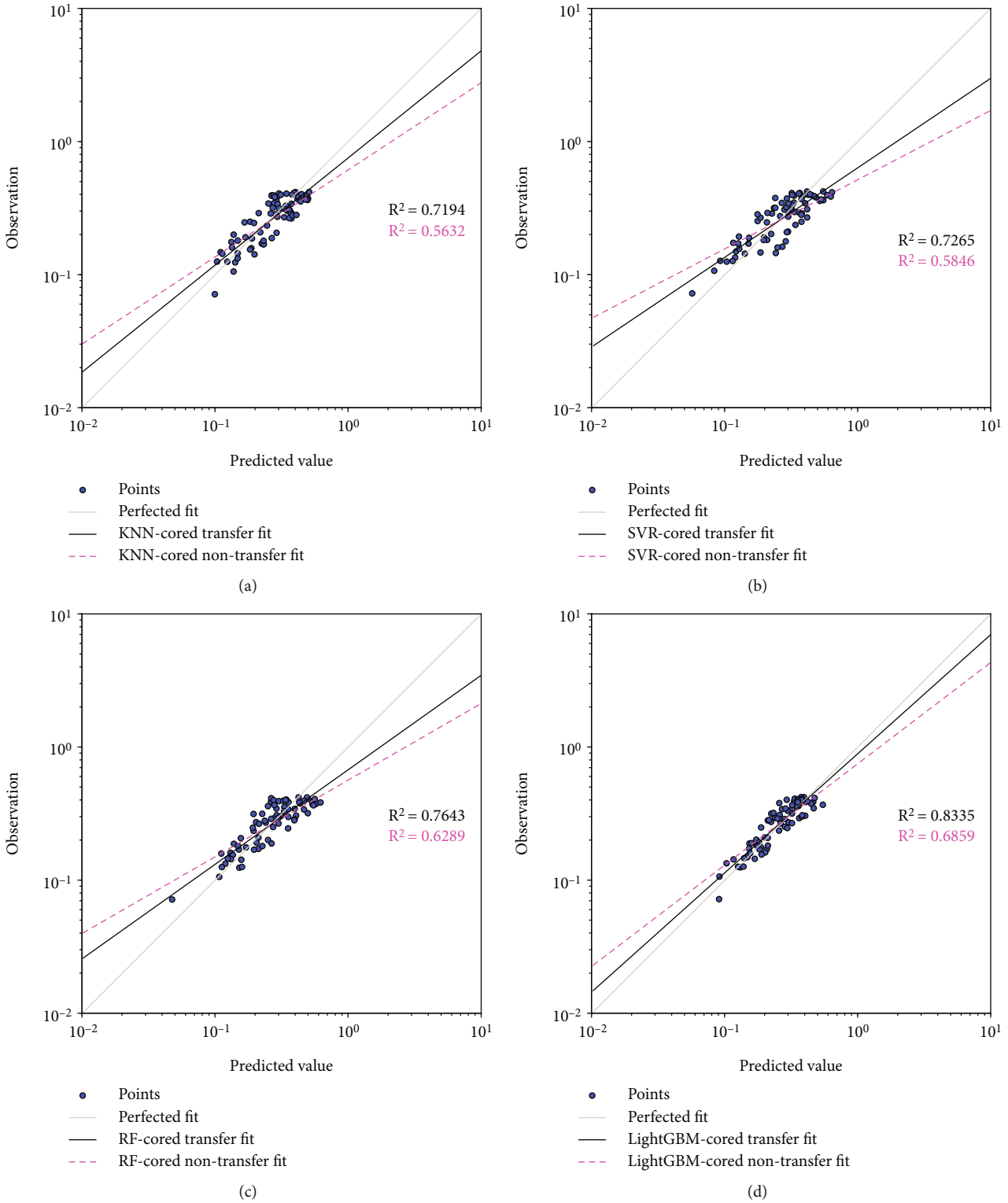


Figure 16: Fitness of permeability results provided by KNN-cored predictor (a), SVR-cored predictor (b), RF-cored predictor (c), and LightGBM-cored predictor (d) in the third experiment. KNN = k-nearest neighbors; SVR = support vector regression; RF = random forest; LightGBM = light gradient boosting machine; RMSE = root-mean-square error.

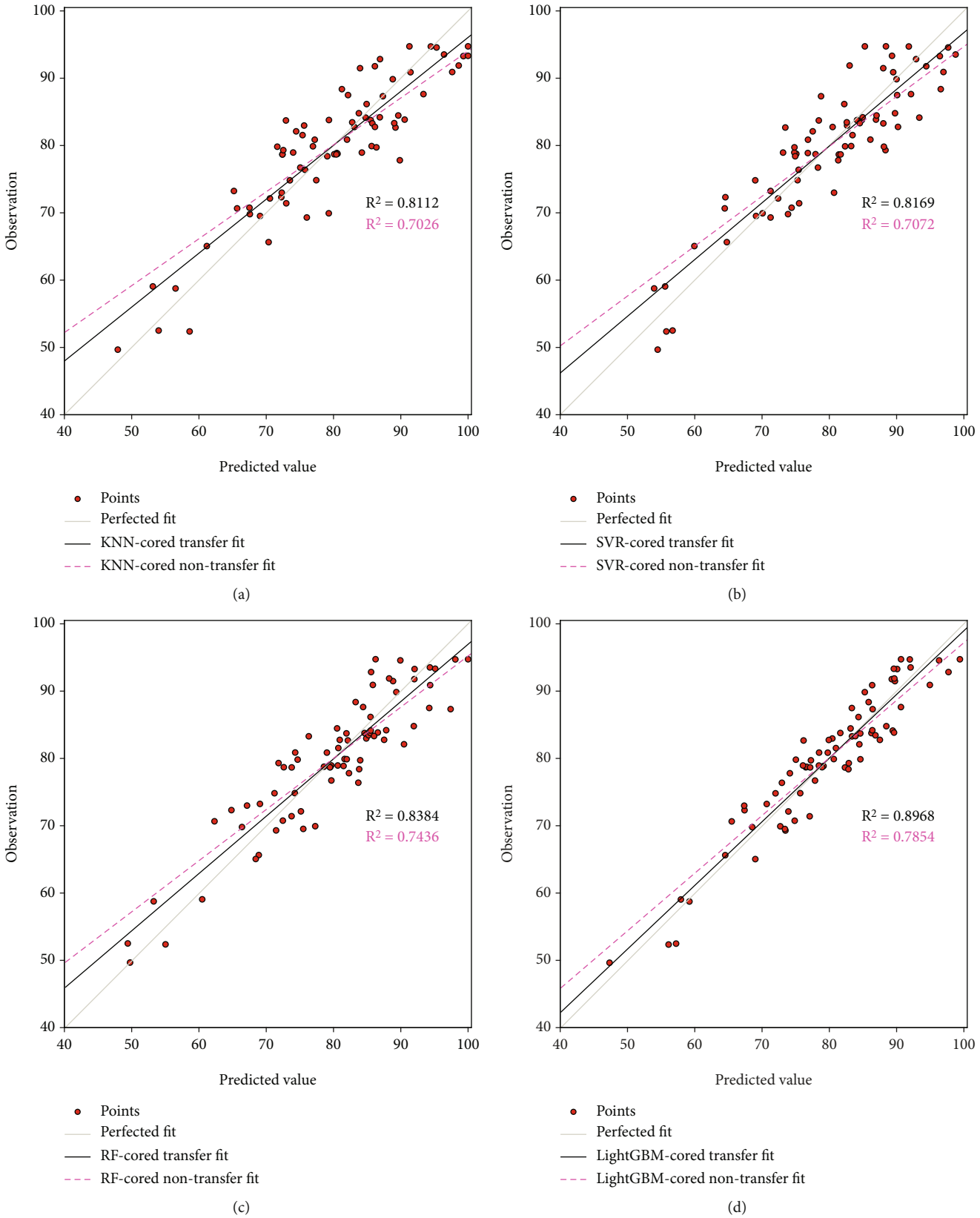


Figure 17: Fitness of water saturation results provided by KNN-cored predictor (a), SVR-cored predictor (b), RF-cored predictor (c), and LightGBM-cored predictor (d) in the third experiment. KNN = k-nearest neighbors; SVR = support vector regression; RF = random forest; LightGBM = light gradient boosting machine; RMSE = root-mean-square error.

more intelligent solver for the petrophysical regression in comparison with other three competitors. Consequently, since a stronger predicting capability and a better robustness are evidenced for the proposed predictor, CRBM-Bayes-LightGBM combined with transfer learning deserves a more widespread application in the petrophysical regression.

4. Conclusion

Given a comprehensive as well as comparative analysis of the experiment results, some critical points regarding the working performance of four employed predictors in the regression of porosity, permeability, and water saturation are summarized as follows:

(1) To create a fast and effective regression and meanwhile to guarantee the fitting quality for LightGBM, the dimensionality of inputs and the setting of hyperparameters should be reduced and optimized, respectively. CRBM and Bayes are introduced to address the dimensional reduction and parametric optimization, and through several tests, the integration of them is proved both reasonable and functional for LightGBM

(2) For KNN-, SVR-, RF-, and LightGBM-cored predictors, there exist three kinds of conclusive information: (1) training more learning samples indeed can rise an enhancement on the predicting capability of any predictor; (2) based on the training of a small-volumetric dataset, the petrophysical regression implemented by any predictor will be rather unreliable; (3) under the support of transfer learning, any predictor established from the training of a small set of learning samples will become capable to produce the expected results for the regression of three target reservoir characters

(3) No matter what kind of dataset there uses, compared to KNN-, SVR-, and RF-cored predictors, the proposed predictor always presents with a stronger computing capability and a better robust nature, then becoming a preferential selection for the petrophysical regression and accordingly deserving a more widespread application in the field of logging interpretation

Since the essence of other reservoir characters such as pore pressure and index of brittleness also can be viewed as a logging-based regression, there could have a deeper probe for the proposed predictor in the petrophysical regression. Therefore, in the future study, it is worth further improving the computing capability of LightGBM-cored predictor and then making a new breakthrough in the petrophysical regression.

Abbreviations

AC: Acoustic log
AT10: Resistivity of formation measured by array induction log at 10-inch logging depth
AT20: Resistivity of formation measured by array induction log at 20-inch logging depth
AT30: Resistivity of formation measured by array induction log at 30-inch logging depth
AT60: Resistivity of formation measured by array induction log at 60-inch logging depth
AT90: Resistivity of formation measured by array induction log at 90-inch logging depth
Bayes: Bayesian optimization
CART: Classification and regression tree
CD: Contrastive divergence
CDF: Cumulative distribution function
CNL: Compensated neutron log
CRBM: Continuous restricted Boltzmann machine
DBN: Deep belief network
DEN: Density log
EFB: Exclusive feature bundling
EI: Expected improvement
EL: Ensemble learning
GBDT: Gradient boosting decision tree
GOSS: Gradient-based one-side sampling
GP: Gaussian process
GP-UCB: Gaussian process-upper confidence bound
GR: Gamma ray
IQR: Inner quartile range
KNN: K-nearest neighbors
LIF: Lower inner fence
LightGBM: Light gradient boosting machine
ML: Machine learning
MSE: Mean squared error
PCA: Principal component analysis
PDF: Probability distribution function
PE: Photoelectric absorption cross-section index
PI: Probability of improvement
PSO: Particle swarm optimization
RBM: Restricted Boltzmann machine
RF: Random forest
RMSE: Root-mean-square error
RS: Random search
SI: Swarm intelligence
SNR: Signal-to-noise ratio
SP: Spontaneous potential
SVR: Support vector regression
UIF: Upper inner fence
XGBoost: Extreme gradient boosting.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of 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.

Authors' Contributions

Conceptualization was contributed by Shenghan Zhang and Yufeng Gu; methodology was contributed by Yufeng Gu;

software was contributed by Yufeng Gu; validation was performed by Yufeng Gu; formal analysis was contributed by Yinshan Gao and Yufeng Gu; investigation was performed by Xinxing Wang; resources was contributed by Shenghan Zhang; data curation was contributed by Shenghan Zhang; writing—original draft preparation was performed by Yufeng Gu; writing—review and editing was performed by Daoyong Zhang and Liming Zhou; visualization was contributed by Yufeng Gu; supervision was performed by Shenghan Zhang; project administration was performed by Shenghan Zhang. All authors have read and agreed to the published version of the manuscript.

References

[1] M. Kennedy, "Log analysis part I: porosity," *Developments in Petroleum Science*, vol. 62, pp. 181–207, 2015.

[2] A. Timur, "An investigation of permeability, porosity, and residual water saturation relationship for sandstone reservoirs," *The Log Analyst*, vol. 9, no. 4, pp. 8–17, 1968.

[3] M. H. Waxman and L. J. M. Smits, "Electrical conductivities in oil-bearing shaly sands," *SPE Reprint Series*, vol. 55, pp. 107–122, 2003.

[4] G. E. Archie, "Introduction to petrophysics of reservoir rocks," *AAPG Bulletin*, vol. 34, no. 5, pp. 943–961, 1950.

[5] P. C. Carman, "Fluid flow through granular beds," *Transactions Institution of Chemical Engineers*, vol. 15, pp. 150–166, 1937.

[6] J. Konzey, "Uber die kapillare Leitung des Wassers im Boden," *Sitz. b. Sitzungberichte, Abt. Ea, Mathematik, Astronomie, Physik Und Meteorologie*, vol. 136, no. a, pp. 271–306, 1927.

[7] W. C. Krumbein and G. D. Monk, "Permeability as a function of the size parameters of unconsolidated sand," *Transactions of the AIME*, vol. 151, no. 1, pp. 153–163, 1943.

[8] P. H. Nelson, "Permeability-porosity relationships in sedimentary rocks," *The Log Analyst*, vol. 35, no. 3, pp. 38–62, 1994.

[9] A. Poupon and J. Leveaux, "Evaluation of water saturation in shaly formations," *The Log Analyst*, vol. 12, no. 4, pp. 3–8, 1971.

[10] P. F. Worthington, "The uses and abuses of the Archie equations, 1: the formation factor-porosity relationship," *Journal of Applied Geophysics*, vol. 30, no. 3, pp. 215–228, 1993.

[11] M. R. J. Wyllie, A. R. Gregory, and L. W. Gardner, "Elastic wave velocities in heterogeneous and porous media," *Geophysics*, vol. 21, no. 1, pp. 41–70, 1956.

[12] L. Chen, W. Lin, P. Chen, S. Jiang, L. Liu, and H. Hu, "Porosity prediction from well logs using back propagation neural network optimized by genetic algorithm in one heterogeneous oil reservoirs of Ordos Basin, China," *Journal of Earth Science*, vol. 32, no. 4, pp. 828–838, 2021.

[13] A. Nasseri, M. J. Mohammadzadeh, and S. Hashemtabatabaee, "Evaluating Bangestan reservoirs and targeting productive zones in Dezful embayment of Iran," *Journal of Geophysics and Engineering*, vol. 13, no. 6, pp. 994–1001, 2016.

[14] J. A. Vargas-Guzmán, "Spatial modeling of heterogeneous initial water saturation," *Journal of Petroleum Science and Engineering*, vol. 58, no. 1-2, pp. 283–292, 2007.

[15] M. Wang, H. Tang, F. Zhao et al., "Controlling factor analysis and prediction of the quality of tight sandstone reservoirs: a case study of the He8 member in the eastern Sulige gas field, Ordos Basin, China," *Journal of Natural Gas Science and Engineering*, vol. 46, pp. 680–698, 2017.

[16] M. A. Ahmadi, M. R. Ahmadi, S. M. Hosseini, and M. Ebadi, "Connectionist model predicts the porosity and permeability of petroleum reservoirs by means of petro-physical logs: application of artificial intelligence," *Journal of Petroleum Science and Engineering*, vol. 123, pp. 183–200, 2014.

[17] L. N. Osli, N. Y. Yakub, M. R. Shalaby, and M. A. Islam, "Log-based petrophysical analysis of Khatatba formation in Shoushan Basin, North Western Desert, Egypt," *Geosciences Journal*, vol. 22, no. 6, pp. 1015–1026, 2018.

[18] S. M. T. Qadri, M. A. Islam, and M. R. Shalaby, "Application of well log analysis to estimate the petrophysical parameters and evaluate the reservoir quality of the Lower Goru Formation, Lower Indus Basin, Pakistan," *Geomechanics and Geophysics for Geo-Energy and Geo-Resources*, vol. 5, no. 3, pp. 271–288, 2019.

[19] D. A. Wood, "Predicting porosity, permeability and water saturation applying an optimized nearest-neighbour, machine-learning and data-mining network of well-log data," *Journal of Petroleum Science and Engineering*, vol. 184, article 106587, 2020.

[20] M. A. Ahmadi, S. Zendehboudi, A. Lohi, A. Elkamel, and I. Chatzis, "Reservoir permeability prediction by neural networks combined with hybrid genetic algorithm and particle swarm optimization," *Geophysical Prospecting*, vol. 61, no. 3, pp. 582–598, 2013.

[21] M. A. Ahmadi and Z. Chen, "Comparison of machine learning methods for estimating permeability and porosity of oil reservoirs via petro-physical logs," *Petroleum*, vol. 5, no. 3, pp. 271–284, 2019.

[22] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Communications of the ACM*, vol. 18, no. 9, pp. 509–517, 1975.

[23] A. A. Adeniran, A. R. Adebayo, H. O. Salami, M. O. Yahaya, and A. Abdulraheem, "A competitive ensemble model for permeability prediction in heterogeneous oil and gas reservoirs," *Applied Computing and Geosciences*, vol. 1, article 100004, 2019.

[24] Q. Guo, T. Zhuang, Z. Li, and S. He, "Prediction of reservoir saturation field in high water cut stage by bore- ground electromagnetic method based on machine learning," *Journal of Petroleum Science and Engineering*, vol. 204, article 108678, 2021.

[25] C. Li, L. Jiang, and J. Wu, "Distance and attribute weighted k-nearest-neighbor and its application in reservoir porosity prediction," *Journal of Information and Computational Science*, vol. 6, no. 2, pp. 845–851, 2009.

[26] B. Schölkopf, A. J. Smola, R. C. Williamson, and P. L. Bartlett, "New support vector algorithms," *Neural Computation*, vol. 12, no. 5, pp. 1207–1245, 2000.

[27] A. F. Al-Anazi and I. D. Gates, "Support vector regression for porosity prediction in a heterogeneous reservoir: a comparative study," *Computers and Geosciences*, vol. 36, no. 12, pp. 1494–1503, 2010.

[28] N. Moosavi, M. Bagheri, M. Nabi-Bidhendi, and R. Heidari, "Fuzzy support vector regression for permeability estimation of petroleum reservoir using well logs," *Acta Geophysica*, vol. 70, no. 1, pp. 161–172, 2022.

[29] S. R. Na'imi, S. R. Shadizadeh, M. A. Riahi, and M. Mirzakhanian, "Estimation of reservoir porosity and water

saturation based on seismic attributes using support vector regression approach," *Journal of Applied Geophysics*, vol. 107, pp. 93–101, 2014.

[30] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.

[31] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[32] Y. Ao, H. Li, L. Zhu, S. Ali, and Z. Yang, "The linear random forest algorithm and its advantages in machine learning assisted logging regression modeling," *Journal of Petroleum Science and Engineering*, vol. 174, pp. 776–789, 2019.

[33] P. Behnoud far, P. Hosseini, and A. Azizi, "Permeability determination of cores based on their apparent attributes in the Persian Gulf region using naive Bayesian and random forest algorithms," *Journal of Natural Gas Science and Engineering*, vol. 37, pp. 52–68, 2017.

[34] D. A. Otchere, T. O. A. Ganat, J. O. Ojero, B. N. Tackie-Otoo, and M. Y. Taki, "Application of gradient boosting regression model for the evaluation of feature selection techniques in improving reservoir characterisation predictions," *Journal of Petroleum Science and Engineering*, vol. 208, article 109244, 2022.

[35] T. Chen and C. Guestrin, "XGBoost: a scalable tree boosting system," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016.

[36] G. Ke, Q. Meng, T. Finley et al., "LightGBM: a highly efficient gradient boosting decision tree," *Advances in Neural Information Processing Systems*, pp. 3147–3155, 2017.

[37] K. Zhou, Y. Hu, H. Pan et al., "Fast prediction of reservoir permeability based on embedded feature selection and LightGBM using direct logging data," *Measurement Science and Technology*, vol. 31, no. 4, article 045101, 2020.

[38] F. Hadavimoghaddam, M. Ostadhassan, M. A. Sadri, T. Bondarenko, I. Chebyshev, and A. Semnani, "Prediction of water saturation from well log data by machine learning algorithms: boosting and super learner," *Journal of Marine Science and Engineering*, vol. 9, no. 6, article 666, p. 045101, 2021.

[39] K. Weiss, T. M. Khoshgoftaar, and D. D. Wang, "A survey of transfer learning," *Journal of Big Data*, vol. 3, p. 9, 2016.

[40] H. W. J. Reeve, T. I. Cannings, and R. J. Samworth, "Adaptive transfer learning," *Annals of Statistics*, vol. 49, no. 6, pp. 3618–3649, 2021.

[41] H. Chen and A. F. Murray, "Continuous restricted Boltzmann machine with an implementable training algorithm," *IEEE Proceedings: Vision, Image and Signal Processing*, vol. 150, no. 3, pp. 153–159, 2003.

[42] M. Seeger, "Gaussian processes for machine learning," *International Journal of Neural Systems*, vol. 14, no. 2, pp. 69–106, 2004.

[43] J. Tukey, "Mathematics and the picturing of data," in *Proceedings of the International Congress of Mathematicians*, pp. 523–532, Vancouver, 1975.

[44] Y. Bengio, "Learning deep architectures for AI," *Foundations and Trends in Machine Learning*, vol. 2, no. 1, pp. 1–27, 2009.

[45] Y. Chen and W. Wu, "Application of one-class support vector machine to quickly identify multivariate anomalies from geochemical exploration data," *Geochemistry: Exploration, Environment, Analysis*, vol. 17, no. 3, pp. 231–238, 2017.

[46] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Computation*, vol. 14, no. 8, pp. 1771–1800, 2002.

[47] O. Dekel, R. Gilad-Bachrach, O. Shamir, and L. Xiao, "Optimal distributed online prediction using mini-batches," *Journal of Machine Learning Research*, vol. 13, pp. 165–202, 2012.

[48] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian optimization of machine learning algorithms," *Advances in Neural Information Processing Systems*, pp. 2951–2959, 2012.

[49] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: theory and practice," *Neurocomputing*, vol. 415, pp. 295–316, 2020.

[50] M. Bagnardi and A. Hooper, "Inversion of surface deformation data for rapid estimates of source parameters and uncertainties: a Bayesian approach," *Geochemistry, Geophysics, Geosystems*, vol. 19, no. 7, pp. 2194–2211, 2018.

[51] R. Wang, Y. Chi, L. Zhang, R. He, Z. Tang, and Z. Liu, "Comparative studies of microscopic pore throat characteristics of unconventional super-low permeability sandstone reservoirs: examples of Chang 6 and Chang 8 reservoirs of Yanchang Formation in Ordos Basin, China," *Journal of Petroleum Science and Engineering*, vol. 160, pp. 72–90, 2018.

[52] K. Zhang, R. Liu, Z. Liu, B. Li, J. Han, and K. Zhao, "Influence of volcanic and hydrothermal activity on organic matter enrichment in the Upper Triassic Yanchang Formation, southern Ordos Basin, Central China," *Marine and Petroleum Geology*, vol. 112, article 104059, 2020.

[53] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, no. 6, pp. 1129–1159, 1995.

[54] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 2, no. 1-3, pp. 37–52, 1987.

[55] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *Journal of Machine Learning Research*, vol. 13, pp. 281–305, 2012.

[56] P. Comon, "Independent component analysis, a new concept?," *Signal Processing*, vol. 36, no. 3, pp. 287–314, 1994.

[57] R. C. Eberhart and Y. Shi, "Particle swarm optimization: developments, applications and resources," in *Proceedings of the IEEE Conference on Evolutionary Computation*, pp. 81–86, Seoul, Korea (South), 2001.

[58] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *IEEE International Conference on Neural Networks*, pp. 1942–1948, 1995.

[59] S. A. Yasear and K. R. Ku-Mahamud, "Review of the multi-objective swarm intelligence optimization algorithms," *Journal of Information and Communication Technology*, vol. 20, no. 2, pp. 171–211, 2021.