Article

Research on a Coal Seam Gas Content Prediction Method Based on an Improved Extreme Learning Machine

Shuicheng Tian 1,2, Lei Ma 1,2,* , Hongxia Li 2,3, Fangyuan Tian 2,3 and Junrui Mao 1,2

1 College of Safety Science and Engineering, Xi’an University of Science and Technology, Xi’an 710054, China; tianse@xust.edu.cn (S.T.); 21120089020@stu.xust.edu.cn (J.M.)
2 Institute of Safety and Emergency Management, Xi’an University of Science and Technology, Xi’an 710054, China; lihx@xust.edu.cn (H.L.); tfy@xust.edu.cn (F.T.)
3 School of Management, Xi’an University of Science and Technology, Xi’an 710054, China
* Correspondence: 22120089028@stu.xust.edu.cn

Abstract: With the rapid advancement of artificial neural network (ANN) algorithms, many researchers have applied these methods to mine gas prediction and achieved numerous research achievements. It is of great significance to study methods that can accurately predict the gas content for the prevention of gas disasters in mining areas. In order to enhance the accuracy, stability, and generalization capability of the gas content prediction model, the GASA-KELM prediction model was established using the GASA algorithm to improve the KELM initial parameter assignment method, and the prediction model based on BPNN and SVM was established under the same conditions. The experimental results show that the GASA-BPNN model failed to achieve the desired outcome within 800 iterations. On the other hand, the GASA-SVM and GASA-KELM models accomplished the goal in significantly fewer iterations, taking only 673 and 487 iterations, respectively. Moreover, the overall average relative errors of the cross-validated gas content predictions were 15.74%, 13.85%, and 9.87% for the three models, respectively. Furthermore, the total average variance of the test set was 3.99, 2.76, and 2.05 for the GASA-BPNN, GASA-SVM, and GASA-KELM models, respectively. As a result, compared with other ANN models, the GASA-KELM model demonstrates higher accuracy, stronger prediction stability, and generalization ability in the practical application. This novel model provides a basis for accurately predicting gas content and proposing effective regional gas management measures.

Keywords: gas content prediction; BP neural network (BPNN); support vector machine (SVM); kernel extreme learning machine (KELM)

1. Introduction

Coal resources play a crucial role as an energy source in China and have contributed immensely to the country’s economic development [1,2], and given China’s current energy structure, coal is expected to maintain its dominant position in the energy supply for the foreseeable future [3]. Based on statistics from the National Bureau of Statistics, China is endowed with abundant coal resources, with proven coal reserves of 249.23 billion tons. Additionally, China holds the position of the world’s largest producer and consumer of coal. In 2021, the total primary energy production in China was recorded at 4.33 billion tons of standard coal, with coal accounting for 67% of the overall energy structure. The total energy consumption was 5.24 billion tons of standard coal, with coal consumption contributing to 56.0% of the total primary energy consumption. Ensuring the healthy, stable, and sustainable development of the coal industry is crucial to maintaining the country’s energy security [4]. However, significant disparities in coal resource endowments, complex geological conditions, and deep burial depths across different regions pose substantial obstacles to achieving this goal. China’s robust demand for coal has resulted in an
increase in coal mining depths, and mining operations now face high gas pressure, elevated ground stress, and heightened gas content, posing significant threats to mine safety. Although the number of gas accidents has decreased significantly in recent years, the number of fatalities remains high compared to other nations. The issue of mine gas is a major problem that restricts the capacity of coal production, affects the safety of workers, and impedes the economic benefits of coal mines [5,6]. The unclear distribution pattern and occurrence status of CBM, as well as the failure to implement effective gas prevention and control measures, are the primary causes of most gas accidents. Coal seam gas extraction is not only a source of clean and natural energy but also a means to address the pressing issue of gas control [7]. Undoubtedly, methane, a crucial element of coal mine gas, is a potent greenhouse gas [8]. Its ozone-depleting capacity is sevenfold higher than that of carbon dioxide, and its heat-trapping potential is 25 to 30 times higher than an equivalent volume of carbon dioxide. Hence, it holds immense importance to investigate techniques for predicting coalbed methane content from multiple standpoints, such as mitigating gas calamities and safeguarding human lives and property, exploiting and harnessing clean energy, and conserving the environment.

In recent years, with the rapid advancement of machine learning and intelligent algorithms, novel methods such as artificial neural networks (ANNs), support vector machines (SVMs), and extreme learning machines (ELMs) have presented new avenues for tackling high-dimensional, nonlinear, and complex function optimization problems [9]. Hanbo Zheng et al. utilized the SVM to construct a model for predicting the dissolved gas content in power transformers [10]. Feng Yu et al. developed a short-term natural gas load forecasting model using a back propagation neural network (BPNN) [11]. Jiaxing Xin et al. employed a BPNN to identify the deformation characteristics of ACM oil and gas pipelines [12]. Bohan Cao et al. used the ANN to establish a shallow gas identification model for deep water drilling [13]. Machine learning algorithms have garnered widespread application and have yielded fruitful research outcomes in coal seam gas content prediction, coal and gas outburst prediction, and gas outflow prediction. Lin Haifei et al. utilized particle swarm optimization (PSO) to optimize the initial weights and thresholds of the BPNN to create a PSO-BPNN gas content prediction model. In comparison with the multiple linear regression model and the BPNN model, the PSO-BPNN model demonstrated the highest prediction accuracy [14]. Ma Lei et al. implemented the genetic algorithm (GA) and simulated annealing (SA) algorithms to optimize the BPNN for constructing a GASA-BPNN gas content prediction model. During the application process, the model showed a more robust generalization ability for new samples, faster parameter training speed, and higher prediction accuracy [15]. Yaqin Wu et al. introduced an adaptive learning rate to the BPNN and established the GASA-BPNN coal and gas outburst prediction model, which demonstrated good prediction performance during field applications [16]. Zhang Ruilin et al. proposed a coal and gas outburst potential risk level prediction model that combines fault tree and ANN coupling. By utilizing qualitative and quantitative data to solve the model, they made the relationship between geological factors and gas outburst potential risk more evident [17]. Xie Xuecai et al. utilized an improved fruit fly optimization algorithm (IFOA) to optimize the general regression neural network (GRNN), leading to the development of the IFOA-GRNN model for predicting coal and gas outbursts. The IFOA-GRNN model displays the desirable attributes of low prediction errors, high stability, and rapid convergence speed during practical applications [18]. Qian Meng et al. proposed a coal seam gas content prediction model based on an SVM and PSO. The results of practical applications revealed that the PSO-SVR model outperforms both the ANN model and the ordinary support vector regression (SVR) model, particularly with a limited number of samples [19]. Zhang Sirui et al. improved the grey prediction model by incorporating a BPNN, ultimately establishing an enhanced gas concentration prediction model based on grey theory and the BPNN. The simulation results indicate that this model significantly improves the prediction accuracy of the gas concentration [20]. Zhenhua Yang et al. introduced an improved residual gas content prediction method based on the
drilling cutting index and the bat algorithm-optimized ELM. In comparison with the BPNN, SVM, and ELM, this novel method exhibits superior accuracy and effectively uncovers the nonlinear relationship between the drilling cutting index and residual gas content [21]. Liming Qiu et al. established a protrusion risk prediction model based on a convolutional neural network. This model explores the correlation between post-explosion gas concentration changes and coal seam protrusion risks, which is crucial in improving coal and gas outburst prediction accuracy [22]. Xiang Wu et al. proposed a gas outburst prediction model based on the grey relation analysis (GRA) and adaptive algorithm-optimized SVM. Their study demonstrates that the new model exhibits better performance than the SVM and PSO-SVM outburst prediction models [23].

With regard to predicting coalbed methane content, the predominant approach involves leveraging gas geological theory to analyze the factors that influence methane content. By comprehensively considering these factors and employing mathematical techniques to establish a functional mapping relationship between the influencing factors and the target of prediction, accurate predictions can be achieved. The results demonstrate that this method surpasses traditional prediction methods and exhibits a relatively high accuracy rate. To overcome the limitations of ELM, BPNN, and SVM, researchers have integrated intelligent algorithms to optimize their parameters. However, the application of algorithms for parameter optimization is based on a random search algorithm framework, which still has room for improvement in terms of functionality. During the process of developing prediction models, researchers commonly rely on a single algorithm to optimize model parameters. Nevertheless, using a single algorithm may lead to limitations such as being vulnerable to local optima, exhibiting poor generalization ability, and achieving low accuracy when addressing intricate issues. To overcome the limitations of single algorithm optimization, the GASA hybrid optimization algorithm with complementary advantages is designed by using a divide-and-conquer strategy by comprehensively exploiting the differences and complementarities of different intelligent algorithms. The GASA algorithm is used to optimize KELM, SVM and BPNN to build three gas content prediction models to obtain a gas content prediction model with faster iteration speed, higher prediction accuracy, and stronger generalization ability.

2. Development of Prediction Models for Coal Seam Gas Content

2.1. Theoretical Analysis and Performance Evaluation of the GASA Optimization Algorithm

2.1.1. Theoretical Analysis of the GASA Optimization Algorithm

The GA is a stochastic search algorithm that mimics the genetic mechanism of nature and Darwin’s theory of biological evolution [24,25]. The GA uses a coding space to represent the parameter space of the problem and evaluates the fitness of individuals in the population based on a fitness function. This approach simulates biological selection and genetic mechanisms through genetic operations to generate new individuals who outperform the previous generation. Through repeated iterations, the algorithm approaches the global optimum solution. Figure 1 shows the flowchart of the GA algorithm.

The SA algorithm is a random search algorithm that is based on the Monte Carlo iterative solution strategy [26,27]. Its starting point is inspired by the similarity between the annealing process of solid materials and combinatorial optimization problems. The SA algorithm starts with a high initial temperature and applies the Metropolis sampling criterion, which accepts suboptimal solutions in the neighborhood with a certain probability. This allows the algorithm to effectively avoid becoming stuck in local optima in the early stage. In the later stage, the algorithm improves its convergence efficiency by rejecting suboptimal solutions with a high probability. This overcomes the problem of the algorithm becoming easily trapped in local optimal solutions and is crucial to the SA algorithm’s ability to converge globally. The SA algorithm process is depicted in Figure 1.
The GASA algorithm is designed to combine the advantageous features of SA and GA, namely, SA’s gradually decreasing probability jump and GA’s survival of the fittest, to overcome the challenge of becoming trapped in local minima during the search process. Structurally, while GA simultaneously searches the population using the neighborhood function, SA only concentrates on a single individual at a time. The GASA algorithm combines these approaches by executing SA sequentially on each individual in the GA population, diversifying the neighborhood search structure of each individual and enhancing the algorithm’s search capability and efficiency.

The GASA algorithm adheres to the process illustrated in Figure 2. The optimization process can be summarized as follows [16]:

1. **GA Initialization:** Set the population size $N$ and initialize the population $P_0$. Set the maximum number of generations $M$ and the genetic iteration number $K$ to 1;
2. **Defining the fitness function**;
3. **Assess the fitness value of the $P_0$: Verify if the stopping criterion for the GA has been attained. If the criterion has indeed been met, then it yields the most superior solution. If not, proceed with steps (4) to (9) accordingly;
4. **$P_0$ performs genetic crossover and mutation to generate a fresh population, $P_{K0}$, and subsequently assesses the fitness value of $P_{K0}$**;
5. **Initializing SA parameters:** Set the initial solution $P_{K0}$ of the population as the initial solution of SA, set $i = 0$, set the initial temperature $T = T_i$ (sufficiently high), determine the length of the Metropolis chain $L$ at each state $T$, and set the iteration count in the chain $Q = 1$.
6. **At the present temperature $T$, and for $Q = 1, 2, 3, \ldots L$, iterate through steps (7) to (9) repeatedly.**
7. **For every member of the current population, induce a random perturbation to generate a fresh population, and subsequently evaluate the fitness of this new population.**
8. **For each member of the current population, determine the disparity in fitness between the fresh population and the current population, represented by $\Delta E$. Determine whether $P = \exp \left(-\frac{\Delta E}{T}\right) \geq \text{rand}$, and if so, accept the new solution as well; if not, reject the new solution.**
(9) Increment $Q$ by 1. Verify if $Q > L$. If this condition is satisfied, then increment $i$ by 1, and reduce the temperature using the temperature annealing function such that $T = T_{i-1}$, where $T < T_{i}$. Subsequently, verify if the annealing stopping criteria have been met. If so, then increment $K$ by 1, produce the optimal group solution $P_K$, and return to step (3). If the annealing stopping criteria have not been met, then set $Q = 1$, and return to step (7). If $Q > L$ is not met, then return to step (7). The annealing stopping criterion is typically established as the stopping temperature.

(10) Once the genetic operations have been applied to the $P_K$ population to produce the $P_{K0}$ population, the $P_{K0}$ population is employed as the initial solution for the SA algorithm to create $P_{K+1}$. If the new $P_{K+1}$ population does not meet the stopping criteria for the GA, then the population formed by the SA algorithm, $P_{K+1}$, is adopted as the starting population for the GA to partake in the iterative optimization.

Figure 2. Flowchart of the GASA algorithm.

2.1.2. Performance Testing of the GASA Algorithm

The GASA is a random search algorithm, and Rastrigin’s function was chosen to test its optimization performance due to its numerous local optima, which can mislead algorithms. Therefore, it is an ideal choice for testing the performance of algorithms. Figure 3 displays a three-dimensional plot of the function. The mathematical expression for this function is as follows:

$$Ras(x) = 20 + x_1^2 + x_2^2 - 10(\cos 2\pi x_1 + \cos 2\pi x_2)$$  \hspace{1cm} (1)

where $x_1 \in [-5, 5], x_2 \in [-5, 5]$
Figure 3. Three-dimensional graph of the test function.

To optimize the function, the GASA algorithm’s parameters can be customized before executing the algorithm. In addition, the PSO, GA, and SA algorithms can also be utilized individually for function optimization. A performance test through experimental analysis can be conducted to compare the performance of each algorithm. Given that these algorithms are all stochastic search algorithms, it is advisable to conduct multiple experiments to avoid significant accidental errors. Therefore, each algorithm can be tested 20 times under identical environmental conditions. The termination condition for each algorithm can be set to exceed 200 iterations or reach the global minimum value during optimization. The outcome of these experiments is presented in Table 1.

Table 1. The table of optimization results for the test function.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Test Function</th>
<th>Average Result</th>
<th>Variance of Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSO</td>
<td>Rastrigin's function</td>
<td>$9.2472 \times 10^{-4}$</td>
<td>3.1547</td>
</tr>
<tr>
<td>GA</td>
<td></td>
<td>$7.9003 \times 10^{-3}$</td>
<td>3.7519</td>
</tr>
<tr>
<td>SA</td>
<td></td>
<td>$9.1873 \times 10^{-2}$</td>
<td>7.6823</td>
</tr>
<tr>
<td>GASA</td>
<td></td>
<td>$5.6935 \times 10^{-4}$</td>
<td>2.0524</td>
</tr>
</tbody>
</table>

After analyzing the test outcomes presented in Table 1, it is apparent that the algorithms did not achieve the set objectives in every one of the 20 optimization runs. Additionally, the number of iterations and optimization results varied for each algorithm. Upon examining the performance metrics of average optimization value, variance of optimization results, and average number of iterations, it is evident that the GASA algorithm exhibits superior function optimization capabilities and global search stability compared to the PSO, GA, and SA algorithms.

Figure 4 illustrates the curve of the best individual’s average fitness function value during the function optimization process. The GASA algorithm reaches the optimal value at the 78th iteration, the PSO reaches the optimal value at the 102nd iteration, and the GA and SA algorithms converge at the 156th and 176th iterations, respectively. However, the fitness function value quality of the latter two algorithms is inferior to that of the former two. Consequently, when compared to the PSO, SA, and GA algorithms, the GASA algorithm demonstrates better performance and efficiency in the complex function parameter optimization process.
2.2. Development of a Gas Content Prediction Model Based on the GASA-BPNN Algorithm

BP theory was originally proposed by Werbos in 1974, and it served as a cornerstone for the development of artificial neural networks [26,28]. In 1986, Rumelhart and McClelland introduced the error backpropagation learning algorithm, which is used for training multilayer neural networks. The neural network trained using the BP algorithm is referred to as the BP neural network (BPNN). The BPNN is a multilayer feedforward neural network that is trained using the error backpropagation algorithm. It comprises an input layer, several hidden layers, and an output layer, with each layer being connected through different weight parameters in a fully connected manner. Research has demonstrated that a single hidden layer BPNN has the ability to arbitrarily approximate any complex nonlinear system, as depicted in Figure 5.
An insufficient number of hidden layer nodes in a BPNN can result in underfitting and low prediction accuracy, while an excessive number of nodes can lead to overfitting. The range of nodes is determined based on empirical Formula (2) and found to be [4, 12]. The optimal number of nodes is selected by comparing the average relative prediction error of the model, and the results are presented in Table 2. The table demonstrates that the error is minimized when the number of nodes is \( W = 12 \), indicating that 12 nodes are optimal. Research has revealed that a three-layer BPNN can effectively approximate any complex nonlinear system; therefore, the GASA-BPNN prediction model is structured as a 6-12-1 BPNN.

\[
W = \sqrt{c + q} + B
\]  

(2)

The variables in the equation are defined as follows: \( W \) denotes the count of nodes in the hidden layer, \( c \) represents the number of nodes in the input layer, \( q \) signifies the number of nodes in the output layer, and \( B \) is an integer ranging from 1 to 10.

Table 2. Number of hidden layer nodes and the average relative error table.

<table>
<thead>
<tr>
<th>W</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relative Error/%</td>
<td>27.9</td>
<td>30.84</td>
<td>21.29</td>
<td>19.73</td>
<td>24.18</td>
<td>19.62</td>
<td>16.35</td>
<td>21.87</td>
<td>15.22</td>
</tr>
</tbody>
</table>

Figure 6 depicts the process of applying the model, while Figure 7 shows the GASA-BPNN gas content prediction model. The application steps are described below in detail [16]:

1. To apply the prediction model, the data are first normalized to the range of [0, 1]. Subsequently, the dataset is stratified and sampled into 10 mutually exclusive subsets. Each time, one subset is chosen as the test set, while the remaining subsets are used as training sets for 10 rounds of training and testing. Figure 8 illustrates the schematic diagram of the sample division process.
Figure 6. Flowchart of the GASA algorithm for initializing BP neural network parameters.

Figure 7. Diagram of the gas content prediction model.
(2) The initialization of the GA population involves encoding the 97 parameters using floating point number encoding rules. Each individual in the population represents a unique set of weights and thresholds. The initial population, $P_0$, consists of 50 individuals who are randomly generated.

(3) The GA parameters are set as follows: The maximum number of genetic iterations is set to 800, $K$ is initialized to 1, and the GA stops when the maximum number of iterations is reached. The crossover probability is set to 0.7, and the mutation probability is set to 0.005.

(4) The fitness function is defined based on the MSE of the prediction, whereby a smaller MSE value yields a higher fitness value for the individual.

(5) The initial population $P_0$ was subjected to fitness evaluation, and the individuals were ranked based on their respective fitness values.

(6) Stopping criteria for GASA: Check if the maximum number of genetic iterations has been reached. If so, then terminate the process and proceed to steps (7)–(11). Otherwise, increment $K$ by 1 and proceed to step (12).

(7) BPNN Initialization: The solution obtained from step (6) is assigned to the BPNN. The maximum number of training iterations is set to $H = 800$, and the iteration counter is initialized to $S = 0$.

(8) The input dataset is processed by propagating the data forward through the layers of the BPNN. The input for the $f$th neuron in the hidden layer is computed using Equation (3), and the input for the $j$th output node is computed using Equation (4).

$$\gamma_f = \sum_{v=1}^{S} v_{vf} x_v$$

where $\gamma_f$ represents the input of the hidden layer neuron $f$ and $v_{vf}$ represents the weight of the connection between the input layer neuron $v$ and the hidden layer neuron $f$.

$$\beta_j = \sum_{f=1}^{S} w_{jf} b_f$$

where $\beta_j$ is the input of output node $j$; $w_{jf}$ is the weight of the connection between intermediate layer node $f$ and output layer node $j$; and $b_f$ is the output of intermediate layer node $f$.

(9) The BPNN weights and thresholds are updated by backpropagating the MSE to each node, and the parameters are adjusted using the error-adaptive learning rate gradient descent method described in Equation (5). If the error approaches the target value with less fluctuation, then the model training direction is correct, and the learning rate can be increased. However, if the error increases beyond the allowed range, then the model training direction is incorrect, and the learning rate should be reduced.

$$\mu(S + 1) = \begin{cases} S_{in} \mu(S), & E(S + 1) < E(S) \\ S_{de} \mu(S), & E(S + 1) > E(S) \end{cases}$$

where $\mu(S + 1)$ is the learning rate when the number of iterations is $S+1$; $\mu(S)$ is the learning rate when the number of iterations is $S$; $S_{in}$ is the incremental coefficient; $S_{de}$ is the decremental coefficient; $E(S + 1)$ is the error when the number of iterations is $S + 1$; and $E(S)$ is the error when the number of iterations is $S$.

(10) BPNN stopping criterion: Verify whether the termination criterion has been met, where the minimum error is defined as the stopping condition. If the criterion is
satisfied, then the network completes its learning phase, and the model is established, advancing to step (11). Otherwise, examine if H is equal to S. If true, then return to step (7). Otherwise, increase S by one and go back to step (8).

(11) The performance evaluation of the model involves testing the model on the test set. The evaluation metrics used to assess the model’s performance are the MSE, iteration number, and relative prediction error.

(12) The selection process in the GA involves the use of the roulette selection method, which serves to screen individuals based on their fitness values. Specifically, for an individual \( x_0 \) with a fitness value of \( f_0 \), the probability of \( x_0 \) being selected is given by

\[
p_u = \frac{f_u}{\sum_{j=1}^{N} f_u}.
\]

(13) The process of GA crossover involves randomly selecting two individuals, \( x_1 \) and \( x_2 \), from the population \( P_x \) and generating new individuals, \( x'_1 \) and \( x'_2 \), through the arithmetic crossover operation outlined in Equation (6).

\[
\begin{align*}
    x'_1 &= \lambda_1 x_1 + \lambda_2 x_2 \\
    x'_2 &= \lambda_1 x_2 + \lambda_2 x_1
\end{align*}
\]

where \( \lambda_1 + \lambda_2 = 1, \lambda_1 > 0, \) and \( \lambda_2 > 0 \).

(14) For the GA variation, an individual genotype \( X = x_1 x_2 \ldots x_b \ldots x_i \) is randomly selected, and the genetic operation is performed at the mutation point \( x_b \) using Equation (7).

\[
x'_b = \begin{cases} x_b + \Delta(K, UB - x_b), & r \leq 0.5 \\
                     x_b - \Delta(K, x_b - LB), & r > 0.5 \end{cases}
\]

Here, UB and LB denote the upper and lower boundary values of the variable \( x_b \), respectively. \( r \) represents a random number, and \( K \) denotes the evolutionary algebra. Furthermore, \( \Delta(t, y) \) is a function defined as follows:

\[
\Delta(t, y) = y \left[ 1 - r^{(1-K)^q} \right]
\]

The nonconsistency control parameter \( q \) (set to \( q = 0.8 \)) is used in the above equation.

(15) Evaluate the fitness of \( P_{x_0} \): Calculate the fitness value of \( P_{x_0} \) and designate it as the initial population of the annealing algorithm.

(16) SA parameter initialization: All genetically operated individuals undergo annealing.

The initial temperature \( T \) is set to 100 °C, the cooling factor is set to 0.98, and the stopping criterion is set such that \( Q = 1 \) and the Markov chain length is \( L = 60 \).

(17) A novel solution is produced by perturbing each member of the population, computing the disparity between the fitness values of each member in the old and new populations, and determining whether to adopt the new solution based on the Metropolis sampling criterion.

(18) Increment \( Q \) by 1 and check if it is in the current Markov chain. If it is in the chain, then proceed to step (17). If it is not in the chain, then decrease the temperature and check if the stopping condition for the SA is met. If the stopping condition is met, then the SA algorithm ends and proceeds to step (4). Otherwise, set \( Q = 1 \) and proceed to step (17).

2.3. Establishment of a Gas Content Prediction Model Based on the GASA-SVM Algorithm

The theory of SVM encompasses optimal classification hyperplanes, kernel functions, and margin theory. The SVM offers several advantages, such as suitability for handling small sample sizes, robust generalization ability, and simple structure. In a wide range of domains, including medicine, electricity, and economics, SVMs have found practical applications for pattern recognition and regression problems [29,30].

The fundamental tenet of SVM is to identify the optimal classification hyperplane in the sample space that can segregate various classes of samples while minimizing the empirical and structural risks associated with the classifier. To illustrate the principle of SVM
for linear classification in two dimensions, consider the example shown in Figure 9. The orange circles denote one class of samples, while the blue circles correspond to another class. The classification line for these samples is represented by H, whereas $H_1$ and $H_2$ are two parallel lines that pass through the points of the two classes that are closest to the classification line. The distance between $H_1$ and $H_2$ is known as the classification margin. To enhance the model’s generalization ability, it is crucial to maximize the classification margin, indicating that the greater the margin, the more robust the model’s generalization ability to unobserved examples and the higher the accuracy of the model’s prediction.

![Figure 9. Schematic diagram of the optimal classification surface.](image)

Sample $D = \{(x_1, y_1), (x_2, y_2), ..., (x_m, y_m)\}, y_i \in \mathbb{R}$, conventional regression models evaluate prediction error by directly computing the difference between the predicted value of the model and the actual value $y$. A fitting error of zero is achieved only when the predicted value matches $y$ precisely. However, support vector regression uses a different method to calculate errors. SVR permits a maximum error of $\varepsilon$ between the predicted value and the measured value, assuming that the deviation between the predicted value and the measured value is zero. Figure 10 illustrates this concept, with the red line representing the true value and the black circle representing the predicted value. A sample is accurately predicted if the predicted values of the training set fall within the interval band created by the central line of $f(x)$ with a width of $2\varepsilon$, resulting in a prediction error of zero.

![Figure 10. Schematic diagram of support vector regression.](image)

The selection of the kernel width parameter and regularization factor has a profound impact on the predictive performance of SVM. However, traditional grid search algorithms have limitations, such as undefined value ranges and large computational costs. To overcome these challenges, a GASA algorithm is proposed to initialize the core
parameters of SVM. The optimized parameters are then assigned to SVM, and the GASA-SVM gas content prediction model is established by training the SVR machine with the original data. Since the process of initializing SVM and ELM parameters using the GASA algorithm is similar to the process of initializing BPNN parameters, in the following text, only the key steps of the initialization process for SVM and KELM are retained.

The process of constructing the GASA-SVM prediction model is depicted in Figure 11. The steps involved in building the model are as follows:

1. The 10-fold cross-validation method is used to split the data into test and training sets;
2. GA initialization: The parameters σ and c of the kernel function are encoded using the floating-point number encoding rule. A population of 50 individuals is randomly initialized, and the population size and GA parameters such as maximum generation and crossover probability are set;
3. Fitness evaluation function: The MSE between the predicted gas content and the actual output is minimized as the fitness evaluation function. The randomly initialized individuals from step (2) are evaluated. The genetic stopping condition is checked. If met, then the optimal individual is output, decoded, and assigned to SVM. GASA completes the optimized SVM, which is then trained again using the sample data to establish the model. If the genetic stopping condition is not met, then genetic crossover and genetic mutation are performed. The resulting individuals are used as the initial generation population of the SA algorithm;
4. SA algorithm initialization: Individuals in the population that did not meet the genetic stopping condition in step (3) are used as the initial individuals in the SA algorithm. Each individual undergoes simulated annealing. The initial parameters of SA, such as the initial temperature, temperature decay function, and Metropolis chain length, are set. Essentially, an SA algorithm is concatenated under each individual in the GA population;
5. Interference population: Each individual in the population generates a new population using the fitness evaluation function in step (3). The fitness function value is evaluated, and the difference in fitness function value between new and old populations is calculated. Finally, the Metropolis sampling criterion is used to determine whether the old solution should be replaced by the new solution;
6. Check if the SA stopping condition is met. If it is met, then end the annealing and use this population as the next generation population in the GA in step (3). If it is not met, then return to step (5).
2.4. Development of a Gas Content Prediction Model Based on the GASA-KELM Algorithm

Multilayer feedforward neural networks (FFNNs) have been widely used in linear and nonlinear system identification due to their excellent global approximation performance [31]. However, most traditional FFNN learning algorithms are based on the gradient descent method to modify all the weights and thresholds of the entire network. This has the disadvantages of requiring a large number of iterations, which makes it difficult to select an appropriate learning rate. To overcome the shortcomings of traditional FFNN algorithms, Huang Guangbin proposed a new algorithm called ELM for solving single-hidden-layer feedforward neural networks [32]. Compared with traditional algorithms, ELM randomly initializes the connection weights and thresholds of FFNN and does not participate in learning and correction. The unique optimal solution can be obtained by adjusting the number of hidden layer nodes.

The proposed model for predicting gas content in coal seams based on the improved ELM introduces the radial basis function kernel due to the nonlinear nature of the gas content prediction system. The initial kernel parameters, which include the bandwidth of the Gaussian kernel and regularization factor of KELM, have a significant impact on the model’s predictive performance. The issue of randomly assigned kernel parameters leading to a non-full rank output matrix can weaken the model’s ability to generalize to new data. To address these issues, a hybrid heuristic algorithm is constructed to optimize the kernel parameter $\sigma$ and regularization factor $C$ of ELM.

Figure 11 illustrates the process of constructing the GASA-KELM model for predicting gas content in coal seams. The model building process involves the following steps:

1. Normalize the dataset and split it into training and testing sets using 10-fold cross-validation.

2. Initialize the GA by encoding the kernel function parameters $\sigma$ and penalty factor $c$ for the KELM, randomly initializing the population, and setting the population size and GA parameters.
(3) Define the fitness evaluation function as the MSE between the predicted and actual values of the model, and evaluate the randomly initialized individuals in the population from step (2). Determine if the genetic stopping condition is met. If so, then output the optimal individual, decode it, and assign it to KELM. Complete the model establishment by training the sample data again. The training process and model performance evaluation are discussed in the following section. If the genetic stopping condition is not met, then perform genetic operations, and use the resulting individuals as the initial population for SA.

(4) Initialize the SA algorithm by using the population individuals from step (3) that did not meet the genetic stopping condition as the initial population for the SA algorithm. Each individual undergoes simulated annealing. Set the initial parameters for SA, such as the initial temperature, temperature decay function, and Metropolis chain length, which correspond to an SA algorithm linked to each population individual in the GA.

(5) Update the population using the neighborhood function to generate a new population. Evaluate the fitness function value using the evaluation function from step (3), calculate the difference in fitness values between the new and old populations, and determine whether to replace the old solution based on the Metropolis criterion.

(6) Determine whether the annealing stopping condition is met. If it is met, then complete the annealing operation and use this population as the next generation population in the GA in step (3). If it is not met, then return to step (5).

3. Field Application of Gas Content Prediction Models

3.1. Overview of the Jiulishan Mine in Jiaozuo

The Jiulishan Mine is situated in the heart of the Jiaozuo mining area, approximately 18 km from Jiaozuo city. The mine spans an area of approximately 18.7 km², with a north-south width of approximately 3.4 km and an east-west length of approximately 5.5 km. Its construction began in July 1970, and it commenced simple production in April 1983. The mine was designed to have a production capacity of 900,000 tons per annum, with a rated production capacity of 1 million tons per year. The vertical shaft mining method and combined development of upper and lower levels are employed in the mine. The coal seam that is mined is the Shanxi Formation II-1 coal seam, which has a simple structure and stable occurrence and is a medium-gray, low-sulfur, high-quality anthracite with a thickness ranging from 0.92 m to 8.13 m and an average thickness of 5.15 m. Its recoverable reserves index is 97.5%. The Jiulishan Mine is known for its coal and gas outburst, and its primary regional gas control measures include the combination of bottom rock roadway cross-layer predrainage of coal seam gas in the premining area and the drilling of in-seam boreholes to predrain coal seam gas in the mining area.

The Jiulishan Mine employs a central parallel and diagonal mixed ventilation system, utilizing a mechanical extraction method. The main and auxiliary shafts, in addition to the West Ventilation Shaft, are designated as intake airways, while the East and South Ventilation Shafts function as return airways. At present, the mine’s total intake air volume stands at 13,500 m³/min, while the total exhaust air volume is 13,820 m³/min. The absolute gas emission rate of the mine is 43.99 m³/min, with a corresponding relative gas emission rate of 33.41 m³/t. The mine’s location can be seen in Figure 12.
3.2. Establishment of the Parameter System for the Gas Content Prediction Model

3.2.1. Establishing a Dataset for Gas Content Prediction

Based on an analysis of the gas geological conditions and distribution patterns in the 15th mining area of the Jiulishan Coal Mine in Henan Province, China, we have identified eight factors that can quantitatively affect gas content ($X_0$, m$^3$·t$^{-1}$): coal seam depth ($X_1$, m), coal seam thickness ($X_2$, m), dip angle coefficient ($X_3$), overlying rock thickness ($X_4$, m), surrounding rock equivalent coefficient ($X_5$), fault complexity coefficient ($X_6$), fold complexity coefficient ($X_7$), and floor elevation ($X_8$, m) [33,34]. In the 15 mining areas, 290 sets of data on gas content and influencing factors were selected as the experimental dataset. The training sample data for the model are shown in Table 3.

Table 3. Part of the model training sample data.

<table>
<thead>
<tr>
<th>Sample Number</th>
<th>$X_0$/($m^3$·t$^{-1}$)</th>
<th>$X_1$/(m)</th>
<th>$X_2$/(m)</th>
<th>$X_3$</th>
<th>$X_4$/(m)</th>
<th>$X_5$</th>
<th>$X_6$</th>
<th>$X_7$</th>
<th>$X_8$/(m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>17.01</td>
<td>407.12</td>
<td>6.44</td>
<td>0.0198</td>
<td>254.09</td>
<td>0.4185</td>
<td>0.0008</td>
<td>0.0005</td>
<td>-324.81</td>
</tr>
<tr>
<td>2</td>
<td>25.32</td>
<td>350.41</td>
<td>6.08</td>
<td>0.0191</td>
<td>209.54</td>
<td>0.9000</td>
<td>0.1416</td>
<td>0.1812</td>
<td>-265.47</td>
</tr>
<tr>
<td>3</td>
<td>25.63</td>
<td>327.86</td>
<td>5.74</td>
<td>0.0354</td>
<td>191.74</td>
<td>0.4508</td>
<td>0.3062</td>
<td>0.0001</td>
<td>-243.64</td>
</tr>
<tr>
<td>4</td>
<td>15.27</td>
<td>438.26</td>
<td>4.84</td>
<td>0.0213</td>
<td>241.67</td>
<td>0.4000</td>
<td>0.0016</td>
<td>0.0001</td>
<td>-353.7</td>
</tr>
<tr>
<td>5</td>
<td>18.99</td>
<td>363.62</td>
<td>5.79</td>
<td>0.0440</td>
<td>219.25</td>
<td>0.5663</td>
<td>0.1003</td>
<td>0.0004</td>
<td>-279.48</td>
</tr>
<tr>
<td>6</td>
<td>12.53</td>
<td>400.92</td>
<td>3.61</td>
<td>0.0193</td>
<td>256.23</td>
<td>0.7740</td>
<td>0.0264</td>
<td>0.0048</td>
<td>-315.06</td>
</tr>
<tr>
<td>7</td>
<td>22.03</td>
<td>524.6</td>
<td>5.35</td>
<td>0.0122</td>
<td>267.32</td>
<td>0.5109</td>
<td>0.3255</td>
<td>0.0025</td>
<td>-437.3</td>
</tr>
<tr>
<td>8</td>
<td>21.46</td>
<td>517.3</td>
<td>4.36</td>
<td>0.0171</td>
<td>114.52</td>
<td>0.3855</td>
<td>0.0853</td>
<td>0.0115</td>
<td>-430</td>
</tr>
<tr>
<td>9</td>
<td>24.46</td>
<td>437.4</td>
<td>4.81</td>
<td>0.0224</td>
<td>121.89</td>
<td>0.2370</td>
<td>0.1189</td>
<td>0.0137</td>
<td>-350.7</td>
</tr>
<tr>
<td>10</td>
<td>24.03</td>
<td>444</td>
<td>7.07</td>
<td>0.0165</td>
<td>188.67</td>
<td>0.6074</td>
<td>0.0352</td>
<td>0.0084</td>
<td>-359.82</td>
</tr>
<tr>
<td>11</td>
<td>31.01</td>
<td>305.42</td>
<td>5.39</td>
<td>0.0403</td>
<td>207.99</td>
<td>0.3394</td>
<td>0.1738</td>
<td>0.0002</td>
<td>-217.27</td>
</tr>
<tr>
<td>12</td>
<td>19.17</td>
<td>326.7</td>
<td>5.43</td>
<td>0.0546</td>
<td>241.06</td>
<td>0.5873</td>
<td>0.0032</td>
<td>0.0117</td>
<td>-242.16</td>
</tr>
<tr>
<td>13</td>
<td>27.63</td>
<td>530</td>
<td>10.59</td>
<td>0.0412</td>
<td>334.41</td>
<td>0.4105</td>
<td>0.0421</td>
<td>0.0004</td>
<td>-440.9</td>
</tr>
<tr>
<td>14</td>
<td>20.41</td>
<td>501.7</td>
<td>5.34</td>
<td>0.0159</td>
<td>128.22</td>
<td>0.3534</td>
<td>0.0266</td>
<td>0.0010</td>
<td>-414.5</td>
</tr>
<tr>
<td>15</td>
<td>9.67</td>
<td>284</td>
<td>5.62</td>
<td>0.0166</td>
<td>144.48</td>
<td>0.3299</td>
<td>0.0338</td>
<td>0.0012</td>
<td>-195.1</td>
</tr>
<tr>
<td>16</td>
<td>18.86</td>
<td>438</td>
<td>5.07</td>
<td>0.0154</td>
<td>95.09</td>
<td>0.4837</td>
<td>0.0984</td>
<td>0.0029</td>
<td>-350.4</td>
</tr>
<tr>
<td>17</td>
<td>16.71</td>
<td>400.92</td>
<td>3.61</td>
<td>0.0278</td>
<td>256.23</td>
<td>0.7740</td>
<td>0.0264</td>
<td>0.0048</td>
<td>-317.1</td>
</tr>
<tr>
<td>18</td>
<td>12.74</td>
<td>400.92</td>
<td>3.61</td>
<td>0.0315</td>
<td>256.23</td>
<td>0.7740</td>
<td>0.0009</td>
<td>0.0001</td>
<td>-308.68</td>
</tr>
</tbody>
</table>
3.2.2. Primary Controlling Factors of Gas Content Based on Grey Correlation Analysis

The prediction of gas content is a challenging nonlinear prediction problem that is affected by various factors. Grey system theory is a suitable method for such problems with limited data, and grey correlation analysis can determine the extent to which each reference sequence affects the parent sequence. The quantitative ordering of correlations provides a clear comprehension of the relationship between various influencing factors and helps identify the principal controlling factors of gas content.

The GRA method is used to screen the main controlling factors for the model’s input. The steps of the grey relational calculation are as follows [35]:

1. Setting reference sequence and comparison sequence: Reference sequence: Gas content ($X_0$), Comparison sequence: Eight influencing factors of coal seam gas content.
2. Data preprocessing and normalization: According to Formula (8), the data will be normalized.

$$x'_i(h) = \frac{x_i(h) - x_{\min}}{x_{\max} - x_{\min}}$$ (8)

where $x_i(h)$ is the value of the $l$-th evaluation index of the sample with the number $h$; $h$ represents the sample number, $h = 1, 2, \ldots, n$; $l$ represents the evaluation index, $l = 1, 2, \ldots, m$; and $x_{\max}$ and $x_{\min}$ are the maximum and minimum values of the evaluation index, respectively.

3. To calculate the correlation coefficient: The calculation of correlation coefficient is obtained according to Formula (9).

$$\xi_e(k) = \frac{\min_{k} \min_{e} |x_0(k) - x_e(k)| + \rho \max_{k} \max_{e} |x_0(k) - x_e(k)|}{|x_0(k) - x_e(k)| + \rho \max_{e} |x_0(k) - x_e(k)|}$$ (9)

The symbol $\xi_e(k)$ denotes the correlation coefficient of the comparison sequence $x_e$ with respect to the reference sequence $x_0$ on index $k$, where $k$ ranges from 1 to $n$. The parameter $\rho$, which takes a value between 0 and 1, is the resolution coefficient.

4. To compute the correlation degree: Calculate the correlation degree using Formula (10).

$$r_e = \frac{1}{n} \sum_{k=1}^{n} w_k \xi_e(k)$$ (10)

where $n$ is the number of samples, $r_e$ is the correlation degree of the comparison sequence $x_e$ with the reference sequence $x_0$, and $w_k$ is the weight of the indicator.

5. Ranking of correlation and determination of input parameters

Table 4 presents the results of the correlation analysis, revealing that the correlation coefficients of the influencing factors $X_3$ and $X_4$ fall below 0.5, indicating a low correlation. Consequently, these factors are eliminated. This brings the number of input layer nodes to 6, as the data dimension in Table 3 is reduced from 8 to 6. The six highly correlated gas content influencing factors are the only inputs for the model, whereas the other two factors are no longer involved in the modeling process.

Table 4. The results of grey correlation analysis.

<table>
<thead>
<tr>
<th>Factors Influencing</th>
<th>$X_1$</th>
<th>$X_2$</th>
<th>$X_3$</th>
<th>$X_4$</th>
<th>$X_5$</th>
<th>$X_6$</th>
<th>$X_7$</th>
<th>$X_8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$r_e$</td>
<td>0.6698</td>
<td>0.5623</td>
<td>0.2908</td>
<td>0.3176</td>
<td>0.5024</td>
<td>0.6117</td>
<td>0.5102</td>
<td>0.5785</td>
</tr>
</tbody>
</table>
3.3. Parameter Optimization and Testing of the Model

The parameter initialization process involves utilizing the fitness function of the GASA algorithm, which is the MSE between the predicted and actual values, to optimize the initial parameters of the three gas content prediction models. To circumvent the potential issue of stochastic errors that may arise in random search algorithms, the GASA was employed to optimize the parameters of each model for 100 iterations. Figure 13 depicts the average evaluation function values during the parameter optimization process. As observed in the figure, the target requirement was not met within 800 iterations while optimizing BPNN parameters using GASA. Hence, the threshold and weight corresponding to the minimum fitness value were selected as the optimal initial parameters and assigned to the BPNN model to complete the parameter initialization process.

As the SVM and KELM models underwent initial optimization of kernel parameters and penalty factors, optimal initial parameters were discovered by the two models in the 673rd and 487th iterations, respectively. Once these parameters were decoded and assigned values, the initialization process was complete. The results depicted in Figure 13 reveal that the optimization speed and quality of the SVM and KELM parameters with GASA were markedly superior to those of the BPNN, which has a larger number of parameters to optimize.

After initializing each model with optimal initial parameter values, we trained the final models separately using the data collected in Table 3. To evaluate the predictive performance of the GASA-BPNN, GASA-SVM, and GASA-KELM models for gas content prediction, we performed 10-fold cross-validation ten times under identical conditions. In each iteration, the training set and test set were input into the model for 100 iterations of training and prediction. The prediction results for the ten test sets in the 10-fold cross-validation are presented in Table 5. As observed from Table 5, the variance of the average relative error and the total average relative error for the coiled methane prediction by the GASA-SVM and GASA-KELM models in each of the ten test sets is lower than that of the GASA-BPNN model. This suggests that the accuracy and stability of the SVM and KELM models for methane content prediction are superior to those of the GASA-BPNN model.

![Figure 13. Average mean square error convergence curves.](image_url)
In the final stage of the study, 12 sets of samples were designated as validation sets, and the three models were utilized to predict the gas content of these samples. The predictive performance of the 12 validation samples, which were included in the test set of all 10 simulated tests, was analyzed, and the prediction outcomes of the three models were compared. The results of the validation sample predictions for the three models are presented in Figure 14.

![Figure 14. The average prediction results of the test set.](image)

Based on the graph, it can be observed that the predicted values of GASA-KELM are in closer proximity to the actual values than those of GASA-BPNN and GASA-SVM. Moreover, the GASA-KELM model displays smaller average relative errors and exhibits less fluctuation in both average relative and absolute errors. Furthermore, the accuracy and stability of the GASA-SVM model surpass those of the GASA-BPNN model. These

<table>
<thead>
<tr>
<th></th>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
<th>A6</th>
<th>A7</th>
<th>A8</th>
<th>A9</th>
<th>A10</th>
<th>Average/</th>
<th>Variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>BPNN</td>
<td>19.46</td>
<td>12.59</td>
<td>16.6</td>
<td>7.53</td>
<td>15.32</td>
<td>17.9</td>
<td>17.82</td>
<td>12.06</td>
<td>20.89</td>
<td>17.21</td>
<td>15.738</td>
<td>3.99</td>
</tr>
<tr>
<td>SVM</td>
<td>13.81</td>
<td>15.3</td>
<td>8.74</td>
<td>15.11</td>
<td>18.29</td>
<td>13.76</td>
<td>10.03</td>
<td>15.58</td>
<td>14.71</td>
<td>13.15</td>
<td>13.848</td>
<td>2.76</td>
</tr>
</tbody>
</table>

Table 5. Ten-fold cross-validation of the prediction results.
findings suggest that the GASA-KELM model possesses the most robust prediction stability and highest prediction accuracy for gas content. The prediction performance of GASA-SVM for gas content is only second to that of GASA-KELM, while the performance of GASA-BPNN for coal seam gas content prediction is relatively subpar in comparison.

3.4. Application of the Model in Engineering and Evaluation of Its Predictive Performance

The developed model was successfully applied for on-site prediction of gas content in the 15th mining area of the Jiulishan Coal Mine in Jiaozuo, Henan. However, it should be noted that the model was developed based on the gas content and related influencing factor data specific to the 15th mining area. This is because different geological units have their own gas geological laws, and the main factors influencing gas content can vary greatly among different mining areas, zones, seams, and mines. Therefore, if the model needs to be applied in a different location, then it is necessary to reanalyze local gas content influencing factors and collect relevant data to retrain the model with new information. Before applying the gas content prediction model on site in the 15th mining area, it is essential to collect gas content influencing factor data from the measurement point. All influencing factors should be carefully recorded, and five sample datasets, as shown in Table 6, could be used as a reference.

Table 6. Field application sample data sheet.

<table>
<thead>
<tr>
<th>Sample Number</th>
<th>$X_0$($m^3\cdot t^{-1}$)</th>
<th>$X_1$(m)</th>
<th>$X_2$(m)</th>
<th>$X_3$</th>
<th>$X_4$(m)</th>
<th>$X_5$</th>
<th>$X_6$</th>
<th>$X_7$</th>
<th>$X_8$(m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>24.03</td>
<td>449</td>
<td>4.75</td>
<td>0.0107</td>
<td>199.44</td>
<td>0.3299</td>
<td>0.1264</td>
<td>0.0027</td>
<td>-364</td>
</tr>
<tr>
<td>2</td>
<td>17.65</td>
<td>402</td>
<td>5.48</td>
<td>0.0285</td>
<td>183.86</td>
<td>0.3810</td>
<td>0.0298</td>
<td>0.0199</td>
<td>-308.6</td>
</tr>
<tr>
<td>3</td>
<td>24.22</td>
<td>305.1</td>
<td>5.39</td>
<td>0.0291</td>
<td>135.74</td>
<td>0.4302</td>
<td>0.0895</td>
<td>0.0615</td>
<td>-218.9</td>
</tr>
<tr>
<td>4</td>
<td>19.71</td>
<td>481</td>
<td>6.24</td>
<td>0.0166</td>
<td>118.10</td>
<td>0.5123</td>
<td>0.0135</td>
<td>0.0047</td>
<td>-394.2</td>
</tr>
<tr>
<td>5</td>
<td>22.98</td>
<td>387.8</td>
<td>5.55</td>
<td>0.0243</td>
<td>98.24</td>
<td>0.5808</td>
<td>0.0494</td>
<td>0.0012</td>
<td>-295.1</td>
</tr>
</tbody>
</table>

Upon inputting the data of the five influencing factors of gas content, as listed in Table 6, into the trained model for prediction, the resulting predictions were analyzed for the five samples when they were all in the test set during 10 simulations. The mean predicted results of the three models for the on-site gas content prediction application process are presented in Figure 15.
Based on the results presented in Figure 15, it can be observed that the GASA-KELM model exhibits superior performance in predicting gas content in coal seams, with a maximum relative error of 16.59% and a minimum relative error of 7.93%. The average relative and absolute errors for the prediction are 10.6% and 2.28%, respectively. In comparison, the GASA-SVM model achieves a maximum relative error of 19.89% and a minimum relative error of 9.01%, with average relative and absolute errors of 13.04% and 2.73%, respectively. Meanwhile, the GASA-BPNN model yields a maximum relative error of 20.64% and a minimum relative error of 10.14%, with average relative and absolute errors of 14.31% and 3.18%, respectively. Taking into account both the relative and absolute errors in the prediction, it can be inferred that the GASA-KELM model is more effective in generalizing to new sample data and provides higher accuracy and stability in predicting gas content in new data samples. Thus, it is better suited to meet the goals and requirements for predicting gas content. The GASA-SVM model performs comparably to the GASA-KELM model in terms of accuracy and generalization ability, while the GASA-BPNN model exhibits relatively lower accuracy than the other two models.

4. Conclusions

The main conclusions are as follows:

1. After conducting verification, Rastrigin’s function was optimized 20 times using the PSO, GA, SA, and GASA algorithms under the same conditions. The algorithms completed the iterative optimization at the 102nd, 156th, 176th, and 78th iterations. The average optimization values of the four algorithms were $9.2472 \times 10^{-4}$, $7.9003 \times 10^{-3}$, $9.1873 \times 10^{-2}$, and $5.6935 \times 10^{-4}$, with respective variances of 3.1547, 3.7519, 7.6823, and 2.0524. After considering the average optimization results over 20 iterations, the
variance of the optimization results, and the average number of iterations, the GASA designed in this paper exhibits stronger capabilities in optimizing complex functions and providing stable global search performance compared to the PSO, GA, and SA algorithms. Furthermore, the GASA algorithm demonstrates a more efficient optimization speed and higher optimization accuracy for complex functions compared to single algorithms, effectively avoiding the issue of optimization algorithms being prone to local optima.

2. In the process of constructing the GASA-BPNN prediction model, the GASA failed to meet the target requirements within 800 iterations. Conversely, during the construction of the GASA-SVM and GASA-KELM gas content prediction models, the GASA was able to discover the optimal initial parameters during the 673rd and 487th iterations, respectively. This disparity can be attributed to the fact that the number of parameters to be optimized in BPNN is significantly greater than in SVM and KELM. As a result, the optimization process for the GASA-SVM and GASA-KELM models was much faster and produced higher-quality results than the BPNN model.

3. During 10-fold cross-validation, the GASA-BPNN, GASA-SVM and GASA-KELM models yielded average relative errors of 15.74%, 13.85%, and 9.87%, respectively. The corresponding variances of the 10 cross-validation results were 3.99, 2.76 and 2.05. Notably, in comparison with the GASA-SVM and GASA-BPNN models, the GASA-KELM model displayed superior accuracy and stability in predicting gas content. Subsequently, the GASA-KELM model was tested on twelve additional samples, which further revealed the model’s exceptional performance in terms of prediction accuracy and generalization ability to new sample data.

4. The developed GASA-KELM model proves to have significant advantages over other ANN models in terms of high accuracy in gas content prediction, stability in prediction, and strong generalization ability when applied to the gas content prediction case of the Jiulishan Mine’s 15-mining area. These advantages are essential for the accurate prediction of gas content and for formulating effective regional gas management strategies.

**Author Contributions:** Conceptualization, S.T. and L.M.; methodology, J.M. and L.M.; software, L.M.; validation, H.L.; formal analysis, L.M.; investigation, J.M.; resources, S.T. and F.T.; data curation, L.M.; writing—original draft preparation, L.M. and J.M.; writing—review and editing, S.T., F.T. and H.L.; visualization, J.M., F.T. and H.L.; supervision, S.T., F.T. and H.L.; project administration, S.T.; funding acquisition, H.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Natural Science Foundation of China (Grant No. U1904210, Grant No. 51874237, and Grant No. 71273208).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data used to support the findings of this study are available from the corresponding author upon request (22120089028@stu.xust.edu.cn).

**Acknowledgments:** The authors would like to thank the Jiulishan Mine of Henan Coking Coal Energy Co., Ltd.

**Conflicts of Interest:** The authors declare no conflict of interest.

**References**


Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.