Prediction of Blasting Vibration Intensity by Improved PSO-SVR on Apache Spark Cluster[1]

Yunlan Wang, Jing Wang, Xingshe Zhou, Tianhai Zhao and Jianhua Gu

School of Computer Science, Center for High Performance Computing
Northwestern Polytechnical University
Xi'an, Shaan xi, China
wangyl@nwpu.edu.cn

Abstract. In order to predict blasting vibration intensity accurately, support vector machine regression (SVR) was adopted to predict blasting vibration velocity, vibration frequency and vibration duration. The mutation operation of genetic algorithm (GA) is used to avoid the local optimal solution of particle swarm optimization (PSO). The improved PSO algorithm is used to search for the best parameters of SVR model. In the experiments, the improved PSO-SVR algorithm was realized on the Apache Spark platform. The execution time and prediction accuracy of the sadovski method, the traditional SVR algorithm, the neural network (NN) algorithm and the improved PSO-SVR algorithm were compared. The results show that the improved PSO-SVR algorithm on Spark is feasible and efficient, and the SVR model can predict the blasting vibration intensity more accurately than other methods.

Keywords: the Prediction of Blasting Vibration Intensity, Improved PSO-SVR Algorithm, Spark.

1   Introduction

In the blasting project, predicting the blasting vibration intensity accurately plays an important role in controlling the impact of blasting vibration. The blasting vibration intensity can be estimated by blasting vibration velocity, which is widely used around the world. In practice, sadovski formula is used to calculate blasting vibration velocity[1]. However, the method is not accurate because of the complex environment and many unknown factors in blasting. In order to predict velocity more accurately, Lv et al. used the non-linear regression method to calculate the parameters of the sadovski formula[2]. Shi et al. proposed to use the SVR model to predict velocity and compared SVR with the neural network (NN) method and sadovski method. The results showed that SVR turned out to be a better prediction method[3]. However, the parameters of SVR are empirically set. So it is unreliable to determine the blasting vibration velocity by the traditional SVR method.

With the further study of blasting vibration, it has been found that blasting vibration frequency plays an important role in the destruction of buildings. When the vibration frequency is close to the inherent frequency of the building, resonance phenomenon may occur and the building can be easily destroyed. In addition, the vibration duration is an important attribute of blasting vibration intensity[4]. Therefore, we use vibration velocity, frequency and duration to predict the blasting vibration intensity, which is better to guide engineering blasting activities. Many scholars used NN that has three nodes in output layer to predict the above three variables simultaneously, and experiments showed that the relative error of NN was lower than other methods[5][6][7][8]. However, NN method is easy to get the local minimum, and the key parameters , such as hidden layer nodes and learning rate, need to be manually set. Especially when there are abnormal points in the blasting data, the over-fitting feature will reduce the accuracy and the stability of NN model.

The work of this paper is as follows: (1) we combine genetic algorithm (GA) to adjust move direction of particles in PSO, and adopt the appropriate fitness function and encoding method; (2) we use improved PSO to search for the best parameters of SVR model, and use the best SVR model to predict the blasting vibration velocity, frequency and duration; (3) based on the blasting vibration data, we complete the improved PSO-SVR algorithm on the Apache Spark computing cluster, and compare prediction accuracy and time performance with other blasting vibration prediction methods. The results show that the improved PSO-SVR algorithm is more accurate, and it is feasible to predict blasting vibration intensity. Meanwhile, the algorithm is more efficient on the Spark cluster than on single node.

2 Improved PSO-SVR Algorithm

We use three algorithms which include support vector machine regression (SVR), particle swarm optimization (PSO) and genetic algorithm (GA). The SVR is used to predict the blasting vibration intensity, PSO is used to optimize the parameters of SVR, and GA is used to improve the PSO.

2.1 Support vector machine regression

Support vector machine regression(SVR) is used to solve the non-linear regression problem. SVR has the following characteristics compared with other methods: (1) a few data can determine the optimal space, so it is not easy to be over-fitted; (2) the abnormal points of training data result in limited impact on the optimal space, thus the SVR model is stable. However, the prediction accuracy depends on the parameters of SVR model, including penalty parameter, insensitive loss coefficient, kernel function and kernel parameter.

1) Penalty parameter: The penalty parameter is used to present the interval error and decide the complexity of the SVR model that is controlled by the number of

support vectors. Small penalty parameter means that there is a relatively large interval, thus the resulting model is relatively simple.

2) Insensitive loss coefficient: The insensitive loss coefficient is used to measure the interval error of each data sample. It also controls the complexity of the model. The larger the parameter is, the fewer the number of support vectors obtained and the simpler the SVR model is.

3) Kernel function: The original feature space maps to the new feature space through the kernel function. Different kernel functions can get different SVR models with different regression functions, so the change of kernel functions will make a big difference in the prediction result of the SVR model[9]. Vol. N. explained the RBF is a better choice for the data without prior knowledge, since blasting vibration data lack of prior knowledge and distribution information[10]. The RBF is shown in formula (1).

$$K(x_i, x_j) = \exp\left(-\gamma \times \|x_i - x_j\|^2\right) \tag{1}$$

4) Kernel parameter: The kernel parameter is related to the distribution characteristics of data. Xiao et al. showed that the performance of the SVR models may vary greatly depending on the different kernel parameters[11]. And Vstvn et al. proved that when the value range is $\gamma = [0.01, 0.2]$, the predicted result of SVR model is well[12].

In summary, the selection of penalty parameter, insensitive loss coefficient, kernel function and kernel parameter largely determine the quality of the SVR model, and these parameters are related to specific data. Therefore, PSO algorithm is used to optimize parameters of SVR model, and make the prediction error of SVR model smallest. Thus the SVR model based on the blasting vibration data is more accurate.

2.2 Particle swarm optimization algorithm

Particle swarm optimization (PSO) was proposed by Dr. Eberhart and Kennedy in 1995 [13], which was used to simulate foraging behavior of birds. In the description of PSO, each bird is treated as a particle, and each particle represents a potential solution in its own position. In each iteration, the particle adjusts the position and velocity according to the optimal position of the individual, the global optimum position and the position of the previous moment. The algorithm stops its iteration until it reaches to the predetermined termination condition.

We define particle's position at the moment t as Xi(t). The i particle's position is shown in formula (2).

$$X_i(t+1) = X_i(t) + V_i(t+1) \tag{2}$$

$X_i(t)$ represents multidimensional vector, and the number of dimensions depends on the number of parameters to be optimized. Velocity $V_i(t+1)$ is shown in formula (3).

$$V_i(t+1) = \omega V_i(t) + c_1 r_1(t)[pbest - X_i(t)] + c_2 r_2(t)[gbest - X_i(t)] \tag{3}$$

$V_i(t+1)$ can be initialized to 0 or a random value within a given range, ω is the inertia weight that describes the particle's ability to retain its inertia. c_1 and c_2 are learning factors which is usually equal to 2, $r_1(t)$ and $r_2(t)$ are random values between 0 and 1. Besides, *pbest* represents the best location of a particle and *gbest* represents the best position of all the particles.

$$p = \{C, \delta, \gamma\} \tag{4}$$

These parameters can be initialized based on their approximate value range. For example, Vstvn et al. gave the range $C = [1,10^8]$, $\delta = [0,0.2]$ and $\gamma = [0.01,0.2]$ [12]. The encoding method makes PSO algorithm be able to optimize multiple parameters simultaneously.

In this paper, the blasting data samples are divided into two parts, one part as training data and another one as test data. The prediction error of the test data can characterize the generalization ability of the SVR model. Therefore, we use the root mean square error (RMSE) function as fitness function to evaluate the quality of particles. The RMSE is shown in formula (5).

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - pre_i)} \tag{5}$$

In above equation, y_i represents the measured value, pre_i represents the predicted value of the SVR model and n is the number of test data samples. The smaller the RMSE is, the better the fitness is.

2.3 Application of genetic algorithm in PSO

The traditional PSO has the possibility of falling into the local optimal solution. The genetic algorithm (GA) can expand the search space through cross operation and mutation operation, and search for the optimal solution to avoid falling into the local optimum. In this paper, we introduce the mutation operation of GA into PSO, the mutation operation is performed on the particle with poor fitness so that the particle can jump out of current search space.

In the algorithm, particles with poor fitness can be defined as follows. For each iteration, when the RMSE of a particle exceeds average RMSE, it can be set as a poor particle, then we change the parameters of the poor particles. At least one parameter should be changed, which is randomly selected. If the fitness value of the changed particle is worse, it is discarded to restore the original position.
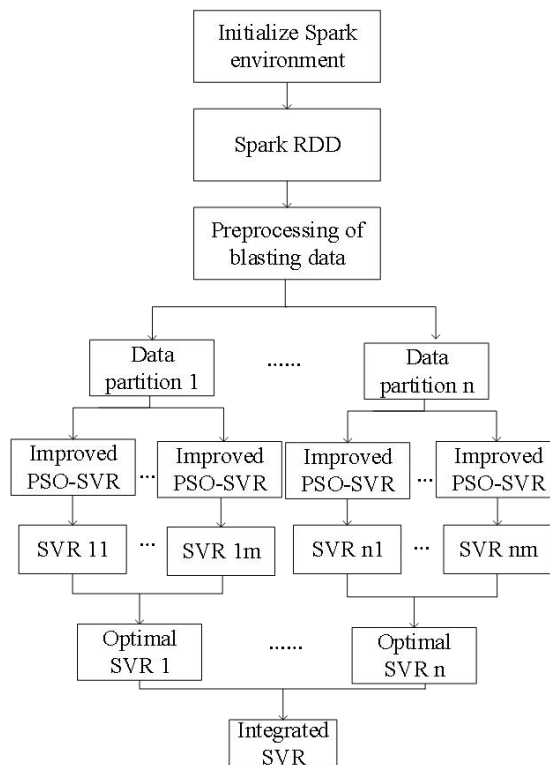
2.4 The steps of improved PSO-SVR algorithm

We use the improved PSO to search for the best parameters of SVR model, then predict blasting vibration intensity with the best SVR model. The steps are as follows:

1) Initialization: Initialize the particle swarm randomly, including population size, initial position and velocity, inertia weight, learn factors and other parameters.

2) Computing fitness value: Compute the fitness value of every particle using the RMSE of the SVR model.

3) Update *pbest* and *gbest*: For each particle, if the current fitness value is better than previous values of this particle, it would be taken as *pbest*. And *pbest* is compared with the best position of other particles, if it is better, then use it as *gbest*.

4) Mutation operation: Select the poor particles to carry out mutation operation, and discard the mutation operation if the fitness value of the particle is worse.

5) Change particle's position: The velocity and position of the particles are updated according to formula (2) and formula (3).

6) Terminate the iteration: If any of the following termination conditions is met:
   a. the maximum number of iterations is reached;
   b. the resulting solution converges;
   c. the desired result is achieved.

the process of the parameters optimization is terminated; otherwise return 2).

3 Parallel Design of improved PSO-SVR on Spark Cluster

Fig. 1. The improved PSO-SVR algorithm on Spark

Spark is a computing engine designed for large-scale data processing, developed by AMP Labs at the UC Berkeley [14]. Master-slave architecture is adopted by it. In spark, the master node is responsible for scheduling tasks, called driver node and the slave node is used to execute the programs, called executor node. They run as separate processes and communicate with each other. Compared to Hadoop, the intermediate results of Spark can be stored in memory, which improves the efficiency of data accessing, so it is suitable for big data mining tasks.

In the case of large population size or large scale data, it will take long time to run PSO algorithm, and sometimes can not get the satisfied results. The improved PSO-SVR algorithm is parallelized on the Spark cluster. As shown in Fig. 1, the main steps of improved PSO-SVR on the Spark cluster are as follows:

1) Initialization of the Spark: Python is used to implement the algorithm and spark-submit script of Spark is used to run the program. The SparkConf object is imported to configure application and SparkContext object is created to access Spark cluster.

2) Data preprocessing: Firstly, the original blasting data is abstracted to resilient distributed dataset (RDD). Secondly, we deal with RDD, including removing duplicate data, filtering data, conversing data and so on, then store the new RDD to Hadoop Distributed File System (HDFS). If necessary, we should cache the data to memory using cache() or persist() method of RDD. After data preprocessing, the quality of blasting data are improved significantly.

3) Train SVR model on data partitions: Before applying a specific algorithm, the data needs to be reasonably partitioned, and the number of RDD partitions should at least be equivalent to the number of CPU cores in the cluster, only in this way we can achieve full parallelism. Then we execute the improved PSO-SVR algorithm on each data partition to obtain multiple SVR models, and finally reserve the optimal SVR model. The process of training SVR model on data partitions is as follows.

   − Initialization: For each data partition, multiple swarm of PSO are randomly initialized, including population size, initialing position and velocity and other parameters.
   − Tasks distribution: Driver node requires resources from the cluster manager and distributes tasks to the executor nodes, then every work node executes algorithm task.
   − PSO optimization: In each iteration of PSO, the particles move according to the position and velocity updating equation, and then carry on mutation operation according to the fitness values of particles.
   − Terminate or not: If the termination condition is satisfied, the training process is ended, and the driver node redistributes the new task to the executor nodes.
   − Terminate tasks: If all the the tasks are completed, the driver node will terminate the executor nodes and release resources through the cluster manager.
   − Return the best SVR: We get multiple SVR models from one data partition and return the best SVR model.

4) Integration of SVR model: The improved PSO-SVR algorithm is implemented on each data partition, and we can get multiple optimal SVR models which meets the user-defined threshold. According to the prediction accuracy of SVR models, these SVR models are integrated into a SVR model using the weighted average method. Then we use the integrated SVR model to predict blasting vibration intensity. The integration method is shown in formula (6) and (7).

$$y^* = \sum_{i=1}^{n} \omega_i y_i \tag{6}$$

$$\omega_i = \frac{ACC_i}{ACC_1 + ACC_2 + \ldots + ACC_n} \tag{7}$$

y^* represents the predicted result of the integrated SVR model, y_i represents the predicted value of every SVR model. ω_i indicates the weight of SVR model, which is related to the accuracy of SVR model.

4 Experiment of blasting vibration intensity prediction

4.1 Experimental environment and data

In the experiment, Spark runs on Hadoop YARN cluster manager. The Spark cluster has four cluster nodes with the same configuration, and the configuration is shown in Table 1. Each node includes two 12-core processors, so it can execute 24 jobs in parallel.

The experiment is based on one thousand of real blasting vibration data samples that provided by remote vibration measurement system developed by Shaanxi China-Blast Safety Web Technology Co., Ltd. Nine attributes of the blasting data is chosen, including the maximum charge per delay, total charge, horizontal distance, dilution time, etc. The properties predicted include blasting vibration velocity, frequency and duration. The blasting data is divided into two parts equally, one part is the training data and the other part is test data.

Table 1. Configuration of single node on Spark

| Software and hardware	Configuration
CPU	Intel (R) Xeon (R) CPU E5-2650 v4 @ 2.20GHz
Memory	128GB
Network card	Gigabit
System disk	480G SSD
Other hard disk	5991.5 GB
Operation system	RedHat Enterprise Linux 6.3 x86_64
Hadoop version	hadoop-2.7.4
Spark version	Spark-2.1.0

4.2 Comparison of prediction accuracy

We use four different methods to predict blasting vibration velocity, frequency and duration, including improved PSO-SVR, NN, traditional SVR and Sadovski method. The parameters of SVR models are showed in Table 2, including the empirical parameters of the traditional SVR model and optimized parameters of the improved PSO-SVR model for velocity, frequency and duration.

Table 2. The parameters of different SVR models

| Model	Attribute	Parameters of SVR			
		C	δ	K	γ
Traditional SVR	velocity	100	0.100	RBF	0.111
	frequency	100	0.100	RBF	0.111
	duration	100	0.100	RBF	0.111
Improved SVR	velocity	24.795	0.101	RBF	0.016
	frequency	74.716	0.056	RBF	0.007
	duration	92.640	0.060	RBF	0.004

As shown in table 2, the parameters of the traditional SVR model has the same empirical values for velocity, frequency and duration. The improved PSO-SVR method results in different parameters for them. The predicted results are shown in Fig. 2, Fig. 3 and Fig. 4. On the abscissa of every figure, thirty samples of test data are selected to show the predicted results.
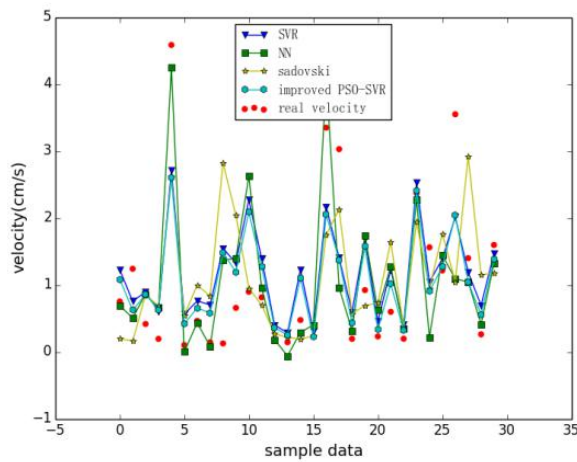
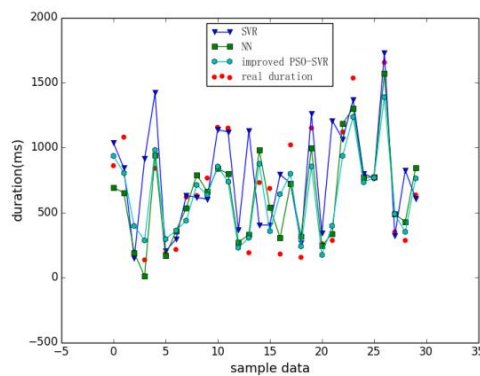**Fig. 2.** The predicted results of blasting vibration velocity

As shown in Fig. 2, the scatter points show the real values of blasting vibration velocity, and the four polylines show the predicted values of four methods, including NN, traditional SVR model, the sadovski method and the improved PSO-SVR method proposed in this paper. According to the figure, the velocity's variation trend of the four methods are similar, and the values predicted by NN and improved PSO-SVR method are much closer to the real values.
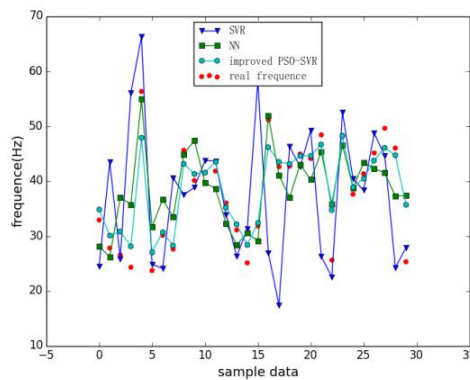
Fig. 3. The predicted results of blasting vibration frequency

As shown in Fig. 3, we use three methods to predict the blasting vibration frequency, including NN method, the traditional SVR method and the improved PSO-SVR method. It can be seen from the figure that the traditional SVR method has a large error between the predicted values and the real values, which is likely because the parameters of the SVR model is unreasonable, while the other two methods are much more precise than traditional SVR.

Fig. 4. The predicted results of blasting vibration duration

As shown in Fig. 4, there are three methods to predict blasting vibration duration, including the NN, the traditional SVR and the improved PSO-SVR. From the figure,

we can see that the variation trend of NN method and improved PSO-SVR method are almost the same as the real values, while the prediction error of SVR method is relatively large.

From the above experimental results, it can be roughly seen that all of the four methods can predict the blasting vibration intensity. In order to evaluate the accuracy of different methods in detail, the relative error of the test data is used. The smaller the relative error is, the higher the prediction accuracy is. The relative error of different methods are shown in Table 3.

Table 3. Relative error of different methods (%)

| Method	Blasting Vibration Intensity		
	Velocity	Frequency	Duration
sadovski	41.7	-	-
SVR	20.3	22.1	24.6
NN	30.2	12.8	11.7
improved PSO-SVR	19.4	8.4	11.5

Table 3 shows the relative errors of the four methods. For the prediction of blasting vibration velocity, the relative errors of SVR and the improved PSO-SVR are much lower than the other two methods. Besides, it can also be seen that the performance of sadovski formula is not good in velocity prediction. For the prediction of frequency and duration, NN and improved PSO-SVR are better than SVR, which means the parameters of SVR need to be determined by blasting data, rather than empirical value. In summary, the improved PSO-SVR algorithm has less error and better prediction ability than other algorithms in the prediction of blasting vibration intensity.

4.3 The comparison of running time on Spark cluster and single node

We achieve the improved PSO-SVR algorithm on the Spark cluster that consist of four nodes. We use ten thousand original blasting data and observe the difference in running time between single node and the Spark cluster.

As shown in Fig. 5, taking the blasting vibration velocity prediction as an example, we compare the running time of the improved PSO-SVR on single node with the Spark cluster of four nodes. When the amount of data is small, the running time on single node is shorter than that on the Spark cluster. The reason is that the initialization, resource allocation, data transmission and nodes communication on Spark cluster. With the data increases, the running time on the Spark cluster is less than single node and their ratio is close to 1/3, thus we infer that the ratio can approach 1/4 when the data is very large. Since there is enough memory at single node, the running time is not affected by memory. But the running time is related to the size of the data and the number of processors. Therefore, the running time on single node linearly increases with the data increases. However, the running time on

the Spark cluster tends to increase slowly because there are four nodes to execute tasks in parallel.
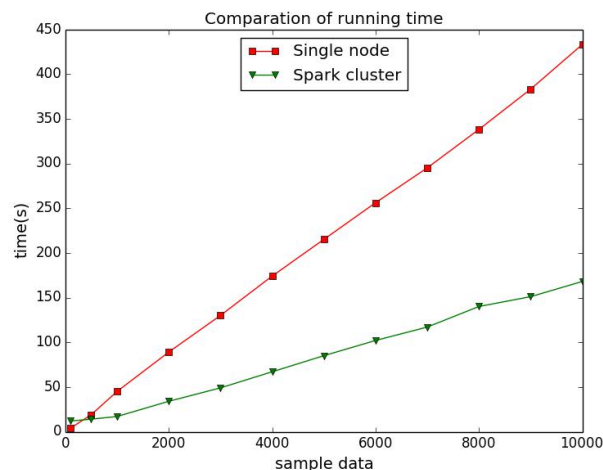
Fig. 5. The running time on single node and Spark cluster

5   Conclusion

Based on the real blasting data, the improved PSO algorithm is adopted to search for the best parameters of the SVR model, and the blasting vibration velocity, frequency and duration is predicted by the optimized SVR model. Results show that the relative prediction error of the improved PSO-SVR method is lower than the other methods. The experiment results also show that the parallel PSO-SVR algorithm on Spark cluster is more efficient than on single node.

However, there are still some problems to be studied in the future. For example, the selection of parameters in the PSO algorithm need to be optimized, and the kernel function of SVR model can be combined with the blasting data and specific application. Since the data is usually stored in multiple data sources such as HDFS and Oracle database, we will study how to access diversity data more quickly from Spark platform.

References

1. Zheng Jinxi. Applicability Research of Sadov's Vibration Formula in analyzing of tunnel blasting vibration velocity. Fujian Construction Science and Technology, 2011 (5): 68-70.
2. Lv Tao, Shi Yongqiang, Huang Cheng, Li Haibo, Xia Xiang, Zhou Qingchun, Li Junru. Study on attenuation parameters of blasting vibration by nonlinear regression analysis. Geomechanics, 2007, 28(9): 1871-1878.

3. Shi Xiuzhi, Dong Kaicheng, Qiu Xianyang, Chen Xiaokang. Analysis of the PPV prediction of blasting vibration based on support vector machine regression. Blasting, 2009, 15(3): 28-30.
4. Chen Shihai, Wei Haixia, Qian Qihu. The study on effect of structure vibration response by blast vibration duration. National Coal Blasting Symposium, 2008.
5. TUMENBAYAR BADRAKH-YERUUL, Xia Anxiong, Zhang Jianhua, Wang Tao. Application of Neural Network based on Genetic Algorithm in Prediction of Blasting Vibration. Blasting 2014 (3): 140-144.
6. Zhi Xiuzhi, Xue Jianguang, Chen Shouru. Study of time and frequency analysis of blasting vibration signal and the prediction of blasting vibration characteristic parameters and damage. Vibration and Shock, 2009, 28(7): 73-76.
7. Wang Jianguo, Huang Yonghui, Zhou Jianming. BP neural network prediction for blasting vibration in open-pit coal mine. 2016 (3): 322-328.
8. Mohamadnejad M, Gholami R, Ataei M. Comparison of intelligence science techniques and empirical methods for prediction of blasting vibrations. Tunnelling and Underground Space Technology, 2012, 28: 238-244.
9. Liu Qingjie, Chen Guiming, Liu Xiaofang, Yang Qing. Genetic algorithm based svm parameter composition optimization. Computer Applications and Software, 2012, 29(4):94-96.
10. Vol. N. Learning With Kernels: Support Vector Machines, Regularization, Optimization, and Beyond/Learning Kernel Classifiers (Book). Journal of the American Statistical Association, 2003, 98:pages. 489-490.
11. Xiao Jian, Yu Long, Bai Yifeng. Survey of the selection of kernels and hyper-parameters in support vector regression.Journal of Southwest Jiaotong University, 2008, 43(3):297-303.
12. Vstvn B, Melssen W J, Oudenhuijzen M, et al. Determination of optimal support vector regression parameters by genetic algorithms and simplex optimization. Anal Chim Acta. Analytica Chimica Acta, 2005, 544(1):292-305.
13. Eberhart R, Kennedy J. A new optimizer using particle swarm theory. 1995.
14. Holden Karau. Learning Spark - Lightning-Fast Big Data Analysis[J]. Oreilly & Associates Inc, 2015.