ACO-RR: Ant Colony Optimization Ridge Regression in Reuse of Smart City System

Qiaoyun Yin[1], Ke Niu[1], Ning Li[1], Xueping Peng[2], and Yijie Pan[3]

[1] Computer School, Beijing Information Science and Technology University
623660923@qq.com,niuke@bistu.edu.cn,ningli.ok@163.com
[2] CAI, FEIT, University of Technology Sydney
xueping.peng@uts.edu.au
[3] Ningbo Institute of Information Technology Application
pyj@nbicc.com

Abstract. With the rapid development of artificial intelligence, governments of different countries have been focusing on building smart cities. To build a smart city is a system construction process which not only requires a lot of human and material resources, but also takes a long period of time. Due to the lack of enough human and material resources, it is a key challenge for lots of small and medium-sized cities to develop the intelligent construction, compared with the large cities with abundant resources. Reusing the existing smart city system to assist the intelligent construction of the small and medium-sizes cities is a reasonable way to solve this challenge. Following this idea, we propose a model of Ant Colony Optimization Ridge Regression (ACO-RR), which is an smart city evaluation method based on the ridge regression. The model helps small and medium-sized cities to select and reuse the existing smart city systems according to their personalized characteristics from different successful stories. Furthermore, the proposed model tackles the limitation of ridge parameters' selection affecting the stability and generalization ability, because the parameters of the traditional ridge regression is manually random selected. To evaluate our model performance, we conduct experiments on real-world smart city data set. The experimental results demonstrate that our model outperforms the baseline methods, such as support vector machine and neural network.

Keywords: smart city system · system reuse · parameter optimization · ant colony · ridge regression.

1 Introduction

In recent years, smart city has become the theme of urban development, which is based on information technology and aims at high-quality and happy city life [1]. Smart cities have played a key role in changing different aspects of human life, involving transportation, health, energy and education [2]. The "intellectualization" of various fields in a city is not enough to make a city smart, but considers the mutual relations between various fields in related cities to realize the intellectualization of the city. Therefore, a smart city is regarded as a system or a whole system [3].

Scholars have solved many problems in different fields through different ways of reuse. Paludo [4] proposed the use of patterns to help the software designer to model business processes. Penzenstadler [5] proposed a concrete artefact model for integrated reuse from requirements to technical architecture, which satisfies documentation demands with respect to functionality and the context assumed by the subsystem. Gasparic [6] performed an industrial exploratory case study to analyze the software reuse process of a medium size company which is a technology leader in a niche market. Kaindl [7] presented an integration of business process and software reuse and reusability in the context of developing software supporting business processes. Oumaziz [8] performed an empirical study of duplications in JavaDoc documentation on a corpus of seven famous Java APIs and proposed a simple but efficient automatic reuse mechanism.

Resolving similar problems by multiplexing the same system is important for reducing costs and shortening time [9]. The construction of smart cities is a very complicated systematic project. There are many smart cities with high degree of intelligence in China. How can we use the existing experience to effectively avoid the waste of resources and other issues when building a smart city? Reuse of smart city system is a good solution. Recently, such methods have not been put forward in the field of smart cities, but in other fields, similar problems were often solved by reusing methods. Ying [10] improved the modeling efficiency and reliability of complex systems by using model reuse methods in the field of Simulation science. Song [11] proposed an aimoes algorithm to solve a multi-objective optimization problem, and introduced an evolutionary scheme that flexibly reused past search experience, further improving the efficiency of the search.

By introducing the reuse of smart city systems into the process of smart city construction, we have solved the problem of waste of resources in the construction of smart cities in small and medium-sized cities and the problem of high manpower and time cost caused by the inability to select the urban objects that conform to the characteristics of the city and guide the construction of the city. In the initial stage of smart city construction, a city needs a set of smart city evaluation index system, which is developed by experts based on the characteristics of the city and specific needs, to evaluate the current wisdom of the smart city and select the appropriate smart city system based on the evaluation results. For different indicator systems, the ACO-RR algorithm that we use to evaluate smart cities can better calculate the weight of each indicator, thus improving the accuracy of automated evaluation of smart city systems. Then choose the appropriate imitation objects suitable for the city and the corresponding smart city system to be applied in the process of intelligent construction to better guide system reuse.

The remainders are organized as follows. In Section 2, we briefly discuss some related work. Then, details about our method are presented in Section 3. In Section 4, we demonstrate the experimental results conducted on real world public datasets. Finally, we conclude our study and prospect our future work in Section 5.

2 Related Work

How to use smart city evaluation system to evaluate the effect of smart city construction and the degree of its development scientifically has been widely concerned by experts and scholars [12]. We can understand the essence of smart city evaluation as ranking and evaluating the level of smart city development [13]. The methods of determining weights in the smart city evaluation process of the existing research include entropy method [14], Analytic Hierarchy Process (AHP) [15], and CRITIC method [16]. The entropy method [14] only calculates the weight from the angle of the index's own information, without considering the influence of the correlation between the indexes on the weight. The Delphi method [17–19] was proposed by Helm and Dark in the 1940s and further developed by Golden and RAND. It first obtains experts' opinions on a certain problem with multiple rounds of communication between an anonymous way, and then sums up the experts' opinions on the results of prediction and evaluation. Delphi with repeated solicitation of opinions has the disadvantage of complex investigation and high resource consumption. When a large number of indicators are needed for evaluation, AHP [15] can not distinguish the importance of different indicators. CRITIC [20] without specific analysis of the correlation between indicators makes the results of different problems deviate greatly.

Zhang [21] used Principal Component Analysis (PCA) method to rank and analyze the smart cities. However, PCA does not consider the influence of the relationship between independent variables and dependent variables on the accuracy of smart city evaluation results. Many researches begin to use machine learning methods to extract information from data to solve various ranking prediction problems. To approach this challenge, we research machine learning method in other specialties. Aljouie [22] Introduced support vector regression and ridge top regression models to cross-verify the SNP with the highest ranking, and calculated the correlation coefficient between the true and predicted phenotypes. Ran [23] used support vector machine regression to solve the estimation problem of photovoltaic system output. Hong [24] realized power management and solves the problem of load/load estimation by using support vector regression. Yang [25] solved a difficult problem with the medical field by using Decision Tree model to predict portal vein prostatitis vein thrombosis after acute pancreatitis. In the field of smart city evaluation, in 2019, Ma [26] assessed the level of intelligent city construction in 2011-2016 by trained neural network model with smart city data. Compared with resource-rich big cities, the intelligent construction of small and medium-sized cities is a difficult challenge. How to accurately evaluate smart cities, and then help small and medium-sized cities choose smart city systems that meet their own characteristics to assist the city's intelligent construction is particularly important. We propose an evaluation model ACO-RR.

The principle of ACO-RR is based on ridge regression algorithm and ant colony algorithm is used to find the optimal ridge parameters in the process of training model. It solves the influence of multi-collinearity of data onto sorting results and avoids the randomness of manual determination of parameters in traditional ridge regression algorithm. we have proved that ACO-RR algorithm can be used in the task of accurate evaluation of smart cities under different smart city evaluation systems.

3 Proposed ACO-RR Model

3.1 ACO-RR Algorithm Flow

ACO-RR realizes automatic parameter tuning of the model by using group intelligence to find the appropriate ridge parameters. As shown in Fig. 1. Firstly, the solution space is divided, the information concentration and other parameters in each space are initialized, and ants are randomly placed in each space. The second step are to calculate the transition probability of each ant transferring to other space based on the existing information and the information concentration on the path, and determine whether to transfer the ants according to the magnitude of the probability value. In addition, we need to update the information concentration on the path after the ant has transferred. At this time, the search solution space will gradually shrink. We can finally find the optimal solution by repeating the above process. The core of the whole algorithm is the selection of the objective function. The ultimate goal of the algorithm is to make the model learn the intrinsic information reflected by different evaluation index systems, and find out the influencing factors reflecting the evaluation indicators. Therefore, the objective function of the algorithm is set as the accuracy function of the ridge parameter for smart city evaluation.

3.2 Algorithm Principle

The difficulty of ridge regression is how to determine the parameters of the model, but the traditional ridge regression has a certain randomness by manually selecting parameters, which makes the results less accurate. In order to solve this problem, the ant colony algorithm is introduced to optimize the parameters, which improves the robustness of the algorithm and the accuracy of the evaluation. Ant colony algorithm is a simulation optimization algorithm that finds the optimal solution by simulating the ant foraging behavior. It can quickly converge to the optimal solution by using the group wisdom of the ant colony to solve the optimal problem. During the search process, ants continuously release pheromones so that individuals in the ant colony can acquire information of other ants by capturing pheromones to realize information transmission. Each ant calculates the transfer probability of the corresponding space according to the pheromone concentration of each transferable space,and then decide which space to transfer to according to the probability values of different spaces. The

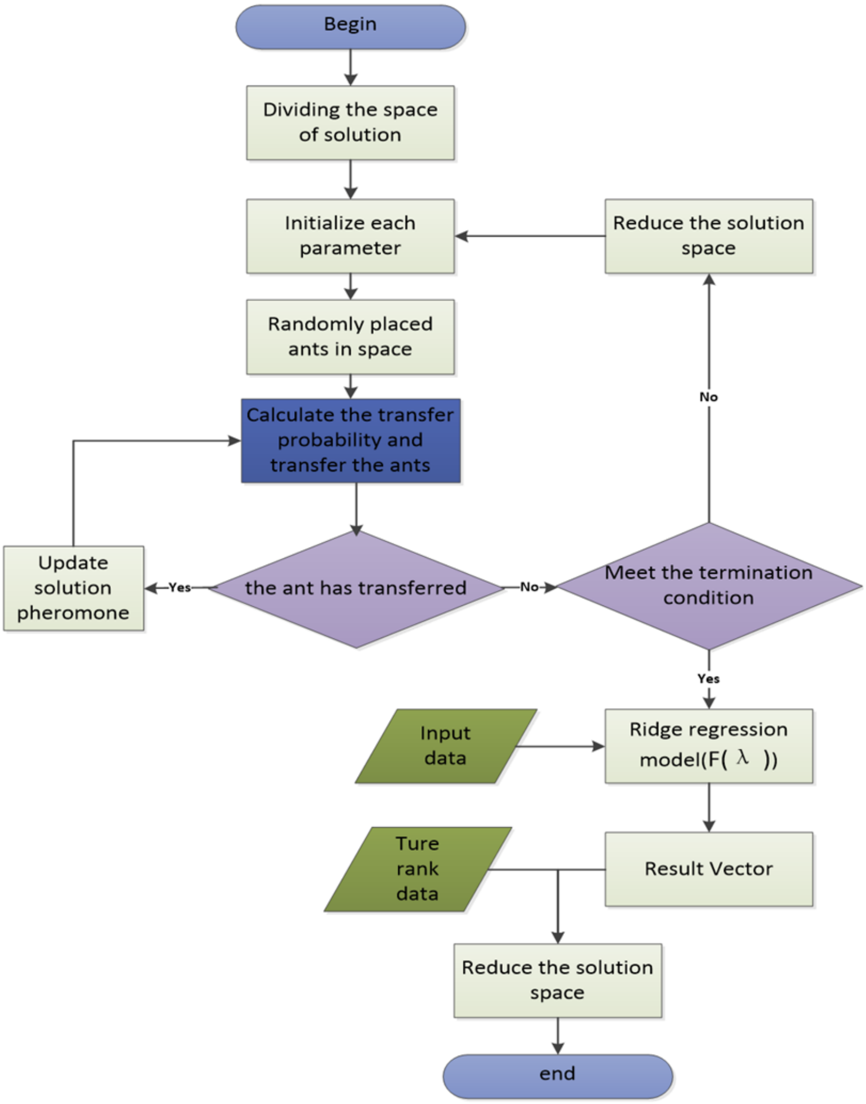
Begin
Dividing the space of solution
Initialize each parameter
Reduce the solution space
Randomly placed ants in space
No
Calculate the transfer probability and transfer the ants
Update solution pheromone
Yes
the ant has transferred
No
Meet the termination condition
Yes
Input data
Ridge regression model(F(λ))
Ture rank data
Result Vector
Reduce the solution space
end

Fig. 1. The main flow chart of the ACO-RR algorithm.

pheromone of the solution space needs to be updated after the transfer. Repeat the iteration of the above process until the ants no longer transfer, and finally the ant colony will gather near the optimal solution. The ant colony algorithm was originally proposed to solve the discrete optimization problem, while the parameter optimization of ridge regression is a continuous optimization problem. Therefore, the ACO-RR algorithm proposed in this paper is to optimize the parameters in the ridge regression model based on the idea of using ant colony algorithm to solve the extremum of multimodal functions.

Assuming that the optimization problem function is $f(x), x \in [a, b]$, algorithm divides the domain $[a, b]$ into n subintervals, equation 1 is the length of the divided interval. Each interval is denoted as $I_1, ..., I_i, ..., I_n$, and equation 2 is denoted as solution space I. If the middle position of the interval is represented by X_i, which can be calculated by equation 3. Ants in ant colony calculate the transition probability value to each interval in the transferable subinterval set according to pheromone of each interval and the transition cost v from their current position to the next interval, and select the interval with the largest probability value for transition. Ants will release pheromones in a certain interval after transferring to that interval.

$$\iota = (a - b)/n \tag{1}$$

$$I_i = [a + (i - 1) * \iota, a + i * \iota] \tag{2}$$

$$X_i = a + (i - 1/2) * \iota \tag{3}$$

Initialize the Position and Interval Pheromone Concentration of Ant Colony m is the number of ants in the ant colony, and n is the number of divided solution spaces. In the first step, all ants are randomly placed in each solution space to complete the initialization of ant colony positions. In addition, the pheromone amount released by each ant at its location is expressed by *const*, and then we can obtain the initial information concentration of each space.

Transferable Set T_{si} (Transfer set i) is defined as a set of transferable subspaces of ant i as shown in equation 4, where k is used to limit the ant transfer range. We take the midpoint position X_i to calculate the corresponding target value $f(X_i)$ and determine whether the current space is the optimal solution space.

$$T_{si} = \begin{cases} \{I_1, ..., I_i, ..., I_{i+k}\}, & 1 \leq i \leq k \\ \{I_{i-k}, ..., I_i, ..., I_{i+k}\}, & k < i \leq n - k \\ \{I_{i-k}, ..., I_i, ..., I_n\}, & n - k < i \leq n \end{cases} \tag{4}$$

Ant Colony Transfer Each ant selects a transferable subspace with a certain limit, and selects the subspace to be transferred by comparing the magnitudes of the spatial transition probabilities. If $f(X_i) - f(X_j) < 0$, the transfer occurs, otherwise the ant does not transfer. Heuristic function is defined as B_{ij} because the transfer cost of solution space i and solution space j is $|f(X_i) - f(X_j)|$. $P_{ij}(t)(s)$ indicates that the ant t is transferred from subinterval I_i to subinterval I_j at the s th iteration as shown in equation 6, where A_j is the pheromone content of interval j at the current moment, and φ is information inspiration. ϕ is the expected heuristic factor.

$$B_{ij} = |f(X_i) - f(X_j)| \tag{5}$$

$$P_{ij}(t)(s) = \begin{cases} 0, & otherwise \\ \dfrac{A_j^{\varphi}(s)B_{il}^{\phi}}{\sum_{l \in T_{si}} A_l^{\varphi}(s)B_{il}^{\phi}}, & j \in T_{si} \end{cases} \tag{6}$$

Pheromone Update The information increment of ant colony corresponding to subinterval j after completing one transfer is $\bigtriangledown A_j^t(s)$. If ant t transfers from I_i to I_j, ant t will release a certain amount of pheromone $\bigtriangledown A_j^t(s)$ on I_j in the process of iteration s, c_1 is a normal number. while all ants complete one transfer, the algorithm updates the information concentration of all subintervals as shown in formula 8, where χ is the pheromone volatilization factor, and $A_j(s)$ represents the information concentration of current interval j.

$$A_j^t(s) = c_1(f(x_i) - f(x_j)) \tag{7}$$

$$A_j(s+1) = (1-\chi)A_j(s) + \bigtriangledown A_j(s) \tag{8}$$

Narrow Solution Space After several times of ant colony transfer, all ants will concentrate in the interval near the maximum value, and there are no ants in other intervals. These ant - containing intervals are taken out as new solution spaces, and then the above steps are repeated until the refined space is small enough. We found that the position where the ant stayed at this time was the position of the required optimal solution.

3.3 Specific Steps of the Algorithm

At present, all existing smart cities have their own smart city systems. For other cities, which are going to build smart cities, effective use of existing experience to solve the problem of smart city construction and realize the reuse of smart city system can not only improve the construction process and efficiency, but also avoid the waste of a large number of human and material resources, and also help to control the overall and direction of construction. The core of smart city system reuse is to use the evaluation results to help small and medium-sized cities that need smart city construction to choose a suitable smart city system. Therefore, the accuracy of evaluation results is the key issue.

4 Experiments

According to the given evaluation index system, we established ACO-RR model to evaluate the smart city, and verified the accuracy of the model through a number of experiments.

Compared with the experimental results of other algorithms, we found that ACO-RR had higher accuracy and stronger stability. The experiment consisted of three parts: in order to verify whether the algorithm always has high accuracy for smart city evaluation under different index systems, the first part of the experiment is a comparative experiment based on three different smart city evaluation systems in 2013, 2014 and 2015 respectively. The second part of the experiment mainly considers the influence of different data samples on the accuracy of the algorithm. In addition, we use the third part of experiments to verify the change trend of algorithm accuracy with the change of sample size.

4.1 Experimental Data

Our experimental data come from the evaluation report on the development level of China's smart cities issued by the Chinese Academy of Social Sciences and Other Smart City Research Center in 2013, 2014 and 2015. The extraction of data in the report were collected in the following four ways: Literature review, Network surveys, Telephone surveys and Technical inspections. Since the smart city data in recent years have not been published, we used the smart city data sets of 2013, 2014 and 2015 as the experimental data of different evaluation index systems. Through comparing the results of several groups of comparative experiments, we found that ACO-RR algorithm was more suitable for solving the problem of smart city evaluation with different evaluation index systems. Sample 1 is the smart city data of 2013, sample 2 is the smart city data of 2014, and sample 3 is the smart city data of 2015.

Partial sample data of the experiment are shown in Table 1. The focus of this paper is not on the construction of the index system, we will not describe the construction and composition of the evaluation index system. Because the corresponding smart city evaluation index systems in the smart city development level evaluation reports of different years are different, the published three-year city data and the evaluation index system of corresponding years are used for subsequent experiments. Before the experiment, The data need to be standardized in order to eliminate the influence caused by different dimensions of different indexes.

4.2 Experimental Methods and Processes

Experimental Methods

Decision Tree Decision Tree is a common class of machine learning methods. First, we select the optimal partitioning attribute at the splitting node, and use the value of this attribute to partition the sample to generate many new

Table 1. Partial Sample Data

Indx	Wuxi	Pudong	Ningbo	Shanghai	Hangzhou	Beijing
Index1	71.4	102.9	80.85	89.25	78.75	71.4
Index2	94.5	94.5	63	73.5	94.5	73.5
Index3	36.75	31.5	21	31.5	26.25	21
Index4	157.5	126	147	126	105	157.5
Index5	36.75	21	36.75	52.5	42	21
Index6	96.6	86.1	96.6	92.4	84	94.5
Index7	84	47.25	73.5	73.5	68.25	73.5
Index8	47.25	36.75	47.2	21	31.5	21
Index9	42	105	73.5	42	42	52.5
Index10	77.7	81.9	54.6	56.7	46.2	100.8
Index11	52.5	94.5	42	94.5	84	105
Index12	105	52.5	81.9	79.8	77.7	75.6
Index13	42	84	42	63	105	42
Index14	105	126	126	115.5	115.5	126
Index15	105	63	105	63	73.5	84
Index16	73.5	73.5	84	63	73.5	63

splitting nodes. Then, we repeat the above process of each new splitting node, and finally generate a Decision Tree with strong generalization ability. In the experimental part, the Decision Tree model was applied to solve the regression prediction problem by dividing the feature space into several units with specific output. Each partition checked the values of all the features in the current set and chose the best value as the segmentation point according to the criterion of minimum square error.

SVR Support Vector Machine is proposed for the principle of structural risk minimization, which overcomes the disadvantages of traditional models caused by the principle of empirical risk minimization and has strong generalization ability. Owing to the regression problem of smart city evaluation problem, we applied the method of SVR to make empirical analysis. The goal of SVR is to find a regression hyperplane to minimize the distance from all sample data to the plane. Given training samples $D = (X_1, Y_1), (X_2, Y_2), ..., (X_n, Y_n)$, model will learn an $f(x)$ as close as possible to Y. Assuming that there is a maximum ϵ deviation between $f(x)$ and Y that we can tolerate, the loss is calculated when the absolute value of the difference between $f(x)$ and Y is greater than ϵ. This is equivalent to building a spacing band with a width of 2ϵ with $f(x)$ as the center. The training sample will be considered to be the predicted correct sample if it falls within this interval band.

Neural Network Neural network is essentially a neural network with error back propagation. It consists of an input layer, an output layer and an implicit layer, wherein the implicit layer may have more than one layer. The learning process of neural networks with different hidden layers is similar. Firstly, the connection weights of each hierarchical structure in the network are initialized, and the training sample data is selected to calculate the mean square deviation of the final output data and the actual data through the hierarchical structure of the network to determine whether the next sample needs to be selected to continue iteration. Then reverse error propagation is carried out to further correct the error. The initial weights of each layer of the neural network are randomly generated, and the slow training speed of the network leads to the local minimum problem.

Experimental Processes The experimental data were standardized by the specific gravity method. According to the results of literature research, we selected Decision Tree [27] and SVR [22, 23] algorithms that had achieved good results in other fields, and neural network [26] algorithms used in the field of smart city evaluation as the comparison algorithms to verify the effectiveness of ACO-RR in this paper.

In the first part, we conducted three groups of experiments by using Decision Tree, SVR, neural network and ACO-RR on three different samples. In each group of experiment, the data were first divided into training sets and test sets. The model trained by the training sets was used to evaluate smart cities in the test sets. Pearson correlation coefficient between the evaluation results vector of the model and the actual ranking result vector of the city was calculated as the accuracy of the model. In order to prevent possible differences caused by data sets, the same training set and test set were used to train their models respectively with four algorithms in the intra-group experiments, and the trained models were used to evaluate smart cities. In this part, there were a total of 12 experiments in three groups. The first group consisted of experiments 1-1, 1-2, 1-3, and 1-4. The four experiments were based on sample 1 using Decision Tree, SVR, and neural network. Sample 1 was used as experimental data in the four experiments, and then Decision Tree, SVR, and neural network and ACO-RR algorithms were respectively used for comparative experiments. The second set of experiments consisted of experiments 1-5, 1-6, 1-7, 1-8, the four experiments in this group used sample 2 as experimental data to compare the effectiveness of the four algorithms. The third group consisted of experiments 1-9, 1-10, 1-11, 1-12. Similarly, these four experiments were comparative experiments of four algorithms based on sample 3.

The second part of the experiment consisted of four groups of 16 experiments. First, we randomly divided sample 3 into four times to obtain four sets of test sets and training set samples of the same scale, and the data sets obtained by the four divisions were represented by D1, D2, D3, and D4. We used the above four algorithm training models to evaluate the smart city, and finally calculated the accuracy of each model separately. The first set of experiments consisted of experiments 2-1, 2-2, 2-3, and 2-4. The four experiments were based on the data set D1 using Decision Trees, SVR, neural networks, and ACO-RR algorithms for comparison experiments. The difference between the three sets of experiments and the first set of experiments was only that the data sets were different. The second set of experiments was based on the comparative experiment of the data set D2 through four algorithms, and the third and fourth sets were based on the data sets D3 and D4 respectively.

The third part of the experiment consisted of 12 groups of 48 experiments. In each of the 12 sets of experiments, we conducted four experiments using the above four algorithms to train the model and evaluate the smart city. Throughout the third part of the experiment, we gradually increased the number of data samples in the training set to observe the change trend of accuracy rate of different algorithms with the change of data sample size.

4.3 Experimental Results and Analysis

We used the third-party package in Python to construct the model of comparative experiments, and set the default parameter value to the parameter value of the model. The parameters of SVR model such as C value set to 1.0, cache size value set to 200, degree value set to 3, epsilon value set to 0.1, kernel value set to RBF. By observing the ridge regression parameters of ACO-RR algorithm, we found that the parameters of the model constructed under different index systems will be different. For example, the ridge parameter under sample 1 is 0.65, while that under sample 2 is 0.33. In fact, ACO-RR algorithm can quickly and effectively realize intelligent city evaluation because the algorithm can automatically find such differences.

The accuracy of the results of all experiments was shown in Table 2. We measured the accuracy of the evaluation results of the algorithm model by calculating the Pearson correlation coefficient between the two vectors. The larger the Pearson correlation coefficient was, the closer the evaluation results of the model were to the real ranking of the city, which also showed the higher the accuracy of the algorithm. The comparison results of algorithm accuracy under different index systems were shown in Fig. 2. The left-to-right bar chart in the figure showed the accuracy of Decision Tree, SVR, neural network and ACO-RR algorithm in evaluating smart cities based on different evaluation systems respectively. It is not difficult to find that the Pearson correlation coefficient of ACO-RR algorithm is always the highest compared with the other three algorithms, which further illustrates the applicability of this calculation in the field of smart city evaluation. In the second part of the experiment, the results obtained by different division methods of data are shown in Fig. 3. The figure shows that the accuracy of SVR, neural network and Decision Tree algorithms are greatly affected by the difference of samples under different sample data with

Table 2. Algorithm Accuracy under Different Index Systems

Year	Decision Tree	SVR	neural_network	ACO-RR
2013	0.96	0.87	0.87	0.99
2014	0.97	0.92	0.89	0.99
2015	0.98	0.97	0.96	0.99

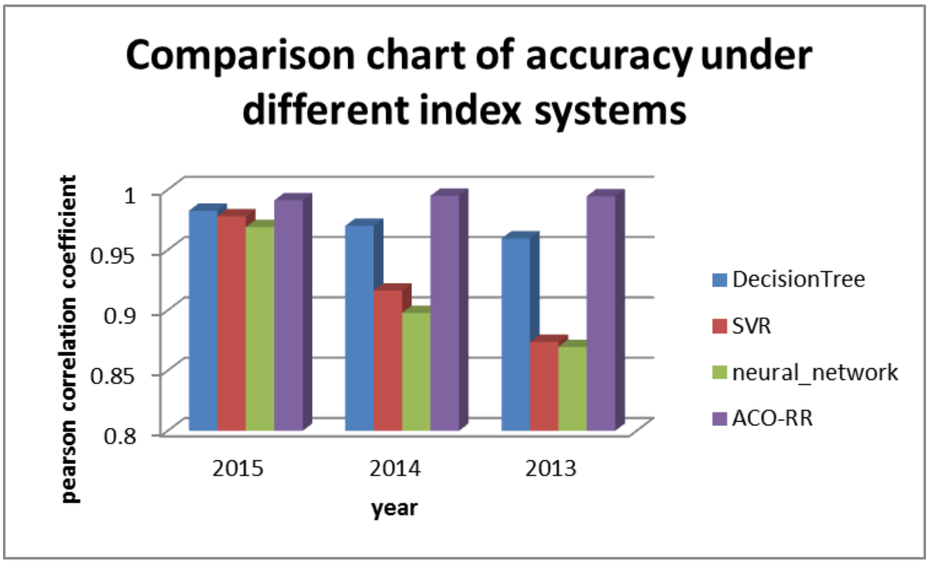

Fig. 2. Comparison chart of accuracy under different index systems

the same data size, while the accuracy of ACO-RR algorithm is basically stable, which shows that the stability of the algorithm is higher than that of the other three algorithms.

The experimental results in the third part are shown in Fig. 4. No matter how the sample size changes, the accuracy rate obtained by ACO-RR algorithm is always higher than that of the other three algorithms, thus verifying that ACO-RR algorithm can evaluate smart cities more accurately. From the curve trend in the figure, we find that the accuracy of all algorithms has been improved and gradually converge with the increasing sample size. However, ACO-RR always has the highest accuracy rate, which indicates that the algorithm is more suitable for solving smart city evaluation problems. Through the above experimental results, we come to the conclusion that SVR, Decision Tree and neural network algorithms are highly sensitive to data sets. When the data sets change, the accuracy of the model changes greatly. Neural network is actually an optimization method of local search in which the weights in the network are gradually adjusted along the direction of local improvement. It is suitable for solving a complex nonlinear problem. In fact, the distribution characteristics of smart city evaluation index data are not suitable for evaluation by neural network. Because the network often learns too many sample details to make the learned model unable to reflect the rules contained in the sample. In fact, network based on data distribution characteristic of smart city evaluation index usually learn too many sample details, resulting in that learned model cannot reflect the rule contained in the samples. Therefore, neural networks are not suitable for the evaluation of smart cities. However, the support vector machine algorithm mainly determines

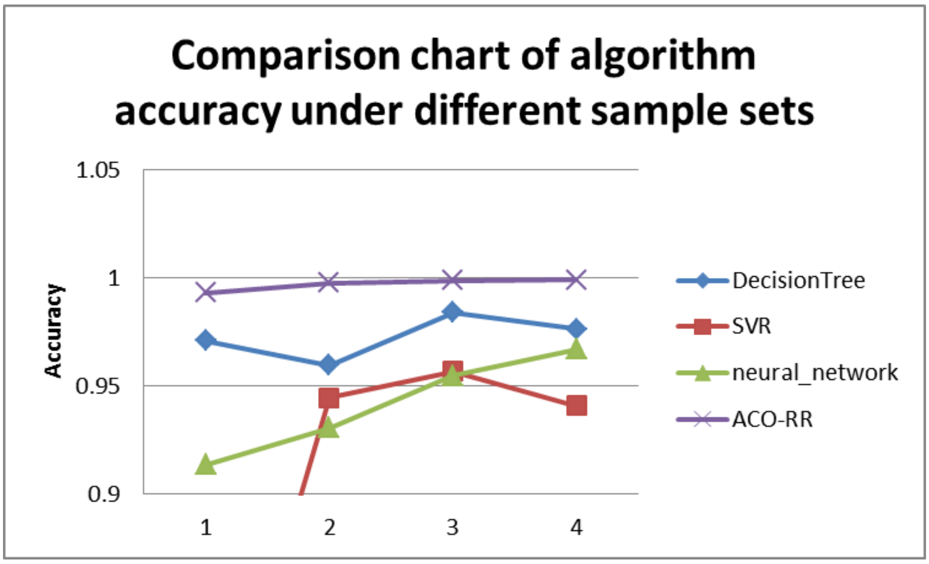

Fig. 3. Comparison chart of algorithm accuracy under different sample sets

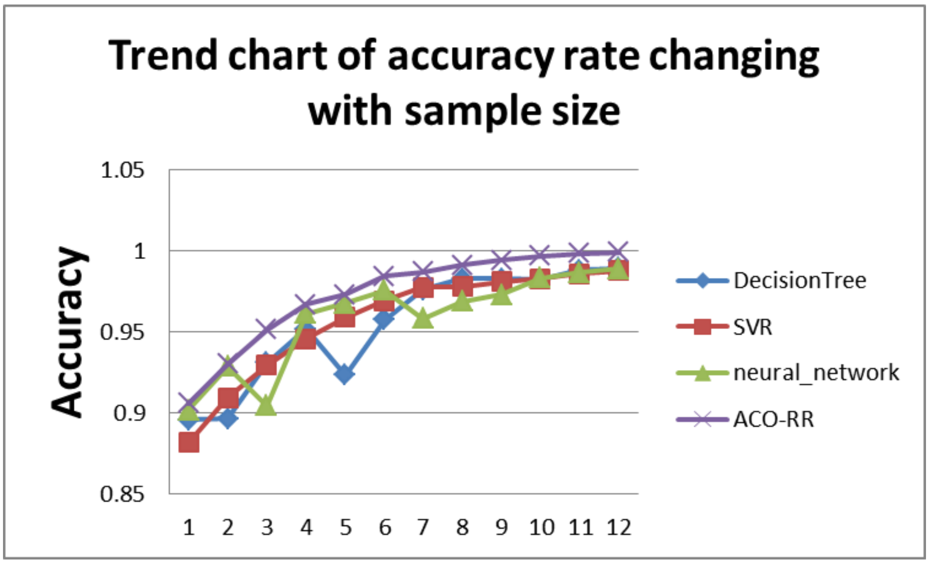

Fig. 4. Trend chart of accuracy rate changing with sample size

whether the output data of the model is close to the actual data by dividing the region band. As the errors in the region band can be accepted, the evaluation accuracy cannot reach higher accuracy. The low accuracy of decision trees is mainly due to the fact that a small change in the data may lead to the generation of a completely different tree, and the decision trees are easily over-fitted by dividing with features as nodes. Through many groups of experimental results, it is proved that ACO-RR algorithm not only has higher accuracy but also has stronger applicability and stability in solving the problem of smart city evaluation. In the early stage of the construction of a smart city, we can use ACO-RR algorithm to evaluate the existing smart city under the index system formulated by experts that conforms to the city's own characteristics. According to the evaluation results, an appropriate smart city system is selected and applied to the construction of smart cities. By reusing the smart city system, the problem of insufficient resources in small and medium-sized cities is solved to a certain extent, meanwhile, a large amount of manpower and material resources are saved, and the construction efficiency is further improved.

5 Conclusion

The algorithm proposed in this paper uses the regularization of ridge regression to solve the multicollinearity problem of data, and introduces the optimization of ant colony algorithm to solve the randomness problem of artificial parameters. We have proved that ACO-RR algorithm can always capture the changes of information and maintain high accuracy despite the changes in the index system. When a small and medium-sized city is ready to build a smart city, the first step is to use the evaluation system given by the city experts to evaluate the existing smart city combined with the algorithm in this paper, then select the corresponding smart city system according to the smart city with the highest degree of intelligence, and finally reuse the smart city system in the construction process. This not only solves the problem of waste of resources in the construction of smart cities, reduces the development time and labor costs, but also further improves the reliability and efficiency.

In the experiment, different intelligent city index systems were used for model training, and the results show that ACO-RR algorithm is more suitable for smart city evaluation than other algorithms no matter how the smart city index system change. Other experimental results of smart city evaluation under different data samples show that the proposed algorithm is more stable than other algorithms. In addition, through a variety of comparison algorithms, it can be proved that ACO-RR algorithm can improve the accuracy of evaluation results for smart cities. Therefore, with the continuous expansion and rapid development of smart cities, we can try to evaluate smart cities with this algorithm. At present, the main method of smart city evaluation is to weight the index data of each city, calculate the score of smart degree of the city and rank it. The evaluation of smart city is also a multi-objective decision-making problem. In the future work, we will focus on the study of whether some comprehensive evaluation models can solve the evaluation of smart city. With the gradual acceleration of the development of smart cities, the diversity of data will be further considered when evaluating smart cities in the future.

6 Acknowledgment

This paper is supported by the National Key R & D Program of China (No. 2018YFB1004100), the Beijing Education Commission Research Project of China(No. KM201911232004) and the National Natural Science Foundation of China (No. 61672105).

References

1. Pan, J. G., Lin, Y. F.,Chuang, S. Y., Kao,Y. C.: From Governance to Service-Smart City Evaluations in Taiwan. International Joint Conference on Service Sciences (74), 334–337 (2011)
2. Hashem, I. A. T. , Chang, V. , Anuar, N. B., et al: The Role of Big Data in Smart City. International Journal of Information Management **36**(5), 748–758 (2016)
3. Osman-Shahat,M. A.: A Novel Big Data Analytics Framework for Smart Cities. Future Generation Computer Systems (91), 620–633 (2019)
4. Paludo M., Burnett R., Jamhour E. Patterns Leveraging Analysis Reuse of Business Processes. In: Frakes W.B. (eds) Software Reuse: Advances in Software Reusability. ICSR 2000. Lecture Notes in Computer Science, vol 1844. Springer, Berlin, Heidelber (2000).
5. Penzenstadler B., Koss D. High Confidence Subsystem Modelling for Reuse. In: Mei H. (eds) High Confidence Software Reuse in Large Systems. ICSR 2008. Lecture Notes in Computer Science, vol 5030. Springer, Berlin, Heidelberg (2008).
6. Gasparic M., Janes A., Sillitti A., Succi G. An Analysis of a Project Reuse Approach in an Industrial Setting. In: Schaefer I., Stamelos I. (eds) Software Reuse for Dynamic Systems in the Cloud and Beyond. ICSR 2015. Lecture Notes in Computer Science, vol 8919. Springer, Cham (2015).
7. Kaindl H., Popp R., Hoch R., Zeidler C. Reuse vs. Reusability of Software Supporting Business Processes. In: Kapitsaki G., Santana de Almeida E. (eds) Software Reuse: Bridging with Social-Awareness. ICSR 2016. Lecture Notes in Computer Science, vol 9679. Springer, Cham (2016).
8. Oumaziz M.A., Charpentier A., Falleri JR., Blanc X. Documentation Reuse: Hot or Not? An Empirical Study. In: Botterweck G., Werner C. (eds) Mastering Scale and Complexity in Software Reuse. ICSR 2017. Lecture Notes in Computer Science, vol 10221. Springer, Cham (2017).
9. Yao, Y., Liu, M., Du, J., et al: Design of A Machine Tool Control System for Function Reconfiguration and Reuse in Network Environment. Robotics and Computer-Integrated Manufacturing(56), 117–126 (2019)
10. Liu, Y.Zhang, L.,LaiLi, Y. J.: Study on Model Reuse for Complex System Simulation. Scientia Sinica, (2018)
11. Song, S., Gao, S., Chen, X., et al: AIMOES: Archive Information Assisted Multi-objective Evolutionary Strategy for Ab Initio Protein Structure Prediction. Knowledge-Based Systems, (2018)

12. Luo,S. L., Xia, H. X.: Reflections on the Evaluation of Smart Cities from the Perspective of Capability Maturity. Scientific research management **s1**, (2018)
13. Gu, D. D., Qiao, W.: Research on the Construction of Evaluation Index System for Smart Cities in China. Future and Development **35**(10), 79–83 (2012)
14. Liu, W. Y., Wang, H. L., Liu, K. G., Zhou, X. X.: Applying the Combination Model of Entropy Weight and TOPSIS to Construct the Evaluation System of Smart City - Taking Beijing, Tianjin and Shanghai as An Example to Explore. Modern City Research **1**, 31–36 (2015)
15. Qi, J. Q., Ba, Y. Q.: Smart City Construction Evaluation System Study Based on the Specialists Method and Analytic Hierarchy Process Method. International Conference on Smart City and Systems Engineering (115), 149–152 (2016)
16. Liu, Y. L., Cao, W. J.: Application of CRITIC-GREY Comprehensive Evaluation Method in Quality Evaluation of Medical Work. China Health Statistics **33**(6), 991–993 (2016)
17. Wu, C. Z., Ma, L. L., Zhang, B. G., Hong, Z. Z.: Study on Indicators Choosing for Navigation Safety Assessment of Three Gorges Reservoir Areas based on Delphi method. Asia-Pacific Conference on Information Processing , 282–285 (2009)
18. Zeng, G., Li, H.: Method and Application of Modern Epidemiology. Beijing: Joint Press of Beijing Medical University and Beijing Xiehe Medical University, 250–259(1994)
19. Hu,C. P.,Yang, J. : Delphi Method in Building A Government Performance Indicators System to the Township Government-as An Example. Journal of Shaanxi Institute of Administration **4**(21), 12–15 (2007)
20. Diakoulaki, D., Mavrotas, G.,Papayannakis, L.: Determining Objective Weight-s in Multiple Criteria Problems : the Critic Method. Computers and Operations Research **22**(3), 763–770 (1995)
21. Zhang, N.,Sheng W.: Research on the Development of Intelligent Cities Based on Principal Component Analysis and Entropy Method. Journal of Urban Science (03), 30–335 (2018)
22. Aljouie, A., Roshan, U.: Prediction of Continuous Phenotypes in Mouse, Fly, and Rice Genome Wide Association Studies with Support Vector Regression SNPs and Ridge Regression Classifier. IEEE International Conference on Machine Learning and Applications, (2016)
23. Ran, L. I., Guang-Min, L. I.: Photovoltaic Power Generation Ooutput Forecasting Based on Support Vector Machine Regression Technique. Electric Power (02),(2008)
24. Hong, W. C.: Electric Load Forecasting by Support Vector Model. Applied Mathematical Modelling **33**(5), 2444–2454 (2009)
25. Yang, F., Kun, G., Jian, H.: Predicting the Incidence of Portosplenomesenteric Vein Thrombosis in Patients With Acute Pancreatitis Using Classification And Regression Tree (CART) Algorithm. Journal of Critical Care **(**39), 124-130 (2017)
26. Ma, Q. T., Shang, G. Y., Jiao, X. X.: Research on Evaluation of Smart City Construction Level Based on BP Neural Network. Practice and Understanding of Mathematics **48**(14), 64–72 (2018)
27. Ramezankhani, R., Sajjadi, N., Nezakati, R. E., et al: Application of Decision Tree for Pprediction of Cutaneous Leishmaniasis Incidence Based on Environmental and Topographic Factors in Isfahan Province. Geospatial Health **13**(1), 664 (2018)