NUMERIC PREDICTION OF DISSOLVED OXYGEN STATUS THROUGH TWO-STAGE TRAINING for CLASSIFICATION-DRIVEN REGRESSION

PENGFEI GUO[1,2], HAN LIU[3,*], SHUANGYIN LIU[2,4], LONGQIN XU[2,4]

[1]College of Computational Science, Zhongkai University of Agriculture and Engineering, Guangzhou,510225, China
[2]Intelligent Agriculture Engineering Research Center of Guangdong Higher Education Institues, Guangzhou, 510225, China
[3]School of Computer Science and Informatics, Cardiff University, Cardiff CF24 3AA, United Kingdom
[4]College of Information Science and Technology, Zhongkai University of Agriculture and Engineering, Guangzhou, 510225, China
E-MAIL: guopfzhku@gmail.com, liuh48@cardiff.ac.uk, hdlsyxlq@126.com, xlqlw@126.com

Abstract:

Dissolved oxygen of aquaculture is an important measure of the quality of culture environment and how aquatic products have been grown. In the machine learning context, the above measure can be achieved by defining a regression problem, which aims at numerical prediction of the dissolved oxygen status. In general, the vast majority of popular machine learning algorithms were designed for undertaking classification tasks. In order to effectively adopt the popular machine learning algorithms for the above-mentioned numerical prediction, in this paper, we propose a two-stage training approach that involves transforming a regression problem into a classification problem and then transforming it back to regression problem. In particular, unsupervised discretization of continuous attributes is adopted at the first stage to transform the target (numeric) attribute into a discrete (nominal) one with several intervals, such that popular machine learning algorithms can be used to predict the interval to which an instance belongs in the setting of a classification task. Furthermore, based on the classification result at the first stage, some of the instances within the predicted interval are selected for training at the second stage towards numerical prediction of the target attribute value of each instance. An experimental study is conducted to investigate in general the effectiveness of the popular learning algorithms in the numerical prediction task and also analyze how the increase of the number of training instances (selected at the second training stage) can impact on the final prediction performance. The results show that the adoption of decision tree learning and neural networks lead to better and more stable performance than Naive Bayes, K Nearest Neighbours and Support Vector Machine.

Keywords:

Machine learning; Regression; Dissolved oxygen;

1. Introduction

Dissolved oxygen of aquaculture is an important index that reflects the quality of culture environment and growth status of aquatic product [1]. Dissolved oxygen content control and management is a crucial section to obtain the high quantity and quality of final product in aquaculture [2]. However, the content of dissolved oxygen is susceptible impacted by multi-factors such as temperature, wind speed, wind direction, rainfall, aquatic metabolism and human activity. This property makes the system of forecasting the content of dissolved oxygen which expresses complex, dynamic, nonlinear, non-stationary and large time delay [3]. It is an interesting project to model this complex structure system and make it interpretable. Some numerical or physics methods have been adopted to study the water quality. However, it is difficult to develop sufficiently accurate models by using traditional statistical methods for the forecasting dissolved oxygen content because of the natural noise of data, missing background information, incomplete monitoring data, inaccurate initial conditions, and limited spatial resolution [4]. Recently, the drawback of the forecasting dissolved oxygen content is overcome by using artificial intelligence (AI) techniques as efficient tools for imitating complex nonlinear systems and detecting the "bad" and outlier information of data stream. As a black box model system, although the AI method usually do not consider internal operation mechanisms, they can also build a relationship between the input features and the output response to predict

the water quality. Hatzikos etal. reported in [5] the forecasting of seawater quality indicators, such as dissolved oxygen, pH, water temperature and turbidity by using neural networks with nonlinear active neurons, but the model has a large number of parameters which make it hard to obtain a stable solution [5].

Faruk D.Ö. utilized a hybrid model which is combined by artificial neural networks (ANN) and auto-regressive integrated moving average (ARIMA) to predict time series of water quality data. However, the drawback of this model is the lack of mentioning how to set the layers and the neurons of each layer in ANN which makes the model difficult to get extended for other data sets [6]. Han et al. proposed in [7] a flexible structure radial basis function (RBF) neural network whose hidden neurons can be added or removed online through taking neuron activity and mutual information (MI). The experimental results obtained using this model showed effectiveness for water quality prediction [7]. Liu et al. presented the least squares support vector regression (LSSVR) model whose parameters are optimized by using the ant colony algorithm (ACA) for dissolved oxygen content prediction. This model partially solved the local minimum problem [3]. Liu et al. analyzed the dissolved oxygen content in river crab culture by a hybrid least squares support vector regression (LSSVR) model with optimal parameters selected by improved particle swarm optimization (IPSO) algorithm. The use of this model led to advances in the prediction accuracy compared with using the standard support vector regression (SVR) and BP network [8]. Chen et.al proposed a hybrid model combined by principal component analysis(PCA) and the long short-term memory (LSTM) neural network to forecast the dissolved oxygen content for aquaculture. This model showed better prediction performance than BPNN, PSO-BP, ELM and LSSVM [9].

In this paper, we propose a two-stage training approach, which involves transforming a regression problem into a classification problem and then transforming it back to a regression problem. In order to achieve the above proposed approach, the original data set needs to be manipulated by discretizing the continuous (numeric) target attribute, such that popular machine learning algorithms can be adopted to train classifiers on the manipulated data set. Furthermore, the classification outcome for each test instance is taken further to predict the numeric value of the target attribute of the instance through instance-based learning algorithms, such as K Nearest Neighbours (KNN) for regression.

The rest of this paper is organized as follows: Section 2 provides a review of related work on regression methods. In Section 3, we describe the procedure of the proposed two-stage training approach in details. In Section 4, we provide details on the study data and describe the experimental setup and results. This paper is concluded in Section 5 by highlighting the contributions and suggesting further directions.

2. Related work

Time series prediction is a special type of regression tasks. However, classification plays the dominant role in the field of machine learning. Since most of the widely available and efficient methods deal with classification, it is an important and interesting direction to extend the methods of classification to handle regression problems. The idea of transforming a regression problem into a classification problem was originally proposed by Weiss S. & Indurkhya N. with the rule-based regression system [10, 11]. They adopted the P-class algorithm to discretize the continuous target attribute into class intervals as a part of a learning system. Their experiments showed that highly accurate prediction results were obtained by converting a regression problem into a classification problem. Torgo L. et al. followed this direction to present three methods of discretizing the continuous target attribute for regression by classification: Equally probable intervals (EP), Equal width intervals (EW) and K-means clustering (KM) [12]. Based on the above discretisation methods, Torgo L. et al. reported a flexible wrapper approach for the selection of class intervals in their paper [13]. Bibi S. et al applied the regression via classification method on software defect estimation which obtained lower regression error than the standard regression approaches on both real-world data sets [14]. Janssen F. and Fürnkranz J. presented a rule-based learning algorithm by dynamically defining a region around the target value predicted by the rule to discretize the target continuous attribute into class intervals leading to a high improvement of the prediction accuracy [15]. Ahmad A. proposed an ensemble model by Extreme Randomized Discretisation (ERD) for discretizing the target continuous attribute into class intervals. The authors proved that the ensembles for regression via classification performed better than regression via classification with the equal width discretisation method [16].

So far, in the regression via classification problem, a regression problem was solved by converting it into a classification problem. Then, the problem can be solved by training any classifiers for classifying an instance to one of the dicretized target intervals. This regression via classification method is comprised by two important stages:

(1) At the training stage, various discretisation methods are adopted to obtain target class intervals in order to learn a clas-

sification model.

(2) At the testing (prediction) stage, the mean of the numeric target attribute values of the training instances in the output class interval is obtained as the predicted numeric value of the test instance.

3. Proposed two-stage training approach for classification driven regression

In this section, we propose a two-stage approach for classification-driven regression. In order to enable the use of the proposed approach, the original data set D that involves a continuous (numeric) target attribute needs to be manipulated by discretizing the target attribute, i.e., the numeric values of the target attribute need to be put into several intervals and each interval represents a class, such that a regression problem is transformed into a classification problem.

Following the discretization of the target attribute, a classifier h can be trained on the manipulated data set D' at the first stage of training and the trained classifier h is then used to classify each test instance x_t into one (c_k) of the intervals obtained through the previous discretization of the target attribute. Based on the classification outcome ($x_t \in c_k$), all the training instances in D' that belong to the interval c_k are recorded. Through using the IDs of the recorded training instances, we can retrieve the same instances (with numeric target attribute values) in D, which are selected to form a new training set D_1 at the second stage of training for regression.

Based on the training instances in D_1, the KNN algorithm is adopted to predict the numeric value of the target attribute of each test instance x_t, i.e., the k training instances that are closest to the test instance x_t are selected as the nearest neighbours and the numeric values of the target attribute of the nearest neighbours are averaged to obtain the predicted numeric value of the test instance x_t.

The whole procedure of the proposed two-stage approach for classification-driven regression is illustrated in Algorithm 1, which generally enables it to adopt any popular machine learning algorithms for classification-driven regression tasks. However, it is crucial that a suitable learning algorithm is employed for training a classifier to effectively classify each test instance into an interval, such that it would be more likely to reduce the error in numerical prediction of the value of the target attribute. In general, if the discretization of the numeric target attribute is done effectively, it would be more likely to achieve more accurate classification of each instance. While each of the intervals is reasonably small, the likelihood of reducing the error of the numeric prediction of the target attribute value of each instance would be higher, as long as the classification at the first step is done accurately. In Section 4, we will investigate the effectiveness of various machine learning algorithms employed for undertaking classification at the first stage and analyze how the classification results obtained at the first step can impact the performance of numeric prediction at the second stage.

Algorithm 1: Procedure of two-stage training

Input : a training set D_l, a test set D_p
Output: a set of predicted numeric values NV

Initialize: Manipulating D_l and D_p into D'_l and D'_p, respectively, by discretizing the target attribute A_t;
train classifier h on D'_l;
for *each test instance* x'_p *in* D'_p **do**
  classify x'_p into an interval c_k by using h;
  for *each training instance* x'_l *in* D'_l **do**
    if $x'_l \in c_k$ **then**
      retrieve x_l from D_l;
      add x_l into the new training set D_{l2} used at the second stage of training for regression;
    end
  end
  for *each training instance* x_{l2} *in* D_{l2} **do**
    calculate the distance between x_{l2} and x_p (the current test instance in D_p;
  end
  find the k training instances that are closest to x_p;
  add the k training instances into the set D_{nn} of nearest neighbours;
  for *each nearest neighbour* x_{nn} *in* D_{nn} **do**
    add the numeric value nv_{tnn} of the target attribute A_t of x_{nn} into the summed value sum_t;
  end
  predict the numeric value nv_{tp} by averaging the numeric values of the target attribute A_t of the k nearest neighbours, i.e., $nv_{tp} = \frac{sum_t}{k}$;
end

4. Experimental study

In this section, we report an experimental study to investigate the effectiveness of classification-driven regression using various machine learning algorithms and different settings of the K value of the KNN algorithm. In particular, the details on data collection and preparation are provided in Section 4.1 and the experimental setup and results are presented in Section 4.2.

4.1. Study area data source

The raw data used in this study was acquired by the Digital Wireless Monitoring System of Aquaculture Water Quality. The DWM system is installed at the zone of technology application and demonstration at the Yixing base of intelligent aquaculture management systems of China Agricultural University in Jiangsu province, China. Four towns with 120 river crab farms, containing approximately 700 ha of river crab ponds, were selected to be monitored. Each experimental pond was approximately 3.44 ha, and the average water level was approximately 12 m. In order to forecast the dissolved oxygen content, we choose the data of dissolved oxygen content and three key relative factors of it: Water temperature, PH value and Dissolved oxygen saturation.

4.2. Experimental setup and results

In this experimental study, we adopt the unsupervised method of discretization of continuous attributes [17] for obtaining several intervals for the target attribute and putting the numeric value of the target attribute of each instance into one of the intervals. In this way, the original data set has been manipulated to enable the use of popular machine learning algorithms for training classifiers. In particular, we adopt five popular machine learning algorithms, namely, C4.5, Naive Bayes (NB), KNN and Multi-layer Perceptron (MLP), for training classifiers on the manipulated data set. Each test instance is classified to one of the intervals at the first stage and then the numeric target attribute value of the test instance will be predicted by averaging the values of the target attribute of the training instances that are closest to the test instance in the interval predicted at the first stage. In other words, the KNN algorithm is adopted for numeric prediction of the target attribute value of each test instance, while inside the interval to which the test instance classified at the first stage, K training instances that are closest to the test instance are selected as the nearest neighbours by using the Euclidean distance function. The average of the target attribute values of the K nearest neighbours is taken as the predicted value of the target attribute of the test instance.

The experiments are conducted using hold-out testing, where 70% of the instances in the data set are randomly selected for training and the rest (30%) of the data set is used for testing. The random partitioning of the data set is repeated 10 times. The average accuracy obtained over the 10 runs is taken for comparison of different learning algorithms in terms of their effectiveness in classifying test instances to specific intervals at the first stage. Similarly, the average root mean squared error (RMSE) obtained over the 10 runs is taken to analyze how the classification results obtained at the first stage using each specific learning algorithm impact the regression performance at the second stage.

In terms of parameters setting of the learning algorithms, C4.5 is adopted to train unpruned decision trees, i.e., the trained decision trees are not simplified using a pruning algorithm. For the KNN algorithm, the value of K is set to 1, and the Euclidean distance function is used to measure the distance between the test instance and each of training instances, where the nearest neighbours are equally weighted for instance-based reasoning. SVM classifiers are trained by using the polynomial kernel alongside a multinomial logistic regression model with a ridge estimator. The other parameters are set as follows: batch size= 100; c=1.0; epsilon= 1.0E-12; tolerance parameter= 0.001. The parameters of MLP are set as follows: number of hidden layers= (number of attributes + number of classes)/2; learning rate= 0.3; momentum= 0.2; number of epochs= 500; batch size= 100. At the second step for regression, the value of k for KNN is set from 1 to 15 to investigate the impacts of different values on the performance of regression.

TABLE 1. Classification accuracy at the first step

K	C4.5	NB	KNN	SVM	MLP
1	0.933	0.862	0.869	0.815	0.948
2	0.937	0.860	0.866	0.828	0.940
3	0.937	0.862	0.867	0.833	0.943
4	0.934	0.863	0.862	0.829	0.942
5	0.937	0.863	0.869	0.822	0.943
6	0.940	0.868	0.863	0.826	0.941
7	0.931	0.865	0.865	0.824	0.943
8	0.931	0.864	0.868	0.826	0.948
9	0.934	0.863	0.865	0.824	0.947
10	0.931	0.861	0.864	0.827	0.943
11	0.937	0.863	0.857	0.822	0.945
12	0.934	0.861	0.862	0.828	0.946
13	0.936	0.864	0.869	0.836	0.943
14	0.936	0.865	0.863	0.831	0.939
15	0.934	0.865	0.865	0.820	0.943

The results on classification accuracy obtained at the first stage by using various learning algorithms are shown in Table 1. The random partitioning of the data set over 10 runs is done independently, while each of the 15 values of the parameter k is set. Therefore, the classification accuracy obtained using each learning algorithm shows very small variation, while different values of k are set. However, this may be useful to see if the small variation due to random data partitioning has a real

impact on the regression performance.

It can be seen from Table 1 that the adoption of C4.5 and MLP generally leads to better classification performance than the use of NB, KNN and SVM, no matter which one of the 15 values of k is selected. The results also show that the performance of MLP looks marginally better than the one of C4.5, but the adoption of MLP usually results in higher computational complexity and worse model interpretability than the use of C4.5. Therefore, the C4.5 algorithm is more recommended for such a classification-driven regression task to interpret how the classification outcome obtained at the first stage for each test instance can be effectively used for regression (i.e., predicting the numeric value of the target attribute of the test instance).

TABLE 2. Root mean squared error at the second step

K	C4.5	NB	KNN	SVM	MLP
1	0.071	0.081	0.079	0.105	0.056
2	0.073	0.081	0.080	0.101	0.061
3	0.068	0.079	0.084	0.104	0.055
4	0.071	0.081	0.087	0.108	0.058
5	0.071	0.084	0.083	0.105	0.055
6	0.074	0.086	0.088	0.111	0.061
7	0.072	0.089	0.093	0.106	0.062
8	0.070	0.089	0.092	0.110	0.060
9	0.070	0.087	0.091	0.113	0.064
10	0.078	0.089	0.091	0.118	0.060
11	0.074	0.087	0.096	0.115	0.066
12	0.078	0.091	0.096	0.116	0.060
13	0.077	0.090	0.093	0.115	0.066
14	0.075	0.086	0.094	0.114	0.063
15	0.085	0.088	0.087	0.121	0.068

The results on RMSE obtained at the second stage by using KNN with different values of K are shown in Table 2. In general, the phenomenon on regression results is consistent to the one on classification results, while different algorithms are compared. In other words, the regression results show again that the performance of C4.5 and MLP is better than the one of NB, KNN and SVM and the performance of C4.5 is slightly worse than the one of MLP.

In terms of the impacts of different K values on the regression performance, the regression results show somewhat the tendency that the RMSE rate is gradually increased as the increase of the K value. For example, the highest RMSE rate is obtained while $k = 15$, for C4.5, SVM and MLP. For both NB and KNN, the highest RMSE rate is obtained while $k = 12$. Moreover, the average RMSE obtained while $k \in [11, 15]$ is consistently higher the average RMSE obtained while $k \in [1, 10]$. For the five learning algorithms, while the 15 values of k are set, the performance of SVM shows the highest variation, i.e., the difference between the highest and lowest RMSE rates is 0.020, whereas the performance of NB shows the lowest variation, i.e., the difference between the highest and lowest RMSE rates is 0.012.

Overall, the results shown in Tables 1 and 2 indicate that the proposed two-stage training approach shows its effectiveness in undertaking classification-driven regression, while the classification results obtained at the first stage really make an impact on the regression results obtained at the second stage. In other words, it is crucial to ensure that a suitable learning algorithm is employed for training a classifier on the manipulated training set that involves the discretized target attribute to achieve more effective classification of each test instance, such that it would be more likely to reduce the error of numeric prediction of the target attribute value of the test instance. Moreover, the discretization of the numeric target attribute also plays an important role in increasing the effectiveness of the proposed approach in both classification and regression tasks.

5. Conclusions

In this paper, we have proposed a two-stage training approach. At the first stage, the numeric target attribute is discretized into a nominal one with several intervals obtained. In this way, a classification task can be undertaken by adopting popular machine learning algorithms to predict the interval to which a new instance belongs. Furthermore, within the interval to which the new instance is classified at the first stage, K nearest neighbours are selected for training at the second stage to predict the target attribute value of each test instance.

We have also conducted an experimental study to investigate in general how effective the popular learning algorithms can be applied in the numerical prediction task and also analyze how the increase of the number of selected neighbours can impact on the final prediction performance. The results show that the adoption of the C4.5 and MLP algorithms leads to better and more stable performance than NB, KNN and SVM, while different values of the parameter K are set.

In future, we will investigate granular computing techniques [18] to achieve the numeric prediction through multi-level division of the interval to which a new instance is classified, such that the selection of K nearest neighbours as the secondary training set can be done in more depth. We also explore the use of optimization techniques for determining the value of K for KNN in the setting of instance-based regression.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant 61871475, in part by Foundation for High-level Talents in Higher Education of Guangdong Province under Grant 2017GCZX001, 2017KTSCX094, 2017KTSCX095, in part by the general program of technology project of Guangzhou 201707010221.

References

[1] J. Q. Wang, X. d. Zhang, M. Nie, C.Z. Fu, J.K. Chen, B. Li, Exotic Spartina alterniflora provides compatible habitats for native estuarine crab Sesarma dehaani in the Yangtze River estuary, Ecological Engineering, Volume 34, pp. 5764, 2008.

[2] O. Zmora, A. Findiesen, J. Stubblefield, V. Frenkel, Y. Zohar, Large-scale juvenile production of the blue crab, Aquaculture, Volume 244, pp. 129139, 2005.

[3] S. Y. Liu., L. Q. Xu, D.L Li., L. H. Zeng, Dissolved oxygen prediction model of eriocheir sinensis culture based on least squares support vector regression optimized by ant colony algorithm, Trans. Chin. Soc. Agric. Eng., Volume 28, pp. 167175, 2012.

[4] O. Kisi and M. Cimen, A wavelet-support vector machine conjunction model for monthly stream flow forecasting, J. Hydrol., Volume 399, pp. 132140, 2011.

[5] E. Hatzikos, L. Anastasakis, N. Bassiliades, I. Vlahavas, Simultaneous prediction of multiple chemical parameters of river water quality with tide, Proceedings of the 2nd International Scientific Conference on Computer Science, IEEE Computer Society, Bulgarian Section, 2005.

[6] D.Ö. Faruk, A hybrid neural network and ARIMA model for water quality time series prediction, Eng. Appl. Artif. Intell, Volume 23, pp. 586594, 2010.

[7] H. G. Han, Q. L. Chen, J. F. Qiao, An efficient self-organizing RBF neural network for water quality prediction, Neural Networks, Volume 24, pp. 717-725, 2011.

[8] S. Y. Liu , L. Q. Xu, D. L. Li, Q. C. Li, Y. Jiang, H. J. Tai, L. H. Zeng, Prediction of dissolved oxygen content in river crab culture based on least squares support vector regression optimized by improved particle swarm optimization, Computers and Electronics in Agriculture, Volume 95, pp. 82-91, 2013.

[9] Y. Y. Chen, Q. Q. Cheng, X. M. Fang, H. H. Yu, D. L. Li, Principal component analysis and long short-term memory neural network for predicting dissolved oxygen in water for aquaculture, Trans. Chin. Soc. Agric. Eng., Volume 34, pp. 183191, 2018.

[10] S. Weiss and N. Indurkhya, Rule-base Regression, Proceedings of the 13th International Joint Conference on Artificial Intelligence, pp. 1072-1078, 1993.

[11] S. Weiss and N. Indurkhya, Rule-based Machine Learning Methods for Functional Prediction, Journal Of Artificial Intelligence Research, volume 3, pp. 383-403, 1995.

[12] L. Torgo and J. Gama, Regression by classification, Borges D.L., Kaestner C.A.A. (eds) Advances in Artificial Intelligence. SBIA 1996. Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence), Springer, Berlin, Heidelberg, 1996.

[13] L. Torgo and J. Gama, Regression using classification algorithms, Intelligent Data Analysis, Volume 1, pp. 275-292, 1997.

[14] S. Bibi, G. Tsoumakas, I. Stamelos, I. Vlahavas, Regression via Classification applied on software defect estimation, Expert Systems with Applications, Volume 34, pp. 2091-210, 2008.

[15] F. Janssen and J. Fürnkranz, Heuristic rule-based regression via dynamic reduction to classification, Proceedings of the Twenty-Second international joint conference on Artificial Intelligence, pp. 1330-1335, 2011.

[16] A. Ahmad, S. Halawani, I. Albidewi, Novel ensemble methods for regression via classification problems, Expert Systems with Applications, Volume 39, pp. 63966401, 2012.

[17] Y. Yang and G. I. Webb, "Proportional k-interval discretization for naive-bayes classifiers", In: 12th European Conference on Machine Learning, vol. 2167, pp. 564-575, 2001.

[18] H. Liu and M. Cocea, Granular Computing Based Machine Learning: A Big Data Processing Approach, Studies in Big Data, vol. 35, Springer, 2018.