ISSN 1064-5624, Doklady Mathematics, 2021. © The Authors(s), 2021. This article is an open access publication.
Russian Text © The Author(s), 2021, published in Doklady Rossiiskoi Akademii Nauk. Matematika, Informatika, Protsessy Upravleniya, 2021, Vol. 499, pp. 63–66.

COMPUTER SCIENCE

Two-Level Regression Method Using Ensembles of Trees with Optimal Divergence

Academician of the RAS **Yu. I. Zhuravlev**[a,*], **O. V. Senko**[a,**], **A. A. Dokukin**[a,***], **N. N. Kiselyova**[b,****], and **I. A. Saenko**[c,*****]

Received June 17, 2021; revised June 17, 2021; accepted June 19, 2021

Abstract—The article discusses a new two-level regression analysis method in which a corrective procedure is applied to optimal ensembles of regression trees. Optimization is carried out based on the simultaneous achievement of the divergence of the algorithms in the forecast space and a good approximation of the data by individual algorithms of the ensemble. Simple averaging, random regression forest, and gradient boosting are used as corrective procedures. Experiments are presented comparing the proposed method with the standard decision forest and the standard gradient boosting method for decision trees.

Keywords: regression, collective methods, bagging, gradient boosting

DOI: 10.1134/S1064562421040177

INTRODUCTION

Regression modeling methods based on computing more accurate collective forecasts from predictions made by a set (ensemble) of less accurate and simpler original algorithms have been widely used in modern machine learning. These methods include random regression forest and methods based on adaptive or gradient boosting. An important role in the construction of collective algorithms is played by the method of obtaining the original ensemble of so-called weak algorithms. A theoretical analysis shows that the generalization ability can be improved by using an ensemble of algorithms that not only have high accuracy, but also produce maximally diverging forecasts [1]. The low correlation between forecasts potentially makes it possible to achieve a more accurate algorithmic approximation, which objectively ensures the most accurate forecast with the use of a bounded number of algebraic operations [2, 3]. In the random regression forest method, the divergence of forecasts is achieved by training the algorithms of the ensemble on different samples generated from the original training sample with the use of bootstrap [4]. In the gradient boosting method [5], an ensemble is generated sequentially. At every iteration step, the ensemble is supplemented with trees approximating the first derivatives of the loss function with respect to the variables corresponding to collective forecast.

Another important component is the method used to compute collective forecast, which can also be interpreted as a result of mutual correction of forecasts. In the random regression forest method, correction is carried out by simple computation of average forecasts.

Another possible method for organizing a corrective procedure is the stacking scheme, in which the outputs of the ensemble algorithms are treated as input features of an algorithm computing an output corrected forecast [6, 7]. As a rule, the efficiency of stacking is low as applied to the computation of collective decisions from sets of weak algorithms produced by random forest generation procedures. It can be assumed that the low efficiency is caused by the insufficient divergence of the weak algorithms in the forecast space in the case of standard ensemble generation methods.

The goal of this paper is to study the efficiency of a two-level method for improving the generalization ability, which involves the construction of an ensemble consisting of algorithms characterized by high-degree divergence in the forecast space and a good approximation of the target variable. Simple averaging and stacking are used as a corrective procedure.

[a] *Federal Research Center "Computer Science and Control," Russian Academy of Sciences, Moscow, 119333 Russia*
[b] *Baikov Institute of Metallurgy and Materials Science, Russian Academy of Sciences, Moscow, 119334 Russia*
[c] *Faculty of Computational Mathematics and Cybernetics, Lomonosov Moscow State University, Moscow, 119991 Russia*
**e-mail: zhur@ccas.ru*
***e-mail: senkoov@mail.ru*
****e-mail: dalex@ccas.ru*
*****e-mail: kis@imet.ac.ru*
******e-mail: i.a.saenko@mail.ru*

TWO-LEVEL METHOD

Let $\{A_1(X),\ldots,A_{k-1}(X)\}$ be an ensemble of algorithms predicting the value of a variable y from a given vector X of variables $x_1,\ldots,x_n$. It is assumed that the algorithms of the ensemble are trained on a sample $S=\{(X_1,y_1),\ldots,(X_m,y_m)\}$. Preliminarily, a baseline regression analysis method is chosen, which usually represents a regression tree model. Define $L_k(X)=\frac{1}{k}\sum_{i=1}^{k}A_i(X)$ and $Q_k(X)=\frac{1}{k}\sum_{i=1}^{k}A_i^2(X)$. According to the declared goal, an ensemble is constructed by simultaneously minimizing the criterion

$$\Phi_E(A_1(X),\ldots,A_k(X))=\frac{1}{mk}\sum_{i=1}^{k}\sum_{j=1}^{m}\left(y_j-A_i(X_j)\right)^2,$$

which estimates the average approximation error of y by the vector X, and maximizing the criterion

$$\Phi_V(A_1(X),\ldots,A_k(X))=\frac{1}{mk}\sum_{i=1}^{k}\sum_{j=1}^{m}\left(L_k(X_j)-A_i(X_j)\right)^2,$$

which is the variance of the forecasts produced by the algorithms of the ensemble.

The problem of simultaneously minimizing Φ_E and maximizing Φ_V can be reduced to the minimization of

$$\Phi_G=(1-\mu)\Phi_E-\mu\Phi_V,$$

where $\mu\in[0,1]$ determines the contribution of the heterogeneity of the ensemble in the terms of the forecast variance.

Let D_E^k and D_V^k denote the variations in the functionals Φ_E and Φ_V occurring when the ensemble is supplemented with an algorithm A_{k+1}.

$$\begin{aligned}D_E^k&=\Phi_E(A_1(X),\ldots,A_{k+1}(X))-\Phi_E(A_1(X),\ldots,A_k(X))\\&=\left(\Phi_E(A_1(X),\ldots,A_k(X))*k\right.\\&\quad\left.+\frac{1}{m}\sum_{j=1}^{m}(y_j-A_{k+1}(X_j))^2\right)\frac{1}{k+1}\\&\quad-\Phi_E(A_1(X),\ldots,A_k(X))=\frac{1}{m(k+1)}\\&\quad\times\sum_{j=1}^{m}\left(y_j-A_{k+1}(X_j)\right)^2-\frac{1}{k+1}\Phi_E(A_1(X),\ldots,A_k(X))\\&=\frac{1}{m(k+1)}\sum_{j=1}^{m}\left(y_j-A_{k+1}(X_j)\right)^2-C_E,\end{aligned}$$

where C_E does not depend on$A_{k+1}(X)$.

To compute D_V, we use the well-known variance expression

$$\sum_{j=1}^{m}\sum_{i=1}^{k}\left(L_k(X_j)-A_i(X_j)\right)^2=\sum_{j=1}^{m}(Q_k(X_j)-L_k^2(X_j));$$

$$\begin{aligned}D_V^k&=\Phi_V(A_1(X),\ldots,A_{k+1}(X))-\Phi_V(A_1(X),\ldots,A_k(X))\\&=\frac{1}{m}\sum_{j=1}^{m}(Q_{k+1}(X_j)-L_{k+1}^2(X_j))\\&\quad-\frac{1}{m}\sum_{j=1}^{m}\left(Q_k(X_j)-L_k^2(X_j)\right)=\frac{1}{m(k+1)}\\&\quad\times\sum_{j=1}^{m}\left(-Q_k(X_j)+A_{k+1}^2(X_j)\right.\\&\quad\left.-\frac{1}{k+1}\left(kL_k(X_j)+A_{k+1}(X_j)\right)^2+(k+1)L_k^2(X_j)\right)\\&=\frac{k}{m(k+1)^2}\sum_{j=1}^{m}(A_{k+1}^2(X_j)-2L_k(X_j)A_{k+1}(X_j))+C_V,\end{aligned}$$

where C_V does not depend on $A_{k+1}(X)$.

An approach for minimizing Φ_G is to reduce this problem to the search for an algorithm A_{k+1} (added to the ensemble) that minimizes the functional

$$\begin{aligned}D_G^k&=(1-\mu)D_E^k-\mu D_V^k\\&=\frac{1-\mu}{m(k+1)}\sum_{j=1}^{m}\left(y_j-A_{k+1}(X_j)\right)^2\\&\quad-\frac{\mu k}{m(k+1)^2}\sum_{j=1}^{m}(A_{k+1}^2(X_j)-2L_k(X_j)A_{k+1}(X_j))+C_G,\end{aligned}$$

where C_G does not depend on $A_{k+1}(X)$. An algorithm to be added to the ensemble at the step $k+1$ is constructed by combining bagging and a gradient boosting procedure and consists of the following two stages:

1. At the first stage, a random number generator is applied to the original sample S to produce a sample with replacement S_{k+1}^0, which is then used to train the algorithm A_{k+1}^0.

2. At the second stage, an algorithm A_{k+1} to be added to the ensemble is constructed. The algorithm A_{k+1} computes the forecast of y at the point X using the formula

$$A_{k+1}(X)=A_{k+1}^0(X)-\varepsilon G_{k+1}(X),$$

where $G_{k+1}(X)$ is an algorithm predicting the gradient of the functional $D_G^k(A_k(X_1),\ldots,A_{k+1}(X_m))$ at the point $(A_{k+1}^0(X_1),\ldots,A_k^0(X_m))$.

The algorithm G_{k+1} is trained on the sample $\left\{\left(X_1,\left.\frac{\partial D_G^k}{\partial A_{k+1}(X_1)}\right|_{A_{k+1}^0(X_1)}\right),\ldots,\left(X_m,\left.\frac{\partial D_G^k}{\partial A_{k+1}(X_m)}\right|_{A_{k+1}^0(X_m)}\right)\right\}$.

Table 1. Experimental results

	Reference	Forest	Boosting	Average
$I2/m$, a	0.943 ± 0.0012	0.948 ± 0.0031	0.941 ± 0.005	0.94 ± 0.0025
$I2/m$, b	0.692 ± 0.0087	0.78 ± 0.0038	0.746 ± 0.012	0.758 ± 0.0088
$I2/m$, c	0.847 ± 0.004	0.875 ± 0.0052	0.856 ± 0.0051	0.87 ± 0.0063
$I4/m$, c	0.99 ± 0.0015	0.993 ± 0.0006	0.993 ± 0.0007	0.994 ± 0.0003
$Fm3(-)m$, a	0.636 ± 0.0019	0.639 ± 0.0026	0.615 ± 0.0034	0.631 ± 0.0041
$P2_1/n$, a	0.593 ± 0.0075	0.575 ± 0.005	0.575 ± 0.0052	0.572 ± 0.0041
$P2_1/n$, b	0.928 ± 0.0004	0.903 ± 0.0019	0.895 ± 0.0025	0.906 ± 0.0009
$P2_1/n$, c	0.441 ± 0.0047	0.516 ± 0.0103	0.439 ± 0.013	0.54 ± 0.008
$R3(-)$, c	0.319 ± 0.0146	0.343 ± 0.013	0.364 ± 0.0178	0.346 ± 0.0134
A_3BHal_6, Tm	0.903 ± 0.0001	0.893 ± 0.0015	0.89 ± 0.0014	0.895 ± 0.0009
$ABHal_3$, Tm	0.861 ± 0.02	0.874 ± 0.022	0.881 ± 0.021	0.871 ± 0.021

It is easy to show that

$$\frac{\partial D_G^k}{\partial A_{k+1}(X_i)} = \frac{-2(1-\mu)}{m(k+1)}(y_i - A_{k+1}(X_i)) - \frac{\mu 2k}{m(k+1)^2}(A_{k+1}(X_i) - L_k(X_i)).$$

EXPERIMENTS

The method was implemented using the Python language with the help of the scikit-learn library [8]. The baseline trees were constructed by applying the BaggingRegressor method. At the second level, we used GradientBoostingRegressor (which is referred to hereafter as boosting) or RandomForestRegressor (forest) or the results of the baseline methods were averaged (average). GradientBoostingRegressor was also used as a reference technique for estimating the efficiency of the proposed method.

The developed method was used to predict the parameters of the crystal lattice of the complex inorganic compound $A_2^{2+}B^{+3}C^{+5}O_6$ and the melting points of the halides A_3BHal_6 and $ABHal_3$. For various space symmetry groups, we present results for some of the parameters admitting sufficiently reliable forecast, namely: the parameters a, b, and c for monoclinic space groups ($I2/m$ and $P2_1/n$); the parameter c for tetragonal ($I4/m$) and hexagonal ($R3(-)$) groups; and the parameter a for the cubic group $Fm3(-)m$. The accuracy was evaluated using the standard characteristic r^2, or the determination coefficient, which was computed by applying cross validation. Since algorithms in the bagging procedure are generated to a large degree at random, the solution results for the same problem change from experiment to experiment. Accordingly, for each problem, Table 1 presents the value of r^2 averaged over 10 experiments.

CONCLUSIONS

The results given in Table 1 show that, in most cases, the proposed two-level method yields better results than the standard gradient boosting algorithm, which prompts further research in this direction. We intend to explore the possibility of choosing an optimal gradient descent step size (in terms of certain criteria) in correcting bagging-generated algorithms.

FUNDING

This work was supported in part by the Russian Foundation for Basic Research, project nos. 18-29-03151, 20-01-00609, and 21-51-53019.

REFERENCES

1. A. A. Dokukin and O. V. Senko, Comput. Math. Math. Phys. **55** (3), 526–539 (2015).
2. Yu. I. Zhuravlev, Kibernetika, No. 4, 14–21 (1977).
3. Yu. I. Zhuravlev, Kibernetika, No. 6, 21–27 (1977).
4. L. Breiman, Mach. Learn. **24**, 123–140 (1996).
5. T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. (Springer, New York, 2009), pp. 337–384.
6. D. H. Wolpert, Neuron Networks **5** (2), 241–259 (1992).
7. L. Breiman, Mach. Learn. **24**, 49–64 (1996).
8. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, J. Mach. Learn. Res. **12**, 2825–2830 (2011).

Translated by I. Ruzanova