Bagged Averaging of Regression Models

S. B. Kotsiantis, D. Kanellopoulos, I. D. Zaharakis
Educational Software Development Laboratory
Department of Mathematics
University of Patras, Greece
sotos@math.upatras.gr, dkanellop@teipat.gr, jzaharak@cti.gr

Abstract. Linear regression and regression tree models are among the most known regression models used in the machine learning community and recently many researchers have examined their sufficiency in ensembles. Although many methods of ensemble design have been proposed, there is as yet no obvious picture of which method is best. One notable successful adoption of ensemble learning is the distributed scenario. In this work, we propose an efficient distributed method that uses different subsets of the same training set with the parallel usage of an averaging methodology that combines linear regression and regression tree models. We performed a comparison of the presented ensemble with other ensembles that use either the linear regression or the regression trees as base learner and the performance of the proposed method was better in most cases.

1 Introduction

Several algorithms have been proposed for the design of ensemble of regression models [4]. Mechanisms that are used to make ensemble of regression models include: i) Using different subset of training data with a single machine learning method, ii) Using different training parameters with a single learning method, iii) Using different machine learning methods.

Even though many algorithms of ensemble creation have been proposed, there is as yet no obvious picture of which method is best. One notable successful adoption of ensemble learning in a distributed scenario is the meta-learning framework. It offers a way to mine regression models from homogeneously distributed data. In this approach, supervised learning techniques are first used to build regression models at local data sites; then meta-level models are generated using the locally learned concepts. This paper explores an efficient method for constructing ensembles that can take place in a distributed way. The idea is simple: use different subsets of the same training set with the parallel usage of an averaging methodology at each site that combines a linear regression model [6] and a regression tree algorithm [10].

Please use the following format when citing this chapter:
Kotsiantis, Sotiris, Kanellopoulos, Dimitris, Zaharakis, Ioannis, 2006, in IFIP International Federation for Information Processing, Volume 204, Artificial Intelligence Applications and Innovations, eds. Maglogiannis, I., Karpouzis, K., Bramer, M., (Boston: Springer), pp. 53–60

Using averaging methodology, we expect to obtain better results because both theory and experiments show that averaging helps most if the errors in the individual regression models are not positively correlated [9]. In fact, the comparison with other ensembles that use either the linear regression or regression tree algorithm on 30 standard benchmark datasets showed that the proposed ensemble had on the average better performance.

Section 2 presents the most well-known methods for building ensembles, while section 3 discusses the proposed ensemble method. Experiment results and comparisons of the presented combining method in a number of datasets with other ensembles that also use as base learner either the regression tree or the linear regression model are presented in section 4. We conclude in Section 5 with summary and further research topics.

2 Ensembles of Regression Models

Bagging [2] is a ``bootstrap" ensemble method that creates individuals for its ensemble by training each regression model on a random redistribution of the training set. Each regression model's training set is generated by randomly drawing, with replacement, N examples - where N is the size of the original set; many of the original examples may be repeated in the resulting training set while others may be left out. After the construction of several regression models, averaging the predictions of each regression model performs the final prediction. Breiman [2] made the important observation that instability (responsiveness to changes in the training data) is a prerequisite for bagging to be effective.

Another method that uses different subset of training data with a single data mining method is the boosting approach [5]. Boosting is similar in overall structure to bagging, except that it keeps track of the performance of the learning algorithm and concentrates on instances that have not been correctly learned. Instead of choosing the t training instances randomly using a uniform distribution, it chooses the training instances in such a manner as to favor the instances that have not been accurately learned. After several cycles, the prediction is performed by taking a weighted average of the predictions of each regression model, with the weights being proportional to each regression model's performance on its training set. Additive Regression is a practical version of the boosting approach [7].

Another approach for building ensembles of regression models is to use a variety of learning algorithms on all of the training data and combine their predictions. When multiple regression models are combined using averaging methodology, we expect to obtain good results based on the belief that the majority of experts are more likely to be correct in their decision when they are close in their opinions [9].

Stacked generalization [3], or Stacking, is a more sophisticated approach for combining predictions of different learning algorithms. Stacking combines multiple regression models to induce a higher-level regression model with improved performance. In detail, the original data set constitutes the level zero data and all the base regression models run at this level. The level one data are the outputs of the base regression models. A learning algorithm is then used to determine how the

outputs of the base regression models should be combined, using as input the level one data.

3 Proposed Methodology

Bagging uses an averaging technique which is unable to take into account the heterogeneity of the instance space. When majority of the base regression models give a wrong prediction for a new instance then the average value will result in a wrong prediction [8]. The problem may consist in discarding base regression models that are highly accurate in a restricted region of the instance space because this accuracy is swamped by their inaccuracy outside the restricted area. It may also consist in the use of regression models that are accurate in most of the space but still unnecessarily confuse the whole committee in some restricted areas of the space. To overcome this problem we have suggested the bagged averaging using two learning algorithms: the linear regression (LR) model and a regression tree (RT) algorithm. There is a reason that makes us believe the one method acts as a complement to the other. Perlich et al. [12] have proved that the corresponding classification models: logistic regression and decision trees act as a complement to each other. The algorithm is briefly described in Fig. 1.

MODEL GENERATION

Let n be the number of instances in the training data.

For each of t iterations (t=10 in our experiments):

- Sample n instances with replacement from training data.
- Built two regression models (LR, RT) from the sample
- Store the resulting models.

APPLICATION PHASE

For each of the t models:

Predict value of instance by averaging the two learning algorithms (LR, RT)

Return the average value of the predicted values.

Fig. 1. The proposed ensemble

As it is well known, Regression Trees produce decision trees with numeric output for leaf nodes rather than categorical output. M5 is one of the most well-known algorithms for regression tree induction [13] and for this reason it was used for our model.

It has been observed that for bagging, an increase in committee size (sub-regression models) usually leads to a decrease in prediction error, but the relative impact of each successive addition to a committee is ever diminishing. Most of the effect of each technique is obtained by the first few committee members [11]. For this reason, we used 10 sub-regression models for the proposed algorithm.

It must be also mentioned that the proposed ensemble is easily distributed and parallelized. The computations required to obtain the regression models in each bootstrap sample are independent of each other. Therefore we can assign tasks to each processor in a balanced manner. By the end each processor has obtained a part

of the Bagged Averaging ensemble. In the case we use the master-slave parallel programming technique, the method starts with the master splitting the work to be done in small tasks and assigning them to each slave (LR and RT regression models). Then the master performs an iteration in which if a slave returns a result (this means it finished its work) then the master assigns it another task if there are still tasks to be executed. Once all the tasks have been carried out the master process obtains the results and orders the slaves to finish since there are not more tasks to be carried out. This parallel and distributed execution of the presented ensemble achieves almost linear speedup.

4 Comparisons and Results

For the comparisons of our study, we used 30 well-known datasets mainly from domains from the UCI repository [1]. These datasets cover many different types of problems having discrete, continuous and symbolic variables.

The most well known measure for the degree of fit for a regression model to a dataset is the correlation coefficient. If the actual target values are $a_1, a_2, \ldots a_n$ and the predicted target values are: $p_1, p_2, \ldots p_n$ then the correlation coefficient is given by the formula:

$$R=\frac{S_{PA}}{\sqrt{S_P S_A}} \text{ where } S_{PA}=\frac{\sum_i (p_i-\bar{p})(a_i-\bar{a})}{n-1},\ S_P=\frac{\sum_i (p_i-\bar{p})^2}{n-1},\ S_A=\frac{\sum_i (a_i-\bar{a})^2}{n-1}.$$

In order to calculate the regression models' correlation coefficient, the whole training set was divided into ten mutually exclusive and equal-sized subsets and for each subset the regression model was trained on the union of all of the other subsets. Then, cross validation was run 10 times for each algorithm and the average value of the 10-cross validations was calculated (10x10 cross-validation). It must be mentioned that we used the free available source code for the most algorithms by [14].

In the following tables, we represent with "v" that the proposed ensemble (Bagged Averaging) looses from the specific ensemble. That is, the specific algorithm performed statistically better than the proposed according to t-test with $p<0.05$. Furthermore, in Tables, "*" indicates that Bagged Averaging performed statistically better than the specific ensemble according to t-test with $p<0.05$. In all the other cases, there is no significant statistical difference between the results (Draws).

In the last rows in all tables one can see the aggregated results in the form (a/b/c). In this notation "a" means that the proposed ensemble is significantly more accurate than the compared algorithm in a out of 30 datasets, "c" means that the proposed ensemble is significantly less accurate than the compared algorithm in c out of 30 datasets, while in the remaining cases (b), there is no significant statistical difference between the results. In the following Tables, we also present the average correlation coefficient of all tested dataset for each ensemble.

For both Bagging and Boosting, much of the reduction in error appears to have occurred after ten to fifteen regression models. But boosting continues to measurably

improve their test-set error until around 25 regression models [11]. For this reason, we used 25 sub-regression models for our experiments. Firstly, we compare the presented methodology with bagging and boosting version of LR (using 25 sub-regression models). Secondly, we compare the presented methodology with bagging and boosting version of M5 (using 25 sub-regression models). In the last rows of the Table 1 one can see the aggregated results.

Table 1. Comparing Bagged Averaging ensemble with bagging and boosting version of LR and M5

	Bagged Averaging	Bagging LR	Bagging M5	Boosting LR	Boosting M5
auto93.names	0.81	0.79*	0.80	0.83v	0.80
autoHorse.names	0.95	0.95	0.89*	0.95	0.91*
autoMpg.names	0.93	0.93	0.91*	0.93	0.91*
autoPrice.names	0.90	0.89*	0.89*	0.89*	0.91v
baskball	0.59	0.61v	0.51*	0.62v	0.44*
bodyfat.names	0.99	0.99	0.97*	0.99	0.97*
breastTumor	0.29	0.29	0.26*	0.30	0.18*
cholesterol	0.20	0.20	0.19	0.19*	0.06*
cleveland	0.71	0.72v	0.66*	0.71	0.63*
cloud	0.92	0.93v	0.85*	0.93v	0.86*
cpu	0.96	0.96	0.89*	0.95	0.92*
echoMonths	0.70	0.70	0.70	0.71v	0.69
elusage	0.89	0.87*	0.85*	0.86*	0.85*
fishcatch	0.96	0.97v	0.91*	0.97v	0.96
housing	0.89	0.85*	0.88*	0.85*	0.89
hungarian	0.69	0.71v	0.63*	0.72v	0.61*
lowbwt	0.79	0.79	0.79	0.79	0.78*
meta	0.44	0.40*	0.43	0.38*	0.25*
pbc	0.60	0.59	0.52*	0.60	0.50*
pollution	0.76	0.75	0.68*	0.76	0.67*
pwLinear	0.89	0.87*	0.86*	0.86*	0.90v
quake	0.07	0.06*	0.07	0.06*	0.01*
sensory	0.45	0.38*	0.48v	0.39*	0.45
servo	0.87	0.85*	0.85*	0.85*	0.84*
sleep	0.66	0.65	0.60*	0.62	0.57*
stock	0.97	0.93 *	0.97	0.93 *	0.99 v
strike	0.52	0.53v	0.49*	0.53v	0.47*
triazines	0.43	0.37 *	0.48	0.38 *	0.44
veteran	0.45	0.46	0.39*	0.48v	0.34*
wisconsin	0.36	0.34 *	0.30 *	0.33 *	0.24 *
W-D-L		*6/12/12*	*1/8/21*	*8/10/12*	*3/6/21*
Average correlation coefficient	*0.69*	*0.68*	*0.66*	*0.68*	*0.63*

The presented ensemble has significantly higher correlation coefficient than bagging LR in 12 out of the 30 datasets, while it has significantly lower correlation coefficient in 6 datasets. At this point, it must be also mentioned that the proposed ensemble and the bagging version of LR with 25 sub-regression models need similar training times (more detailed evaluation in quantitative terms will be presented in a future paper). In addition, the presented ensemble has significantly higher correlation coefficient than boosting LR in 12 out of the 30 datasets, whilst it has significantly lower correlation coefficient in 8 datasets.

Moreover, the presented ensemble has significantly higher correlation coefficient than bagging regression tree algorithm –M5– in 21 out of the 30 datasets, while it has significantly lower correlation coefficient in one dataset. In addition, the presented ensemble has significantly higher correlation coefficient than boosting M5 in 21 out of the 30 datasets whilst it has significantly lower correlation coefficients in 3 datasets.

To sum up, on the average the presented ensemble has higher correlation coefficient than the other well-known ensembles that use only the LR algorithm about 2%. Moreover, on the average the performance of the presented ensemble is more accurate than the other well-known ensembles that use only the M5 algorithm from 5% to 8%. What is more, the presented ensemble needed much less time for training than bagging and boosting version of M5 algorithm (more detailed evaluation in quantitative terms will be presented in a future paper).

Subsequently, we compare the presented methodology with other well-known ensembles that use either LR or M5 as base regression models. We compare the proposed methodology with:

- Stacking methodology [3]. We used LR, M5 as base regression models and LR as meta-level regression model.
- Averaging methodology using LR, M5 as base regression models [9]

In the last rows of the Table 2 one can see the aggregated results. The presented ensemble has significantly higher correlation coefficient than averaging in 12 out of the 30 datasets, whilst it has significantly lower correlation coefficient in 3 datasets. It must be also mentioned that on the average the performance of the presented ensemble is more accurate than averaging about 2%.

Similarly, the proposed ensemble has significantly higher correlation coefficient than Stacking in 10 out of the 30 datasets, while it has significantly lower correlation coefficient in 7 datasets. The average relative correlation coefficient improvement of the proposed ensemble is about 2% better in relation to Stacking.

To sum up, the presented methodology of combining LR and M5 algorithms could be an off-the self method-of-choice for a regression task where there is no a priori knowledge available about the domain and the primary goal is to develop an regression model with lowest possible error.

5 Conclusions

It is known that if we are only concerned for the best possible correlation coefficient, it might be difficult or impossible to find a single regression model that performs as

well as a good ensemble of regression models. In this study, we built an ensemble of regression models using two different learning methods: the Linear Regression and the M5 algorithm.

Table 2. Comparing Bagged Averaging ensemble with Stacking and Averaging ensembles

	Bagged Averaging	Averaging	Stacking
auto93.names	0.81	0.84v	0.83v
autoHorse.names	0.95	0.95	0.95
autoMpg.names	0.93	0.93	0.93
autoPrice.names	0.90	0.90*	0.90*
baskball	0.59	0.59	0.61v
bodyfat.names	0.99	0.98*	0.99
breastTumor	0.29	0.28	0.28
cholesterol	0.20	0.18*	0.15*
cleveland	0.71	0.68*	0.71
cloud	0.92	0.91*	0.93
cpu	0.96	0.95*	0.94*
echoMonths	0.70	0.71v	0.71v
elusage	0.89	0.89	0.87*
fishcatch	0.96	0.96	0.97v
housing	0.89	0.89	0.89
hungarian	0.69	0.67*	0.72v
lowbwt	0.79	0.79	0.79
meta	0.44	0.42*	0.36*
pbc	0.60	0.59*	0.60
pollution	0.76	0.74	0.74
pwLinear	0.89	0.89	0.89
quake	0.07	0.06*	0.04*
sensory	0.45	0.43*	0.42*
servo	0.87	0.87	0.86*
sleep	0.66	0.64	0.61*
stock	0.97	0.97	0.98 v
strike	0.52	0.51*	0.52
triazines	0.43	0.45	0.45
veteran	0.45	0.46v	0.47v
wisconsin	0.36	0.35	0.31 *
W-D-L		*3/15/12*	*7/13/10*
Average correlation coefficient	*0.69*	*0.68*	*0.68*

While ensembles provide very accurate regression models, too many regression models in an ensemble may limit their practical application. To be feasible and competitive, it is important that the learning algorithms run in reasonable time. In our method, we limit the number of sub-regression models to 20. It was proved after a number of comparisons with other ensembles, which use either M5 or LR as base models, that the Bagged Averaging methodology gives better correlation coefficient in most cases. In a future research project we will also examine the product rule for

combining LR and RT. In addition, more experiments based on varying number of sub-regression models are needed for the proposed approach.

Accessing and analyzing data from a ubiquitous computing device offer many challenges. For example, ubiquitous data mining (UDM) introduces additional cost due to communication, computation, security, and other factors. For the proposed method, a learning algorithm can take the form of a software agent in order the proposed model to be used in a ubiquitous environment. Of course, some problems such as agent interaction, cooperation, collaboration, negotiation and organizational behavior should earlier be solved. These are the research topics we are currently working on and hope to report our findings in the near future.

References

1. C.L. Blake, C.J. Merz, UCI Repository of machine learning databases. Irvine, CA: University of California, Department of Information and Computer Science (1998). [http://www.ics.uci.edu/~mlearn/MLRepository.html]
2. L. Breiman, Bagging Predictors. Machine Learning, 24(3) (1996) 123-140.
3. L. Breiman, Stacked Regression. Machine Learning, 24 (1996):49-64.
4. T.G. Dietterich, Ensemble methods in machine learning. In Kittler, J., Roli, F., eds.: Multiple Classifier Systems. LNCS Vol. 1857, Springer (2001) 1–15
5. N. Duffy, D. Helmbold, Boosting Methods for Regression, Machine Learning, 47, (2002) 153–200.
6. J. Fox, Applied Regression Analysis, Linear Models, and Related Methods, ISBN: 080394540X, Sage Pubns (1997).
7. J. Friedman, Stochastic Gradient Boosting, Computational Statistics and Data Analysis 38 (2002) 367-378.
8. Y. Grandvalet, Bagging Equalizes Influence, Machine Learning, Volume 55(3) (2004) 251 – 270.
9. N.L. Hjort, G. Claeskens, Frequentist Model Average Estimators, Journal of the American Statistical Association, 98 (2003) 879-899.
10. Y. Morimoto, H. Ishii, S. Morishita, Efficient Construction of Regression Trees with Range and Region Splitting, Machine Learning, 45(3) (2001) 235-259.
11. D. Opitz, R. Maclin, Popular Ensemble Methods: An Empirical Study, Artificial Intelligence Research, 11 (1999): 169-198, Morgan Kaufmann.
12. C. Perlich, F. J. Provost, J. S. Simonoff, Tree Induction vs. Logistic Regression: A Learning-Curve Analysis. Journal of Machine Learning Research 4 (2003) 211-255
13. Y. Wang, I. H. Witten, Induction of model trees for predicting continuous classes, In Proc. of the Poster Papers of the European Conference on ML, (1997) 128–137.
14. I. Witten, E. Frank, Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations, Morgan Kaufmann, San Mateo (2000).

Appendix: Acknowledgements

The Project is Co-Funded by the European Social Fund & National Resources - EPEAEK II.