Bagging Classification Models with Reduced Bootstrap

Rafael Pino-Mejías[1,2], María-Dolores Cubiles-de-la-Vega[2], Manuel López-Coello[3], Esther-Lydia Silva-Ramírez[3], and María-Dolores Jiménez-Gamero[2]

[1] Centro Andaluz de Prospectiva, Avda. Reina Mercedes, s/n, 41012 Sevilla, Spain
[2] Departamento de Estadística e Investigación Operativa, Facultad de Matemáticas
Universidad de Sevilla, Avda. Reina Mercedes, s/n, 41012 Sevilla, Spain
{rafaelp,cubiles,dolores}@us.es
[3] Departamento de Lenguajes y Sistemas Informáticos, E. Superior de Ingeniería
Universidad de Cádiz, C/ Chile 1, 11002 Cádiz, Spain
{manuel.coello,esther.silva}@uca.es

Abstract. Bagging is an ensemble method proposed to improve the predictive performance of learning algorithms, being specially effective when applied to unstable predictors. It is based on the aggregation of a certain number of prediction models, each one generated from a bootstrap sample of the available training set. We introduce an alternative method for bagging classification models, motivated by the reduced bootstrap methodology, where the generated bootstrap samples are forced to have a number of distinct original observations between two values k_1 and k_2. Five choices for k_1 and k_2 are considered, and the five resulting models are empirically studied and compared with bagging on three real data sets, employing classification trees and neural networks as the base learners. This comparison reveals for this reduced bagging technique a trend to diminish the mean and the variance of the error rate.

1 Introduction

Bagging (Bootstrap Aggregating) is a method proposed by Breiman [1] to improve the performance of prediction models. Given a model, bagging draws B independent bootstrap samples from the available training set, fits a particular model to each bootstrap sample, and finally it aggregates the B models by computing the mean (regression) or majority voting (classification). Bagging is a very effective procedure when applied to unstable learning algorithms such as classification and regression trees and neural networks. The empirical success of the first published works has been confirmed by theoretical results as we can see in [2], where bagging is shown to smooth hard decision problems, yielding smaller variance and mean squared error (MSE). These results have been derived for classification and regression trees, but the variance and MSE reduction effect of bagging is not necessarily true for other models, as it is shown in [3] for U-statistics.

Bagging averages models constructed over nearby empirical distributions corresponding to replacement samples from the training set. However, if we consider other classes of neighborhoods of the empirical distribution of the original sample, or if we

A. Fred et al. (Eds.): SSPR&SPR 2004, LNCS 3138, pp. 966–973, 2004.

vary the method to carry out the aggregating process, a more general bagging is defined. The use of robust location measures, as the median, is an example of the second approach. For the first approach, we could draw samples with or without replacement, and sample sizes not necessarily equal to the training set size would also be considered, as is the case for Subbagging (Subsample Aggregating) in [2].

If we maintain the replacement sampling process, a generalization is motivated by the following reasoning of [4]: bootstrap samples are simple random samples of size n selected with replacement from the original n sized sample, so not all bootstrap samples are equally informative, due to the randomness associated to the number of distinct original observations in the bootstrap sample. The variability of this number is neither necessary nor desirable, having negative effects on the performance of the bootstrap technique in certain applications. For example, the bootstrap does not provide a consistent estimator for the variance of the median, but an alternative bootstrap resampling scheme which solves that inconsistency is presented in [5]. We propose to consider this alternative bootstrap procedure, namely the reduced bootstrap, as the sampling algorithm for bagging. In section 2 we present this new method, while section 3 is devoted to some empirical comparisons with the usual bagging procedure, resuming the main conclusions and the future work in section 4.

2 Reduced Bootstrap

We consider a classification problem where a training set $\mathbf{D}=\{U_i=(X_i,Y_i),\ i=1,\ldots,n\}$ is available. X_i is a realization of a multidimensional predictor variable and Y_i contains the label of the class of the case i, for example an element of $\{1,2,\ldots,K\}$ for a K-class problem. Given a classification model g, depending on a set of parameters to be optimized, bagging was defined in [1] as follows:

Definition 1. Algorithm Bagging
Fix B
For b=1,2,…,B
Draw a bootstrap sample, i.e., a simple random sample with replacement
$\mathbf{D}^* = (U_1^*,\ldots,U_n^*)$ taken from $\mathbf{D}$.
Fit g to $\mathbf{D}^*$, obtaining g_b.
Next b.

The aggregate model g_{agg} is defined by voting of the B computed models:

$$g_{agg}(x) = \arg\max_j f_j(x) \tag{1}$$

$$f_j(x) = \#\{g_b(x) = j\} \tag{2}$$

We must note that the bootstrap procedure inside bagging is really what Efron called in [6] the bootstrap Method II, used to approximate a theoretical distribution by Monte Carlo simulation. However, this simulation process is affected by a series of errors and variabilities, as is formalized in [7]. For this reason, several alternative techniques have been proposed, as those recorded by [4], [8], [9].

In [7] we defined a variation of Efron´s method II based on the outlier bootstrap sample concept, namely OBS, that is based on only considering those bootstrap samples having a number of distinct original observations d_n greater or equal to some value computed from the distribution of such random variable d_n. Several empirical studies carried out in [7] showed closer estimations of the parameters under study and a reduction of the standard deviations of such estimations. These results were theoretically confirmed in [10].

In this paper we consider a generalization of the OBS method, that consists of drawing bootstrap samples verifying $k_1 \le d_n \le k_2$ for some $1 \le k_1 \le k_2 \le n$. We will name RB (Reduced Bootstrap) to this method. This way, only αn^n bootstrap samples are considered, where $\alpha = P[k_1 \le d_n \le k_2]$. The use of RB inside a bagging procedure lets us to define Bagging with Reduced Bootstrap. We will name Rbagging the resulting procedure.

Definition 2. Algorithm Rbagging.
Fix B, k_1, k_2
For b=1,2,…,B
Draw a reduced bootstrap sample, i.e., a bootstrap sample $\mathbf{D}^* = (U_1^*, \ldots, U_n^*)$
with $k_1 \le d_n^* \le k_2$, taken from **D**
Fit g to **D***, obtaining g_b
Next b.

The resulting aggregated model is also computed as in (1) and (2). To obtain a bootstrap sample **D*** with $k_1 \le d_n^* \le k_2$, we propose the next algorithm.

Definition 3. Algorithm Reduced Bootstrap Sampling.

1. Select a random sample of size k_2 without replacement from $\{1,\ldots,n\}$, say I_1
2. Select a random sample of size k_1 without replacement from I_1, say I_2
3. Draw a random sample of size n-k_1 with replacement from I_1 say I_3
4. Let $L=(l_1,\ldots,l_n)$ be a vector whose components are obtained by randomly permuting the string formed by concatenating I_2 and I_3
5. The sample obtained taking the elements of **D** indexed by $(l_1,\ldots,l_n)$ is a bootstrap sample $\mathbf{D}^* = (U_1^*,\ldots,U_n^*)$, with $k_1 \le d_n^* \le k_2$.

In [5], six choices for k_1 and k_2 are proposed in a study about the consistent estimation of the variance of the sample median, including the usual bagging as a particular case. Because of its good performance, we have used these selections to study the performance of Rbagging. The six resulting methods are presented in table 1, being identified by RB1,…, RB6, where p=1-1/e, q=1-p. Note that RB1 is the original method II presented by Efron, while RB2 and RB6 are particular cases of the OBS method.

Table 1. The six selections for k_1 and k_2

Method	k_1	k_2
RB1	1	n
RB2	$[np-(npq)^{1/2}]+1$	n
RB3	$[np-(npq)^{1/2}]+1$	$[np+(npq)^{1/2}]$
RB4	$[np]+1$	$[np]+1$
RB5	$[np+(npq)^{1/2}]+1$	$[np+(npq)^{1/2}]+1$
RB6	$[np+(npq)^{1/2}]+1$	n

3 Numerical Results

We have made an empirical comparison of the six considered methods over three real data sets. Two unstable classification models, classification trees and neural networks, are used as the base algorithm. R system [11] has been the selected computational tool for our study, whereas the tree and nnet libraries have provided us with the implementation of classification trees and multiplayer perceptrons, respectively. Tree library is based on the CART methodology [12] proposed by Breiman. Nnet library fits single-hidden-layer neural networks by a quasi-Newton method (also known as a variable metric algorithm), specifically that published simultaneously in 1970 by Broyden, Fletcher, Goldfarb and Shanno. We have used the logistic activation function in the hidden layer and the identity function as the activation function for the output layer, selecting the hidden layer size by cross validation.

3.1 Fragile X Syndrome Data

Fragile X is the most common inherited cause of mental impairment and the most common known cause of autism. In 1991, the gene (called FMR1) that causes Fragile X was discovered. In individuals with Fragile X, a defect in FMR1 (a "full mutation") shuts the gene down. Symptoms of fragile X include: mental impairment, ranging from learning disabilities to mental retardation, attention deficit and hyperactivity, anxiety and unstable mood, autistic-like behaviors, long face, large ears, flat feet, and hyperextensible joints, especially fingers. A DNA based test to diagnose Fragile X was developed in 1992. This test is quite accurate, and it can detect both carriers and fully-affected individuals. However it can be very expensive, and for this reason, an automatic classification model would be acknowledged, motivating a study conducted in Andalusia, Spain, where 100 FMR1 mutated children and 72 children with Fragile X symptoms but not mutated were selected, being the last 72 the control cases. From the 61 recorded variables, we selected those variables retained by a stepwise logistic regression analysis performed with SPSS v11.0, reducing to 9 the number of predictor variables.

We randomly divided the data set into training (80%) and test (20%) sets, and we applied the six bagging procedures with B=100 to the classification tree and mutilayer perceptron with 12 hidden nodes, computing the error rate (percent of incorrectly classified cases) for both data sets for each method. The whole procedure were independently repeated 50 times. Table 2 shows the mean and standard deviation of the 50 test error rates for each bagging procedure, where "raw" denotes no bagging.

Table 2. Fragile X Syndrome data. Mean and standard deviation of the 50 test error rates for each procedure

	Classification trees		Multilayer perceptron	
Method	Mean	S.D.	Mean	S.D.
Raw	5.652	5.111	5.403	4.592
RB1	5.977	5.043	5.607	4.919
RB2	5.225	4.584	5.285	4.675
RB3	5.799	5.524	5.388	4.571
RB4	5.225	4.828	5.669	4.958
RB5	5.448	4.629	5.375	5.203
RB6	5.577	4.388	5.329	4.549

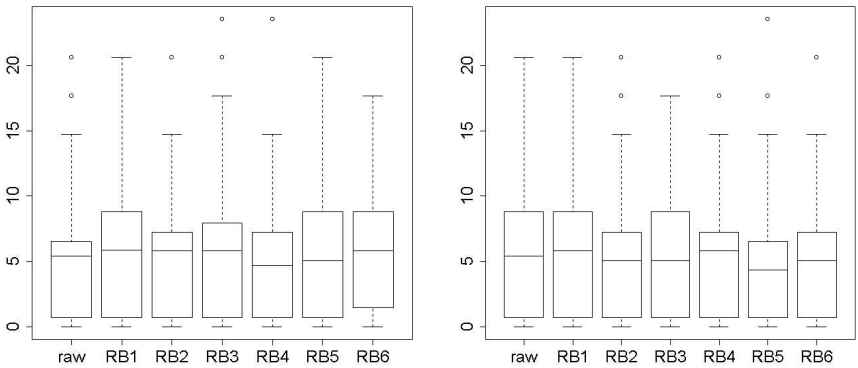

Fig. 1. Distribution of the 50 test mean error rates for the *raw* and *bagged* classification trees (left) and multilayer perceptrons with 12 hidden nodes (right) for the fragile X syndrome data

We can see in table 2 and figure 1 that for both models the mean error rate is increased when bagging is applied. However, for classification trees reduced bagging 2, 4, 5 and 6 yield a lower mean error rate, accompanied by a lower variance of the error rate. A similar comparative performance is observed for the multiplayer perceptron, with the exception of an increase in the variability of RB5 (motivated by two outliers), though RB3 also offers a reduction in the mean and standard deviation of the test error rate. We must note that RB2, a reduced bagging based on OBS bootstrap, produces the minimum mean error rate and a clear reduction of the variability, for both classification models.

3.2 Forensic Glass Data

The forensic glass dataset has 214 points from six classes with nine measurements, and provides a fairly stiff test for classification methods. As in 3.1, we randomly divided the data set into training (80%) and test (20%) sets, applying the six bagging procedures with B=100 to the classification tree and mutilayer perceptron with 15 hidden nodes, also computing the error rate (percent of incorrectly classified cases) for both data sets for each method. The whole procedure were independently repeated 50 times, and the main results are exhibited in table 3 and figure 2.

Table 3. Forensic glass data. Mean and standard deviation of the 50 test error rates for each procedure

	Classification trees		Multilayer perceptron	
Method	Mean	S.D.	Mean	S.D.
Raw	32.662	3.211	50.761	13.141
RB1	23.441	2.334	39.627	10.892
RB2	23.239	2.441	39.243	24.924
RB3	23.621	2.178	39.426	10.010
RB4	23.622	2.156	39.042	9.326
RB5	23.337	2.561	40.572	9.964
RB6	23.620	2.357	39.624	9.153

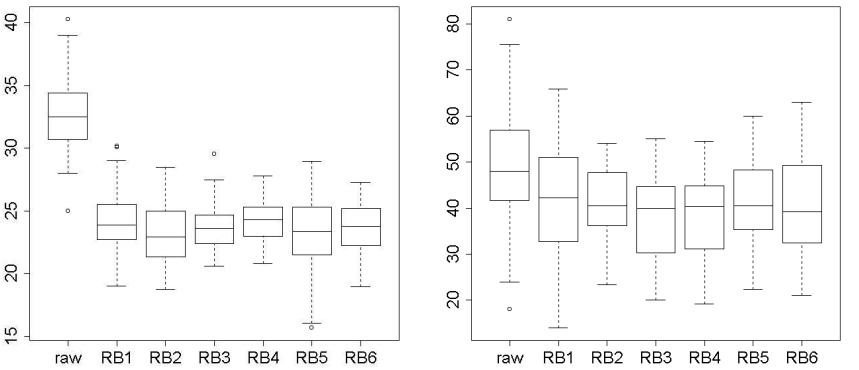

Fig. 2. Distribution of the 50 test mean error rates for the *raw* and *bagged* classification trees (left) and multilayer perceptrons with 15 hidden nodes (right) for the forensic glass dataset

Figure 2 (left) shows the box-and-whisker plots for the classification tree, where a clear reduction in the mean error rate is observed for all the six bagging procedures. However, a slight additional reduction with RB2 and RB5 is observed, though last method has a greater variability. Figure 2 (right) contains a similar representation for the multiplayer perceptron. The bests results are also achieved by RB2, with a great reduction in the mean percent error rate and in its variability.

3.3 South Africa Heart Disease Data

This dataset, utilized in [13], contains 463 cases selected from a larger retrospective sample of males in a heart-disease high-risk region of the Western Cape, South Africa. The target is the absence/presence of a coronary heart disease, existing nine predictor variables. A similar study as in 3.1 and 3.2 were conducted: we randomly divided the data set into training (80%) and test (20%) sets, applying the six bagging procedures with B=100 to the classification tree and mutilayer perceptron with 12 hidden nodes, also computing the error rate (percent of incorrectly classified cases) for both data sets for each method. The whole procedure were also independently repeated 50 times. Table 4 shows the mean and standard deviation of the 50 test error rates for each bagging procedure, and the whole distributions are plotted in the figure 3.

Table 4. South Africa heart disease data. Mean and standard deviation of the 50 test error rates for each procedure

	Classification trees		Multilayer perceptron	
Method	Mean	S.D.	Mean	S.D.
Raw	33.217	4.398	34.434	4.783
RB1	31.173	4.350	34.134	5.042
RB2	30.693	4.572	34.326	4.143
RB3	30.608	4.355	34.413	4.378
RB4	30.630	4.295	34.413	4.433
RB5	31.021	4.082	34.086	4.313
RB6	30.562	4.082	34.108	4.358

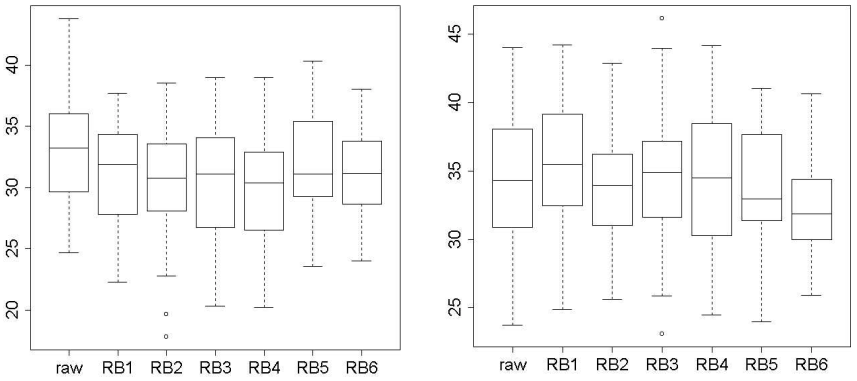

Fig. 3. Distribution of the 50 test mean error rates for the *raw* and *bagged* classification trees (left) and multilayer perceptrons with 12 hidden nodes (right) for the South Africa heart disease dataset

We see that the five reduced bagging procedures yield a mean percent error rate lower than the raw and bagged classification trees, standing out the reduced bagging 6, which also provides the minimum standard deviation, as it is confirmed by the

figure 3 (left). For the multiplayer perceptron the bagging procedure is not so clearly improved, but the procedures rbagging 5 and 6 provide the lowest mean values, accompanied by a standard deviation lower than that achieved by the usual bagging procedure. This better performance, particularly for rbagging 6, is more clearly illustrated in the figure 3 (right).

4 Concluding Remarks

The alternative bagging methodology based on reduced bootstrap sampling shows good and hopeful results. It has outperformed the usual bagging in our empirical study over real data sets, at least one rbagging method which offers a lower mean and variance of the test error rate is found for each data set.

However, a further study may be realized following some guidelines, for example: the theoretical study of the properties of rbagging, the development of criteria to select the parameters k_1 and k_2, a comparison with other ensemble methods, to analyze the effect of rbagging over other learning algorithms, and the application to prediction problems.

References

1. Breiman, L.: Bagging Predictors. Mach. Learn. 24 (1996) 123–140
2. Bühlman, P., Yu, B.: Analyzing Bagging. Ann. Stat. 30 (4) (2002) 927-961
3. Buja, A., Stuetzle, W.: The effect of bagging on variance, bias, and mean squared error. Preprint, AT&T Labs-Research (2000)
4. Rao, C.R., Pathak, P.K., Koltchinskii, V.I.: Bootstrap by sequential resampling. J. Statist. Plan. Infer. 64 (1997) 257-281
5. Jiménez-Gamero, M.D., Muñoz-García, J., Pino-Mejías, R.: Reduced bootstrap for the median. Stat. Sinica (2004) (in press)
6. Efron, B.: Bootstrap methods: another look at the jackknife. Ann. Stat. 7 (1979) 1-26
7. Muñoz-García, J., Pino-Mejías, R., Muñoz-Pichardo, J.M., Cubiles-de-la-Vega, M.D.: Identification of outlier bootstrap samples. J. Appl. Stat. 24 (3) (1997) 333-342
8. Hall, P.: Antithetic resampling for the bootstrap. Biometrika (1989) 713-724
9. Johns, M.V.: Importance sampling for bootstrap confidence intervals. J. Am. Stat. Assoc. 83 (1988) 709-714
10. Jiménez-Gamero, M.D., Muñoz-García, J, Muñoz-Reyes, A., Pino-Mejías, R.: On Efron´s method II with identification of outlier bootstrap samples. Computation. Stat. 13 (1998) 301-318
11. Ihaka, R. & Gentleman, R.: R: A Language for Data Analysis and Graphics. J. Comput. Graph. Stat. 5 (1996) 299-314
12. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: Classification and Regression Trees. Wadsworth (1984)
13. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning. Springer (2001)