ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8. Available from http://www.i6doc.com/en/.

Improving the Random Forest Algorithm by Randomly Varying the Size of the Bootstrap Samples for Low Dimensional Data Sets

Md Nasim Adnan and Md Zahidul Islam

Centre for Research in Complex Systems (CRiCS)
School of Computing and Mathematics, Charles Sturt University
Bathurst, NSW 2795, Australia
madnan@csu.edu.au, zislam@csu.edu.au.

Abstract. The Random Forest algorithm generates quite diverse decision trees as the base classifiers for high dimensional data sets. However, for low dimensional data sets the diversity among the trees falls sharply. In Random Forest, the size of the bootstrap samples generally remains the same every time to generate a decision tree as the base classifier. In this paper we propose to vary the size of the bootstrap samples randomly within a predefined range in order to increase diversity among the trees. We conduct an elaborate experimentation on several low dimensional data sets from UCI Machine Learning Repository. The experimental results show the effectiveness of our proposed technique.

1 Introduction

In the last four decades, ensemble of classifiers have been extensively applied in the arena of data mining for pattern understanding [1]. Decision forest is an ensemble of decision trees where an individual decision tree acts as the base classifier and the classification is performed by taking a vote based on the predictions made by each decision tree of the decision forest [2]. Random Forest [3] is a popular state-of-the-art decision forest building algorithm that is essentially a combination of Bagging [4] and Random Subspace [5] algorithms. Bagging generates new training data set $\boldsymbol{D}_i$ iteratively where the records of $\boldsymbol{D}_i$ are chosen randomly from the training data set $\boldsymbol{D}$. $\boldsymbol{D}_i$ contains the same number of records as in $\boldsymbol{D}$. Thus some records of $\boldsymbol{D}_i$ can be chosen multiple times and some records may not be chosen at all. This approach of generating a new training data set is known as bootstrap sampling. On an average 63.2% of the original records are present in a bootstrap sample [6]. The Random Subspace algorithm is then applied on each bootstrap sample $\boldsymbol{D}_i (i = 1, 2, \ldots, T)$ in order to generate T number of trees for the forest. The Random Subspace algorithm randomly draws a subset of attributes (subspace) $\boldsymbol{f}$ from the entire attribute space $\boldsymbol{m}$ in order to determine the splitting attribute for each node of a decision tree.

It is important to note that the decision forest accuracy is dependent on both the individual tree accuracy and the diversity among the decision trees. This means that optimization on any one of the two factors does not essentially deliver the best decision forest accuracy. Ideally, an algorithm should generate the

ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8. Available from http://www.i6doc.com/en/.

best individual trees with lowest similarity (highest diversity) [5]. The individual tree accuracy and the diversity among the decision trees depend on the size of $\boldsymbol{f}$. If $\boldsymbol{f}$ is sufficiently small then the chance of having the same attributes in different subspaces is low, thus the trees in a forest tend to become less similar. However, a sufficiently small $\boldsymbol{f}$ may not guaranty the presence of adequate number of attributes with high classification capacity that may decrease individual tree accuracy. Consequently, the individual tree accuracy can be increased with relatively larger number of attributes in $\boldsymbol{f}$ by sacrificing diversity. In literature, the number of attributes in $|\boldsymbol{f}|$ is commonly chosen to be $int(\log_2 |\boldsymbol{m}|) + 1$ [3].

As a matter of fact, the values of $int(\log_2 |\boldsymbol{m}|)+1$ increase very slowly relative to the increase of $|\boldsymbol{m}|$. In theory, the proportion of $|\boldsymbol{f}|$ to $|\boldsymbol{m}|$ gradually decreases with the increase of $|\boldsymbol{m}|$. Now, let us assume that we have a low dimensional data set consisting of 8 attributes. Thus the splitting attribute is determined from randomly selected subspace of $int(\log_2 8) + 1 = 4$ attributes. As a result, we can have only two subspaces with completely different attributes. Thus the chance of having the similar attributes in different subspaces becomes high resulting in decreasing diversity. In literature, empirical studies have proven that Bagging [4] is a simple component of Random Forest that generally increases diversity among the base classifiers [3], [4], [7]. In Bagging, the bootstrap samples contain the same number of records as in the training data set [3], [4]. In reality, this prescription is arbitrary and is not necessarily optimal in terms of prediction accuracy for any ensemble. In fact, bootstrap samples (with replacement) containing either 60% or 80% of the unique records from the training data set may contribute to better overall ensemble accuracy [8]. However, the optimal sample configuration can be very different for different data sets [8]. This indicates that the number of records in the bootstrap samples should not be fixed beforehand, independently of the training data sets. In line with these facts, we propose to vary the size of the bootstrap samples randomly within a predefined range in order to induce stronger diversity among the trees to help increase the ensemble accuracy of Random Forest specially for low dimensional data sets.

2 Improving the Random Forest Algorithm for Low Dimensional Data Sets

At first, we randomly determine the percentage of the number of unique records to be fetched from the training data set by generating a random number within the range of 60 and 80. We select this particular range because we find both bootstrap samples (with replacement) containing either 60% or 80% of the unique records from the training data set may contribute to better overall ensemble accuracy [8]. We next calculate the exact number of the unique records to be drawn from the training data set using the randomly generated percentage. For example, let we get 70 (within 60 and 80) as the percentage for unique records. Let the training data set contains 1000 records. So the number of unique records to be fetched randomly from the training data set will be $\frac{70}{100} \times 1000 = 700$. After selecting a certain percentage (within 60 to 80) of unique records from the

ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8.
Available from http://www.i6doc.com/en/.

training data set, we reselect 30% records randomly from the already selected unique records and add them to complete generating the proposed bootstrap sample. For the ongoing example, 700 unique records are randomly selected from the 1000-record training data set. Then 30% of the 700 unique records are reselected randomly. Thus we reselect $\frac{30}{100} \times 700 = 210$ records. In total, the proposed bootstrap sample will contain $(700 + 210) = 910$ records. The number of records in the proposed bootstrap samples varies with the selection of the percentage of the number of unique records. Assuming we have 1000 records in a training data set as before, we next present how the final size of the proposed bootstrap samples can vary based on different percentage values in Table 1.

Table 1: Variation of the Size for Bootstrap Samples

No. of Records	Unique Records in (%)	No. of Unique Records	30% Reselection	Final No. of Records	Increase/ Decrease
1000	60	600	180	780	-22.00%
1000	70	700	210	910	-09.00%
1000	80	800	240	1040	+04.00%

In literature, we find that bootstrap samples containing 60% or 80% of the unique records (with replacement) may perform better than the standard bootstrap samples; yet the optimal sample size can be very different for different data sets [8]. However, it is very difficult to optimize the exact size for the bootstrap samples for every training data set since the search spaces (in this case the total number of records of the training data set) are huge. Decision trees generated from the bootstrap samples having fewer number of unique records from the training data set exhibit strong diversity being individually less accurate. On the other hand, bootstrap samples that are very similar to the original training data set with respect to the number of unique records may generate decision trees with better individual accuracy but not diverse enough to be able to correct the generalization errors. As a solution to these scenarios, we intend to provide a good balance between individual accuracy and diversity through our proposed technique in order to improve the ensemble accuracy. Our approach not only varies the number of the unique records but also the size of the bootstrap samples in order to extract stronger diversity from the bootstrap samples.

3 Experimental Results

We conduct an elaborated experimentation on seven (07) natural data sets that are publicly available from the UCI Machine Learning Repository [9]. The data sets used in the experimentation are listed in Table 2. All the data sets shown in Table 2 have less than ten (10) non-class attributes that cover almost every well known low dimensional data sets available in [9].

ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8. Available from http://www.i6doc.com/en/.

Table 2: Description of the data sets

Data Sets	No. of Attributes	No. of Records	No. of Classes
Balance Scale	04	625	3
Liver Disorders	06	345	2
Glass	09	214	6
Hayes-Roth	04	160	3
Iris	04	150	3
Lenses	04	24	3
PIMA	08	768	2

We implement Random Forest using different types of bootstrap samples. When Random Forest uses bootstrap samples with 60% unique records, we call it 60% RF. Thus we call the original Random Forest as 63.2% RF. In this way, Random Forest on bootstrap samples with 80% unique records is called 80% RF. However, as our proposed bootstrap samples may have variable number of unique records we call it Proposed RF. We generate 100 fully grown trees (no pruning is applied) for each ensemble since the number is considered to be large enough to ensure convergence of the ensemble effect [10] and use majority voting is used to aggregate the results. All the results reported in this paper are obtained using 10-fold-cross-validation (10-CV). The best results are emphasized through **bold-face**.

Ensemble accuracy is obviously the most important performance indicator for any ensemble such as Random Forest. In Table 3 we present ensemble accuracies for 63.2% RF, 60% RF, 80% RF and Proposed RF for all the data sets considered.

Table 3: Ensemble Accuracy

Data Set Name	63.2% RF	60% RF	80% RF	Proposed RF
Balance Scale	80.5040	81.2980	78.5550	**81.4480**
Liver Disorders	71.4800	71.2960	70.4540	**71.5540**
Glass	74.1150	74.1140	75.5430	**78.3230**
Hayes-Roth	64.9210	68.8700	64.3590	**71.9480**
Iris	**96.0000**	95.3330	95.3330	**96.0000**
Lenses	78.3330	78.3330	78.3330	**83.3330**
PIMA	75.9460	75.4340	75.5770	**77.1560**
Average	77.3284	77.8111	76.8791	**79.9660**

From Table 3, we see that the Proposed RF outperforms the other prominent variants for all the data sets considered (including one tie). These results clearly demonstrates the effectiveness of the Proposed RF. To figure out the reason

ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8. Available from http://www.i6doc.com/en/.

behind this improvement we compute Average Individual Accuracy (AIA) and Average Individual Kappa (AIK) as was done in literature [11]. We report the results in Table 4 and Table 5.

Table 4: Average Individual Accuracy

Data Set Name	63.2% RF	60% RF	80% RF	Proposed RF
Balance Scale	64.8844	64.9384	**65.3832**	65.0357
Liver Disorders	60.1019	59.2888	**60.9583**	59.8780
Glass	60.5181	59.9337	**63.2001**	62.5301
Hayes-Roth	53.9205	54.4273	**54.7269**	55.0924
Iris	93.3063	93.2795	**93.9862**	93.5863
Lenses	66.4000	65.7667	**67.2833**	66.6333
PIMA	69.9518	69.5045	**70.5357**	70.3253
Average	67.0118	66.7341	**68.0105**	67.5830

Table 5: Average Individual Kappa

Data Set Name	63.2% RF	60% RF	80% RF	Proposed RF
Balance Scale	0.4322	**0.4327**	0.4758	0.4541
Liver Disorders	0.3122	**0.3043**	0.3761	0.3478
Glass	0.5338	**0.5120**	0.5610	0.5418
Hayes-Roth	0.3316	0.3452	0.3431	**0.3189**
Iris	**0.9097**	0.9223	0.9259	0.9189
Lenses	0.3125	**0.3101**	0.3510	0.3616
PIMA	0.4897	**0.4772**	0.5322	0.5104
Average	0.4745	**0.4720**	0.5093	0.4934

Usually these two performance indicators AIA and AIK are in conflict. From Table 4 and Table 5, we see that when a method is low in diversity then it is high in AIA. For example, 80% RF has the highest AIA value and thus has the lowest diversity among it's trees. On the contrary, 60% RF has the lowest AIA value with highest diversity. But none of these two have the highest ensemble accuracy. This means that optimizing any of these two objectives does not necessarily result in the best ensemble accuracy. Ideally a method should generate trees with highest AIA and lowest AIK. However, in theory and in practice this dual optimization cannot be attained simultaneously [5]. The results reported in this paper indicate that the Proposed RF improves the ensemble accuracy considerably by maintaining a better balance between AIA and AIK.

ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8. Available from http://www.i6doc.com/en/.

4 Conclusion

Original Random Forest algorithm falls short in generating diverse decision trees specially for low dimensional data sets. To help Random Forest be more versatile we propose a new technique that not only varies the number of the unique records but also the size of the bootstrap samples in order to extract stronger diversity from the bootstrap samples. The results presented in this paper show great potential of the proposed technique. Further, we plan to apply our technique on some of the latest forest building algorithms such as Rotation Forest [12].

References

[1] Zhang, L., Suganthan, P. N.: *Random Forests with ensemble of feature spaces*. Pattern Recognition, vol. 47, pp. 3429-3437, 2014.

[2] Tan, P., Steinbach, M., Kumar, V.: *Introduction to Data Mining*. Pearson Education, Boston, U.S.A., 2006.

[3] Breiman, L.: *Random Forests*. Machine Learning, vol. 45, no. 1, pp. 5-32, 2001.

[4] Breiman, L.: *Bagging predictors*. Machine Learning, vol. 24, no. 2, pp. 123-140, 1996.

[5] Ho, T. K.: *The Random Subspace Method for Constructing Decision Forests*. IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 20, no. 1, pp 832-844, 1998.

[6] Han, J., Kamber, M.: *Data Mining Concepts and Techniques*, 2nd ed. Morgan Kaufmann, San Francisco, U.S.A., 2006.

[7] Quinlan, J. R.: *Bagging, Boosting and C4.5*. Proceedings of the 13th National Conference on Artificial Intelligence, pp. 725-730, Cambridge, MA, 1996.

[8] Munoz, G. M., Suarez, A.: *Out-of-bag estimation of the optimal sample size in bagging*. Pattern Recognition, vol. 43, pp. 143-152, 2010.

[9] UCI Machine Learning Repository: *http://archive.ics.uci.edu/ml/datasets.html*. (Last Accessed: 11 Jun 2014)

[10] Geurts, P., Ernst, D., Wehenkel, L.: *Extremely randomized trees*. Machine Learning, vol. 63, pp. 3-42, 2006.

[11] Amasyali, M. F., Ersoy, O. K.: *Classifier Ensembles with the Extended Space Forest*. IEEE Transaction on Knowledge and Data Engineering, vol. 26, pp. 549-562 (2014)

[12] Rodriguez, J. J., Kuncheva, L. I., Alonso, C. J.: *Rotation Forest: A New Classifier Ensemble Method*. IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 28, pp. 1619-1630, 2006.