The final publication is available at Springer via http://dx.doi.org/10.1007/978-3-642-38326-7_15

Volume 7885 of the series Lecture Notes in Computer Science, pp 99-103

Enhancing Random Forests Performance in Microarray Data Classification

Nicoletta Dessì, Gabriele Milia, Barbara Pes
Università degli Studi di Cagliari, Dipartimento di Matematica e Informatica,
Via Ospedale 72, 09124 Cagliari, Italy
{dessi, milia.ga, pes}@unica.it

Abstract. Random forests are receiving increasing attention for classification of microarray datasets. We evaluate the effects of a feature selection process on the performance of a random forest classifier as well as on the choice of two critical parameters, i.e. the forest size and the number of features chosen at each split in growing trees. Results of our experiments suggest that parameters lower than popular default values can lead to effective and more parsimonious classification models. Growing few trees on small subsets of selected features, while randomly choosing a single variable at each split, results in classification performance that compares well with state-of-art studies.

Keywords: Microarray data classification, Random Forests, Feature selection.

1 Introduction

As observed in [1], the random forest performance tends to decline when the number of features is huge and the proportion of truly informative features is small, such as with gene expression data. Thus, applying random forests in microarray data analysis presents an interesting research goal due to the additional issue of reducing the contribution of trees whose nodes are populated by non-informative features.

Pre-filtering features is a popular procedure that has proved to be useful to face the curse of dimensionality of gene expression data. When applied before growing a random forest, this process has to face an additional issue: asserting values for the two critical parameters of the random forest, i.e. the number of variables randomly chosen at each split, namely *mtry*, and the number of the trees in the forest, namely *ntree*.

This paper evaluates the effects of a filtering process on the predictive performance of a random forest classifier as well as on the choice of its critical parameters. Using two popular microarray datasets, we carried out classification experiments by growing random forests both on the whole set of features and on different subsets of pre-filtered features: different parameter settings were explored in order to investigate the optimal trade-off between the number of trees and the number of variables randomly chosen at each split. Our results suggest that growing few trees on small subsets of pre-filtered features, with only one variable randomly chosen at each split, presents results which compare very well with state-of-art studies in literature.

2 Experiments and Results

We experimented with two public microarray datasets: *Leukemia* [2] and *Colon* [3]. The overall analysis was performed using the Weka data mining environment [4]. For performance estimation, we used a standard cross-validation procedure (LOOCV), as in the majority of the papers, though it has been observed that a cross-validation setting can produce overoptimistic results on small sample size domains [5]. The performance was evaluated using the AUC (area under the ROC curve) metric in order to synthesize the information of sensitivity and specificity.

The experiments were divided into two classes:

1. *Tuning on the whole dataset.* We grew different random forests within the following parameters values: (i) *ntree* = 10, 20, 30, 50, 100, 200, 300, 500, 1000, 1500; (ii) *mtry* = 1, 2, 3, 5, 10, 20, 30, 40, 50, 80. Both the choices (i) and (ii) aim to finely explore parameters values smaller than the common default values.
2. *Tuning on filtered subsets.* First, we ranked the features of the original dataset using two popular ranking methods, i.e. *Information Gain* (IG) and *Chi Squared* (χ^2). Based on their outputs, we selected different subsets of highly-ranked features denoted in the following as TOP10 (i.e. the first 10 top-ranked features), TOP20 (i.e. the first 20 top-ranked features) and so on. Then, we used these subsets for growing random forests within the following parameter configurations: (i) *ntree* = 10, 20, 30, 50, 100, 200, 300; (ii) *mtry* = 1, 2, 3, 5, 10, 20, 30.

Results about tuning on the whole dataset. Fig.1 and Fig. 2 show, for different values of *mtry*, the effects of changes in the parameter *ntree* on the AUC. As asserted by [6], the behavior of AUC is asymptotic: as the number of trees increases, the AUC value converges to a limit. Interestingly, in both *Leukemia* and *Colon*, we observed this asymptotic trend for *ntree* > 100, while previous studies [7][8] on microarray datasets made use of *ntree* values in the order of thousands. Globally, results in Fig. 1 and Fig. 2 suggest that, even on high-dimensional domains, the choice *ntree* = 100 can be quite adequate, with further increases having negligible effects and smaller values leading to more unstable AUC performance.

As regards the influence of *mtry* parameter on random forest behavior, Fig. 1 and Fig. 2 show that, for small values (≤ 50) of *ntree*, the choice of high values of *mtry* (*mtry* >= 30 for *Leukemia* and *mtry* >= 5 for *Colon*) results in higher values of AUC. This seems to suggest that, when we choose to grow a forest with a small number of trees, we need to set higher values for *mtry* in order to increase the probability of randomly selecting informative variables. On the other hand, if the forest is sufficiently large (*ntree* >= 100), the influence of *mtry* parameter decreases. In particular, no improvement in AUC performance can be observed when setting values of *mtry* > 20 and *mtry* > 10 for *Leukemia* and *Colon* respectively. Hence, as previously observed for the *ntree* parameter, the common default setting of *mtry* = sqrt(M) [7][8], where M is the dataset dimensionality, seems to be unnecessary large, with smaller values ensuring a good predictive performance at a lower computational cost.

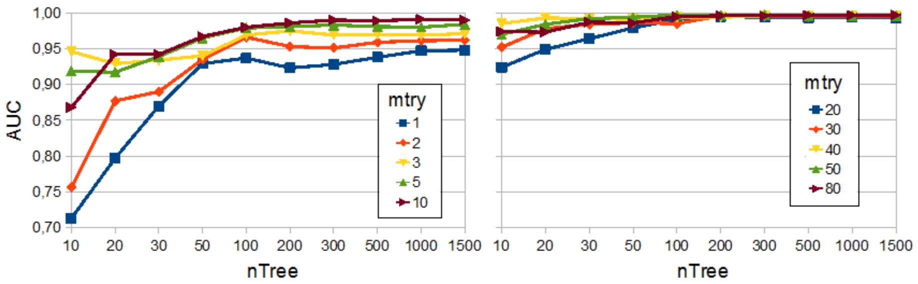

Fig. 1. Tuning on *Leukemia* dataset: AUC versus *ntree* for *mtry* = 1, 2, 3, 5, 10 (left) and *mtry* = 20, 30, 40, 50, 80 (right).

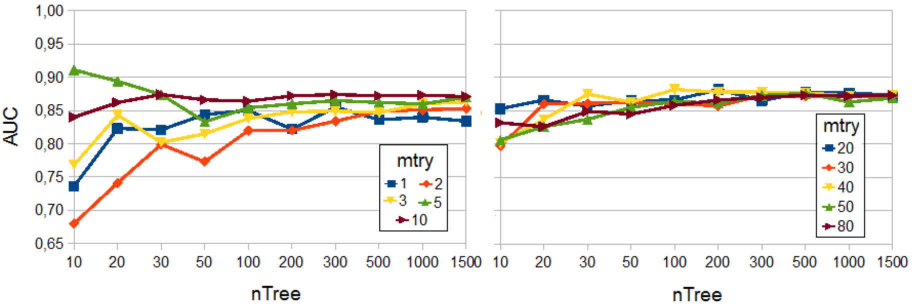

Fig. 2. Tuning on *Colon* dataset: AUC versus *ntree* for *mtry* = 1, 2, 3, 5, 10 (left) and *mtry* = 20, 30, 40, 50, 80 (right).

Results about tuning on filtered subsets. As said before, we applied two ranking methods (IG and χ^2) and, for each ranking method, we performed tuning experiments on pre-filtered subsets of increasing size (TOP10, TOP20, etc). Table 1 summarizes the "optimal" values of both parameters *ntree* and *mtry*, i.e. the lowest values leading, on a given subset, to the best AUC result. As we can see, in most cases, the value *mtry* = 1 is sufficient to maximize the predictive performance of random forests. The optimal number of trees is also quite low, especially for *Leukemia*, where the AUC is maximized with at most 30 random trees. More trees (a few hundred at most) can be needed for *Colon* which is recognized to be a more noisy dataset. Results in Table 1 globally confirm what previously observed on the overall datasets: parameter values lower than common default values can lead to effective and more parsimonious classification models. Although surprising, the goodness of the choice *mtry* = 1 is also supported (for datasets of low-moderate dimensionality, as the pre-filtered datasets here considered) by some considerations reported in [6].

Additionally, the pre-filtering process significantly improves the predictive performance. As regards *Leukemia*, our experiments gave excellent AUC results in all the subsets from TOP10 to TOP500. Only for larger subsets (TOP1000), the AUC decreases if the number of random trees is not sufficiently large, as we can see in Fig. 3.a, where the AUC behavior is shown for some subsets filtered by IG (an analogous trend has been registered for χ^2) within the "optimal" setting *mtry* = 1.

Table 1. Optimal values of *mtry* and *ntree* for pre-filtered subsets of increasing size, as obtained by IG and χ^2 ranking methods, for both *Leukemia* and *Colon* datasets.

Pre-filtered subset	**Leukemia**				**Colon**			
	IG		χ^2		**IG**		χ^2	
	mtry	***ntree***	***mtry***	***ntree***	***mtry***	***ntree***	***mtry***	***ntree***
TOP10	1	30	1	20	1	30	10	20
TOP20	1	10	1	10	1	10	1	200
TOP30	1	10	1	10	1	20	1	10
TOP50	1	10	1	20	10	10	1	10
TOP100	1	20	1	20	1	100	1	200
TOP300	1	30	1	20	1	100	1	300
TOP500	1	10	1	20	1	200	3	50

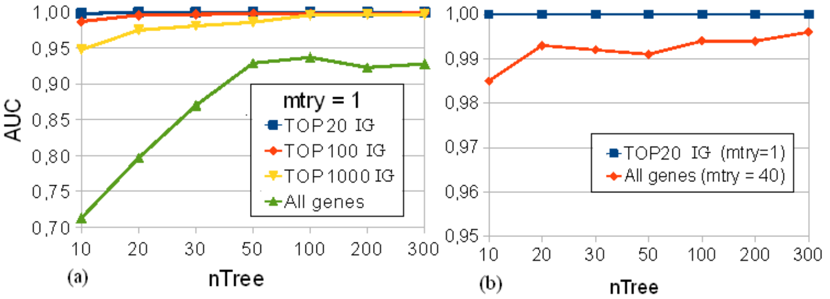

Fig. 3. *Leukemia* dataset: (a) AUC versus *ntree* for some pre-filtered subsets and for the whole dataset (*mtry* = 1 for all the curves); (b) AUC versus *ntree* for the subset TOP20 (*mtry* = 1) and for the whole dataset (*mtry* = 40).

Again, we notice the asymptotic behavior of AUC. The effectiveness of pre-filtering is considerable as the random forests grown on the reduced subsets greatly outperform the random forests built on the original dataset. However, the setting *mtry* = 1, optimal for the filtered subsets, is not so optimal for the whole dataset, where the best AUC performance is registered for *mtry* ≥ 30, as shown in Fig.1. Hence, a further demonstration of the effectiveness of the pre-filtering process is given in Fig. 3.b where the performance on the TOP20 subset (*mtry* = 1) is compared with the performance on the whole dataset, based on *mtry* = 40 (this value corresponds to the "best" AUC curve in Fig.1). The advantages deriving from pre-filtering are confirmed by the analysis on *Colon* dataset (here omitted for the sake of space).

Finally, Table 2 shows the effectiveness of our approach when compared to the most cited studies that applied random forests to microarray data [7] [8]. In particular, [8] reports an error rate of 0,051 for the *Leukemia* dataset (in a slightly different version) using the random forest method with *mtry* = sqrt(M) and *ntree* = 5000 and without a preliminary gene selection. Within the same settings, the error rate reported for *Colon* is 0.127. By integrating a variable selection approach, the best error rates given in [8] for *Leukemia* and *Colon* are 0,075 and 0,159 respectively. In [7], the AUC performance for *Colon* is 0.867 on the full dataset and 0,917 with gene selection; here, the best-performing configuration is selected among the following values of parameters: *ntree* = 500, 1000, 2000 and *mtry* = 0,5·sqrt(M), 1·sqrt(M), 2·sqrt(M).

Table 2. Our best results on *Leukemia* and *Colon*, both in terms of AUC and accuracy.

Dataset	**On the full set of genes**		**Using a filtered subset**	
	AUC	*Accuracy*	*AUC*	*Accuracy*
Leukemia	0,997	0,986	1,00	1,00
Colon	0,911	0,855	0,939	0,903

Acknowledgments. This research was supported by RAS, Regione Autonoma della Sardegna (Legge regionale 7 agosto 2007, n. 7), in the project "*DENIS: Dataspaces Enhancing the Next Internet in Sardinia*".

3 Conclusions

The experimental analysis performed on two public microarray datasets reveals that a pre-filtering process positively impacts both on random forest performance and on its optimal parameterization, leading to very effective and more parsimonious classification models. Our future research will address a further potentiality of the random forest method: it can be used not only for classification but also for feature selection, due to its capacity of deriving a variable importance index.

References

1. Amaratunga, D., Cabrera, J., Lee, Y.S.: Enriched random forest. Bioinformatics, vol.24, pp. 2010--2014 (2008)
2. Golub, T.R., et al.: Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. Science, 286:531--537 (1999)
3. Alon, U., et al.: Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. PNAS, vol. 96, pp. 6745--6750 (1999)
4. http://www.cs.waikato.ac.nz/ml/weka/
5. Braga-Neto, U. and Dougherty, E.: Is cross-validation valid for small-sample microarray classification? Bioinformatics, 20, pp. 374--380 (2004)
6. Breiman, L.: Random forests. Machine Learning, vol. 45, pp. 5--32 (2001)
7. Statnikov, A., Wang, L., Aliferis, C.F.: A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification. BMC Bioinformatics 9:319 (2008)
8. Dìaz-Uriarte, R., Alvarez de Andrés, S.: Gene selection and classification of microarray data using random forest. BMC Bioinformatics 7:3 (2006)