Variable Selection Using Mean Decrease Accuracy And Mean Decrease Gini Based on Random Forest

Hong Han[1]
School of Engineering Science
University of Chinese Academy of Science
Beijing, China
hanhong14@mails.ucas.ac.cn

Xiaoling Guo[2] and Hua Yu[*]
School of Engineering Science
University of Chinese Academy of Science
Beijing, China
{guoxiaoling & yuh}@ucas.ac.cn

Abstract—**Variable selection is very important for interpretation and prediction, especially for high dimensional datasets. In this paper, a new method is proposed based on Random Forest (RF) to select variables using Mean Decrease Accuracy (MDA) and Mean Decrease Gini (MDG). We also use dichotomy method to screen variables, which is proved to perform very fast. Experiments on 10 microarray datasets show that the new method is proficient and robust. In addition, we compared the proposed method with other variable selection methods, and the results demonstrated that our proposed method is more robust and more powerful in both accuracy and CPU time.**

Keywords-Variable selection; High dimensionality; Random Forest; Dichotomy; Mean Decrease Accuracy; Mean Decrease Gini

I. INTRODUCTION

Variable selection becomes more and more vital in statistical learning recently, and wide attention has been paid since variables are remarkably increasing to thousand or even more, especially in microarray dataset. Therefore new techniques are proposed to address these challenging tasks involving many irrelevant and redundant variables and often comparably few training examples [1]. The advantages of variable selection are embodied in three aspects:

- Redundant, noisy or unreliable variables may impair the performance of the final prediction.
- The cost of data gathering and storage will decrease and computational speed will increase with the number of variables decreasing.
- The understandability of machine learning model will be significantly improved.

A. A brief overview

Random Forest, proposed by Breiman [2], is a popular approach in applied statistics due to its easy applicability to classification and regression problems. It shows high predictive accuracy and applicability even in high-dimensional problems with highly correlated variables, which often occurs in bioinformatics [3]. Many researchers use it to screen variables and reduce dimensionality, such as Genur, Poggi, & Tuleau-Malot [4], Ishak [5], Janitza, Tutz & Boulesteix [6], Hapfelmeier & Ulm [7]. Zhou, Zhou & Li [8] proposed a random forest-based feature selection algorithm that incorporates the feature cost into the base decision tree construction process to produce low-cost feature subset, and the real feature cost needs to be estimated by experts. However, literature may not be always reliable, the expert's interpretation may cause bias and interesting results about new findings may be ignored [9-10]. A key advantage of random forest variable importance measures, as compared to univariate screening methods, is that they cover the impact of each predictor variable individually as well as in multivariate interactions with other predictor variables [3]. Lunetta, Hayward, Segal & Eerdewegh [11] found that the interactions between variables can be detected more efficiently by means of random forests than by some univariate screening methods like Fisher Exact test. There are also many other variable selection methods. Park & Kim proposed a sequential random k-nearest neighbor feature selection method [12]. Nguyen & Torre pointed out an optimal feature selection for support vector machines [13]. In fact, variable selection is divided into three types: embedded, filter, and wrapper. The embedded approach combines variable selection and the accuracy of the prediction into its procedure. The filter approach does not depend on the proposed model when evaluating variable importance. The wrapper approach takes the prediction performance into account when calculating the variable scores.

Most of authors use permuting out-of-bag (OOB henceforth) error or impurity as the rule to estimate the importance of one variable, but they only consider one of the two measures. In this paper, we take both of them into account, that is to say, we not only consider the importance of OOB error but also concern the impurity of the variable. Thus, we consider two indices: Mean Decrease Accuracy (MDA) and Mean Decrease Gini (MDG) for classification. As for regression, there are also

978-1-4673-9904-3/16/$31.00 ©2016 IEEE

two indices: %IncMSE and IncNodePurity, which are similar to MDA and MDG, standing for the importance of OOB error and the impurity of the variable, respectively. In this paper, we mainly focus on classification problems using random forest since random forest is more powerful in classification than regression. But in order to identify that our proposed method is also appropriate to regression problems, we will also illustrate an example at the end of paper. The reasons why we choose these two indices to assess the importance of one variable are in four aspects: 1) The OOB error gives fair estimation compared to the usual alternative test set error even if it is considered to be a little bit optimistic. 2) The Gini index is not only suitable for classification but also for regression. 3) The two indices MDA and MDG are all default output of the Random Forest procedure, so it is very convenient to use. 4) Using both of them is more robust than using any single one of them.

In this paper, we mainly use R package randomForest to accomplish our proposed method, for more information about this package you can refer to [14-15].

B. Outline of this paper

The paper is arranged as follows. Section 2 presents the new variable selection method using MDA and MDG (MDAMDG method henceforth). In addition, we give examples to illustrate the detailed procedure of the new method. In section 3, experiments are conducted using 10 microarray datasets and we compare the results with other variable selection methods. To identify MDAMDG is suitable for regression problems as well, we do an experiment on a dataset called ozone. Finally, in section 4 we make a conclusion and put forward our future work.

II. New variable selection method

A. The background of MDA and MDG

In this subsection, we will give the background of the variable importance (VI henceforth) measures. The index MDA, utilizes permuting OOB samples to compute the importance of the variable. The OOB sample is the set of observations which are not used for building the current tree. It is used to estimate the prediction error and then to evaluate variable importance [4]. The OOB error importance is defined as follows: For each tree, the prediction error on the out-of-bag portion of the data is recorded (error rate for classification, mean square error (MSE henceforth) for regression). Then the same is done after permuting each predictor variable. The differences between the two are then averaged over all trees. We give a general equation to define this:

$$VI_j = \frac{1}{ntree}\sum_{t=1}^{ntree}(\mathrm{EP}_{tj} - \mathrm{E}_{tj}) \tag{1}$$

Here

- *ntree* denotes the number of trees in the forest.
- E_{tj} denotes the OOB error on tree t before permuting the values of X_j
- EP_{tj} denotes the OOB error on tree t after permuting the values of X_j

The idea underlying this *VI* can be traced to [6]. The larger of MDA value, the more important of the variable. The second measure uses index MDG, which is the total decrease in node impurities from splitting on the variable, averaged over all trees. For classification, the node impurity is measured by the Gini index. For regression, it is measured by residual sum of squares. We should also denote that the larger of MDG value, the purer of the variable.

B. Procedure

A two-step procedure of the new variable selection method called MDAMDG is shown as follows.

Step 1. Ranking and scoring using MDA and MDG.

- Run *Random Forest* algorithm and return MDA and MDG of each variable.
- Rank every variable using MDA and MDG, respectively, score each variable (high value of MDA or MDG with high scores), and compute the total score of each variable, reorder them by the total score.

Step 2. Selecting variables using dichotomy method.

- Select the first 50 percent high score variables as the new variables.
- Run *Random Forest* using the new variables and return the error rate.

Implementing the two-step procedure iteratively until the number of variables lower than one or the error rate does not decrease (or the error rate increases much which we cannot accept).

Of course, we have to admit that this is a sketch of procedure and more details are needed to be commentated. In next subsection, we will illustrate this procedure on three real-world datasets in detail.

C. Detailed results on three datasets

To explain the above proposed procedure, we first implement the new method MDAMDG on three high dimensional real datasets called lymphoma (62 observations and 4026 variables), prostate (102 observations and 6033 variables), and colon (62 observations and 2000 variables). The detailed results are listed in three tables named Table 1, Table 2 and Table 3. Here, we should denote that the error rate is presented in percentage for better observation. All computational experiments in this paper are performed using R (R 3.2.2) software on a personal laptop equipped with Intel Core i5 2.50 GHz CPU, 4.0GB usable RAM and Microsoft Windows 7 Professional.

TABLE I. THE DETAILED RESULTS ON LYMPHOMA DATASET USING MDAMDG

Iteration	Error Rate (%)	CPU Time(s)	The Number of Selected Variables
0	3.33	2.10	4026
1	3.33	1.05	2013
2	3.11	0.53	1007
3	3.89	0.31	504
4	0	0.21	252
5	0	0.16	126
6	0	0.13	63
7	0	0.12	32
8	0	0.11	16
9*	0*	0.11*	8*
10	4.60	0.11	4

* The word with star represents the best result

TABLE II. THE DETAILED RESULTS ON LYMPHOMA DATASET USING MDAMDG

Iteration	Error Rate (%)	CPU Time(s)	The Number of Selected Variables
0	8.43	6.04	6033
1	8.33	3.12	3017
2	7.20	1.61	1509
3	6.90	0.81	755
4	6.84	0.49	378
5	5.91	0.33	189
6	5.42	0.25	95
7	5.75	0.21	48
8	4.95	0.19	24
9*	4.95*	0.18*	12*
10	6.95	0.18	6

* The word with star represents the best result.

TABLE III. THE DETAILED RESULTS ON COLON DATASET USING MDAMDG

Iteration	Error Rate (%)	CPU Time(s)	The Number of Selected Variables
0	16.90	1.19	2000
1	17.15	0.58	1000
2	16.50	0.33	500
3	13.97	0.19	250
4	12.69	0.13	125
5*	13.58*	0.10*	63*
6	14.27	0.08	32

* The word with star represents the best result.

Let us detail the main procedure of the proposed method.

- Variable ranking. We first run *Random Forest* algorithm on the original variables, and return MDA and MDG value for each variable. Then rank them in increasing orders using MDA and MDG.
- Variable scoring. We score each variable with two grades, one for MDA, and the other for MDG. In MDAMDG method, we consider the two indices as equally important, so the rule of scoring is identical. We give the more important variable a higher score, and the less one a lower score. As for MDA, as mentioned in Section 2.1, a bigger value matches more importance. Thus the variable with high value gains a high score. For instance, if variable X_1 has the biggest MDA value among all variables with a total amount of 2000, then we award it 2000 scores. If variable X_{100} has the smallest value among all variables, then we award it only one score. On condition that two variables have the same value, scores are given according to the ranking respectively, and MDG as well. Then, we compute the total scores according to the calculated two scores as above. At last we reorder the variables using the total scores in decreasing orders.
- Variable selection. Every iteration, we use the dichotomy method to select variables. That is, we choose the first 50 percent variables with the highest scores as the new variables, if the number are decimal, rounding up. As we can see from Tables 1-3, the number of variables is decreased about a half at every iteration. Then run *Random Forest* using the new variables, and return the error rate. If the error rate is not increasing (or increasing few with an acceptable range), we continue the above procedure, and if not, then stop. In this paper, we give a threshold σ (0.01 in this paper), representing a tradeoff between the number of variables and the error rate. If the increase of the error rate does not exceed the threshold, we still continue the procedure. We can see form Table 1 that in the third iteration, the error rate is increased, but we still continue the process, because the error rate increases about 0.78%, which is lower than $\sigma = 0.01$ in this paper. Here, we should denote that we compare the increasing error rate with the minimal error rate, instead of the error rate of the last iteration. For example, in the sixth iteration in table 3, we compare the error rate 14.27% with the forth iteration, not the fifth iteration, and the increasing error rate is 14.27% - 12.69% = 1.58% > 0.01, so we stop.

From the three tables, as the number of iterations increases, the error rate is decreasing (though sometimes with a little increase which is acceptable), and the CPU time and the number of variables are also decreasing. It is easy to find that it only takes a few steps to get the final result due to the dichotomy method. For the worst case, we need log$P$ times at most, where P represents the total number of the original variables. For more comparisons, we also completed the method of SRKNN, proposed by Hee Park & Bum Kim [12], on these three datasets, and the results are shown as in Fig. 1-6.

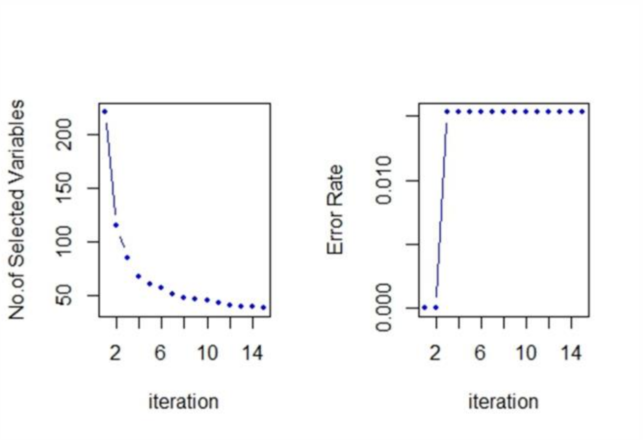

Figure 1. The number of selected variables (left) and the error rate (right) of SRKNN method on lymphoma dataset

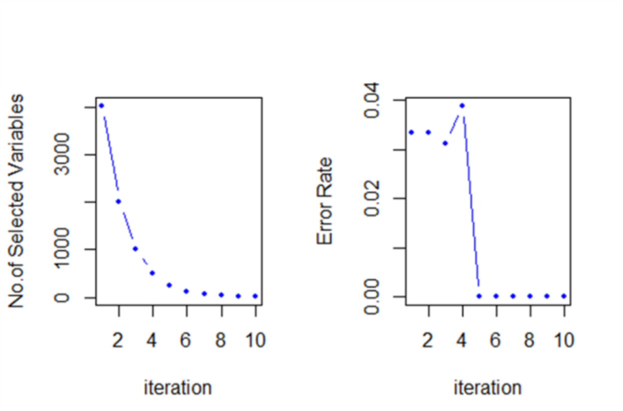

Figure 2. The number of selected variables (left) and the error rate (right) of MDAMDG method on lymphoma dataset

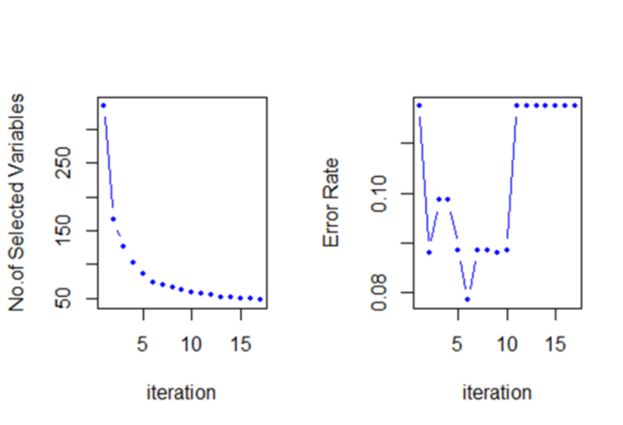

Figure 3. The number of selected variables (left) and the error rate (right) of SRKNN method on prostate dataset.

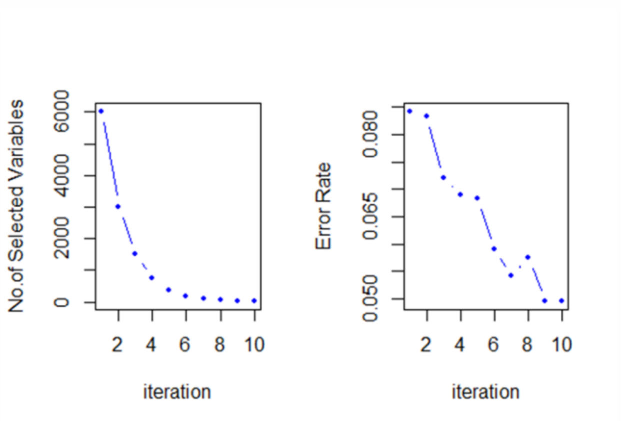

Figure 4. The number of selected variables (left) and the error rate (right) of MDAMDG method on prostate dataset

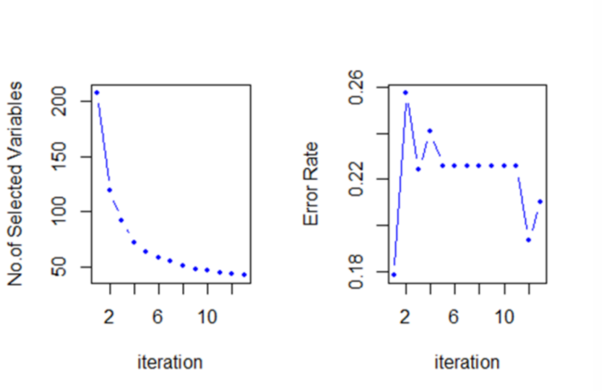

Figure 5. The number of selected variables (left) and the error rate (right) of SRKNN method on colon datase

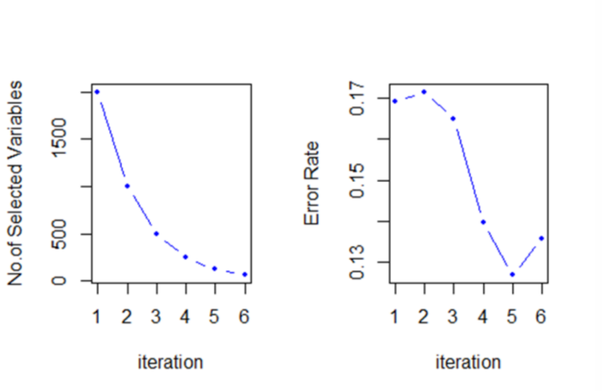

Figure 6. The number of selected variables (left) and the error rate (right) of MDAMDG method on colon datase

From Fig. 1, 3, 5, we can see that as the number of iterations increases, the error rate of SRKNN method is not always decreasing, though the number of selected variables decreasing. What's more, with occasionally error rate increasing, the final result is not necessarily the best one, which can be proved from Fig. 1 and Fig. 3 that the final error rate is the biggest one. That is because SRKNN method is not merely concerned about the error rate or accuracy, but the number of selected variables is its final termination criterion. But MDAMDG method mainly focuses on error rate with considering the number of selected variables. So in Fig. 2, 4, 6, it can be observed that with the iteration increasing, error rate is decreasing (though sometimes increasing but it is acceptable according to the threshold), the number of selected variables is also decreasing. Although the number of selected variables is more than the SRKNN method in the first few iterations, the number of final selected variables are smaller than SRKNN in most cases. Readers can refer to Table 1-4.

III. EXPERIMENTS

A. Datasets and parameter setting

To identify the proficiency of our proposed method, we performed our method on 10 high dimensional datasets involving thousands of features, which are also used in SRKNN method proposed by Hee Park & Bum Kim [12] and Genuer, Poggi & Tuleau-Malot [4]. The URL of these datasets can be obtained from [12], in this paper, we use the second URL of the paper listed.

The MDAMDG method does not need to set many parameters. The only parameter we need to set is σ, the acceptable increasing error rate, which is set to be 0.01 in this paper. Every time we run the *Random Forest*, we use the R package default value of *mtry* (the number of input variables randomly chosen at each split), *ntree* (the number of trees in the forest), i.e. *mtry* are the arithmetic square root of *P* for classification and *P*/3 for regression (*P* is the total number of the original variables), and *ntree* are 500.

B. Experimental results and comparison

For more persuasive, we perform 5-fold cross validation and run 30 times to calculate the classification error rate on 10 datasets, and averaged error rate is returned as the final value both on our proposed method and SRKNN method. The results are shown in Table 4. For comparison, we also listed the SRKNN result and the method of random forest, which is mentioned in the paper of SRKNN. In order to keep consistency, we also keep two decimals and the error rate of SRKNN and Random Forest we use the results of [12].

TABLE IV. COMPARATIVE PERFORMANCES WITH VARIABLE SELECTION METHODS USING MDAMDG, SRKNN AND RANDOM FOREST IN TERMS OF ERROR RATE

Dataset	MDAMDG	SRKNN	Random Forest
Nci	0.07 (41) *	0.36 (33)	0.38 (60)
Brain_tumor2	0 (11) *	0.22 (23)	0.24 (100)
Adenocarcinoma	0.04 (5) *	0.05 (13)	0.08(5)
Lymphoma	0 (8) *	0.02 (20)	0 (79) *
Leukemia	0 (1) *	0.04(25)	0.03 (2)
Breast3	0.17 (39) *	0.43 (44)	0.33 (10)
Breast2	0.11 (10) *	0.46 (15)	0.34 (5)
Prostate	0.05 (12) *	0.10 (23)	0.07 (5)
SRBCT	0 (19) *	0.02 (35)	0.02 (22)
Colon	0.14 (63) *	0.18 (18)	0.15 (4)

* The word with star represents the best result and the number of selected variables is shown in parentheses

Table 4 indicates that our proposed MDAMDG method is more powerful than SRKNN and Random Forest. In each dataset, the error rate of our proposed method is the smallest. For lymphoma dataset, although the error rate of MDAMDG is the same as Random Forest, but the number of final selected variables is much smaller than Random Forest. Besides, the number of final selected variables in MDAMDG method is smaller than SRKNN and Random Forest in most cases.

C. A regression example

In order to identify our proposed method is also suitable for regression problems, we apply it to an ozone dataset, which is also used in [4]. This dataset can be obtained from the R package mlbench. This dataset has 336 observations and 12 variables, but it has 123 missing observations, so the final samples used in our method are 203 observations. The result is shown in Table 5.

TABLE V. THE RESULT ON OZONE DATASET USING %INCMSE AND INCNODEPURITY

Iteration	MSE	Number of variables
0*	5.56*	12 (all variables) *
1	17.76	6 (V9, V8, V12, V1, V5, V11)
2	16.58	5 (V9, V8, V12, V1, V11)
3	24.90	3 (V9, V8, V12)

* The word with star represents the best result and the selected variables are shown in parentheses.

From Table 5, it can be indicated that the best choice is not to reduce variables. The original is the best for that the number of original variables are not very large, which have only 12 variables. For comparison, we listed the selected variables in each iteration, and the first five important variables are same as Genuer, Poggi & Tuleau-Malot [4].

IV. CONCLUSIONS AND PERSPECTIVES

In this paper, we proposed a new method to select variables using Random Forest, which is also suitable for regression problems. Experiments on 10 microarray datasets show that MDAMDG method is very powerful and robust as a result of using two significant indices. In addition, we use dichotomy to screen variables, which is very fast and controllable. Besides, no redundant parameters need to be set except for a tradeoff between the number of variables and the error rate. But the optimal rule to set it deserves further research. So we will focus on optimizing the parameter σ in future researches. Moreover, we will search for more efficient methods to select variables than MDAMDG method as well.

ACKNOWLEDGMENT

This work was generously supported by the National Natural Science Foundation of China Grant Numbers 70971128 and 71450009. The authors would like to thank two anonymous reviewers for their insights and suggestions for improving this manuscript.

REFERENCES

[1] I. Guyon and A. Elisseeff, An Introduction to Variable and Feature Selection. Journal of Machine Learning Research, 2003, 3(6): 1157-1182.

[2] L. Breiman, Random forests. Machine Learning, 2001, 45(1): 5-32.

[3] C. Strobl, A.-L. Boulesteix, T. Kneib, T. Augustin and A. Zeileis, Conditional variable importance for random forests. BMC Bioinformatics, 2008, 9(14): 1-11.

[4] R. Genuer, J. M. Poggi and C. Tuleau-Malot, Variable selection using random forests. Pattern Recognition Letters, 2010, 31(14): 2225-2236.

[5] A.B. Ishak, Variable selection using support vector regression and random forest: A comparative study. Intelligent Data Analysis, 2016, 20(1): 83-104.

[6] S. Janitza, G. Tutz and A.-L. Boulesteix, Random forest for ordinal responses: Prediction and variable selection. Computational Statistics and Data Analysis, 2016, 96: 57-73.

[7] A. Hapfelmeier and K. Ulm, A new variable selection approach using Random Forests. Computational Statistics and Data Analysis, 2013, 60(4): 50-69.

[8] Q. Zhou, H. Zhou and L. Tao, Cost-sensitive feature selection using random forest: Selecting low-cost subsets of informative features. Knowledge-Based Systems, 2016, 95: 1-11.

[9] S. Walter and H. Tiemeier, Variable selection: current practice in epidemiological studies. European Journal of Epidemiology, 2009, 24(12): 733-736.

[10] S. Greenland, Invited Commentary: Variable Selection versus Shrinkage in the Control of Multiple Confounders. American Journal of Epidemiology, 2008, 167(5): 523-529.

[11] K.L. Lunetta, L. Hayward, J. Segal and P.V. Eerdewegh, Screening large-scale association study data: exploiting interactions using random forests. BMC Genetics, 2004, 5(2): 1-13.

[12] C.H. Park and S.B. Kim, Sequential random k-nearest neighbor feature selection for high-dimensional data. Expert Systems with Applications, 2015, 42(5): 2336-2342.

[13] M.H. Nguyen and F. de la. Torre, Optimal feature selection for support vector machines. Pattern Recognition, 2010, 43(3): 584-591.

[14] L. Breiman and A. Cutler, Random forests. Berkeley, 2005.

[15] A. Liaw and M. Wiener, Classification and regression by random forest. R News, 2002, 2(3): 18-22.