BIOINFORMATION

open access

Supplementary material:

Performance Measurement:

$$\left(1-\frac{1}{n}\right)^{n} \approx e^{-1} \approx 0.368$$

$$acc_{boot} = \frac{1}{b}\sum_{i=1}^{b}(0.632 \cdot c0_i + 0.368 \cdot acc_s)$$

Where acc_s is the resubstitution error estimate on the full dataset (the error on the training set).

Table 1: Main characteristics of the microarray datasets used

Dataset	Genes	Patients	Classes	Reference
Adenocarcinoma	9868	76	2	**[6]**
Brain	5597	42	5	**[7]**
Breast2	4869	77	2	**[8]**
Breast3	4869	95	3	**[8]**
Colon	2000	62	2	**[9]**
Leukemia	3051	38	2	**[10]**
Lymphoma	4026	62	3	**[11]**
NCI60	5244	61	8	**[12]**
Prostate	6033	102	2	**[13]**
SRBCT	2308	63	4	**[14]**

Table 2: Error rates estimated using. 632 bootstrap for different methods.

Dataset Name	SVM	KNN	DLDA	Current Method
Adenocarcinoma	0.203	0.174	0.194	0.1629
Brain	0.138	0.174	0.183	0.1803
Breast2	0.325	0.337	0.331	0.3282
Breast3	0.380	0.449	0.370	0.3318
Colon	0.147	0.152	0.137	0.1243
Leukemia	0.014	0.029	0.020	0.0582
Lymphoma	0.010	0.008	0.021	0.0380
NCI60	0.256	0.317	0.286	0.2729
Prostate	0.064	0.100	0.149	0.0554
SRBCT	0.017	0.023	0.011	0.0239

Automated Dataset Input Function:

In the current R package for random forest gene selection, the dataset format for input as well as processing is not mentioned and cause severe confusion to the users. Besides that, the method for inputting the dataset which is mostly in text file format required further processing to cater to the function parameters and format for usability of the gene selection process. Therefore, an automated dataset input and formatting functions has been created to ease the access of loading and using the dataset of the microarray gene expression based on text files input. The standard dataset format used for this package has two separate text files, which are data file and class file. These files need to be inputted into the R environment before further processing can be done. The method and steps for the automated dataset input is described in **Figure 2**. The steps have been created as an R function which is included inside the package and can be used directly for the loading of the dataset. The function takes two parameters, which are the data file name with extension and class file name with extension.

Step 1: Input data name and class name.
Step 2: Error checking for valid file name, extension and file existence.
Step 3: Read data file into R workspace.
Step 4: Data processing.
Step 5: Transpose data.
Step 6: Read length of class/sample.
Step 7: Read class file into R workspace.
Step 8: Create class factor.
Step 9: Load both data and class for function variable access.

Figure 2: Steps required for the automated dataset input and formatting in R environment

Selection of Smallest Subset of Genes with Lowest OOB Error Rates:

```
While backward elimination process = TRUE
        If current OOB error rates < = previous OOB error rates
                Set lowest error rate as current OOB error rates
                Set no of variables selected
        End If
End While
```

Figure 3: Method used for tracking and storing the lowest OOB error rates

Selection of Biggest Subset of Genes with Lowest OOB Error Rates:

```
While looping all the subset with lowest OOB error rate
        If Current no of selected genes >= Previous no of selected genes
                Set Biggest subset = Current number of selected genes
        End If
End While
```

Figure 4: Method used for selecting the biggest subset of genes with lowest OOB error rates

Setting the Minimum Number of Genes to Be Selected:

The input for the minimum number of genes to be selected during the gene selection process is merged with the existing functions as an extra parameter input that has a default value of 2. The selected minimum values are used during the backward elimination process which takes place in determining the best subset of genes based on out of bag (OOB) error rates. At each time of a loop for selecting the best subset of genes, random forest backward elimination of genes is carried out by removing the unwanted genes gradually at each loop based on the *fraction.dropped* values selected. Therefore, as the no of loop increases, the no of genes in the subset decreases leaving the most informative genes inside the subset, as less informative genes are removed. The minimum no of genes specified is checked at each loop and if the total number of genes for a subset is less than the specified value, the loop is terminated leaving behind all the subsets.

```
While backward elimination process
        If length of variables <= to minimum required variables
                Break
```

Figure 5: Method used for terminating the loop once the desired number of variables achieved

Results Comparison:

Further evaluation and comparison has been carried out to analyze the microarray data with other existing methods such as Diagonal Linear Discriminant Analysis (DLDA), K nearest neighbor (KNN) and Support Vector Machines (SVM) with Linear Kernel. These three methods are carried out without the use of variable selection. The error rates comparison between these methods with the improved Random Forest gene selection method (EvarSelRF) is presented in the **Table 2.**