

Package ‘RandForestGUI’

February 14, 2012

Title RandForestGUI

Version 1.0

Date 2010-07-26

Author Rory Michelland <rory.michelland@gmail.com>, Genevieve Grundmann <grundman@biomserv.univ-lyon1.fr>

Maintainer Rory Michelland <rory.michelland@gmail.com>

Description A graphical user interface program to analyze microbial profiles (taxonomic sequences, fingerprints and microarrays) based on conditional inference random forest

Depends

party,randomForest,RColorBrewer,tcltk,tcltk2,MASS,vegan,Hmisc,clusterSim,rpart,car,sciplot

License GPL-2

Repository CRAN

Date/Publication 2010-08-27 06:23:37

R topics documented:

Internal functions	2
RandForestGUI	2

Index	5
--------------	----------

Internal functions *Internal functions for the RandForestGUI package*

Description

Internal functions for the RandForestGUI package

Details

These are not to be called by the user

Author(s)

Rory Michelland <rory.michelland@gmail.com>

See Also

[RandForestGUI](#)

RandForestGUI *Graphical user interface of RandForestGUI package, a program to analyze microbial profile (taxonomic sequences, fingerprints and microarrays) based on conditional inference random forest*

Description

A Tcl/Tk GUI to use functions of the RandForestGUI package.

Menu bar

The menu bar at the top of the window is used to launch all the functions of the package.

The Management menu is used to import microbial profile and their corresponding categorical variables (ordinal and nominal), save and load a RandForestGUI project, and select microbial profiles :

- Import microbial profiles is used to import microbial profiles table present either in an external ASCII file (.txt, .csv etc) or in R environment as an object.
- Import categorical variables is used to import categorical variables corresponding to the microbial profiles. These categorical variables can be present either in an external ASCII file (.txt, .csv etc) or in R environment as an object.
- Save is used to save or save as a RandForestGUI project in a specific directory.
- Load is used to import a RandForestGUI project previously saved.

- Selection of microbial profiles is used to select specific microbial profiles to perform random forest.

The Unsupervised random forest menu is used to perform conditional inference unsupervised random forest and determine clusters with K-means and PAM (Partitioning Around Medoids):

- Choose unsupervised mode permits to choose between the Addcl1 sampling method (synthetic data are added by randomly sampling from the product of empirical marginal distributions of the variables) and Addcl2 sampling method (synthetic data are added by randomly sampling from the hyperrectangle that contains the observed data, that is, the variables of synthetic observations have a uniform distribution with range determined by the minimum and maximum of the corresponding observed variable).

- Determine the best mtry permits to determine the best mtry for the unsupervised random forest analysis on your data set. The accuracy of random forest depends of the mtry value.

- Calculate and plot unsupervised random forest is a generic function to calculate and/or to plot unsupervised random forest.

- Determine microbial cluster using K-means or PAM and Davies Bouldin index determines microbial clusters with the k-means or PAM procedures and the best number of microbial clusters using Davies Bouldin index.

The Supervised random forest menu is used to perform conditional inference supervised random forest and calculate taxonomic units importances:

- Determine the best mtry permits to determine the best mtry for the supervised random forest analysis on your data set. The accuracy of random forest depends of the mtry value.

- Calculate and plot supervised random forest is a generic function to calculate and/or to plot supervised random forest.

- taxonomic unit importance calculates the mean decrease accuracy of taxonomic units.

- Significance of taxonomic units importance calculates p values of taxonomic units mean decrease accuracy using Monte Carlo permutations.

- Backward taxonomic units selection performed a backward selection method by iteratively deleting the less important taxonomic units according to their mean decrease in accuracy and at each step calculating a new random forest. This calculation is needed to choice the top sufficient taxonomic units of the next menu 'Supervised random forest with probes selection'

The Supervised random forest with probes selection menu is used to perform conditional inference supervised random forest with the top sufficient taxonomic units:

- Top sufficient taxonomic units determines a small set of taxonomic units sufficient to discriminate microbial clusters. It leads to identify taxonomic units specifics to each cluster. The random forest model with the lowest number of taxonomic units and the lowest out of bag was retained. To comfort the robustness of this approach, two subsequent analyses were performed. A Pearson correlation was calculated between the initial random forest distances (all taxonomic units) and the new random forest distances and a filtered method using with the Kruskal Wallis test performed for each selected taxonomic unit. The choice of the number of taxonomic units is determined according to the lowest out of bag and the best correlation.

- Best mtry determination permits to determine the best mtry for the supervised random forest analysis with the to sufficient taxonomic units. The accuracy of random forest depends of the mtry value.

- Calculate/plot supervised random forest is a generic function to calculate and/or to plot supervised random forest with the top sufficient taxonomic units.

- information about top sufficient taxonomic units is a generic function usefull to obtain mean, standard deviation and dominance index of top sufficient taxonomic units in each microbial clusters.

The Supplementary analysis menu provides helpfull supplementary analysis.

- Diversity calculates several diversity index (Richness, Number of individual, Dominance, Shanon, Simpson, Eveness, Equitability and berger parker dominance)

- Decision tree calculates a conditional inference decision based on the top sufficient taxonomic units (see ctree function from party package).

The Miscellaneous menu is composed of various tools to improve the use of RandForestGUI package.

- Language allows languages to be changed. For instance, only USA english, french, german and Italian can be selected.

- Memory increase permits to increase the RAM memory allocated to R. If RandForestGUI frequently bugs, this function is most of time the solution!

- Background plot colors permits to change the background color of RandForestGUI plots.

Author(s)

Rory Michelland <rory.michelland@gmail.com>

Examples

```
## Not run:
## One of these lines allows the StatFingerprints program to be started
RandForestGUI()
RandforestGUI()
randforestGUI()
randForestGUI()
Randforestgui()
randforestgui()
randForestgui()
RandforestGui()
randforestGui()
randForestGui()
RF()
rf()
Rf()

## End(Not run)
```

Index

- *Topic **microarray,sequence,fingerprint,random for-est,multivariate,conditional inference**
 - Internal functions, 2
- choicefactGUI (Internal functions), 2
- choiceGUI (Internal functions), 2
- cluster_det_URF (Internal functions), 2
- correlation.RF.init.vs.fitted (Internal functions), 2
- Davies_Bouldin_URF (Internal functions), 2
- dev_delete (Internal functions), 2
- diversity.calc.RF (Internal functions), 2
- fig_par (Internal functions), 2
- graphfact.RF (Internal functions), 2
- heat.mapGUI.RF (Internal functions), 2
- import_variable.RF (Internal functions), 2
- Internal functions, 2
- language.RF (Internal functions), 2
- legend_color_scale (Internal functions), 2
- loadGUI.RF (Internal functions), 2
- MDA.RF (Internal functions), 2
- mds.fitted.RF (Internal functions), 2
- mds.RF (Internal functions), 2
- mds.URF (Internal functions), 2
- memory.increase (Internal functions), 2
- misclassification_ini (Internal functions), 2
- new.SRF.OOB (Internal functions), 2
- OOB.RF (Internal functions), 2
- perm.MDA.init (Internal functions), 2
- plot_adjust.numb.var (Internal functions), 2
- plot_univ (Internal functions), 2
- RandForestGUI, 2, 2
- RandforestGUI (RandForestGUI), 2
- RandforestGui (RandForestGUI), 2
- Randforestgui (RandForestGUI), 2
- randForestGUI (RandForestGUI), 2
- randForestGui (RandForestGUI), 2
- randForestgui (RandForestGUI), 2
- randforestGUI (RandForestGUI), 2
- randforestGui (RandForestGUI), 2
- randforestgui (RandForestGUI), 2
- RF (RandForestGUI), 2
- Rf (RandForestGUI), 2
- rf (RandForestGUI), 2
- RISA2RF.RF (Internal functions), 2
- saveGUI.RF (Internal functions), 2
- selection.RF (Internal functions), 2
- several_URF (Internal functions), 2
- SRF.fin.tune.mtryGUI (Internal functions), 2
- SRF.tune.mtryGUI (Internal functions), 2
- SRF_fitted_rpart (Internal functions), 2
- SRF_init_rpart (Internal functions), 2
- SRF_init_VarImp (Internal functions), 2
- supRFGUI.fitted.RF (Internal functions), 2
- supRFGUI.RF (Internal functions), 2
- tk_select (Internal functions), 2
- unsup.mode.RF (Internal functions), 2
- URF_tune_mtry (Internal functions), 2

`waitGUI` (Internal functions), [2](#)