Package 'gaucho'

April 14, 2017

Type Package

Title Genetic Algorithms for Understanding Clonal Heterogeneity and Ordering

Version 1.10.0

Date 2014-04-08

Description Use genetic algorithms to determine the relationship between clones in heterogenous populations such as cancer sequencing samples

biocViews Software,Genetics,SNP,Sequencing,SomaticMutation

VignetteBuilder knitr

Depends R (>= 3.0.0), compiler, GA, graph, heatmap.plus, png, Rgraphviz

Suggests knitr

License GPL-3

Author Alex Murison [aut, cre], Christopher Wardell [aut, cre]

Maintainer Alex Murison <Alexander.Murison@icr.ac.uk>, Christopher Wardell <Christopher.Wardell@icr.ac.uk>

NeedsCompilation no

R topics documented:

BYB1_G07_pruned *BYB1_G07_pruned*

Description

A real data set taken from Lang et al, 2013 (PMID:23873039). See the accompanying vignette for more details.

Format

A `data.frame` with 12 genes (rows) measured at 11 time points (columns)

Author(s)

Alex Murison <`Alexander.Murison@icr.ac.uk`> and Christopher Wardell <`Christopher.Wardell@icr.ac.uk`>

See Also

`ga-class`, `ga`, `gauchoReport`, `gaucho_simple_data`, `gaucho_hidden_data`, `gaucho_synth_data`, `gaucho_synth_data_jittered`, `BYB1_G07_pruned`

gaucho *Genetic Algorithm for Understanding Clonal Heterogeneity and Ordering (GAUCHO)*

Description

Use a genetic algorithm to find the relationships between the values in an input file - the package was written to deal with single nucleotide variants (SNVs) in mixtures of cancer cells, but it will work with any mixture. It will calculate appropriate phylogenetic relationships between clones them and the proportion of each clone that each sample is composed of. For detailed usage, please read the accompanying vignette.

Usage

```
gaucho(observations, number_of_clones, pop_size = 100, mutation_rate = 0.8,
  iterations = 1000, stoppingCriteria = round(iterations/5),
  parthenogenesis = 2, nroot = 0, contamination = 0,
  check_validity = TRUE)
```

Arguments

`observations` Observation data frame where each row represents an SNV and each column represents a discrete sample separated by time or space. Note that the data frame must have column names and row names. Every value must be a proportion between 0 and 1. See details

`number_of_clones` An integer number of clones to be considered

`pop_size` The number of individuals in each generation `mutation_rate` The likelihood of each individual undergoing mutation per generation

`iterations` The maximum number of generations to run

`stoppingCriteria`
The number of consecutive generations without improvement that will stop the algorithm. Default value is 20% of iterations.

`parthenogenesis`
The number of best-fitness individuals allowed to survive each generation

`nroot` Number of roots the phylogeny is expected to have.When nroot=0, a random integer between 1 and the number of clones is generated for each phylogeny

`contamination` Is the input contaminated? If set to 1, an extra clone is created in which to place inferred contaminants

`check_validity` Unless set to false, eliminate any clones with no new mutations, disallow those clones. Increases computational overheads.

Details

The input data should be a data.frame containing proportions of cells that contain a feature. There are a number of ways to create these data, including merging the balance of alleles and copy number of an SNV using the equation min(1,r*CN/(r+R)), where CN is the copy number, r is the number of non-reference reads and R is the number of reference reads. For example, if a site were sequenced to a depth of 100x, with 25 non-reference reads and 75 reference reads and diploid copy number, the result would be min(1,25*2/(25+75)) = 0.5. Therefore, 50% of the cells in the sample contain the SNV. Further details are available in the accompanying vignette.

Value

Returns an object of class ga `ga-class`. Note that the number of clones and number of cases are stored in the unused min and max slots of the output object.

Author(s)

Alex Murison <`Alexander.Murison@icr.ac.uk`> and Christopher Wardell <`Christopher.Wardell@icr.ac.uk`>

See Also

`ga-class`, `ga`, `gauchoReport`, `gaucho_simple_data`, `gaucho_hidden_data`, `gaucho_synth_data`, `gaucho_synth_data_jittered`, `BYB1_G07_pruned`

Examples

```
## The vignette provides far more in-depth explanation and examples ##

## Load the included simple example data
gaucho_simple_data = read.table(file.path(system.file("extdata",package="gaucho"),"gaucho_simple_data.txt")

## Run gaucho using 3 clones and a phylogeny with a single root
solution=gaucho(gaucho_simple_data, number_of_clones=3,nroot=1,iterations=1000)

## Create the four output plots
gauchoReport(gaucho_simple_data,solution,outType="fitness")
gauchoReport(gaucho_simple_data,solution,outType="heatmap")
gauchoReport(gaucho_simple_data,solution,outType="phylogeny")
gauchoReport(gaucho_simple_data,solution,outType="proportion")
```

```
## Output the solution and plots in the current working directory
# gauchoReport(gaucho_simple_data,solution)
```

---

gauchoReport *View solutions contained within gaucho output*

Description

After running gaucho() on data, this function provides a convenient way to view the solutions and also export them as separate text files and images. For detailed usage, please read the accompanying vignette.

Usage

```
gauchoReport(gauchoInput, gauchoOutput, outType = "complete",
  yRange = c(-250, 0), output_file_prefix = "")
```

Arguments

`gauchoInput`	Raw data analysed by gaucho()
`gauchoOutput`	Object of class ga produced by gaucho()
`outType`	Type of output desired - must be one of the following: "complete","fitness","heatmap","phylogeny","p
`yRange`	Y-axis range when plotting fitness of individuals. Default is c(-250,0)
`output_file_prefix`	Optional prefix for all output files

Details

This method reports data for the fittest individual; in the event of there being multiple individuals with identical fitness, up to five individuals will be reported. This function's output is governed by the outType argument. All options except for the default "complete" value result in plotting the desired output to the current R session. When outType=="complete", the following output is created for each individual: the full length string, the phylogeny matrix, the proportion matrix, the presence matrix, a heatmap of the raw data with the assigned clones as coloured bars at the side, a stacked barplot showing the proportion of each clone at each timepoint and a plot showing the phylogenetic relationship between the clones. Note that the colours of the clones are consistent across all plots and that the contamination clone (if present) is always the last clone. Also produced is a plot illustrating the change in fitness as the generations evolved.

Value

Nothing is returned.

Author(s)

Alex Murison <Alexander.Murison@icr.ac.uk> and Christopher Wardell <Christopher.Wardell@icr.ac.uk>

See Also

ga-class, ga, gauchoReport, gaucho_simple_data, gaucho_hidden_data, gaucho_synth_data, gaucho_synth_data_jittered, BYB1_G07_pruned

Examples

```
## The vignette provides far more in-depth explanation and examples ##

## Load the included simple example data
gaucho_simple_data = read.table(file.path(system.file("extdata",package="gaucho"),"gaucho_simple_data.txt")

## Run gaucho using 3 clones and a phylogeny with a single root
solution=gaucho(gaucho_simple_data, number_of_clones=3,nroot=1,iterations=1000)

## Create the four output plots
gauchoReport(gaucho_simple_data,solution,outType="fitness")
gauchoReport(gaucho_simple_data,solution,outType="heatmap")
gauchoReport(gaucho_simple_data,solution,outType="phylogeny")
gauchoReport(gaucho_simple_data,solution,outType="proportion")

## Output the solution and plots in the current working directory
# gauchoReport(gaucho_simple_data,solution)
```

gaucho_hidden_data *gaucho_hidden_data*

Description

A synthetic data set provided to illustrate how to use gaucho. See the accompanying vignette for more details.

Format

A `data.frame` with 5 mutations (rows) measured at 3 time points (columns)

Author(s)

Alex Murison <`Alexander.Murison@icr.ac.uk`> and Christopher Wardell <`Christopher.Wardell@icr.ac.uk`>

See Also

`ga-class`, `ga`, `gauchoReport`, `gaucho_simple_data`, `gaucho_hidden_data`, `gaucho_synth_data`, `gaucho_synth_data_jittered`, `BYB1_G07_pruned`

gaucho_simple_data *gaucho_simple_data*

Description

A very simple synthetic data set provided to illustrate how to use gaucho. See the accompanying vignette for more details.

Format

A `data.frame` with 3 genes (rows) measured at 3 time points (columns)

Author(s)

Alex Murison <Alexander.Murison@icr.ac.uk> and Christopher Wardell <Christopher.Wardell@icr.ac.uk>

See Also

ga-class, ga, gauchoReport, gaucho_simple_data, gaucho_hidden_data, gaucho_synth_data, gaucho_synth_data_jittered, BYB1_G07_pruned

gaucho_synth_data *gaucho_synth_data*

Description

A synthetic data set provided to illustrate how to use gaucho See the accompanying vignette for more details.

Format

A data.frame with 90 genes (rows) measured at 4 time points (columns)

Author(s)

Alex Murison <Alexander.Murison@icr.ac.uk> and Christopher Wardell <Christopher.Wardell@icr.ac.uk>

See Also

ga-class, ga, gauchoReport, gaucho_simple_data, gaucho_hidden_data, gaucho_synth_data, gaucho_synth_data_jittered, BYB1_G07_pruned

gaucho_synth_data_jittered *gaucho_synth_data_jittered*

Description

A synth data set provided to illustrate how to use gaucho. See the accompanying vignette for more details.

Format

A data.frame with 90 genes (rows) measured at 4 time points (columns), with added noise.

Author(s)

Alex Murison <Alexander.Murison@icr.ac.uk> and Christopher Wardell <Christopher.Wardell@icr.ac.uk>

See Also

ga-class, ga, gauchoReport, gaucho_simple_data, gaucho_hidden_data, gaucho_synth_data, gaucho_synth_data_jittered, BYB1_G07_pruned

Index