Package 'MSCsimtester'

August 2, 2019

Type Package

Title Multispecies coalescent gene tree simulator tests

Version 1.0.0

Maintainer The package maintainer <e.allman@alaska.edu>

Description Functions for testing multispecies coalescent gene tree simulators using pairwise distances. See E.S. Allman, H. Banos, and J.A. Rhodes, 'Testing Multispecies Coalescent Simulators Using Summary Statistics,' 2019.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Depends R (>= 2.10), ape (>= 5.0), kSamples, stats

RoxygenNote 6.1.1

R topics documented:

`ADtest` *Perform Anderson-Darling tests comparing sample and and theoretical pairwise distance distributions.*

Description

Takes as input theoretical pairwise distance densities under the MSC and empirical pairwise distances from gene trees in a sample, as returned by the function `pairwiseDist`. Uses the package `kSamples` to perform either one test on the entire dataset or multiple tests on subsamples.

Usage

```
ADtest(distanceDensities, subsampleSize = FALSE)
```

Arguments

distanceDensities
: A list containing values needed for performing Anderson-Darling test(s) on a gene tree sample, as output by pairwiseDist. The first entry of this list is sampleDist, a vector with entries the pairwise distances computed from gene trees in a sample. This can be useful to practitioners. Other entries in this list are documented in the code for pairwiseDist.

subsampleSize
: A positive integer or FALSE. Default is FALSE, to use full sample for one test.

Details

Anderson-Darling tests can perform poorly due to numerical issues when the sample size N is too large, so an optional parameter subsampleSize can be set to create subsamples of smaller size. If subsampleSize is a positive integer, the sample of N gene trees is partitioned into sets of size subsampleSize. Anderson-Darling tests are performed on each subset, comparing them to a random sample of the same size from the theoretical distribution.

For a single test, output of test statistics and p-values are given as text. For multiple tests, output is given as a histogram of p-values. Assuming good fit, the histogram should be approximately uniform.

The Anderson-Darling test compares the empirical distance distribution for the supplied gene tree sample to a sample drawn from the theoretical distribution. The output from the kSamples package will thus say 2 samples are being compared, to test a null-hypothesis that they come from the same distribution. See kSamples documentation for more details. Repeated runs of this function will give different results, since the sample from the theoretical distribution will vary from run to run. Under the null hypothesis p-values for different runs should be approximately uniformly distibuted.

Value

Invisibly returns a sample from the theoretical density, of the same size as the empirical sample.

See Also

pairwiseDist, kSamples-package

Examples

```
stree=read.tree(text="((((a:10000,b:10000):10000,c:20000):10000,d:30000):10000,e:40000);")
pops=c(15000,25000,10000,1,1,1,1,1,12000)
gts=read.tree(file=system.file("genetreeSample",package="MSCsimtester"))
distDen=pairwiseDist(stree,pops,gts,"a","b")
ADtest(distDen)
ADtest(distDen,1000)
```

edgeOrder *Plot species tree, with edge numbers on edges.*

Description

Under the MSC, each edge in the species tree must be assigned a population size. This function displays the species tree with the edges numbered, to aid the user in entering constant population sizes as an appropriately ordered list.

Usage

```
edgeOrder(stree)
```

Arguments

`stree` An object of class `phylo` containing a rooted metric species tree.

See Also

`pairwiseDist`, `rootedTriple`, `plotPops`

Examples

```
stree=read.tree(text="(((a:10000,b:10000):10000,c:20000):10000,d:30000);")
edgeOrder(stree)
pops=c(30000,20000,1,1,1,1,10000)
plotPops(stree,pops)
```

`MSCsimtester` *Functions to test whether simulators of the multspecies coalescent model in phylogenomics give valid gene tree samples.*

Description

The package performs comparisons of certain summary statistics for simulated gene tree samples to theoretical predictions under the multspecies coalescent model. The primary functions are `rootedTriple` for comparison of frequencies of topological rooted triples on gene trees, and `pairwiseDist` and `ADtest` for comparison of the distributions of pairwise distances between taxa on gene trees.

Details

`MSCsimtester` builds on the packages `ape` and `kSamples`.

Required input is a collection of gene trees, stored as a `multiPhylo` object by the `ape` package, and a specification of a rooted species tree with edge lengths in generations, together with constant population sizes for each edge.

For further examples of use and citation purposes, see E.S. Allman, H. Banos, and J.A. Rhodes, 'Testing Multispecies Coalescent Simulators Using Summary Statistics,' 2019.

pairwiseDist *Compute and plot pairwise distance densities.*

Description

Computes theoretical pairwise distance densities under the MSC and empirical pairwise distances from gene trees in a sample. A histogram of empirical values is plotted over the theoretical pdf.

Usage

```
pairwiseDist(stree, popSizes, gtSample, taxon1, taxon2, numSteps = 1000,
  tailProb = 0.01)
```

Arguments

`stree`	An object of class `phylo` containing a rooted metric species tree. Edge lengths are in number of generations.
`popSizes`	An ordered list containing constant population sizes, one entry for each population in the species tree, for a haploid population. Sizes should be doubled for diploids. If `stree` has k edges, then `popSizes` must have k+1 elements, with final entry the size of the population ancestral to the root.
`gtSample`	An object of class `multiPhylo` holding a sample of gene trees from a simulation. Taxon labels on gene trees must be identical to those on `stree`.
`taxon1`	A string specifying one taxon on `stree`.
`taxon2`	A string specifying a second taxon on `stree`, distinct from `taxon1`.
`numSteps`	A positive integer giving the number of values on the x-axis to be sampled for creating and graphing the theoretical pairwise distance density. Default is `numSteps = 1000`. Increasing this value will smooth the plot of the theoretical pairwise distance density.
`tailProb`	The theoretical pairwise distance will be plotted from (0, xMax). The maximum value xMax is either the maximum pairwise distance in the gene tree sample or the x-value subtending a probability of `tailProb` under the pdf. Default is .01.

Details

`numSteps` equally spaced points will be sampled for creating the theoretical pairwise distance density. Default is `numSteps = 1000`.

Value

A list of items needed for Anderson-Darling test(s), for use by `ADtest`. See function code for more details.

See Also

`edgeOrder`, `plotPops`, `ADtest`

Examples

```
stree=read.tree(text="((((a:10000,b:10000):10000,c:20000):10000,d:30000):10000,e:40000);")
pops=c(15000,25000,10000,1,1,1,1,1,12000)
gts=read.tree(file=system.file("genetreeSample",package="MSCsimtester"))
pairwiseDist(stree,pops,gts,"a","b")
```

---

plotPops — *Plot species tree, with population sizes on edges.*

Description

Plot species tree, with population sizes on edges.

Usage

```
plotPops(stree, populations)
```

Arguments

stree	An object of class phylo containing a rooted metric species tree.
populations	A vector of type numeric containing the population sizes for the edges, with last entry the population ancestral to the root.

See Also

pairwiseDist, rootedTriple, edgeOrder

Examples

```
stree=read.tree(text="(((a:10000,b:10000):10000,c:20000):10000,d:30000);")
edgeOrder(stree)
pops=c(30000,20000,1,1,1,1,10000)
plotPops(stree,pops)
```

---

rootedTriple — *Compare expectated frequencies of topological rooted triples under the MSC to empirical ones in a sample.*

Description

For a given species tree with population sizes, this function compares the expected frequencies of rooted triples to empirical frequencies in a sample of gene trees using Chi^2 tests with 2 d.f. The exact and estimated internal branch length (in coalescent units) of the rooted triple in the species tree are also computed for comparison. A single test can be performed on the entire gene tree sample, or multiple tests on subsamples.

Usage

```
rootedTriple(stree, popSizes, gtSample, taxon1, taxon2, taxon3,
  subsampleSize = FALSE)
```

Arguments

stree	An object of class phylo containing a rooted metric species tree. Edge lengths are in number of generations.
popSizes	An ordered list containing constant population sizes for each species tree edge, for a haploid organism. Sizes should be doubled for diploids. If stree has k edges, then popSizes must have k+1 elements, with the final entry for the population ancestral to the root.
gtSample	An object of class multiPhylo holding a sample of gene trees from a simulation. Taxon labels on gene trees must be identical to those on stree.
taxon1	A string specifying one taxon on stree.
taxon2	A string specifying a second taxon on stree, distinct from taxon1.
taxon3	A string specifying a third taxon on stree, distinct from taxon1, taxon2.
subsampleSize	A positive integer or FALSE, giving size of subsamples of gtSample to analyze.

Details

When subsampleSize is FALSE the Chi-squared test is performed using all gene trees in gtSample. Results are reported in tabular form in the console.

When subsampleSize is a positive integer, the N trees in gtSample will be partitioned into subsets of size floor(N/subsampleSize). A Chi-squared test is performed for each subsample. Two histograms are plotted: The first shows p-values for the Chi-squared tests on subsamples. The second shows subsample estimates of the internal branch length for the rooted triple on the species tree, with the true value marked.

This function requires three distinct taxon names, all of which must occur on stree and in each of the gene trees in the sample.

See Also

edgeOrder, plotPops

Examples

```
stree=read.tree(text="((((a:10000,b:10000):10000,c:20000):10000,d:30000):10000,e:40000);")
pops=c(15000,25000,10000,1,1,1,1,1,12000)
gts=read.tree(file=system.file("genetreeSample",package="MSCsimtester"))
rootedTriple(stree,pops,gts,"a","b","c")
rootedTriple(stree,pops,gts,"a","b","c",1000)
```

Index