Package: ksi (via r-universe)

May 17, 2026

Title Phylogeny-based Komogorov-Smirnov Importance Statistic

Description Package underlying Cornwell et al. 2014 J. Ecology.

Author Tempo and Mode in Plant Trait Evolution Working Group

Maintainer Will Cornwell <wcornwell@gmail.com>

URL https://github.com/traitecoevo/ksi

BugReports https://github.com/traitecoevo/ksi/issues

Encoding UTF-8

Version 0.1-3

Depends R (>= 3.1.2)

Imports ape, diversitree (>= 0.9-6)

Suggests knitr, parallel, testthat

License GPL (>=2)

Config/pak/sysreqs libfftw3-dev libgsl0-dev

Repository https://traitecoevo.r-universe.dev

Date/Publication 2017-08-08 06:06:11 UTC

RemoteUrl https://github.com/traitecoevo/ksi

RemoteRef HEAD

RemoteSha d0f0ef4f7240b575f7dd55ce823d471ea7ae73d0

Contents

data.frame.to.vectors 2
ksi 2
ksi.nodeset 4

Index **5**

data.frame.to.vectors *Convert Data Frame to Vectors*

Description

Converts a data.frame with named rows and a number of data columns into a list of vectors, each of which is named with the rownames of the data.frame, and omitting any missing values. This is the format expected by `ksi`.

Usage

```
data.frame.to.vectors(dat)
```

Arguments

`dat`	A data.frame with named rows.

Author(s)

Richard G. FitzJohn

ksi *KSI*

Description

This is the fitting function. The test works by sequentially fitting a series of (possibly nested) nodes and asking "Is the trait distribution for the clade descending from this node different to that of the nodes neighbourhood?". The neighbourhood is defined as all the species that are in the 'partition' of the parent of the focal node that are

- Not descended from that node
- Not descended from a node that was identified in a previous iteration (so, when fitting node i, we exclude all species descended from nodes $1..(i-1)$)

This is basically the same algorithm as MEDUSA uses for fitting diversification rates that vary across a phylogeny.

Usage

```
ksi(tree, dat, depth=10, test=NULL, verbose=TRUE,
    multicore=FALSE, multicore.args=list())
```

Arguments

tree	An phylogeny, of class phylo (ape's phylogeny format). Branch lengths are not required, and are ignored if present. Node labels are recommended. If any nodes lack labels, labels of the form 'node.xxx' will be created. This happens before removing species that do not have trait data so that these node labels will be the same across different analyses.
dat	A named vector of tip states. This can be numeric, a factor (categorical) or logical (TRUE/FALSE).
depth	The number of nested nodes to identify (integer from 1 to the number of nodes in the tree once all taxa that lack state information are removed).
test	Force a particular test to be used. Valid values are • "ks": do a Kolmogorov-Smirnov test on continuous valued distributions. • "chisq": do a Chi-squared test on a contingency table with categorical or binary traits. • "gtest": do a g-test on a contingency table with categorical or binary traits. If NULL (the default) this will be guessed from dat.
verbose	Print a (fairly small) amount of progress information as the fits proceed.
multicore	Logical: if TRUE, uses the multicore package to carry out some of the calculations in parallel.
multicore.args	Arguments to control the behaviour of mclapply, when multicore=TRUE. For example multicore.args=list(mc.cores=4, mc.preschedule=FALSE) specifies that mclapply should use 4 cores, and use its load-balancing algorithm.

Value

The ksi function returns an object of class ksi. This is a list with one element per node, named with the names of the nodes that were fit. Each list element contains the elements:

statistic Value of the statistic against which nodes are ranked (for the Kolmogorov-Smirnov test, this is D scaled by the relative sample sizes – see ?ks.test).

p.value The p-value from the test (often meaninglessly small on large trees or uneven partitions).

n A vector of length 2 with the number of species in the 'target' and 'neighbourhood' of the node, respectively.

In addition, the ksi object (obj, say) has a number of attributes:

attr(obj, "tree") The phylogeny, after adding node labels and dropping species that have no trait data.

attr(obj, "dat") The data, altered to have the same contents and order as attr(obj, "tree")$tip.label.

attr(obj, "contents") A list with one element per node. Each element is a list with elements neighbourhood, target and other, containing the indices of the species that were in each test. These are indices against attr(obj, "dat") for now.

attr(obj, "statistics") A list with one element poer node. Each element is a vector along the nodes (in ape index) with the statistic for each node for that round. Nodes that were used in previous rounds have a value of NA.

attr(obj, "test") Indicates which test (ks, chisq, gtest) was done.

`ksi.nodeset` *Set of Plausible KSI Nodes*

Description

...to write...

Usage

```
ksi.nodeset(obj, idx, alpha=1/20, node=NULL)
ksi.group(obj, idx, alpha=1/20, include=1/5)
```

Arguments

`obj`	Fitted `ksi` object, returned by `ksi`.
`idx`	Index of the node to search around (1, 2, etc).
`alpha`	The quantile to include when defining plausible sets; `alpha=1/20` includes the top 5% of nodes.
`include`	Rank up to this fraction of the nodes in the tree.
`node`	If non-NULL, centres the nodeset at a different node.

Details

...to write...

Value

Vector of node *names*.

Author(s)

Richard G. FitzJohn

See Also

`ksi`

Index

∗ **model**
ksi, 2
∗ **utils**
data.frame.to.vectors, 2
∗ **util**
ksi.nodeset, 4

data.frame.to.vectors, 2

ksi, 2, 2, *4*
ksi.group *(*ksi.nodeset*)*, 4
ksi.nodeset, 4