

Package ‘TITAN2’

December 7, 2020

Type Package

Title Threshold Indicator Taxa Analysis

Version 2.4.1

Description Uses indicator species scores across binary partitions of a sample set to detect congruence in taxon-specific changes of abundance and occurrence frequency along an environmental gradient as evidence of an ecological community threshold. Relevant references include: Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37. King, RS and ME Baker. 2010. Considerations for identifying and interpreting ecological community thresholds. *Journal of the North American Benthological Association* 29(3):998-1008. Baker ME and RS King. 2013. Of TITAN and straw men: an appeal for greater understanding of community data. *Freshwater Science* 32(2):489-506.

VignetteBuilder knitr

Depends R (>= 2.15.0)

Imports parallel, dplyr, ggplot2, glue, cowplot, ggribes, purrr, tibble, snow

Suggests knitr, rmarkdown

License GPL-2

LazyData TRUE

RoxygenNote 7.1.0

Encoding UTF-8

NeedsCompilation no

Author Matthew E. Baker [aut, cph] (<<https://orcid.org/0000-0001-5069-0204>>),
Ryan S. King [aut, cph] (<<https://orcid.org/0000-0002-3159-9816>>),
David Kahle [aut, cph, cre] (<<https://orcid.org/0000-0002-9999-1558>>)

Maintainer David Kahle <david@kahle.io>

Repository CRAN

Date/Publication 2020-12-07 16:30:03 UTC

R topics documented:

boot.titan	2
env.part	4
getivz	5
glades.env	6
glades.taxa	7
glades.titan	7
indval	8
indvalp	9
ivzsums	10
obs.summ	11
permiv	13
plot-cps	14
plot-sumz	16
plot-taxa	19
plot_sumz_density	22
plot_taxa_ridges	25
smallBigBoot	27
sumz.tab	30
tboot	31
titan	33
TITAN2	35
txa.screen	36
Index	38

boot.titan	<i>Controls the allocation of bootstrap replicates</i>
------------	--

Description

A wrapper function for controlling the implementation of bootstrap replicates using the function 'tboot' by sequential or parallel processing.

Usage

```
boot.titan(
  env,
  taxa,
  ivTot = ivTot,
  boot = boot,
  ncpus = ncpus,
  nBoot = nBoot,
  minSplt = minSplt,
  nPerm = 250,
  memory = memory,
  imax = imax,
```

```

    numUnit = numUnit
  )

```

Arguments

env	A vector of values for each sampling location along the environmental gradient.
taxa	A site-by-taxon matrix of taxa counts at each sampling location.
ivTot	A logical indicating whether IndVal scores should be calculated using total relative abundance or the mean relative abundance originally proposed by Dufrene and Legendre (1997). The default is to pass on the argument from the original TITAN function call.
boot	A logical specifying whether or not to implement TITAN's' bootstrap procedure. The default is to use the value specified in the original TITAN function call.
ncpus	An argument specifying the number of processing cores used by the TITAN function call. If ncpus>1 then parallel processing is implemented. The default is to use the value specified in the original TITAN function call.
nBoot	An argument specifying the number of bootstrap replicates. The default is to use the value specified in the original TITAN function call.
minSplit	An argument specifying minimum split size for partitioning along the environmental gradient. The default is to use the value specified in the original TITAN function call.
nPerm	The number of replicates used by the permutation procedure (not to be confused with the number of bootstrap replicates).
memory	A logical indicating whether scratch files should be used to store temporary data in order to conserve active memory during bootstrapping of large data sets. The default is to pass on the argument from the original TITAN function call.
imax	A logical indicating whether taxon-specific change points should be determined by IndVal maxima or z-score maxima (as in TITAN v1.0). The default is to pass on the argument from the original TITAN function call.
numUnit	An argument specifying the number of values along the environmental gradient.

Details

If 'ncpus'>1 evaluates to TRUE, the function employs the package 'snow' to implement parallel processing on multicore processors common in modern desktop computers. With some minor modification it is possible to configure this code to allocate processes to cores on a high-performance computing cluster (i.e., a supercomputer). If 'ncpus'>1 evaluates to FALSE, the function uses 'lapply' to run 'tboot' in sequence 1:nBoot times.

Value

A list of two items:

- bSeqAn index of the sequence of bootstrap replicates. The structure of bSeq will differ for sequential or parallel processing.
- ivz.bt.listItself a list of four items comprising output passed on from function `tboot()`

Author(s)

M. Baker and R. King

References

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

See Also

[tboot\(\)](#), [small.boot\(\)](#), [big.boot\(\)](#), [titan\(\)](#)

env.part

Partitions environmental gradient for analysis

Description

This function compares the number of records in the environmental vector and the number of rows in the taxa matrix to ensure they are compatible. It then uses the rank order of environmental values at each sampling location to determine group membership of all sampling sites at each level of partitioning.

Usage

```
env.part(env, taxa, minSplt = minSplt, messaging = TRUE)
```

Arguments

env	A vector of values for each sampling location along the environmental gradient.
taxa	A site by taxon matrix containing observed counts at each sampling location.
minSplt	The minimum split size used in binary partitioning. The default is to use the argument from the original TITAN function call.
messaging	message user?

Value

A list of seven objects:

- envan environmental vector
- numUnitthe number of sample units in env
- numTxthe number of distinct taxonomic units in taxa
- numClas the number of candidate partitions (numUnit-2*minSplt)
- srtEnva sorted version of the environmental vector
- envclsa vector of environmental values used to distinguish partitions
- eclassa matrix of group membership relative to each partition in envcls

Author(s)

M. Baker and R. King

References

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

getivz	<i>Performs calculation of IndVal z scores from observed and permuted values</i>
--------	--

Description

A wrapper function that calls 'indval' or 'indvals' to obtain observed IndVal scores and 'permiv' to generate permuted values, then calculates z scores and associated p-values.

Usage

```
getivz(
  class,
  spp,
  ivTot = ivTot,
  nPerm = nPerm,
  numClass = numClass,
  imax = imax,
  messaging = FALSE
)
```

Arguments

class	A matrix of binary (0,1) group membership based on partitions of sampling sites ranked along an environmental gradient.
spp	A site-by-taxon matrix of taxa counts at each sampling location.
ivTot	A logical indicating whether IndVal scores should be calculated using total relative abundance or the mean relative abundance originally proposed by Dufrene and Legendre (1997).
nPerm	The number of permutations to be used by 'permiv'.
numClass	The number of classes used to partition samples along the environmental gradient. The default is the total number of observations less two times the minimum split size ('minSplt').
imax	A logical indicating whether taxon-specific change points should be determined using IndVal maxima or z-score maxima (as in TITAN v1.0).
messaging	message user?

Details

This function calls two subfunctions to first calculate IndVals and their associated direction (i.e., a decreasing taxon, group 1, is associated with the left side of any partition, whereas an increasing taxon, group 2, is associated with the right) and second to develop permuted values for each candidate partition. Output includes the indicator direction (1 or 2), z scores, IndVal scores, and associated p value (obtained by the fraction of times an observed IndVal is greater than those obtained from numPerm randomizations of equivalent group sizes). For more detail regarding the relative benefits and potential drawbacks of using imax=T or imax=F, see Baker and King (2013).

Value

A matrix containing four submatrices (the first from [1 : numTxa,], the second from [(numTxa+1) : (2*numTxa) ,], etc.), the first two of which include indicator direction and z scores.

- Group Membership A vector for every taxon showing decreasing (1) or increasing (2) group membership at each value of 'envcls'
- z scores A vector for every taxon showing IndVal z scores at each value of 'envcls'
- IndVals A vector for every taxon showing IndVal scores at each value of 'envcls'
- p values A vector for every taxon showing IndVal p values at each value of 'envcls'

Author(s)

M. Baker and R. King

References

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

Baker ME and RS King. 2013. Of TITAN and straw men: an appeal for greater understanding of community data. *Freshwater Science* 32(2):489-506.

See Also

[indval\(\)](#), [indvals\(\)](#), [indvalp\(\)](#), [indvalps\(\)](#), [permiv\(\)](#), [ivzsums\(\)](#)

glades.env

glades.env title

Description

glades.env description

Author(s)

M. Baker and R. King

References

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

glades.taxa

glades.taxa title

Description

glades.taxa description

Author(s)

M. Baker and R. King

References

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

glades.titan

glades.titan title

Description

glades.titan description

Author(s)

M. Baker and R. King

References

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

indval	<i>IndVal scores based on relative abundance across, and occurrence frequency within, groups</i>
--------	--

Description

Calculates indicator value scores using the original method of Dufrene and Legendre (1997) for all taxa in a site-by-taxon matrix split into two groups along an environmental gradient. A modified version (indvals) calculates indicator value scores using a modification of Dufrene and Legendre (1997), whereby relative abundance is computed as total relative abundance across groups rather than as mean relative abundance.

Usage

```
indval(grp, spp, allscores = FALSE)
indvals(grp, spp, allscores = FALSE)
```

Arguments

grp	A binary vector (0,1) indicating group membership for partitioning of rows in 'spp' according to a ranks of an environmental variable.
spp	A site-by-taxon matrix of taxa counts at each sampling location.
allscores	A logical indicating whether the list of scores returned by the function should be truncated. The default is to return just the largest IndVal (of two, one for each side of the partition) and on which side of the partition the score occurs. If allscores is TRUE, IndVals on both sides of the partition, as well as their relative abundances are also returned.

Details

The modified version of the original IndVal method was developed to address potential for bias in change point identification for highly skewed samples. This and the function 'indvalps' are run when the argument 'ivTot' in the original TITAN function call is set to TRUE. The function 'indval' is expected to be used more frequently, and 'indvals' is provided largely for comparative purposes.

Value

A 2-column matrix with the same nrow as the matrix 'spp'. The first column consists of a 1 (left) or 2 (right) for each taxon indicating on which group produced the largest IndVal score, and the second column contains the actual IndVal score.

Author(s)

M. Baker and R. King

References

Dufrene, M. and P. Legendre. 1997. Species assemblages and indicator species: the need for a flexible asymmetrical approach. *Ecological Monographs* 67:345-366.

See Also

[indvalp\(\)](#), [indvalps\(\)](#), [getivz\(\)](#)

indvalp

Calculate permuted IndVal scores using a group matrix

Description

This function performs the same calculations as 'indval' or 'indvals', but does so using matrix operations on a set of binary group assignments in matrix form. Function 'indvalps' calculates indicator value scores using a modification of Dufrene and Legendre (1997), whereby relative abundance is computed as total relative abundance across groups rather than as mean relative abundance.

Usage

```
indvalp(grpMatrix, spp)
```

```
indvalps(grpMatrix, spp)
```

Arguments

grpMatrix A site-by-permutation matrix of binary (0,1) assignments indicating group membership for successive randomizations of rows of 'spp' according to a ranks of an environmental variable.

spp A site-by-taxon matrix of taxa counts at each sampling location.

Details

Although the 'indval' function is reasonably efficient for a single calculation, when repeated during permutations (default = 250) and again during for each bootstrap replicate (default = 500), small differences in processing time can quickly become unwieldy. Even with the matrix operation, the permutation accounts for > 3/4 of processing time for most data sets (due to the fact that it is repeated for each bootstrap replicate).

The output matrix does not include information about the group membership of indval maxima because this information is not used in the permutation procedure (i.e., only the distribution of IndVal magnitudes is relevant).

Modification of the original IndVal method was developed to address potential for bias in change point identification for highly skewed samples. This and the function 'indvals' are run when the argument 'ivTot' in the original TITAN function call is TRUE. It is expected that 'indval' and 'indvalp' will be used more commonly.

Value

A matrix (ivmax) of IndVal maxima across each partition with nrow equal to the number of permutations and ncol equal to the number of taxa in 'spp'.

Author(s)

M. Baker and R. King

References

Dufrene, M. and P. Legendre. 1997. Species assemblages and indicator species: the need for a flexible asymmetrical approach. *Ecol. Mon.* 67:345-366.

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

See Also

[indval\(\)](#), [indvals\(\)](#), [getivz\(\)](#)

ivzsums

Sum IndVal z scores across taxa

Description

This function uses the output of 'getivz' (and, optionally, similar output from the bootstrap procedure) to sum z scores across taxa associated with each indicator direction (- or +) at each level of the environmental gradient.

Usage

```
ivzsums(allivz)
```

```
ivzsums.f(allivz, sppmax)
```

Arguments

allivz The output matrix from the function 'getivz' that contains four submatrices, the first two of which include indicator direction and z scores.

sppmax A completed summary output table for all taxa, used to ascertain which taxa should be filtered and which retained.

Details

The function selects taxa identified as either increasing or decreasing at each level of the environmental gradient and combines their IndVal z scores to generate an assemblage-wide sum. The sum(z) scores are interpreted as the magnitude of community change at each level of the environmental gradient.

The alias 'ivzsums.f' uses the output of 'getivz' to sum z scores across taxa filtered by purity and reliability associated with each indicator direction (- or +) at each level of the environmental gradient. Filters are provided by the final column ("filter") in the 'sppmax' table that is part of each TITAN object and an argument for 'ivzsums.f'. All taxa with a value of either 1 or 2 are pure and reliable decrease or increase, respectively, and are selected for summation. Filtered sums are used by the 'plot.sumz' function to create estimates of robust community change more precise than the original unfiltered sum(z) in TITAN v1.0.

Value

A matrix of two (z- and z+) parallel vectors with length (nrow) equal to the number of candidate partitions of an environmental gradient.

Author(s)

M. Baker and R. King

References

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

See Also

[getivz\(\)](#), [plot_sumz\(\)](#)

obs.summ

Summarizes results from TITAN's analysis of observed data

Description

This function populates the first seven columns of the 'sppmax' output table with results summarized for each taxon from the output of 'getivz'.

Usage

```
obs.summ(ivzScores, taxa, srtEnv, minSplt = minSplt, imax = imax)
```

Arguments

ivzScores	The product of the 'getivz' function. A data matrix comprised of four submatrices including group membership, z scores, IndVals, and p values.
taxa	A site by taxon matrix with counts observed at each sampling location.
srtEnv	A sorted version of the environmental gradient.
minSplt	The minimum split size used to determine partitions along the environmental gradient. The default is to use the argument from the original TITAN function call.
imax	A logical indicating whether taxon-specific change points should be determined using IndVal maxima or z-score maxima (as in TITAN v1.0).

Details

The items summarized for each taxon include (1) the env value at the IndVal maximum, (2) the env value at the z-score maximum, (3) the taxon's occurrence frequency, (4) group membership (decreaser or increaser), (5) the observed IndVal score, (6) the probability that the taxon's IndVal score could be generated by random chance, (7) the z score. As the values 3-7 are computed for every taxon at each candidate change point, values for the table are determined by the maximum (IndVal or z score) indicated by the value of 'imax'. For further detail regarding the advantages and disadvantages of using imax=T or imax=F, see Baker and King (2013).

Value

The function output consists of a list of three objects:

- obs1a logical indicating decreasing taxa
- obs2a logical indicating increasing taxa
- sppmaxa partially completed summary output table for all taxa

Author(s)

M. Baker and R. King

References

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

Baker ME and RS King. 2013. Of TITAN and straw men: an appeal for greater understanding of community data. *Freshwater Science* 32(2):489-506.

See Also

[getivz\(\)](#), [titan\(\)](#)

permiv	<i>Permutation of group membership for a single candidate partition</i>
--------	---

Description

Randomizes group assignments for all permutations based on ranked values of the environmental gradient, then calls the appropriate matrix-based IndVal function.

Usage

```
permiv(grp, spp, ivTot = ivTot, nPerm = 250)
```

Arguments

grp	A vector of binary (0,1) assignments indicating group membership for a single partition of n rows in 'spp' according to a ranks of an environmental variable.
spp	A site-by-taxon matrix of taxa counts at each sampling location.
ivTot	A logical indicating whether IndVal scores should be calculated using total relative abundance or the mean relative abundance originally proposed by Dufrene and Legendre (1997).
nPerm	The number of permutations to be performed.

Details

This function handles the randomization portion of the permutation procedure, and then serves as a wrapper for 'indvalp' and 'indvalps' (depending on the value of 'ivTot'), which estimate change-point distributions across all permutations. The output is the same as 'indvalp' and 'indvalps' because the function simply passes their products on.

Value

A matrix (ivmax) of IndVal maxima with nrow equal to the number of permutations and ncol equal to the number of taxa in 'spp'.

Author(s)

M. Baker and R. King

References

Dufrene, M. and P. Legendre. 1997. Species assemblages and indicator species: the need for a flexible asymmetrical approach. *Ecol. Mon.* 67:345-366.

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

See Also

[indvalp\(\)](#), [indvalps\(\)](#), [getivz\(\)](#)

`plot-cps`*Plots probability densities of empirical distributions of bootstrapped change points*

Description

This function allows more detailed exploration of taxon-specific response documented by TITAN through analysis of empirical distributions of bootstrapped change points, comparison of those distributions with observed counts, and aggregate (optionally weighted) summaries of those distributions across taxa.

Usage

```
plot_cps(  
  titan.out,  
  taxa.dist = T,  
  z.weights = T,  
  taxaID = NULL,  
  cp.med = F,  
  cp.trace = F,  
  cp.hist = T,  
  stacked = F,  
  xlabel = "Environmental Gradient",  
  xmin = min(titan.out$env),  
  xmax = max(titan.out$envcls) * 1.25,  
  tck = 0.025,  
  bty = "u",  
  ntick = 6,  
  cex = 1.75,  
  cex.axis = 1.75,  
  cex.leg = 1.5,  
  cex.lab = 1.75,  
  write = F,  
  leg.x = 0.8,  
  leg.y = 0.8,  
  leg = TRUE,  
  ...  
)  
  
plotCPs(...)
```

Arguments

`titan.out` A TITAN output object.
`taxa.dist` A logical specifying whether taxon-specific distributions should be plotted.

<code>z.weights</code>	A logical specifying whether taxon-specific or aggregate community distributions should be weighted by their median z scores (median of z-score maxima values across bootstrap replicates).
<code>taxaID</code>	An index specifying whether a particular taxon should be targeted for plotting. A 'NULL' value indicates all taxa should be plotted. Values >0 will select pure and reliable taxa by their row number within the 'sppmax' output table. Character strings may also be used corresponding to the row name within the 'sppmax' output table.
<code>cp.med</code>	A logical specifying whether change point locations should be plotted using the median value across all bootstrap replicates instead of the observed value.
<code>cp.trace</code>	A logical specifying whether IndVals and z scores across all candidate change points should be plotted.
<code>cp.hist</code>	A logical specifying whether histograms of replicate change point PDFs should be plotted.
<code>stacked</code>	A logical specifying whether community level aggregations of change point PDFs are stacked or plotted separately.
<code>xlabel</code>	A character string for the x axis label.
<code>xmin</code>	A graphical argument specifying the value of the x axis minimum.
<code>xmax</code>	A graphical argument specifying the value of the x axis maximum.
<code>tck</code>	A graphical argument specifying the scale of axis tick marks.
<code>bty</code>	A graphical argument.
<code>ntick</code>	A graphical argument specifying the default number of axis tick marks.
<code>cex</code>	A graphical argument specifying the scaling of the figure.
<code>cex.axis</code>	A graphical argument specifying the scaling of the axes.
<code>cex.leg</code>	A graphical argument specifying the scaling of the legend.
<code>cex.lab</code>	A graphical argument specifying the scaling of the labels.
<code>write</code>	A logical specifying whether taxa subsets are written to screen.
<code>leg.x</code>	A graphical argument specifying the x coordinate of the legend.
<code>leg.y</code>	A graphical argument specifying the y coordinate of the legend.
<code>leg</code>	A logical specifying whether or not to plot the legend.
<code>...</code>	An argument for passing generic plotting function parameters.

Details

Following the initial (v1.0) TITAN publications, it was clear that substantial information regarding taxon-specific change points was lost when bootstrapped distributions were represented solely as quantiles (i.e., as in `plot_taxa()` and the 'sppmax' output table). Empirical probability densities allow greater detail and more nuanced interpretation associated with this uncertainty, especially when compared against observed abundance and occurrence. Further, comparison of the summed probability densities and the filtered `sum(z)` plots with the default `sum(z)` output provides additional support of community changes consistent with threshold behavior.

Value

Three types of plots are possible outcomes of this function. The first (`taxa.dist=T`, `taxID=NULL`) is a matrix of histograms showing empirical distributions of bootstrapped change-point locations (as probability densities) for all pure and reliable taxa. The value of the probability densities can be weighted by the median z score for each taxon (`z.weights=T`). The second plot (`taxa.dist=T`, `taxID>0` or a taxon label) overlays a taxon-specific histogram on an abundance scatter plot and the observed change-point location. The third plot (`taxa.dist=F`) shows the sum of probability densities across all pure and reliable taxa, optionally weighted by median z scores (`z.weights=T`) or stacked (`stacked=T`).

Note

Should not be used with output objects from TITAN v1.0.

Author(s)

M. Baker and R. King

References

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

King, RS and ME Baker 2010. Considerations for identifying and interpreting ecological community thresholds. *Journal of the North American Benthological Association* 29(3):998-1008.

See Also

[plot_taxa\(\)](#), [plot_sumz\(\)](#)

Examples

```
data(glades.titan)
plot_cps(glades.titan,
  taxa.dist = FALSE,
  xlabel = "Surface Water TP (ug/l)",
  stacked = TRUE
)
```

plot-sumz

Plots the pattern of community change along an environmental gradient

Description

Creates a plot of community-level sums of taxon-specific change along an environmental gradient and optionally conveys uncertainty associated with the maximum community change derived from decrease or increase.

Usage

```

plot_sumz(
  titan.out,
  filter = FALSE,
  cumfrq = TRUE,
  bootz1 = TRUE,
  bootz2 = TRUE,
  sumz1 = TRUE,
  sumz2 = TRUE,
  xmin = min(titan.out$env),
  xmax = max(titan.out$envcls) * 1.25,
  xlabel = "Environmental Gradient",
  y1label = NULL,
  y2label = "Cumulative Frequency",
  log = "",
  at = NULL,
  tck = 0.025,
  bty = "u",
  ntick = 6,
  prtty = TRUE,
  dig = 1,
  cex = 1.75,
  cex.axis = 1.75,
  cex.leg = 1.5,
  cex.lab = 1.75,
  leg.x = 0.8,
  leg.y = 0.8,
  legend = TRUE,
  pch1 = 16,
  pch2 = 1,
  col1 = "black",
  col2 = "black",
  ...
)

plotSumz(...)

```

Arguments

titan.out	A TITAN output object.
filter	A logical indicating whether only pure and reliable taxa should be used to create the plot. This is the recommended as a check of the unfiltered default to assess whether impure or unreliable taxa are substantially contributing to the distribution of sum(z) scores.
cumfrq	A logical specifying whether cumulative frequencies of sum(z) maxima across bootstrap replicates should be plotted.
bootz1	A logical specifying whether decreasing cumulative frequencies exist or should be plotted.

bootz2	A logical specifying whether increasing cumulative frequencies exist or should be plotted.
sumz1	A logical specifying whether decreasing changes should be plotted.
sumz2	A logical specifying whether increasing changes should be plotted.
xmin	A graphical argument specifying the value of the x axis minimum.
xmax	A graphical argument specifying the value of the x axis maximum.
xlabel	A character string for the x axis label.
y1label	A character specifying the label of the second y axis
y2label	A character specifying the label of the second y axis
log	A graphical argument specifying whether an axis should be log scaled.
at	A graphical argument for controlling placement of the x axis label
tck	A graphical argument specifying the scale of axis tick marks.
bty	A graphical argument.
ntick	A graphical argument specifying the default number of axis tick marks.
prtty	A logical specifying whether pretty() should be used to plot axis labels.
dig	A numeric argument specifying the number of decimal digits in axes.
cex	A graphical argument specifying the scaling of the figure.
cex.axis	A graphical argument specifying the scaling of the axes.
cex.leg	A graphical argument specifying the scaling of the legend.
cex.lab	A graphical argument specifying the scaling of the labels.
leg.x	A graphical argument specifying the x coordinate of the legend.
leg.y	A graphical argument specifying the y coordinate of the legend.
legend	A logical specifying whether or not to plot the legend.
pch1	A graphical argument specifying the type of group 1 symbols.
pch2	A graphical argument specifying the type of group 2 symbols.
col1	A graphical argument specifying the color of group 1 symbols.
col2	A graphical argument specifying the color of group 2 symbols.
...	An argument for passing generic plotting function parameters.

Details

This function assumes that the TITAN object contains bootstrap summaries and filtering information and automatically determines whether this is the case. Without the bootstrap summaries, only unfiltered change magnitudes are plotted.

The original sum(z) plots (v1.0) did not filter taxa using purity and reliability. Because these taxa often have small z scores, they are unlikely to contribute significantly to the sum(z) profiles. However, subsequent investigation has demonstrated that when sufficient numbers of taxa are involved, it is possible for noisy data to generate artifactual peaks in low-magnitude sum(z) profiles or plateaus. Therefore, we recommend evaluating filtered versions of the sum(z) to assess this potential in v2.0.

Value

A plot of sum(z-) and sum(z+) profiles along the environmental gradient.

Note

Should not be used with output objects from TITAN v1.0.

Author(s)

M. Baker and R. King

References

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

King, RS and ME Baker 2010. Considerations for identifying and interpreting ecological community thresholds. *Journal of the North American Benthological Association* 29(3):998-1008.

See Also

[plot_taxa\(\)](#), [plot_cps\(\)](#)

Examples

```
data(glades.titan)
plot_sumz(glades.titan, filter = FALSE)
```

plot-taxa

Plots taxon-specific change points

Description

Creates a plot of taxon-specific change points with optional quantiles conveying uncertainty resulting from bootstrapped samples and optional filtering by pure and reliable taxa.

Usage

```
plot_taxa(  
  titan.out,  
  z1 = TRUE,  
  z2 = TRUE,  
  interval = TRUE,  
  prob95 = FALSE,
```

```

z.med = FALSE,
xlabel = "Environmental Gradient",
log = "",
at = NULL,
xmin = min(titan.out$sppmax[, 8]),
xmax = max(titan.out$sppmax[, 12]) * 1.05,
tck = 0.025,
bty = "u",
ntick = 6,
prtty = TRUE,
dig = 1,
leg.x = 0.8,
leg.y = 10,
cex.taxa = 0.75,
cex = 1.25,
cex.axis = 1.25,
cex.leg = 1.25,
cex.lab = 1.25,
legend = TRUE,
col1 = "black",
fil1 = "black",
col2 = "black",
fil2 = "white",
write = FALSE,
all = FALSE,
...
)

plotTaxa(...)

```

Arguments

titan.out	A TITAN output object.
z1	A logical specifying whether decreasing taxa should be plotted.
z2	A logical specifying whether decreasing taxa should be plotted.
interval	A logical specifying whether quantiles of bootstrapped change points should be plotted.
prob95	A logical specifying whether change-point locations should be plotted on the basis of their 5th (for increasers) and 95th (for decreasers) quantile versus their observed values.
z.med	A logical specifying whether (1) change point magnitudes should be obtained from the median z score across bootstrap replicates and (2) whether the locations should be plotted on the basis of the 50th quantile of change-point locations (i.e., if prob95=FALSE).
xlabel	A character string for the x axis label.
log	A graphical argument specifying whether an axis should be log scaled.

at	A graphical argument controlling the location of the x axis label.
xmin	A graphical argument specifying the value of the x axis minimum.
xmax	A graphical argument specifying the value of the x axis maximum.
tck	A graphical argument specifying the scale of axis tick marks.
bty	A graphical argument specifying the box type around the plot.
ntick	A graphical argument specifying the default number of axis tick marks.
prtty	A logical specifying whether pretty() should be used for axis plotting.
dig	A numeric argument controlling the number of decimal digits in axis labels.
leg.x	A graphical argument specifying the x coordinate of the legend.
leg.y	A graphical argument specifying the y coordinate of the legend.
cex.taxa	A graphical argument specifying the scaling of the taxa names.
cex	A graphical argument specifying the scaling of the figure.
cex.axis	A graphical argument specifying the scaling of the axes.
cex.leg	A graphical argument specifying the scaling of the legend.
cex.lab	A graphical argument specifying the scaling of the labels.
legend	A logical specifying whether or not to plot the legend.
col1	A graphical argument specifying the color of group 1 symbols.
fill1	A graphical argument specifying the color of group 1 fills.
col2	A graphical argument specifying the color of group 2 symbols.
fill2	A graphical argument specifying the color of group 2 fills.
write	A logical specifying whether summary tables are written to screen.
all	A logical specifying whether all taxa with $p < 0.05$ should be plotted.
...	An argument for passing generic plotting function parameters.

Details

The function assumes that TITAN objects contain bootstrap summaries and filtering information and automatically determines whether this is the case. Without bootstrap summaries, only observed change-point locations and z-score magnitudes will be plotted. The plotting function automatically interprets whether observed change-point values were obtained using IndVal or z-score maxima. The interval option is for turning off the intervals for TITAN objects that contain bootstrap information. The prob95 is recommended for communicating uncertainty involving management or policy action, whereas the z.med option is recommended for increasingly robust estimates (by incorporating uncertainty associated with the sample) of taxon-specific change-point locations beyond those provided by the default (i.e., observed values).

Value

A plot of decreasing and/or increasing taxon-specific change points along the environmental gradient.

Note

Should not be used with output objects from TITAN v1.0.

Author(s)

M. Baker and R. King

References

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

King, RS and ME Baker 2010. Considerations for identifying and interpreting ecological community thresholds. *Journal of the North American Benthological Association* 29(3):998-1008.

See Also

[plot_sumz\(\)](#), [plot_cps\(\)](#)

Examples

```
data(glades.titan)
plot_taxa(glades.titan, xlabel = "Surface Water TP (ug/l)")
```

plot_sumz_density *Plot community level change*

Description

Plot community level change

Usage

```
plot_sumz_density(
  titan.out,
  filter = TRUE,
  sumz = TRUE,
  points = FALSE,
  ribbon = TRUE,
  density = TRUE,
  change_points = TRUE,
  sumz1 = TRUE,
  sumz2 = TRUE,
  xlim = c(min(titan.out$env), max(titan.out$envcls)),
  xlabel = "Environmental Gradient",
```

```

y1label = NULL,
y2label = "Density",
alpha1 = 0.65,
alpha2 = 0.5,
col1 = "steelblue4",
col2 = "red",
trans = "identity",
legend.position = c(0.9, 0.9),
...,
axis.text.x,
axis.text.y,
axis.title.x,
axis.title.y
)

```

Arguments

titan.out	A TITAN output object.
filter	A logical indicating whether only pure and reliable taxa should be used to create the plot. This is recommended as a check of the unfiltered default to assess whether impure or unreliable taxa are substantially contributing to the distribution of sum(z) scores.
sumz	Plot sum z values? (Default = TRUE)
points	Plot points on sum z values? (Default = FALSE)
ribbon	Fill in axes under sum z values? (Default = TRUE)
density	Plot densities of change points? (Default = TRUE)
change_points	Plot ranges of change points? These are taken from the titan object. (Default = TRUE)
sumz1	A logical specifying whether decreasing changes should be plotted.
sumz2	A logical specifying whether increasing changes should be plotted.
xlim	x axis limits, e.g. xlim = c(0, 10).
xlabel	A character string for the x axis label.
y1label	A character specifying the label of the second y axis
y2label	A character specifying the label of the second y axis
alpha1, alpha2	Transparency of Z- and Z+ values, respectively.
col1	A graphical argument specifying the color of group 1 symbols.
col2	A graphical argument specifying the color of group 2 symbols.
trans	a scale transformation to be applied to the x-axis through ggplot2. e.g. "log10" or "sqrt".
legend.position	ggplot2 legend position in relative coordinates of sum z plot. (Default = c(.9, .9))
...	An argument for passing generic plotting function parameters.
axis.text.x	Font sizes of respective axis text. Passed as the size argument to the ggplot2 thematic elements of the same name. Defaults to current default ggplot2 theme.

<code>axis.text.y</code>	Font sizes of respective axis text. Passed as the size argument to the <code>ggplot2</code> thematic elements of the same name. Defaults to current default <code>ggplot2</code> theme.
<code>axis.title.x</code>	Font sizes of respective axis text. Passed as the size argument to the <code>ggplot2</code> thematic elements of the same name. Defaults to current default <code>ggplot2</code> theme.
<code>axis.title.y</code>	Font sizes of respective axis text. Passed as the size argument to the <code>ggplot2</code> thematic elements of the same name. Defaults to current default <code>ggplot2</code> theme.

Value

A plot

Note

Should not be used with output objects from TITAN v1.0.

Author(s)

M. Baker and R. King

References

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

King, RS and ME Baker 2010. Considerations for identifying and interpreting ecological community thresholds. *Journal of the North American Benthological Association* 29(3):998-1008.

See Also

[plot_sumz\(\)](#), [plot_cps\(\)](#)

Examples

```
data(glades.titan)

plot_sumz_density(glades.titan)
plot_sumz_density(glades.titan, trans = "log10")
```

plot_taxa_ridges	<i>Taxon change point ridge plots</i>
------------------	---------------------------------------

Description

Taxon change point ridge plots

Usage

```
plot_taxa_ridges(
  titan.out,
  z1 = TRUE,
  z2 = TRUE,
  z1_fill_low = "lightblue",
  z1_fill_high = "black",
  z2_fill_low = "pink",
  z2_fill_high = "red",
  pur.cut = titan.out$arguments[[7]],
  rel.cut = titan.out$arguments[[8]],
  xlabel = "Environmental Gradient",
  n_ytaxa = 90,
  printsp = FALSE,
  grid = TRUE,
  trans = "identity",
  xaxis = !grid,
  xlim,
  bw,
  rel_heights,
  ...,
  axis.text.x,
  axis.text.y,
  axis.title.x,
  axis.title.y
)
```

Arguments

titan.out	A TITAN output object.
z1	A logical specifying whether decreasing taxa should be plotted.
z2	A logical specifying whether increasing taxa should be plotted.
z1_fill_low, z1_fill_high, z2_fill_low, z2_fill_high	Respective fill colors passed to scale_fill_gradient()
pur.cut	pur.cut
rel.cut	rel.cut
xlabel	A character string for the x axis label.

n_ytaxa	The maximum number of taxa to be plotted.
print spp	A logical specifying whether the sppmax table should be printed.
grid	The grid argument of <code>theme_ridges()</code> . Setting this to FALSE removes horizontal lines from the plot, see examples.
trans	a scale transformation to be applied to the x-axis through ggplot2. e.g. "log10" or "sqrt".
xaxis	Logical; should the x-axis be plotted?
xlim	x axis limits, e.g. xlim = c(0, 10).
bw	The bandwidth of used in the kernel density estimate; see <code>density()</code> .
rel_heights	Argument to pass to <code>cowplot::plot_grid()</code> as an override for the defaults
...	Arguments to pass to <code>geom_density_ridges()</code>
axis.text.x, axis.text.y, axis.title.x, axis.title.y	Font sizes of respective axis text. Passed as the size argument to the ggplot2 thematic elements of the same name. Defaults to current default ggplot2 theme.

Value

A plot of decreasing and/or increasing taxon-specific change points along the environmental gradient.

Note

Should not be used with output objects from TITAN v1.0.

Author(s)

M. Baker, R. King, D. Kahle

References

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

King, RS and ME Baker 2010. Considerations for identifying and interpreting ecological community thresholds. *Journal of the North American Benthological Association* 29(3):998-1008.

See Also

`plot_sumz()`, `plot_cps()`

Examples

```
data(glades.titan)

# basic usage
plot_taxa_ridges(glades.titan)
```

```
## Not run: reduce R CMD check time

# manipulating the x axis thematic components
plot_taxa_ridges(glades.titan, grid = FALSE)
plot_taxa_ridges(glades.titan, xaxis = TRUE)

# applying scale transformations
plot_taxa_ridges(glades.titan, trans = "log10", xlim = c(1,150))

# removing z2
plot_taxa_ridges(glades.titan, z2 = FALSE, trans = "sqrt", xlim = c(0, 150))

# more styling
plot_taxa_ridges(glades.titan,
  axis.text.x = 12,
  axis.text.y = 4,
  axis.title.x = 14,
  xlabel = expression(paste("Surface water total phosphorus ("*mu*"g/l)"))
)

## End(Not run)
```

smallBigBoot

Summarizes raw output from TITAN's bootstrap procedure

Description

A function to take output from TITAN's bootstrap procedure and process it for summary output. The default is to perform this processing entirely within active memory, but in the event of overflowing system capacity, an optional program writes temporary files to a scratch directory to circumvent memory limits.

Usage

```
small.boot(
  ivz.bt.list,
  bSeq,
  sppmax,
  obs1,
  obs2,
  nBoot,
```

```

    numClass,
    numUnit,
    ncpus,
    pur.cut,
    rel.cut,
    minSplt
)

big.boot(
  ivz.bt.list,
  bSeq,
  sppmax,
  obs1,
  obs2,
  nBoot,
  numClass,
  numUnit,
  ncpus,
  pur.cut,
  rel.cut,
  minSplt
)

```

Arguments

<code>ivz.bt.list</code>	A list of output from each bootstrap replicate passed from <code>boot.titan()</code> .
<code>bSeq</code>	An index of the sequence of bootstrap replicates.
<code>sppmax</code>	A taxon-specific summary output table for TITAN.
<code>obs1</code>	A binary vector indicating membership in the decreasing group of taxa.
<code>obs2</code>	A binary vector indicating membership in the increasing group of taxa.
<code>nBoot</code>	An argument specifying the number of bootstrap replicates. The default is to use the value specified in the original TITAN function call.
<code>numClass</code>	An argument specifying the number of candidate partitions along the environmental gradient.
<code>numUnit</code>	An argument specifying the number of values along the environmental gradient.
<code>ncpus</code>	An argument specifying the number of processing cores used by the TITAN function call. If <code>ncpus>1</code> then parallel processing is indicated. The default is to use the value specified in the original TITAN function call.
<code>pur.cut</code>	An argument specifying the cutoff value for determining purity. The default is to use the value specified in the original TITAN function call.
<code>rel.cut</code>	An argument specifying the cutoff value for determining reliability. The default is to use the value specified in the original TITAN function call.
<code>minSplt</code>	An argument specifying minimum split size of partitioning along the environmental gradient. The default is to use the value specified in the original TITAN function call.

Details

Use of 'small.boot' versus 'big.boot' is controlled by the argument 'memory' in the original TITAN function call and passed to the wrapper function 'titan'. The two programs have identical functionality, but they accomplish those functions differently to deal with memory limitations.

For sequential processing of the bootstrap, the index 'bSeq' is simply a sequence from 1:nBoot that is printed to the screen. For parallel processing, 'bSeq' is a list of length equal to 'ncpus', where each item is a segment of the sequence allocated to each processing core. Thus, depending on whether 'ncpus' > 1, the value of 'bSeq' is used differently to extract values from the bootstrap output list.

The first part of each function consists of defining output matrices, the second involves extraction of output from the bootstrap list, the third part involves calculating purity, reliability, the median z score, and quantiles of the bootstrapped change points for each taxon. These values are used to complete the 'sppmax' output table and to identify the taxa that meet purity and reliability criteria. The final portion of each function finds the maximum sum(z-), sum(z+), f.sum(z-), and f.sum(z+) for each bootstrap replicate for later estimation of confidence intervals. The final portion of the summary involves calculating the filtered and unfiltered sum(z) scores for each bootstrap replicate from the matrix of z scores and response directions passed from the function boot.titan() within ivz.bt.list

Value

A list of six items:

- sppSub1A vector of taxon index numbers for pure and reliable decreaseers
- sppSub2A vector of taxon index numbers for pure and reliable increaseers
- sppmaxThe completed taxon-specific summary output table for TITAN
- maxSumzA 2-column matrix of environmental values at sum(z-) and sum(z+) maxima across all bootstrap replicates
- maxFsumzA 2-column matrix of environmental values at filtered sum(z-) and sum(z+) maxima across all bootstrap replicates
- metricArrayAn array of group membership, env change points, z scores, and p values for passing to 'plot.IVecdf'

Author(s)

M. Baker and R. King

References

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

Baker ME and RS King. 2013. Of TITAN and straw men: an appeal for greater understanding of community data. *Freshwater Science* 32(2):489-506.

See Also

[boot.titan\(\)](#), [tboot\(\)](#), [titan\(\)](#)

sumz.tab

Summarizes the results of the community-level sum(z) analysis

Description

This function populates the sumz.cp table using the results from function 'ivzsums' and, if 'boot'=TRUE, calls 'ivsums.f' to compute ivz sums filtered by pure and reliable taxa.

Usage

```
sumz.tab(
  ivzScores,
  ivz,
  srtEnv,
  sppmax,
  maxSumz = maxSumz,
  maxFsumz = maxFsumz,
  minSplt = minSplt,
  boot = boot
)
```

Arguments

ivzScores	The product of the 'getivz' function. A data matrix comprised of four submatrices including group membership, z scores, IndVals, and p values.
ivz	The product of the 'ivzsums' function. A data matrix comprised of two parallel vectors of sum(z-) and sum(z+) scores.
srtEnv	A sorted version of the environmental gradient.
sppmax	The taxon-specific summary output table from TITAN, passed to 'ivzsums.f'.
maxSumz	A vector of sum(z) maxima across bootstrap replicates.
maxFsumz	A vector of sum(z) maxima filtered by pure and reliable taxa across bootstrap replicates.
minSplt	The minimum split size used to partition the environmental gradient. The default is to use the argument specified by the original TITAN function call.
boot	A logical indicating whether the bootstrap procedure should be implemented. The default is to use the argument specified by the original TITAN function call.

Details

The function locates the env values of sum(z) maxima, then if 'boot'=TRUE, locates the the env value of the filtered sum(z) and provides bootstrap quantiles of both filtered and unfiltered distributions.

Value

A list with two objects:

- `sumz.cpA` second summary output table from TITAN to accompany `'sppmax'`.
- `ivz.f` The product of the `'ivzsums.f'` function. A data matrix comprised of two parallel vectors of filtered `sum(z-)` and `sum(z+)` scores.

Author(s)

M. Baker and R. King

References

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

King, RS and ME Baker 2010. Considerations for identifying and interpreting ecological community thresholds. *Journal of the North American Benthological Association* 29(3):998-1008.

See Also

[ivzsums\(\)](#), [ivzsums.f\(\)](#), [getivz\(\)](#), [titan\(\)](#)

tboot

Calculate bootstrapped IndVal z scores

Description

This function implements resampling (with replacement) of the observed environmental gradient and site-by-taxon matrix, and then calls the function [getivz\(\)](#) to obtain bootstrapped scores.

Usage

```
tboot(  
  bSeq,  
  env,  
  taxa,  
  ivTot = ivTot,  
  minSplt = minSplt,  
  nPerm = nPerm,  
  memory = memory,  
  imax = imax  
)
```

Arguments

bSeq	An index used to determine the sequence number of the current bootstrap replicate.
env	An environmental gradient.
taxa	A site-by-taxon matrix of taxa counts at each sampling location.
ivTot	A logical indicating whether IndVal scores should be calculated using total relative abundance or the mean relative abundance originally proposed by Dufrene and Legendre (1997). The default is to pass on the argument from the original TITAN function call.
minSplt	The minimum bin size for partitioning along the environmental gradient. The default is to pass on the argument from the original TITAN function call.
nPerm	The number of replicates used by the permutation procedure (not to be confused with the number of bootstrap replicates). The default is to pass on the argument from the original TITAN function call.
memory	A logical indicating whether scratch files should be used to store temporary data in order to preserve RAM during bootstrapping of large data sets. The default is to pass on the argument from the original TITAN function call.
imax	A logical indicating whether taxon-specific change points should be determined by IndVal maxima or z-score maxima (as in Baker and King 2010). The default is to pass on the argument from the original TITAN function call.

Details

Four pieces of information are obtained from every taxon during each bootstrap replicate. If the argument 'imax' is TRUE, bootstrapped change points are identified based on IndVal maxima, whereas if 'imax' is FALSE z-score maxima are used instead. In addition to the IndVal or z score maxima, the value of the environmental gradient, the indicator direction, and the p value are also retained for that point.

In addition to the above metric matrix for each taxon (1), the z scores across all candidate change points are retained from each replicate (2), as well as the response direction (maxgrp) (3) and a sorted version of resampled environmental values (4). These four items are combined as a list object.

Value

A list of four elements:

- `bt.metricsA` matrix with `nrow` equal to number of taxa where the first column is the bootstrapped IndVal or z score maximum, the second is the environmental value, the third is the indicator direction, and the fourth is the p value at that point.
- `ivzsZ` scores for all taxa across candidate change points in the replicate sample
- `bsrtiA` sorted version of the bootstrapped environmental gradient
- `rspdr` response direction (1 or 2) for all taxa across candidate change points in the replicate sample

Author(s)

M. Baker and R. King

References

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

See Also

[getivz\(\)](#), [ivzsums\(\)](#)

titan

Perform a threshold indicator taxa analysis

Description

`titan()` is the primary wrapper function controlling operation of all subroutines (`txa.screen()`, `env.part()`, `getivz()`, `ivzsums()`, `obs.summ()`, `boot.titan()`, `small.boot()/big.boot()`, `sumz.tab()`) apart from plotting functions within TITAN.

Usage

```
titan(  
  env,  
  txa,  
  minSplt = 5,  
  numPerm = 250,  
  boot = TRUE,  
  nBoot = 500,  
  imax = FALSE,  
  ivTot = FALSE,  
  pur.cut = 0.95,  
  rel.cut = 0.95,  
  ncpus = 1,  
  memory = FALSE,  
  messaging = TRUE  
)
```

Arguments

<code>env</code>	A vector of environmental values.
<code>txa</code>	A site by taxon matrix containing observed counts at each sampling location.
<code>minSplt</code>	The minimum split size to be used in partitioning.
<code>numPerm</code>	The number of replicates to be used during permutation.
<code>boot</code>	A logical indicating whether bootstrap resampling should be performed.

nBoot	The number of replicates to be used during bootstrap resampling.
imax	A logical indication whether taxon-specific change points should be determined using IndVal maxima or z-score maxima (as in Baker and King 2010, 2013).
ivTot	A logical indicating whether IndVal scores should be calculated using total relative abundance or the mean relative abundance originally proposed by Dufrene and Legendre (1997).
pur.cut	A proportion specifying the cutoff value for determining purity across all bootstrap replicates.
rel.cut	A proportion specifying the cutoff value for determining reliability across all bootstrap replicates.
ncpus	The number of processing cores to be used during processing. If greater than 1, TITAN will load and use the package 'snow' to perform parallel processing on each core.
memory	A logical indicating whether temporary files should be written to a scratch directory during bootstrap processing to preserve active memory. This function is sometimes necessary for large data files (e.g. more than 400 sampling sites and less than 100 taxa).
messaging	If TRUE, provide progress messages.

Value

A list with 13 items:

- sppmaxDescription of 'comp1'
- sumz.cpDescription of 'comp1'
- envThe vector of environmental values used in the TITAN function call
- taxaThe site-by-taxon matrix used in the TITAN function call
- envlcsA vector of candidate partitions derived from subtracting 'minSpl't' from 'env'
- srtEnvA sorted version of environmental values
- ivzScoresA matrix containing group membership, z scores, IndVals, and p values for each taxon at every candidate partition in 'envlcs'
- ivzA 2-column matrix containing parallel vectors of sum(z-) and sum(z+ scores for every candidate partition in 'envlcs')
- ivz.fA 2-column matrix containing parallel vectors of sum(z-) and sum(z+ scores filtered by pure and reliable taxa for every candidate partition in 'envlcs')
- maxSumzA 2-column matrix of environmental values at sum(z-) and sum(z+) maxima across all bootstrap replicates
- maxFsumzA 2-column matrix of environmental values at filtered sum(z-) and sum(z+) maxima across all bootstrap replicates
- metricArrayAn array of group membership, env change points, z scores, and p values equivalent to 'ivzScores' for each bootstrap replicate
- targSA vector of arguments used in the TITAN function call

Author(s)

M. Baker and R. King

References

- Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.
- King, RS and ME Baker 2010. Considerations for identifying and interpreting ecological community thresholds. *Journal of the North American Benthological Association* 29(3):998-1008.
- Baker ME and RS King. 2013. Of TITAN and straw men: an appeal for greater understanding of community data. *Freshwater Science* 32(2):489-506.

Examples

```
data(glades.env); str(glades.env)
data(glades.taxa); str(glades.taxa)

# small run to illustrate data structure
glades.titan <- titan(glades.env, glades.taxa, minSplt = 5,
  numPerm = 25, boot = TRUE, nBoot = 2, imax = FALSE,
  ivTot = FALSE, pur.cut = 0.95, rel.cut = 0.95, ncpus = 2, memory = FALSE
)
str(glades.titan, give.attr = FALSE)

# typical run
if (FALSE) {
glades.titan <- titan(glades.env, glades.taxa, minSplt = 5,
  numPerm = 250, boot = TRUE, nBoot = 100, imax = FALSE,
  ivTot = FALSE, pur.cut = 0.95, rel.cut = 0.95, ncpus = 7, memory = FALSE
)
}
```

Description

Uses indicator species scores across binary partitions of a sample set to detect congruence in taxon-specific changes of abundance and occurrence frequency along an environmental gradient as evidence of an ecological community threshold.

Details

Relevant references include: Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds.

Methods in Ecology and Evolution 1(1): 25:37. King, RS and ME Baker. 2010. Considerations for identifying and interpreting ecological community thresholds.#' Journal of the North American Benthological Association 29(3):998-1008. Baker ME and RS King. 2013. Of TITAN and straw men: an appeal for greater understanding of community data. Freshwater Science 32(2):489-506.

txa.screen

Screening of site by taxa matrix

Description

Screens site by taxa matrix for potential minimum occurrence and frequency problems.

Usage

```
txa.screen(txa, minSplt = minSplt, messaging = TRUE)
```

Arguments

txa	A site by taxon matrix of sampled counts at each sampling location.
minSplt	The minimum split size used for partitioning. The default is to use the argument form the original TITAN function call.
messaging	If TRUE, provide progress messages.

Details

This function provides a warning for low numbers of observations and data sets with too few observations to make any reasonable interpretation. It also warns users when 100 percent occurrence is detected and how much of the data set meets this criterion, as IndVal based analyses are not ideal for this type of data and we might recommend other techniques (Baker and King 2013). The function also assess whether occurrence frequencies for any taxa fall below 3, and call attention to very small minimum split sizes.

Value

A site by taxon matrix of sampled counts at each sampling location.

Author(s)

M. Baker and R. King

References

Baker, ME and RS King. 2010. A new method for detecting and interpreting biodiversity and ecological community thresholds. *Methods in Ecology and Evolution* 1(1): 25:37.

Baker ME and RS King. 2013. Of TITAN and straw men: an appeal for greater understanding of community data. *Freshwater Science* 32(2):489-506.

Index

- * **IndVal**
 - indval, 8
 - indvalp, 9
 - titan, 33
- * **Indicator~Species**
 - indval, 8
 - indvalp, 9
- * **Indicator~Value**
 - getivz, 5
- * **TITAN**
 - boot.titan, 2
 - env.part, 4
 - getivz, 5
 - indval, 8
 - indvalp, 9
 - ivzsums, 10
 - obs.summ, 11
 - permiv, 13
 - plot-sumz, 16
 - smallBigBoot, 27
 - sumz.tab, 30
 - tboot, 31
 - titan, 33
- * **Threshold~Indicator~Taxa~Analysis**
 - getivz, 5
- * **bootstrap**
 - boot.titan, 2
 - tboot, 31
 - titan, 33
- * **permutation**
 - permiv, 13
 - titan, 33
- * **purity**
 - smallBigBoot, 27
- * **reliability**
 - smallBigBoot, 27
- * **sum(z)**
 - ivzsums, 10
 - plot-sumz, 16
 - smallBigBoot, 27
 - sumz.tab, 30
 - titan, 33
- big.boot (smallBigBoot), 27
- big.boot(), 4, 33
- boot.titan, 2
- boot.titan(), 28, 29, 33
- cowplot::plot_grid(), 26
- density(), 26
- env.part, 4
- env.part(), 33
- geom_density_ridges(), 26
- getivz, 5
- getivz(), 9–13, 31, 33
- glades.env, 6
- glades.taxa, 7
- glades.titan, 7
- indval, 8
- indval(), 6, 10
- indvalp, 9
- indvalp(), 6, 9, 13
- indvalps (indvalp), 9
- indvalps(), 6, 9, 13
- indvals (indval), 8
- indvals(), 6, 10
- ivzsums, 10
- ivzsums(), 6, 31, 33
- ivzsums.f(), 31
- obs.summ, 11
- obs.summ(), 33
- package-TITAN2 (TITAN2), 35
- permiv, 13
- permiv(), 6

plot-cps, 14
plot-sumz, 16
plot-taxa, 19
plot_cps (plot-cps), 14
plot_cps(), 19, 22, 24, 26
plot_sumz (plot-sumz), 16
plot_sumz(), 11, 16, 22, 24, 26
plot_sumz_density, 22
plot_taxa (plot-taxa), 19
plot_taxa(), 15, 16, 19
plot_taxa_ridges, 25
plotCPs (plot-cps), 14
plotSumz (plot-sumz), 16
plotTaxa (plot-taxa), 19

scale_fill_gradient(), 25
small.boot (smallBigBoot), 27
small.boot(), 4, 33
smallBigBoot, 27
sumz.tab, 30
sumz.tab(), 33

tboot, 31
tboot(), 3, 4, 29
theme_ridges(), 26
titan, 33
titan(), 4, 12, 29, 31, 33
TITAN2, 35
txa.screen, 36
txa.screen(), 33