Package 'IgAScores'

September 1, 2020

Title Score Taxon-Level IgA Binding in IgA-Seq Experiments

Version 0.1.2

Description Functions to calculate indices used to score immunoglobulin A (IgA) binding of bacteria in IgA sequencing (IgA-Seq) experiments.
This includes the original Kau and Palm indices and more recent methods as described in Jackson et al. (2020) <doi:10.1101/2020.08.19.257501>.
Additionally the package contains a function to simulate IgA-Seq data and an example experimental data set for method testing.

Depends R (>= 3.5.0)

Suggests knitr, markdown

URL https://doi.org/10.1101/2020.08.19.257501

VignetteBuilder knitr

License GPL-3

Encoding UTF-8

RoxygenNote 7.1.1

NeedsCompilation no

Author Matthew Jackson [aut, cre, cph]

Maintainer Matthew Jackson <mattjackson4@gmail.com>

Repository CRAN

Date/Publication 2020-09-01 08:10:09 UTC

R topics documented:

Index   **13**

---

igaprobability   *IgA Probability*

Description

Calculate the IgA Postive/Negative Probability as described in Jackson et al. (2020, doi: 10.1101/ 2020.08.19.257501).

Usage

```
igaprobability(withinabund, gatesize, presortabund, nazeros = TRUE)
```

Arguments

withinabund	Abundance of the bacteria in the IgA gate under investigation (can be calculated for either the pos/high or neg/low gating) (abundances should sum to 1 not as a %).
gatesize	The fraction of events in the flow cytometer within the gate under investigation (as a decimal fraction not a %).
presortabund	Abundance of the bacteria in whole sample before sorting by IgA (abundances should sum to 1 not as a %).
nazeros	Return NA if the within and tot abundances are both zero. Default is TRUE.

Details

This function calculates the conditional probability that at bacteria will be sufficiently bound/not bound to immunoglobulin A (IgA) to end up in a given IgA gate based on its taxonomy. Calculated on one taxa for one sample.

This uses Bayes' theorem assuming:

- That the relative abundance of a given taxon in the IgA gate under question represents the probability of being that taxa given that it is within the IgA gate (either high or low).

- That the percentage of flow cytometery events binned into the IgA gate represents the probability of any bacteria being within the gate.

- That the abundance of the given taxon in the input sample (or whole fraction) represent the probability that any bacteria is assigned to the taxon. If there is insufficient levels of a taxa in the whole fraction to account for its abundance in the IgA gate, the function assumes all of the taxa fall within this gate (i.e. a probability of 1).

Further details can be found in Jackson et al. (2020, doi: 10.1101/2020.08.19.257501).

Value

A numeric value for the IgA Positive/Negative Probability (depending on the data used for 'withinabund' and 'gatesize') as defined in Jackson et al. (2020, doi: 10.1101/2020.08.19.257501).

Examples

```
igaprobability(withinabund=0.5,gatesize=0.05,presortabund=0.5)
```

---

igaprobabilityratio   *IgA Probability Ratio*

Description

Calculate the IgA Probability Ratio score as described in Jackson et al. (2020, doi: 10.1101/2020.08.19.257501).

Usage

```
igaprobabilityratio(
  posabund,
  negabund,
  possize,
  negsize,
  pseudo = 1e-05,
  scaleratio = TRUE,
  nazeros = TRUE
)
```

Arguments

posabund	Abundance of the bacteria in the IgA positive/high fraction (abundances should sum to 1 not as a %).
negabund	Abundance of the bacteria in the IgA negative/low fraction (abundances should sum to 1 not as a %).
possize	The fraction of events in the flow cytometer classed as IgA positive/high (as a decimal fraction not a %).
negsize	The fraction of events in the flow cytometer classed as IgA negative/low (as a decimal fraction not a %).
pseudo	Pseudo count added to both the IgA positive and negative abundance values prior to calculation. Defaults to 1e-5. Recommend setting to minimum observed abundance in whole dataset.
scaleratio	Should probratio scores be scaled to the pseudo count. Default is TRUE.
nazeros	Return NA if the pos and neg abundances are both zero. Default is TRUE.

Details

This function calculates the ratio of the immunoglobulin A (IgA) positive fraction probability relative to the IgA negative fraction probability for a single taxa in a single sample. These probabilities can individually be calculated using the igaprobability() function. As both calculations have the whole fraction taxon abundance as a denominator it cancels. This means the IgA probability ratio can be calculated without this information. Further details can be found in Jackson et al. (2020, doi: 10.1101/2020.08.19.257501).

Value

A numeric value for the IgA Probability Ratio as defined in Jackson et al. (2020, doi: 10.1101/2020.08.19.257501).

Examples

```
igaprobabilityratio(posabund=0.2,negabund=0.05,possize=0.05,negsize=0.6,pseudo=0.0002)
```

---

igascores	*Generate various scores for IgA binding in IgA-Seq experiments*

Description

Calculate various different IgA-Seq scores across all the taxa and samples in an experiment.

Usage

```
igascores(
  posabunds = NULL,
  negabunds = NULL,
  possizes = NULL,
  negsizes = NULL,
  pseudo = NULL,
  presortabunds = NULL,
  method = "probratio",
  scaleratio = TRUE,
  nazeros = TRUE
)
```

Arguments

posabunds	A dataframe of taxa abundances in the positive/high IgA gate samples. Samples as columns and taxa as rows, column and row names must match across abundance tables.
negabunds	A dataframe of taxa abundances in the negative/low IgA gate samples. Samples as columns and taxa as rows, column and row names must match across abundance tables.

possizes	A named vector containing the fraction of events in the IgA positive gate for each sample, with sample names matching abundance dataframes.
negsizes	A named vector containing the fraction of events in the IgA negative gate for each sample, with sample names matching abundance dataframes.
pseudo	The pseudo count to be used in scores. Default is 1e-5. Recommend setting to minimum observed abundance.
presortabunds	A dataframe of taxa abundances in the whole/initial samples. Samples as columns and taxa as rows, column and row names must match across abundance tables.
method	Method to use to score IgA binding. One of: "probratio","prob","kau","palm". Default is "probratio".
scaleratio	Should probratio scores be scaled to the pseudo count. Default is TRUE.
nazeros	Should taxa with zero abundance in both the posabunds and negabunds (posabunds and presortabunds for prob method) be scored as NA. Default is TRUE.

Details

This function enables calculation of a variety of different indices for scoring immunoglobulin A (IgA) binding to taxa in IgA sequencing (IgA-Seq) experiments. It is designed to be called on dataframes of abundance values, allowing easy calculation of scores across multiple taxa and samples. The igaprobabilityratio(), igaprobability(), kauindex() and palmindex() functions can be used to calculate scores for one taxa and one sample.

Scoring method can be chosen by specifying the method parameter as one of: "probratio", "prob", "kau", "palm" (Defaults to "probratio"). Each method requires different inputs as detailed below:

- probratio - equivalent to igaprobabilityratio() - requires two separate dataframes with iga positive abundances and iga negative abundances, two vectors with the sizes of the iga positive and negative gates per sample, and a pseudo count

- prob - equivalent to igaprobability() - requires a dataframe with iga pos or neg fraction abundances, a vector of iga pos or neg gate size per sample, and a dataframe of taxa abundances in the presort samples

- kau - equivalent to kauindex() - requires two separate dataframes with iga positive abundances and iga negative abundances, and a pseudo count

- palm - equivalent to palmindex() - requires two separate dataframes with iga positive abundances and iga negative abundances, and a pseudo count

Value

A data frame of IgA binding scores for all taxa and samples in the input data frame, generated using the scoring appraoch specified in 'method'.

Examples

```
pab <- data.frame(Samp1=c(0.01,0.02,0.03),Samp2=c(0.05,0.02,0.04))
rownames(pab) <- c("Taxon1","Taxon2","Taxon3")
nab <- data.frame(Samp1=c(0.08,0.2,0.11),Samp2=c(0.05,0.0,0.07))
rownames(nab) <- c("Taxon1","Taxon2","Taxon3")
ps <- c(0.04,0.1)
```

```
ns <- c(0.08,0.4)
preab <- data.frame(Samp1=c(0.1,0.3,0.2),Samp2=c(0.15,0.05,0.2))
rownames(preab) <- c("Taxon1","Taxon2","Taxon3")
igascores(posabunds=pab,negabunds=nab, possizes=ps, negsizes=ns,pseudo=0.009)
igascores(posabunds=pab, possizes=ps, presortabunds=preab, method="prob")
igascores(posabunds=pab, negabunds=nab, pseudo=0.009, method="palm")
igascores(posabunds=pab, negabunds=nab, pseudo=0.009, method="kau")
```

---

kauindex   *Kau Index*

Description

This function calculates the immunoglobulin A (IgA) Index as defined in Kau et al. (2015, doi: 10.1126/scitranslmed.aaa4877) for a single taxon in a single sample.

Usage

```
kauindex(posabund, negabund, pseudo = 1e-05, nazeros = TRUE)
```

Arguments

posabund	The abundance of the bacteria in the IgA positive/high fraction (abundances should sum to 1 not as a %).
negabund	The abundance of the bacteria in the IgA negative/low fraction (abundances should sum to 1 not as a %).
pseudo	Pseudo count added to both the IgA positive and negative fraction values prior to calculation. Defaults to 1e-5. Recommend setting to minimum observed abundance in whole dataset.
nazeros	Return NA if the pos and neg abundances are both zero. Default is TRUE.

Value

A numeric value for the Kau index as defined in Kau et al. (2015, doi: 10.1126/scitranslmed.aaa4877).

Examples

```
kauindex(posabund=0.1,negabund=0.2,pseudo=0.0002)
```

---

oligoMeta | *Metadata for the OligoMM12-Colitis experiment used as an example in the IgAScores package*

Description

Metadata associated with the oligoSpecies data set.

Usage

```
data(oligoMeta)
```

Format

An object of class `"tibble"`.

Details

Metadata for an experiment where mice with a defined gut microbiota (OligoMM12) were either given *Helicobacter hepaticus* and IL10R antibody or the antibody alone (the first developing colitis). These data accompany the species level counts in oligoSpecies and also include a negative extraction control. A subset of the combined condition were not properly colonised by *H.hepaticus* and excluded from later analyses. Further details can be found in Jackson et al. (2020, doi: 10.1101/2020.08.19.257501).

References

To come...

Examples

```
data(oligoMeta)
```

---

oligoSpecies | *Species level counts for the OligoMM12-Colitis experiment used as an example in the IgAScores package*

Description

Data from the colitis model described in Jackson et al. (2020, doi: 10.1101/2020.08.19.257501).

Usage

```
data(oligoSpecies)
```

Format

An object of class `"tibble"`.

Details

Species level counts for an experiment where mice with a defined gut microbiota (OligoMM12) were either given *Helicobacter hepaticus* and IL10R antibody or the antibody alone (the first developing colitis). Metadata for this experiment can be found in oligoMeta. Counts were generated from ASVs from the V4 region of the 16S rRNA gene processed using DADA2 and aligned to the *"RefSeq-RDP16S_v2_May2018.fa.gz"* database. Further details can be found in Jackson et al. (2020, doi: 10.1101/2020.08.19.257501).

References

To come...

Examples

```
data(oligoSpecies)
```

---

palmindex	*Palm Index*

Description

This function calculates the immunoglobulin A (IgA) Index as defined in Palm et al. (2014, doi: 10.1016/j.cell.2014.08.006) for a single taxon in a single sample.

Usage

```
palmindex(posabund, negabund, pseudo = 1e-05, nazeros = TRUE)
```

Arguments

posabund	Abundance of the bacteria in the IgA positive/high fraction.
negabund	Abundance of the bacteria in the IgA negative/low fraction.
pseudo	Pseudo count added to the abundance of the IgA negative fraction if the bacteria is not in that fraction. Defaults to 1e-5. Recommend setting to minimum observed abundance in whole dataset.
nazeros	Return NA if the pos and neg abundances are both zero. Default is TRUE.

Value

A numeric value for the Palm index as defined in Palm et al. (2014, doi: 10.1016/j.cell.2014.08.006).

Examples

```
palmindex(posabund=0.1,negabund=0.2,pseudo=0.0002)
```

---

relabund	*Relative Abundance from Counts*

Description

This function converts values in a dataframe to a fraction/percentage of the sum of their column.

Usage

```
relabund(counttable, percentage = FALSE)
```

Arguments

counttable	Data frame of numeric values with rows as observations and columns as samples.
percentage	Should values be returned as a percentage? i.e multiplied by 100. Default is FALSE (as required for most IgA scoring approaches).

Value

A data frame of the input data normalised by column (to sum to either 1 or 100).

Examples

```
taxcounts <- data.frame(Sample1=c(1,2,10,10),Sample2=c(3,10,5,1))
rownames(taxcounts) <- c("Taxon1","Taxon2","Taxon3","Taxon4")
relabund(taxcounts)
```

---

simulateigaseq	*Simulate an IgA-Seq dataset from a pre-defined set of IgA-binding probabilities*

Description

Simulates IgA-Seq to create datasets with a defined binding distribution that can be used to test scoring method performance

Usage

```
simulateigaseq(
  igavalmeans = NULL,
  igavalsds = NULL,
  nosamples = 10,
  samplingdepth = 1e+05,
  posthresh = 4,
  negthresh = 2,
```

```
    seed = 66,
    betweengroups = FALSE,
    betweenper = 10,
    betweensp = NULL
)
```

Arguments

igavalmeans	A vector of mean IgA values for as many species as you wish to simulate. Will default to an exponentially distributed vector of 10 species.
igavalsds	A vector of standard deviations that will be used to generate IgA value distributions alongside the means. Defaults to 1 for all values.
nosamples	The number of samples to generate simulated data from. Defaults to 10.
samplingdepth	The number of bacteria to simulate in each sample. Defaults to 100000.
posthresh	The IgA value threshold above which a bacteria will be considered IgA positive. Defaults to 4 (which is reasonable with the other defaults). It is recommended to run a simulation twice to determine reasonable thresholds on the first go.
negthresh	The IgA value threshold below which a bacteria will be considered IgA negative. Defaults to 2 (which is reasonable with the other defaults). It is recommended to run a simulation twice to determine reasonable thresholds on the first go.
seed	Seed for random number generation. Has a default so must be changed to rerun simulations.
betweengroups	If TRUE this will modify starting abundances of half of the samples similarly (by adding betweenper% of total counts to a single species) to simulate the case where there is an abundance shift without a change in IgA binding affinity. Defaults to FALSE.
betweenper	Percentage of total counts to add to a species in the second group in the between-groups mode.
betweensp	Species (by index) to increased in between groups simulation. Chosen at random if NULL (default).

Details

This function will generate a simulated immunoglobulin A sequencing (IgA-Seq) data set starting from a list containing the mean (and standard deviations) of IgA binding values expected for each species and cut-offs for defining the IgA positive and negative gates. The input is a vector giving the average IgA value of each species (any arbitrary value that will represent the relative level of IgA binding between the species, ensure standard deviation and cut-offs are in the same magnitude). These values are treated as the means of a normal distribution of IgA binding values for each species. Species counts are generated on a log distribution for a given number of samples at an even depth. For each bacteria in each sample, an IgA binding value is then assigned by sampling from its species IgA value distribution. The value thresholds defining the positive and negative gates are then used to generate positive and negative counts tables of the bacteria whose values fall into these groups. A second mode can also be used (by toggling betweengroups) that will introduce a consistent abundance change in half the samples by increasing one species in them. This can be used to simulate case-control experiments where, as an example, one taxa has bloomed. Further details can be found in Jackson et al. (2020, doi: 10.1101/2020.08.19.257501).

Note: IgA values are simulated for each bacteria in each sample, setting the combination of the samplingdepth, number of species, and number of samples too high will slow the data generation.

Value

A list containing the simulated data set and relevant input parameters.

- presortcounts - A data frame containing simulated species counts for each sample in the pre-sort sample.

- presortabunds - presortcounts as relative abundances.

- poscounts - A data frame containing simulated species counts for each sample in the IgA positive fraction.

- posabunds - poscounts as relative abundances.

- negcounts - A data frame containing simulated species counts for each sample in the IgA negative fraction.

- negabunds - negcounts as relative abundances.

- possizes - A vector of the IgA positive fraction sizes for each sample.

- negsizes - A vector of the IgA negative fraction sizes for each sample.

- igabinding - A long format data frame containing the simulated IgA binding values for all simulated bacteria used to generate the count tables.

- igavalmeans - A vector of the mean IgA values for each species used in the simulation.

- igavalsds - A vector of the standard deviations of the IgA values for each species used in the simulation.

- posthresh - Numeric, the lower threshold used to determine a bacteria is IgA postive in the simulation.

- negthresh - Numeric, the upper threshold used to determine a bacteria is IgA negative in the simulation.

- expgroup - A vector showing class labels for the experimental group of each sample in the experiment. Will be uniform unless doing between group simulations.

- expspecies - Numeric, showing which species was modelled as differentially abundant between experimental groups when carryingout between group simulations.

Examples

```
dat <- simulateigaseq(c(0.1,1,10,15),rep(1,4),posthresh=8,negthresh=4,samplingdepth=100)
```

---

taxnamesplit	*Extract name at a given taxonomic level from a full name in the "p__;c__;o__;f__;g__;s__" format*

Description

This function splits a full taxonomic lineage as a given level and returns the latter half.

Usage

```
taxnamesplit(names, level = "genus")
```

Arguments

names	Name string/ vector of name strings
level	taxonomic level to split at must be in range phylum to species (default is genus).

Value

A string (or vector of strings if input is a vector) containing the second part of the input string split at the given taxonomic level.

Examples

```
taxnamesplit("p__Bacteroidetes;c__Bacteroidia","class")
```

Index