Package ‘OTUbase’

January 15, 2017

Type Package

Title Provides structure and functions for the analysis of OTU data

Description Provides a platform for Operational Taxonomic Unit based analysis

Version 1.24.0

Date 2010-09-10

Author Daniel Beck, Matt Settles, and James A. Foster

Maintainer Daniel Beck <danlbek@gmail.com>

Depends R (>= 2.9.0), methods, S4Vectors, IRanges, ShortRead (>= 1.23.15), Biobase, vegan

Imports Biostrings

License Artistic-2.0

LazyLoad yes

biocViews Sequencing, DataImport

NeedsCompilation no

R topics documented:

OTUbase-package ................................................................. 2
.OTUset-class ................................................................. 2
.TAXset-class ................................................................. 3
abundance ................................................................. 4
accessors ................................................................. 5
assignmentData ............................................................ 6
clusterSamples ............................................................. 7
otherGenerics ............................................................... 8
other_functions .............................................................. 9
readOTUset ................................................................. 10
readTAXset ................................................................. 11
sampleData ................................................................. 12
subOTUset ................................................................. 13
subTAXset ................................................................. 13

Index 15
The OTUBase package: A tool for organizing and manipulating Operational Taxonomic Unit data

Description

The OTUBase Base class for OTU data

Details

Package: OTUbase
Type: Package
Version: 0.1.0
Date: 2010-04-05
License: Artistic-2.0
LazyLoad: yes

OTUbase includes a number of OTUset type classes which provide structure for OTU based data. These classes allow the user to store information that may be useful in the analysis of OTUs. Slots are provided for sequence and quality values, OTU classifications, Sample identifications, and metadata associated with samples and OTUs. In addition, basic functions are provided for the analysis and visualization of the data. In addition to OTU type analysis, classification data is also supported with the TAXset classes.

Author(s)

Daniel Beck - <danlbek@gmail.com>, Matt Settles - <msettles@uidaho.edu>, and James Foster
Maintainer: Daniel Beck - <danlbek@gmail.com>

References

An_introduction_to_OTUbase.pdf

"OTUset" class for OTU data

Description

This class provides a way to store and manipulate operational taxonomic unit data. "OTUset" is inherited by "OTUsetQ", "OTUsetF", and "OTUsetB". The user will want to use "OTUsetQ" when quality data is available, "OTUsetF" when sequence data (without quality data) is available, and "OTUsetB" when only OTU and sample data are available.

Slots

OTUsetB includes Slots id, sampleID,.otuID, sampleData, assignmentData.
OTUsetF includes Slots id sampleID, otuID, sampleData, assignmentData, sread.
OTUsetQ includes Slots id sampleID, otuID, sampleData, assignmentData, sread, quality.
Methods

Methods include:

- **id** provides access to the id slot of object
- **sampleID** provides access to the sampleID slot of object
- **otuID** provides access to the otuID slot of object
- **sampleData** provides access to the sampleData slot of object
- **assignmentData** provides access to the assignmentData slot of object
- **sread** provides access to the sread slot of object
- **quality** provides access to the quality slot of object
- **seqnames** returns the first word of the id line. Intended to extract the sequence name from other sequence information.
- **nsamples** returns the number of samples in an OTUset object
- **notus** returns the number of OTUs in an OTUset object

Examples

```
showClass(".OTUset")
showMethods(class=".OTUset")
showClass("OTUsetQ")
```

Description

This class provides a way to store and manipulate read-classification data. ".TAXset" is inherited by "TAXsetQ", "TAXsetF", and "TAXsetB". The user will want to use "TAXsetQ" when quality data is available, "TAXsetF" when sequence data (without quality data) is available, and "TAXsetB" when only classification and sample data are available.

Slots

- **TAXsetB** includes Slots id, sampleID, tax, sampleData, assignmentData.
- **TAXsetF** includes Slots id, sampleID, tax, sampleData, assignmentData, sread.
- **TAXsetQ** includes Slots id, sampleID, tax, sampleData, assignmentData, sread, quality.

Methods

Methods include:

- **id** provides access to the id slot of object
- **sampleID** provides access to the sampleID slot of object
- **tax** provides access to the tax slot of object
- **sampleData** provides access to the sampleData slot of object
**assignmentData** provides access the assignmentData slot of object

**sread** provides access to the sread slot of object

**quality** provides access to the quality slot of object

**seqnames** returns the first word of the id line. Intended to extract the sequence name from other sequence information.

**nsamples** returns the number of samples in a TAXset object

**show signature** (object = "TAXset"): provides a brief summary of the object, including its class, number of sequences, and number of samples.

---

**Examples**

```
showClass(".TAXset")
showMethods(class=".TAXset")
showClass("TAXsetQ")
```

---

**Description**

abundance generates an abundance table. This table can be either weighted or unweighted.

**Usage**

```
abundance(object, ...)
```

**Arguments**

- **object** An OTUset or a TAXset object
- **...** Additional arguments. These will depend on if the object is an OTUset or a TAXset object.

**Details**

These are other arguments passed to abundance

- **taxCol** If generating the abundance from a TAXset object, taxCol selects the column of the tax dataframe from which to calculate the abundance.
- **assignmentCol** If generating the abundance from an OTUset object assignmentCol will select a column of the assignmentData dataframe to use when calculating abundance. This will override the default of creating an abundance table of the OTUs and instead create an abundance table of a column in the assignmentData dataframe.
- **sampleCol** sampleCol generates the abundance table using a column in the sampleData dataframe instead of the default of using the sampleID.
- **collab** An optional parameter that selects a column of the sampleData dataframe to use when labeling the columns of the abundance table.
- **weighted** By default this is FALSE. When set to TRUE abundance will return proportional abundances.
Value

The returned value will be a data.frame.

Examples

```r
## locate directory with data
dirPath <- system.file("extdata/Sogin_2006", package="OTUbase")

## read in data into OTUset object
soginOTU <- readOTUset(dirPath=dirPath, level="0.03", samplefile="sogin.groups", fastafile="sogin.fasta", otufile="sogin.unique.filter.fn.list", sampleADF="sample_metadata.txt")

## calculate abundance
abundance(soginOTU, collab="Site")
```

---

**accessors**

**Accessor functions for OTUset and TAXset objects**

**Description**

These functions provide access to some of the slots of OTUset and TAXset objects. `otuID` returns the otuID slot of OTUset objects. `sampleID` returns the sampleID slot of both OTUset and TAXset objects. `tax` and `tax<-` return and replace the tax slot of TAXset objects.

**Usage**

```r
sampleID(object, ...)
otuID(object, ...)
tax(object, ...)
tax(object)<-value
```

**Arguments**

- **object**
  - An OTUset or a TAXset object
- **value**
  - The replacement value for `tax`
- **...**
  - Added for completeness. Enables the passing of arguments.

**Value**

- `sampleID` and `otuID` return a character. `tax` returns a data.frame.

**See Also**

`ShortRead`
Examples

```r
## locate directory with data
dirPath <- system.file("extdata/Sogin_2006", package="OTUbase")

## read in data into OTUset object
soginOTU <- readOTUset(dirPath=dirPath, level="0.03", samplefile="sogin.groups", fastafile="sogin.fasta", otufile="sogin.unique.filter.fn.list", sampleADF="sample_metadata.txt")

## get the sampleID slot
sampleID(soginOTU)

## get the otuID slot
otuID(soginOTU)
```

Description

These accessors access and replace the assignmentData slot of OTUbase objects. assignmentData is an AnnotatedDataFrame. assignmentData and assignmentData<- access and replace this AnnotatedDataFrame. assignmentLabels and assignmentLabels<- access and replace the labels of this AnnotatedDataFrame. aData and aData<- access and replace the dataframe component of the AnnotatedDataFrame.

assignmentNames returns the assignment names present in the assignmentData slot.

Usage

```r
aData(object,...)
aData(object)<-value
assignmentData(object,...)
assignmentData(object)<-value
assignmentLabels(object,...)
assignmentLabels(object)<-value
assignmentNames(object,...)
```

Arguments

- `object`: An OTUset or a TAXset object
- `value`: The replacement value for assignmentData or assignmentLabels
- `...`: Added for completeness. Enables the passing of arguments.

Value

- `aData` returns a dataframe. `assignmentData` returns an AnnotatedDataFrame. `assignmentLabels` returns a character. `assignmentNames` returns a character.
clusterSamples

Examples

```r
## locate directory with data
dirPath <- system.file("extdata/Sogin_2006", package="OTUbase")

## read in data into OTUset object
soginOTU <- readOTUset(dirPath=dirPath, level="0.03", samplefile="sogin.groups", fastafile="sogin.fasta", otufile="sogin.unique.filter.fn.list", sampleADF="sample_metadata.txt")

## get the aData dataframe
aData(soginOTU)

## get the assignmentData slot
assignmentData(soginOTU)
```

description

This function is a wrapper for the vegan function vegedist and hclust. It allows the user to cluster samples using a number of different distance measure and clustering methods. Please see the documentation for vegedist and hclust for a more indepth explanation.

Usage

`clusterSamples(object, ...)`

Arguments

- `object` An OTUset or a TAXset object
- `...` Additional arguments. These will depend on if the object is an OTUset or a TAXset object.

Details

These are other arguments passed to `clusterSamples`. For further information on specific arguments, please see `abundance`, `vegedist`, or `hclust`.

- `taxCol` Column of the `tax` slot dataframe on which to cluster (unique to TAXset objects). Passed to the abundance function.
- `assignmentCol` Column of the `assignmentData` dataframe used to classify sequences for clustering. This overrides the default of using the OTUs to cluster samples. This is passed to the abundance function.
- `collab` Specifies a column of the `sampleData` dataframe that will provide the sample labels for the cluster analysis. This is passed to the abundance function.
- `distmethod` The distance method to be used. This value is passed to the vegedist function. The default is the Bray-Curtis distance.
- `clustermethod` The clustering method to be used. This value is passed to the hclust function. The default is `complete` clustering.
Examples

```r
## locate directory with data
dirPath <- system.file("extdata/Sogin_2006", package="OTUbase")

## read in data into OTUset object
soginOTU <- readOTUset(dirPath=dirPath, level="0.03", samplefile="sogin.groups", fastafile="sogin.fasta", otufile="sogin.unique.filter.fn.list", sampleADF="sample_metadata.txt")

## cluster samples
clusterSamples(soginOTU, collab="Site", distmethod="jaccard")
```

otherGenerics

## Description

Various functions. `notus` returns the number of OTUs in an OTUset object. `nsamples` returns the number of samples in either an OTUset or a TAXset object. `seqnames` returns the sequence names of the OTUset or TAXset object without the extra information commonly present with the id.

## Usage

```r
notus(object, ...)
samples(object, ...)
seqnames(object, ...)
```

## Arguments

- **object**
  - An OTUset or a TAXset object.
- **...**
  - Other arguments. These are currently nonfunctional.

## Examples

```r
dirPath <- system.file("extdata/Sogin_2006", package="OTUbase")

## read in data into OTUset object
soginOTU <- readOTUset(dirPath=dirPath, level="0.03", samplefile="sogin.groups", fastafile="sogin.fasta", otufile="sogin.unique.filter.fn.list", sampleADF="sample_metadata.txt")

## get the number of OTUs
notus(soginOTU)

## get the number of samples
nsamples(soginOTU)
```
**Description**

These are other functions available. Caution is advised when using them. Some are still in development and others only work on specific objects (OTUset or TAXset).

**Usage**

```r
getOTUs(object, colnum, value, exact)
getSamples(object, colnum, value, exact)
o_diversity(object, ...)
o_estimateR(object, ...)
```

**Arguments**

- `object`: An OTUset or a TAXset object.
- `colnum`: The column of the sampleData or assignmentData dataframe that contains the value.
- `value`: The desired value.
- `exact`: If exact=T value must match perfectly. If exact=F value will grep instead of match.
- `...`: Other arguments. Often these are passed to abundance

**Details**

- `getOTUs` Returns OTU names that match given values in the assignmentData dataframe.
- `getSamples` Returns sample names that match given values in the sampleData dataframe.
- `o_diversity` Wrapper for vegan's diversity function.
- `o_estimateR` Wrapper for vegan's estimateR function.
- `otuseqplot` Plots the samples according to number of OTUs and number of sequences.
- `otusize` Returns the size of each OTU.
- `otuspersample` Lists the number of OTUs in each sample.
- `rseqplot` Plots the samples by estimated richness and number of sequences.
- `seqspersample` Returns the number of sequences in each sample.
- `sharedotus` Returns the number of OTUs shared between samples.

**Examples**

```r
## locate directory with data
dirPath <- system.file("extdata/Sogin_2006", package="OTUbase")

## read in data into OTUset object
soginOTU <- readOTUset(dirPath=dirPath, level="0.03", samplefile="sogin.groups", fastfile="sogin.fasta", otufile="sogin.unique.filter.fn.list", sampleADF="sample_metadata.txt")

getSamples(soginOTU, colnum="Site", value="Labrador", exact=FALSE)
```
Description

This function reads in data and creates an OTUset object

Usage

```r
readOTUset(dirPath, otufile, level, fastafile, qualfile, samplefile, sampleADF, assignmentADF, sADF.names, aADF.names, rdp, otufiletype)
```

Arguments

- `dirPath`: The directory path were the datafile are located. This is the current directory by default.
- `otufile`: The OTU file. The only format currently supported is the Mothur format for .list files.
- `level`: The OTU clustering level. By default this is 0.03. This level must correspond to levels present in the otufile.
- `fastafile`: The fasta file. This is read in by ShortRead.
- `qualfile`: The quality file. This is read in by ShortRead.
- `samplefile`: The sample file. Currently this must be in Mothur format (.groups).
- `sampleADF`: The sample meta data file. This is in AnnotatedDataFrame format.
- `assignmentADF`: The assignment meta data file (the OTU meta data). This is generally in AnnotatedDataFrame format although it is also possible to read in an RDP classification file if there is only one read classification for each cluster and `rdp=TRUE`.
- `sADF.names`: The column of the sampleADF file that has the sample names.
- `aADF.names`: The column of the assignmentADF file that has the assignment names.
- `rdp`: By default this is FALSE. Change to TRUE if assignmentADF is an RDP classification file. The RDP file must be in the fixed format.
- `otufiletype`: The type of OTU file. Takes values "mothur", "cdhit", and "blastclust". Defaults to "mothur".

Examples

```r
dirPath <- system.file("extdata/Sogin_2006", package="OTUbase")

## read in data into OTUset object
soginOTU <- readOTUset(dirPath=dirPath, level="0.03", samplefile="sogin.groups", fastafile="sogin.fasta", otufile="sogin.unique.filter.fn.list")
soginOTU
```

Description

Function to read in data and create a TAXset object

Usage

readTAXset(dirPath, taxfile, namefile, fastafile, qualfile, samplefile, sampleADF, assignmentADF, sADF.names, aADF.names, type, ...)

Arguments

dirPath The directory path where the datafile are located. This is the current directory by default.
taxfile The classification file. The default format is RDP’s fixed format.
nnamefile A names file in the Mothur format. This is used to add removed unique sequences back into the dataset.
fastafile The fasta file. This is read in by ShortRead.
qualfile The quality file. This is read in by ShortRead.
samplefile The sample file. Currently this must be in Mothur format (.groups).
sampleADF The sample meta data file. This is in AnnotatedDataFrame format.
assignmentADF The assignment meta data file (the OTU meta data) This is in AnnotatedDataFrame format.
sADF.names The column of the sampleADF file that has the sample names.
aADF.names The column of the assignmentADF file that has the assignment names.
type This is the type of taxfile. By default this is the RDP fixed format. However, if type is changed to anything else the read.table function is used to read in the taxfile. In this case the first column of the taxfile must be the sequence names.
... Additional arguments passed to read.table to read in the taxfile.

Examples

dirPath <- system.file("extdata/Sogin_2006", package="OTUbase")

## read in data into TAXset object
soginTAX <- readTAXset(dirPath=dirPath, samplefile="sogin.groups", fastafile="sogin.fasta", taxfile="sogin.unique.fix.tax", namefile="sogin.names")
soginTAX
sampleData

Description

These functions access and replace the sampleData slot of OTUbase objects. sampleData and sampleData<- access and replace the AnnotatedDataFrame sampleData. sampleLabels and sampleLabels<- access and replace the labels of this AnnotatedDataFrame. sData and sData<- access and replace the dataframe component of the AnnotatedDataFrame.

Usage

sData(object,...)
sData(object)<-value
sampleData(object,...)
sampleData(object)<-value
sampleLabels(object,...)
sampleLabels(object)<-value

Arguments

object An OTUset or a TAXset object
value The replacement value for sampleData or sampleLabels
... Added for completeness. Enables the passing of arguments.

Value

sData returns a dataframe. sampleData returns an AnnotatedDataFrame. sampleLabels returns a character. assignmentNames returns a character.

Examples

```r
## locate directory with data
dirPath <- system.file("extdata/Sogin_2006", package="OTUbase")

## read in data into OTUset object
ginOTU <- readOTUset(dirPath=dirPath, level="0.03", samplefile="sogin.groups", fastafile="sogin.fasta", otufile="sogin.unique.filter.fn.list")

## get the sData dataframe
sData(ginOTU)

## get the sampleData slot
sampleData(ginOTU)
```
**subOTUset**

**Description**
Function to get a subset of an OTUset object.

**Usage**

```r
subOTUset(object, samples, otus)
```

**Arguments**
- `object`: An OTUset object
- `samples`: A list of sample names
- `otus`: A list of OTU names

**Value**
`subOTUset` returns an OTUset

**Examples**

```r
## locate directory with data
dirPath <- system.file("extdata/Sogin_2006", package="OTUbase")

## read in data into OTUset object
soginOTU <- readOTUset(dirPath=dirPath, level="0.03", samplefile="sogin.groups", fastafile="sogin.fasta", otufile="sogin.unique.filter.fn.list", sampleADF="sample_metadata.txt")

## get subset of soginOTU
subOTUset(soginOTU, samples=getSamples(soginOTU, colnum="Site", value="Labrador", exact=FALSE))
```

**subTAXset**

**Description**
Function to get a subset of an TAXset object.

**Usage**

```r
subTAXset(object, samples)
```

**Examples**

```r
## locate directory with data
dirPath <- system.file("extdata/Sogin_2006", package="OTUbase")

## read in data into TAXset object
soginTAX <- readTAXset(dirPath=dirPath, level="0.03", samplefile="sogin.groups", fastafile="sogin.fasta", otufile="sogin.unique.filter.fn.list", sampleADF="sample_metadata.txt")

## get subset of soginTAX
subTAXset(soginTAX, samples=getSamples(soginTAX, colnum="Site", value="Labrador", exact=FALSE))
```
subTAXset

Arguments

object  An TAXset object
samples  A list of sample names

Value

subTAXset returns an TAXset

Examples

```r
## locate directory with data
dirPath <- system.file("extdata/Sogin_2006", package="OTUbase")

## read in data into TAXset object
soginTAX <- readTAXset(dirPath=dirPath, samplefile="sogin.groups", fastfile="sogin.fasta", taxfile="sogin.unique.fix.tax", namefile="sogin.names", sampleADF="sample_metadata.txt")

## get subset of soginTAX
subTAXset(soginTAX, samples=getSamples(soginTAX, colnum="Site", value="Labrador", exact=FALSE))
```
Index

*Topic abundance
   abundance.4, 7
   abundance,.OTUset-method(abundance),4
   abundance,.TAXset-method(abundance),4
   accessor,5
   aData(assignmentData),6
   aData,.OTUbase-method(assignmentData),6
   aData<-(assignmentData),6
   aData<-.OTUbase.data.frame-method
       (assignmentData),6
   aData<-.OTUbase-method
       (assignmentData),6
   assignmentData,6
   assignmentData,.OTUbase-method
       (assignmentData),6
   assignmentData<-(assignmentData),6
   assignmentData<-.OTUbase.AnnotatedDataFrame-method
       (assignmentData),6
   assignmentData<-.OTUbase-method
       (assignmentData),6
   assignmentLabels(assignmentData),6
   assignmentLabels,.OTUbase-method
       (assignmentData),6
   assignmentLabels<-(assignmentData),6
   assignmentLabels<-.OTUbase.character-method
       (assignmentData),6
   assignmentLabels<-.OTUbase-method
       (assignmentData),6
   assignmentNames(assignmentData),6
   assignmentNames,.OTUbase-method
       (assignmentData),6
   clusterSamples,7
   clusterSamples,.OTUset-method
       (clusterSamples),7
   clusterSamples,.TAXset-method
       (clusterSamples),7
   clusterSamples-generic
       (clusterSamples),7
   clusterSamples-method(clusterSamples),7
   getOTUs(other_functions),9
   getSamples(other_functions),9
   hclust,7
   notus(otherGenerics),8
   notus,.OTUset-method(otherGenerics),8
   nsamples(otherGenerics),8
   nsamples,.OTUbase-method
       (otherGenerics),8
   o_diversity(other_functions),9
   o_estimateR(other_functions),9
   other_functions,9
   otherGenerics,8
   OTUbase(OTUbase-package),2
   OTUbase-package,2
   otuID(accessors),5
   otuID,.OTUset-method(accessors),5
   otuseqplot(other_functions),9
   OTUset(.OTUset-class),2
   OTUsetB(.OTUset-class),2
   OTUsetB-class(.OTUset-class),2
   OTUsetF(.OTUset-class),2
   OTUsetF-class(.OTUset-class),2
   OTUseqplot(.OTUset-class),2
   otusize(other_functions),9
   otuspersample(other_functions),9
   readOTUset,10
   readTAXset,11
   rseqplot(other_functions),9
   sampleData,12
sampleData,.OTUbase-method
  (sampleData), 12
sampleData<-(sampleData), 12
sampleData<-.OTUbase,AnnotatedDataFrame-method
  (sampleData), 12
sampleData<-.OTUbase-method
  (sampleData), 12
sampleID(accessors), 5
sampleLabels(sampleData), 12
sampleLabels,.OTUbase-method
  (sampleData), 12
sampleLabels<-(sampleData), 12
sampleLabels<-.OTUbase,character-method
  (sampleData), 12
sampleLabels<-.OTUbase-method
  (sampleData), 12
sData(sampleData), 12
sData,.OTUbase-method(sampleData), 12
sData<-(sampleData), 12
sData<-.OTUbase,data.frame-method
  (sampleData), 12
sData<-.OTUbase-method
  (sampleData), 12
seqnames(otherGenerics), 8
seqnames,.OTUbase-method
  (otherGenerics), 8
seqspersample(other_functions), 9
sharedotus(other_functions), 9
ShortRead, 5
subOTUset, 13
subTAXset, 13
tax(accessors), 5
tax,.TAXset-method(accessors), 5
tax<-(accessors), 5
tax<-.TAXset,data.frame-method
  (accessors), 5
tax<-.TAXset-method(accessors), 5
TAXset(.TAXset-class), 3
TAXsetB(.TAXset-class), 3
TAXsetB-class(.TAXset-class), 3
TAXsetF(.TAXset-class), 3
TAXsetF-class(.TAXset-class), 3
TAXsetQ(.TAXset-class), 3
TAXsetQ-class(.TAXset-class), 3
vegdist, 7