Package ‘microcontax’

October 13, 2022

Encoding UTF-8

Type Package

Title The ConTax Data Package

Version 1.2

Date 2020-08-10

Author Hilde Vinje, Kristian Hovde Liland, Lars Snipen.

Maintainer Lars Snipen <lars.snipen@nmbu.no>

Description The consensus taxonomy for prokaryotes is a set of data-sets for best possible taxonomic classification based on 16S rRNA sequence data.

License GPL-2

Depends R (>= 3.5.0)

Imports microseq

Suggests microcontax.data

Additional_repositories https://khliland.github.io/drat/

RoxygenNote 7.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2020-08-11 14:50:06 UTC

R topics documented:

microcontax-package 2
contax.trim 2
fullTaxonomy 4
genusLookup 5
getDomain 6
medoids 7
taxonomy.table 8

Index **9**

`microcontax-package` *The ConTax data package*

Description

The consensus taxonomy for prokaryotes is a package of data sets designed to be the best possible for training taxonomic classifiers based on 16S rRNA sequence data.

Usage

```
microcontax()
```

Details

Package:	microcontax
Type:	Package
Version:	1.2
Date:	2020-06-06
License:	GPL-2

Author(s)

Hilde Vinje, Kristian Liland, Lars Snipen.
Maintainer: Lars Snipen <lars.snipen@nmbu.no>

See Also

`microseq`

`contax.trim` *The ConTax data set*

Description

The trimmed version of the ConTax data set.

Usage

```
data(contax.trim)
```

Details

contax.trim is a data.frame object containing 38 781 full-length 16S rRNA sequences. It is the trimmed version of the full data set (see below). Large taxa (many sequences) have been trimmed as described in Vinje et al. (2016) to obtain a data set with a more even representation of the prokaryotic taxonomy.

The contax.full is the full consensus taxonomy data set as described in Vinje et al. (2016). The data set is too large for CRAN and thus available as a separate package microcontax.data. See example below for how to obtain contax.full.

The Header of every sequence starts with a unique tag, in this case the text "ConTax" and some integer. This is followed by a token describing the origin of the sequence. It is typically

"Intersection=SRG"

meaning it is found in both the Silva, RDP and Greengenes data repository. Intersections can also be SR, SG and RG if the sequence was found in two repositories only. The taxonomy information for each sequence is found in the third token. It follows a commonly used format:

"k__<...>;p__<...>;c__<...>;o__<...>;f__<...>;g__<...>;"

where <...> is some proper text. The letters, followed by a double underscore, refer to the taxonomic levels Domain (Kingdom), Phylum, Class, Order, Family and Genus. Here is an example of a proper string:

"k__Bacteria;p__Firmicutes;c__Bacilli;o__Bacillales;f__Staphylococcaceae;g__Staphylococcus;"

As long as this format is used the taxonomy information can be extracted by the supplied extractor-functions getDomain, getPhylum,...,getGenus.

Author(s)

Hilde Vinje, Kristian Hovde Liland, Lars Snipen.

See Also

medoids, getDomain, contax.full.

Examples

```
data(contax.trim)
dim(contax.trim)

# Write to FASTA-file
## Not run:
writeFasta(contax.trim,out.file="ConTax_trim.fasta")

# Install microcontax.data with the BIG contax.full data set
if (!requireNamespace("microcontax.data", quietly = TRUE)) {
 install.packages("microcontax.data")
}
# Load data
data("contax.full", package = "microcontax.data")

## End(Not run)
```

`fullTaxonomy` *The full taxonomy of a genus*

Description

Converts a genus to a string containing the full taxonomy.

Usage

```
fullTaxonomy(genera)
```

Arguments

`genera`	A vector of texts, the genera names to look up.

Details

The argument `genera` must consist of names in the `Genus` column of the data set `taxonomy.table`.

"k__<...>;p__<...>;c__<...>;o__<...>;f__<...>;g__<...>;"

where <...> is some proper text.

Value

A character vector containing the taxonomy information.

Author(s)

Lars Snipen.

See Also

`taxonomy.table`, `genusLookup`.

Examples

```
genera <-  c("Bacillus","Clostridium","Hyphomonas")
fullTaxonomy(genera)
```

genusLookup *Taxonomy lookup*

Description

Extracting taxonomic information from the `taxonomy.table`.

Usage

```
genusLookup(genera, rank = "Phylum")
```

Arguments

`genera`	A vector of texts, the genera names to look up.
`rank`	A single text, the level of the taxonomy to look up.

Details

Function for looking up higher-level taxonomy of specified genera.

The argument `genera` must consist of names in the `Genus` column of the data set `taxonomy.table`.

Value

A character vector containing the taxonomy information. Names in `genera` not recognized will return `NA`. Please note that there are some cases of un-assigned taxonomy at some ranks (Class, Order or Family), this is returned as "unknown".

Author(s)

Hilde Vinje, Lars Snipen.

See Also

`taxonomy.table`.

Examples

```
genus <-  c("Acidilobus","Nitrosopumilus","Hyphomonas")
genusLookup(genus, rank = "Phylum")
genusLookup(genus, rank = "Class")
```

getDomain *Extractor functions for ConTax data*

Description

Extracting taxonomic information from ConTax data sets.

Usage

```
getDomain(header)
getPhylum(header)
getClass(header)
getOrder(header)
getFamily(header)
getGenus(header)
getTag(header)
getTaxonomy(header)
```

Arguments

`header`	A vector of texts, typically the `Header` from a table, containing taxonomy information in the proper format.

Details

The ConTax data sets are tables in the FASTA format (see `readFasta`), where the `Header` column contains texts according to a strict format.

The `header` always starts with a short text, a Tag, which is a unique identifier for every sequence. The function `getTag` will extract this from the `header`.

After the Tag follows one or more tokens. One of these tokens must be a string with the following format:

"k__<...>;p__<...>;c__<...>;o__<...>;f__<...>;g__<...>;"

where <...> is some proper text. Here is an example of a proper string:

"k__Bacteria;p__Firmicutes;c__Bacilli;o__Bacillales;f__Staphylococcaceae;g__Staphylococcus;"

The functions `getDomain`, ..., `getGenus` extracts the corresponding information from the `header`. `getTaxonomy` combines all taxonomy extractors, combines these in a table and imputes missing taxa with parent taxa.

Value

A vector containing the sub-texts extracted from each `header` text, but `getTaxonomy` returns a table with the full taxonomy, one row for each input `header`

Author(s)

Lars Snipen.

See Also

`contax.trim`, `medoids`.

Examples

```
data(contax.trim)
getTag(contax.trim$Header)
getGenus(contax.trim$Header)
getPhylum(contax.trim$Header)
```

`medoids` *The ConTax medoids*

Description

The genus medoids from the ConTax data set.

Usage

```
data(medoids)
```

Details

`medoids` is a `data.frame` object containing the medoide sequences for each genus in the ConTax data sets (both `contax.trim` and `contax.full`).

The medoide sequence in a genus is the sequence having the smallest sum of distance to all other members of the same genus. Thus, it is the sequence closest to the centre of the genus. The medoids can be used as the representative of each genus, e.g. for building trees for the entire taxonomy.

The taxonomy information for each sequence can be extracted from the `Header` column by the supplied extractor-functions `getDomain`, `getPhylum`,...,`getGenus`.

Author(s)

Hilde Vinje, Kristian Hovde Liland, Lars Snipen.

See Also

`contax.full`, `getDomain`.

Examples

```
data(medoids)
summary(medoids)
```

taxonomy.table *Taxonomy look-up table*

Description

A data frame consisting of the taxonomy information used in the ConTax data sets.

Usage

```
data(taxonomy.table)
```

Details

`taxonomy.table` is a `data.frame` consisting of the seven columns Domain, Phylum, Class, Order, Family, Genus and LPSN. The first six are taxonomy informations, the last is "Yes" or "No" indiocating if the Genus listed is also found in the List of prokaryotic names with standing in nomenclature (LPSN) database, see http://www.bacterio.net/.

Each row contains the taxonomy information for a genus, hence the number of rows equals the number of unique genera.

To quickly look-up the higher rank taxonomy for a given genus, see the function `genusLookup`.

Author(s)

Hilde Vinje, Kristian Hovde Liland, Lars Snipen.

See Also

`genusLookup`, `contax.full`, `contax.trim`, `getDomain`.

Examples

```
data(taxonomy.table)
dim(taxonomy.table)
taxonomy.table[1:10,]
genusLookup(taxonomy.table$Genus[1:10], rank = "Family")
```

Index

∗ **package**
    microcontax-package, 2

contax.full, *3*, *7*, *8*
contax.trim, 2, *7*, *8*

fullTaxonomy, 4

genusLookup, *4*, 5, *8*
getClass *(*getDomain*)*, 6
getDomain, *3*, 6, *7*, *8*
getFamily *(*getDomain*)*, 6
getGenus, *3*, *7*
getGenus *(*getDomain*)*, 6
getOrder *(*getDomain*)*, 6
getPhylum, *3*, *7*
getPhylum *(*getDomain*)*, 6
getTag *(*getDomain*)*, 6
getTaxonomy *(*getDomain*)*, 6

medoids, *3*, *7*, 7
microcontax *(*microcontax-package*)*, 2
microcontax-package, 2
microseq, *2*

readFasta, *6*

taxonomy.table, *4*, *5*, 8