Package ‘sequences’

October 14, 2022

Title Generic and Biological Sequences

Version 0.5.9

Description Educational package used in R courses to illustrate object-oriented programming and package development. Using biological sequences (DNA and RNA) as a working example.

Depends methods, Rcpp

LinkingTo Rcpp

Suggests testthat, knitr

License GPL-3

biocViews Infrastructure, DataRepresentation, DataImport

VignetteBuilder knitr

URL `https://github.com/lgatto/sequences/`

Author Laurent Gatto [aut, cre],
Robert Stojnic [aut]

Maintainer Laurent Gatto <`lg390@cam.ac.uk`>

NeedsCompilation yes

Repository CRAN

Date/Publication 2014-12-03 13:58:03

R topics documented:

`sequences-package` *A short demo package featuring biological sequences.*

Description

Dummy package used in an R course to illustrate OO programming and package development. The course is aimed at bioinformaticians and biologists. Relevant illustrative examples used in the package are generic sequences as a top virtual class and specific biological (DNA and RNA) sequences.

Details

Package:	sequences
Type:	Package
License:	GPL-3
LazyLoad:	yes

Author(s)

Laurent Gatto <lg390@cam.ac.uk> Robert Stojnic <rs550@cam.ac.uk> Maintainer: Laurent Gatto <lg390@cam.ac.uk>

`dnaseq` *An* `DnaSeq` *example.*

Description

`dnaseq` is an example instance of an object of class `DnaSeq`. It has been generated by reading the `aDnaSeq.fasta` file (available in `syste.file("extdata",package="sequences")`) with the `readFasta` function.

Usage

```
data(dnaseq)
```

Format

The format is: chr "dnaseq"

Examples

```
data(dnaseq)
dnaseq
print(dnaseq)
```

`DnaSeq-class` *Class "DnaSeq" and "RnaSeq"*

Description

The `DnaSeq` and `RnaSeq` are instances of the virtual `GenericSeq` class for "DNA" and "RNA" respectively. Their alphabets are defined accordingly.

Objects from the Class

Instance of `DnaSeq` and `RnaSeq` can be created by reading `fasta` files using the `readFasta` function.

Slots

See the `GenericSeq` class for details about the generic slots and methods.

Extends

Class `GenericSeq`, directly.

Methods

See `GenericSeq` for inherited methods.

comp `signature(object = "DnaSeq")`: return the complement of the sequence string of the sequence instance.

transcribe `signature(object = "DnaSeq")`: transcribes the `DnaSeq` instance into a `RnaSeq` instance.

Author(s)

Laurent Gatto <lg390@cam.ac.uk>

See Also

See also the `GenericSeq` virutal class.

Examples

```
data("dnaseq")
dnaseq
rnaseq <- transcribe(dnaseq)
rnaseq
```

`gccount` *Count bases in sequence*

Description

Returns the number of 'A', 'C', 'G' and 'T' bases in the 'inseq' sequence string.

Usage

```
gccount(inseq)
```

Arguments

`inseq`	a DNA sequence string.

Details

This function calls a C primitive

Value

A numeric of length 4

Author(s)

Laurent Gatto `<lg390@cam.ac.uk>`

Examples

```
s <- "AAAACCCGGT"
cnt <- gccount(s)
cnt
stopifnot(cnt==table(strsplit(s,"")))
```

`gccount2` *Count bases in sequence (using Rcpp)*

Description

Returns the number of 'A', 'C', 'G' and 'T' bases in the 'inseq' sequence string.

Usage

```
gccount2(inseq)
```

Arguments

`inseq`	a DNA sequence string.

Details

This function calls a C primitive

Value

A numeric of length 4

Author(s)

Robert Stojnic <`rs550@cam.ac.uk`>

Examples

```
s <- "AAAACCCGGT"
cnt <- gccount2(s)
cnt
stopifnot(cnt==table(strsplit(s,"")))
```

`GenericSeq-class` *Class "GenericSeq"*

Description

This virtual class defines a generic sequence object composed by an identifier, a sequence and an alphabet that defines the letters of the sequence. A set of accessor methods and setters are described below.

Objects from the Class

A virtual Class: No objects may be created from it.

Slots

- `id`**:** Object of class `"character"` that identifies the sequence instance.
- `alphabet`**:** Object of class `"character"` that defines all the letters that a valid sequence can be made of.
- `sequence`**:** Object of class `"character"` storing the actual sequence string of the sequence instance.

Methods

- **alphabet** `signature(object = "GenericSeq")`: retrieves the alphabet of valid sequence letters, returned as `characher`.
- **id** `signature(object = "GenericSeq")`: retrieves the identifier of the sequence instance.
- **id<-** `signature(object = "GenericSeq", value = "character")`: sets the identifier of the sequence instance to `value`.
- **length** `signature(x = "GenericSeq")`: returns the length the sequence instance, i.e. the number of characters of the sequence string.
- **print** `signature(x = "GenericSeq")`: prints the sequence of the instance.
- **rev** `signature(x = "GenericSeq")`: return the reverse of the sequence string of the sequence instance.
- **seq** `signature(... = "GenericSeq")`: returns the sequence character string of the instance.
- **seq<-** `signature(object = "GenericSeq", value = "character")`: changes the sequence string of the sequence instance to `value`. The new string must be compatbile with the sequence instance alphabet.
- **show** `signature(object = "GenericSeq")`: displays the object content.
- **[** `signature(x = "GenSeq")`: Subsets the sequence of `x` and returns the correspong subsetted object of same class as `x`.

Author(s)

Laurent Gatto <lg390@cam.ac.uk>

See Also

See `DnaSeq` and `RnaSeq` classes for examples of subclasses and `readFasta` on how to create instances.

Examples

```
showClass("GenericSeq")
```

`readFasta` *Read fasta files.*

Description

Reads sequences data in fasta and create `DnaSeq` and `RnaSeq` instances.

Usage

```
readFasta(infile)
```

Arguments

`infile`	the name of the fasta file which the data are to be read from.

Details

This funtion reads DNA and RNA fasta files and generates valid `"DnaSeq"` and `"RnaSeq"` instances.

Value

an instance of `DnaSeq` or `RnaSeq`.

Author(s)

Laurent Gatto `<lg390@cam.ac.uk>`

See Also

`GenericSeq`, `DnaSeq` and `RnaSeq`.

Examples

```
f <- dir(system.file("extdata",package="sequences"),pattern="fasta",full.names=TRUE)
f
aa <- readFasta(f[1])
aa
```

`Seq-class` *Class* `"Seq"`

Description

The `Seq` class implements a nucleic acid sequence as well as various sequence transformations. It's aim is to illustrate the usage of reference classes.

Extends

All reference classes extend and inherit methods from `"envRefClass"`.

Fields

`id`**:** Object of class `character` defining the sequence identifier.

`alphabet`**:** Object of class `character` defining the sequence alphabet.

`sequence`**:** Object of class `character` with the actual sequence string.

Methods

- **`transcribe()`:** Transcribes a DNA sequence into RNA.
- **`seq()`:** Retuns the sequence string.
- **`comp()`:** Complements the sequence.
- **`rev()`:** Reverses the sequence.
- **`valid()`:** Checks sequence validity.
- **`setAlphabet()`:** Sets the sequence alphabet.
- **`show()`:** Shows the 'Seq' instance.

Author(s)

Laurent Gatto <lg390@cam.ac.uk>

See Also

`DnaSeq` for a S4 implementation of the same subject.

Examples

```
s <- Seq$new(id="My sequence",
             sequence="AGCAGCTACGACT",
             alphabet=c("A","C","G","T"))
s
s$rev()
s
```

Index