Package 'vhcub'

October 12, 2022

Title Virus-Host Codon Usage Co-Adaptation Analysis

Version 1.0.0

Author Ali Mostafa Anwar [aut, cre],
Mohamed Soudy [aut]

Maintainer Ali Mostafa Anwar <ali.mo.anwar@std.agr.cu.edu.eg>

Description Analyze the co-adaptation of codon usage between a virus and its host, calculate various codon usage bias measurements as: effective number of codons (ENc) Novembre (2002) <doi:10.1093/oxfordjournals.molbev.a004201>, codon adaptation index (CAI) Sharp and Li (1987) <doi:10.1093/nar/15.3.1281>, relative codon deoptimization index (RCDI) Puigbò et al (2010) <doi:10.1186/1756-0500-3-87>, similarity index (SiD) Zhou et al (2013) <doi:10.1371/journal.pone.0077239>, synonymous codon usage orderliness (SCUO) Wan et al (2004) <doi:10.1186/1471-2148-4-19> and, relative synonymous codon usage (RSCU) Sharp et al (1986) <doi:10.1093/nar/14.13.5125>. Also, it provides a statistical dinucleotide over- and underrepresentation with three different models. Implement several methods for visualization of codon usage as ENc.GC3plot() and PR2.plot().

License GPL-3

Encoding UTF-8

LazyData true

biocViews

Imports Biostrings, coRdon , ggplot2, seqinr, stringr

RoxygenNote 6.1.1

Suggests testthat

NeedsCompilation no

Repository CRAN

Date/Publication 2019-11-15 12:00:02 UTC

R topics documented:

CAI.values 2
dinuc.base 3
dinuc.codon 4

dinuc.syncodon 5
ENc.GC3plot 6
ENc.values 7
fasta.read 8
GC.content 9
PR2.plot 10
RCDI.values 11
RSCU.values 12
SCUO.values 13
SiD.value 14
vhcub 15

Index **17**

`CAI.values` *Codon Adaptation Index (CAI)*

Description

Measure the Codon Adaptation Index (CAI) Sharp and Li (1987), of DNA sequence.

Usage

```
CAI.values(df.virus, ENc.set.host,
                      df.host,genetic.code = "1",set.len = 5, threshold = 0)
```

Arguments

`df.virus`	a data frame with seq_name and its virus DNA sequence.
`ENc.set.host`	a data frame with ENc values of a host.
`df.host`	a data frame with seq_name and its host DNA sequence.
`genetic.code`	a single string that uniquely identifies a genetic code to use.
`set.len`	a number represents a percent that will be used as reference genes from the total host genes.
`threshold`	optional numeric, specifying sequence length, in codons, used for filtering.

Details

For more information about CAI Sharp and Li, 1987.

Value

A data.frame containing the computed CAI values for each DNA sequences within df.fasta.

Author(s)

Ali Mostafa Anwar <ali.mo.anwar@std.agr.cu.edu.eg> and Mohmed Soudy <MohmedSoudy2009@gmail.com>

Examples

```
# read DNA from fasta file
fasta <- fasta.read("virus.fasta", "host.fasta")
fasta.v <- fasta[[1]]
fasta.h <- fasta[[2]]
# Calculate CAI
enc.df.host <- ENc.values(fasta.h)


cai.df <- CAI.values(fasta.v, enc.df.host, fasta.h)
```

`dinuc.base` *Statistical dinucleotide over- and underrepresentation (base model).*

Description

A measure of statistical dinucleotide over- and underrepresentation; by allows for random sequence generation by shuffling (with/without replacement) of all bases in the sequence.

Usage

```
dinuc.base(df.virus,permutations=500,exact_numbers = FALSE)
```

Arguments

`df.virus`	data frame with seq_name and its DNA sequence.
`permutations`	the number of permutations for the z-score computation.
`exact_numbers`	if TRUE exact analytical calculation will be used.

Details

For more information seqinr.

Value

A data.frame containing the computed statistic for each dinucleotide in all DNA sequences within df.virus.

Author(s)

Ali Mostafa Anwar <ali.mo.anwar@std.agr.cu.edu.eg> and Mohmed Soudy <MohmedSoudy2009@gmail.com>

Examples

```
# read DNA from fasta file
fasta <- fasta.read("virus.fasta", "host.fasta")
fasta.v <- fasta[[1]]
fasta.h <- fasta[[2]]


# Calculate zscore using (base model)
base <- dinuc.base(fasta.v, permutations = 10)
```

`dinuc.codon` *Statistical dinucleotide over- and underrepresentation (codon model).*

Description

A measure of statistical dinucleotide over- and underrepresentation; by allows for random sequence generation by shuffling (with/without replacement) of codons.

Usage

```
dinuc.codon(df.virus,permutations=500,exact_numbers = FALSE)
```

Arguments

`df.virus`	data frame with seq_name and its DNA sequence.
`permutations`	the number of permutations for the z-score computation.
`exact_numbers`	if TRUE exact analytical calculation will be used.

Details

For more information seqinr.

Value

A data.frame containing the computed statistic for each dinucleotide in all DNA sequences within df.virus.

Author(s)

Ali Mostafa Anwar <ali.mo.anwar@std.agr.cu.edu.eg> and Mohmed Soudy <MohmedSoudy2009@gmail.com>

Examples

```
# read DNA from fasta file
fasta <- fasta.read("virus.fasta", "host.fasta")
fasta.v <- fasta[[1]]
fasta.h <- fasta[[2]]


# Calculate zscore using (codon model)
codon <- dinuc.codon(fasta.v, permutations = 10)
```

`dinuc.syncodon` *Statistical dinucleotide over- and underrepresentation (syncodon model).*

Description

A measure of statistical dinucleotide over- and underrepresentation; by allows for random sequence generation by shuffling (with/without replacement) of synonymous codons.

Usage

```
dinuc.syncodon(df.virus,permutations=500,exact_numbers = FALSE)
```

Arguments

`df.virus`	data frame with seq_name and its DNA sequence.
`permutations`	the number of permutations for the z-score computation.
`exact_numbers`	if TRUE exact analytical calculation will be used.

Details

For more information seqinr.

Value

A data.frame containing the computed statistic for each dinucleotide in all DNA sequences within df.virus.

Author(s)

Ali Mostafa Anwar <ali.mo.anwar@std.agr.cu.edu.eg> and Mohmed Soudy <MohmedSoudy2009@gmail.com>

Examples

```
# read DNA from fasta file
fasta <- fasta.read("virus.fasta", "host.fasta")
fasta.v <- fasta[[1]]
fasta.h <- fasta[[2]]


# Calculate zscore using (syncodon model)
syncodon <- dinuc.syncodon(fasta.v, permutations = 10)
```

`ENc.GC3plot` *ENc-GC3 scatterplot.*

Description

Make an ENc-GC3 scatterplot. Where the y-axis represents the ENc values and the x-axis represents the GC3 content. The red fitting line shows the expected ENc values when codon usage bias affected solely by GC3.

Usage

```
ENc.GC3plot(enc.df, gc.df)
```

Arguments

`enc.df`	a data frame with ENc values.
`gc.df`	a data frame with GC3 values.

Details

For more information about ENc-GC3 plot Butt et al., 2016.

Value

A ggplot object.

Author(s)

Ali Mostafa Anwar <ali.mo.anwar@std.agr.cu.edu.eg> and Mohmed Soudy <MohmedSoudy2009@gmail.com>

Examples

```
# read DNA from fasta file
fasta <- fasta.read("virus.fasta", "host.fasta")
fasta.v <- fasta[[1]]
enc.df.virus <- ENc.values(fasta.v)


gc.df <- GC.content(fasta.v)

ENc.GC3plot(enc.df.virus, gc.df)
```

ENc.values *Effective Number of Codons (ENc).*

Description

Measure the Effective Number of Codons (ENc) of DNA sequence. Using its modified version (Novembre, 2002).

Usage

```
ENc.values(df.fasta,genetic.code = "1",threshold=0)
```

Arguments

df.fasta	a data frame with seq_name and its DNA sequence.
genetic.code	a single string that uniquely identifies a genetic code to use.
threshold	optional numeric, specifying sequence length, in codons, used for filtering.

Details

For more information about ENc Novembre, 2002.

Value

A data.frame containing the computed ENc values for each DNA sequences within df.fasta.

Author(s)

Ali Mostafa Anwar <ali.mo.anwar@std.agr.cu.edu.eg> and Mohmed Soudy <MohmedSoudy2009@gmail.com>

Examples

```
# read DNA from fasta file
fasta <- fasta.read("virus.fasta", "host.fasta")
fasta.v <- fasta[[1]]
fasta.h <- fasta[[2]]


# Calculate ENc
enc.df.v <- ENc.values(fasta.v)

enc.df.h <- ENc.values(fasta.h)
```

`fasta.read` *Read fasta formate and convert it to data frame*

Description

Read fasta formate and convert it to data frame

Usage

```
fasta.read(virus.fasta,host.fasta)
```

Arguments

`virus.fasta`	directory path to the virus fasta file.
`host.fasta`	directory path to the host fasta file.

Value

A list with two data frames.

Note

The list with two data.frames; the first one for virus DNA sequences and the second one for the host.

Author(s)

Ali Mostafa Anwar <ali.mo.anwar@std.agr.cu.edu.eg> and Mohmed Soudy <MohmedSoudy2009@gmail.com>

Examples

```
fasta <- fasta.read("virus.fasta", "host.fasta")
fasta.v <- fasta[[1]]
fasta.h <- fasta[[2]]
```

GC.content *GC content*

Description

Calculates overall GC content as well as GC at first, second, and third codon positions.

Usage

```
GC.content(df.virus)
```

Arguments

df.virus	data frame with seq_name and its DNA sequence.

Value

A data.frame with overall GC content as well as GC at first, second, and third codon positions of all DNA sequence from df.virus.

Author(s)

Ali Mostafa Anwar <ali.mo.anwar@std.agr.cu.edu.eg> and Mohmed Soudy <MohmedSoudy2009@gmail.com>

Examples

```
# read DNA from fasta file
fasta <- fasta.read("virus.fasta", "host.fasta")
fasta.v <- fasta[[1]]
fasta.h <- fasta[[2]]


# Calculate GC content
gc.df <- GC.content(fasta.v)
```

PR2.plot *Parity rule 2 (PR2) plot*

Description

Make a Parity rule 2 (PR2) plot, where the AT-bias [A3/(A3 +T3)] at the third codon position of the four-codon amino acids of entire genes is the ordinate and the GC-bias [G3/(G3 +C3)] is the abscissa. The center of the plot, where both coordinates are 0.5, is where A = U and G = C (PR2), with no bias between the influence of the mutation and selection rates.

Usage

```
PR2.plot(fasta.df)
```

Arguments

fasta.df	a data frame with seq_name and its DNA sequence.

Details

For more information about PR2 plot Butt et al., 2016.

Value

A ggplot object.

Author(s)

Ali Mostafa Anwar <ali.mo.anwar@std.agr.cu.edu.eg> and Mohmed Soudy <MohmedSoudy2009@gmail.com>

Examples

```
# read DNA from fasta file
fasta <- fasta.read("virus.fasta", "host.fasta")
fasta.v <- fasta[[1]]
fasta.h <- fasta[[2]]


PR2.plot(fasta.v)
```

RCDI.values *Relative Codon Deoptimization Index (RCDI)*

Description

Measure the Relative Codon Deoptimization Index (RCDI) of DNA sequence.

Usage

```
RCDI.values(fasta.virus, fasta.host, enc.host, set.len= 5)
```

Arguments

fasta.virus	a data frame with virus seq_name and its DNA sequence.
fasta.host	a data frame with host seq_name and its DNA sequence.
enc.host	a data frame of a hosts' ENc values.
set.len	a number represents a percent that will be used as reference genes from the total host genes.

Details

For more information about RCDI Puigbò et al., 2010

Value

A data.frame containing the computed ENc values for each DNA sequences within df.fasta.

Author(s)

Ali Mostafa Anwar <ali.mo.anwar@std.agr.cu.edu.eg> and Mohmed Soudy <MohmedSoudy2009@gmail.com>

Examples

```
# read DNA from fasta file
fasta <- fasta.read("virus.fasta", "host.fasta")
fasta.v <- fasta[[1]]
fasta.h <- fasta[[2]]

# Calculate RCDI

enc.df.host <- ENc.values(fasta.h)


rcdi.df <- RCDI.values(fasta.v, fasta.h, enc.df.host)
```

RSCU.values *Relative Synonymous Codon Usage (RSCU)*

Description

Measure the Relative Synonymous Codon Usage (RSCU) of DNA sequence.

Usage

```
RSCU.values(df.fasta)
```

Arguments

df.fasta	a data frame with seq_name and its DNA sequence.

Details

For more information about ENc Sharp et al., 1986.

Value

A data.frame containing the computed RSCU values for each codon for each DNA sequences within df.fasta.

Author(s)

Ali Mostafa Anwar <ali.mo.anwar@std.agr.cu.edu.eg> and Mohmed Soudy <MohmedSoudy2009@gmail.com>

Examples

```
# read DNA from fasta file
fasta <- fasta.read("virus.fasta", "host.fasta")
fasta.v <- fasta[[1]]
fasta.h <- fasta[[2]]

# Calculate RSCU
RSCU.H <- RSCU.values(fasta.h)
RSCU.V <- RSCU.values(fasta.v)
```

SCUO.values *Synonymous codon usage eorderliness (SCUO)*

Description

Measure the Synonymous Codon Usage Eorderliness (SCUO) of DNA sequence (Wan et al., 2004).

Usage

```
SCUO.values(df.fasta,genetic.code = "1",threshold=0)
```

Arguments

df.fasta	a data frame with seq_name and its DNA sequence.
genetic.code	a single string that uniquely identifies a genetic code to use.
threshold	optional numeric, specifying sequence length, in codons, used for filtering.

Details

For more information about ENc Wan et al., 2004.

Value

A data.frame containing the computed SCUO values for each DNA sequences within df.fasta.

Author(s)

Ali Mostafa Anwar <ali.mo.anwar@std.agr.cu.edu.eg> and Mohmed Soudy <MohmedSoudy2009@gmail.com>

Examples

```
# read DNA from fasta file
fasta <- fasta.read("virus.fasta", "host.fasta")
fasta.v <- fasta[[1]]
fasta.h <- fasta[[2]]

# Calculate SCUO

SCUO.df <- SCUO.values(fasta.v)
```

SiD.value *Similarity Index (SiD)*

Description

Measure the Similarity Index (SiD) between a virus and its host codon usage.

Usage

```
SiD.value(rscu.host,rscu.virus)
```

Arguments

rscu.host	a data frame with RSCU a host codon values.
rscu.virus	a data frame with RSCU a virus codon values.

Details

For more information about SiD Zhou et al., 2013.

Value

A numeric represent a SiD value.

Author(s)

Ali Mostafa Anwar <ali.mo.anwar@std.agr.cu.edu.eg> and Mohmed Soudy <MohmedSoudy2009@gmail.com>

Examples

```
# read DNA from fasta file
fasta <- fasta.read("virus.fasta", "host.fasta")
fasta.v <- fasta[[1]]
fasta.h <- fasta[[2]]
RSCU.H <- RSCU.values(fasta.h)
RSCU.V <- RSCU.values(fasta.v)

# Calculate SiD
SiD <- SiD.value(RSCU.host, RSCU.virus)
```

vhcub *vhcub: A package to analysis the co-adaptation of codon usage between a virus and its host.*

Description

vhcub can calculate various codon usage bias measurements as; effective number of codons (ENc), codon adaptation index (CAI), relative codon deoptimization index (RCDI), similarity index (SiD), synonymous codon usage eorderliness (SCUO) and, relative synonymous codon usage (RSCU). Also, it provides a statistical dinucleotide over- and underrepresentation with three different models. Implement several methods for visualization of codon usage as ENc.GC3plot and PR2.plot.

vhcub functions

fasta.read: read fasta format files and convert it to data.frame.

GC.content: calculates overall GC content as well as GC at first, second, and third codon positions.

RSCU.values: measure the Relative Synonymous Codon Usage (RSCU) of DNA sequence.

SCUO.values: measure the Synonymous Codon Usage Eorderliness (SCUO) of DNA sequence.

RCDI.values: measure the Relative Codon Deoptimization Index (RCDI) of DNA sequence.

CAI.values: measure the Codon Adaptation Index (CAI) Sharp and Li (1987), of DNA sequence.

ENc.values: measure the Effective Number of Codons (ENc) of DNA sequence. Using its modified version.

dinuc.syncodon: measure of statistical dinucleotide over- and underrepresentation; by allows for random sequence generation by shuffling (with/without replacement) of synonymous codons.

dinuc.codon: measure of statistical dinucleotide over- and underrepresentation; by allows for random sequence generation by shuffling (with/without replacement) of codons.

dinuc.base: measure of statistical dinucleotide over- and underrepresentation; by allows for random sequence generation by shuffling (with/without replacement) of all bases in the sequence.

ENc.GC3plot: make an ENc-GC3 scatterplot. Where the y-axis represents the ENc values and the x-axis represents the GC3 content. The red fitting line shows the expected ENc values when codon usage bias affected solely by GC3.

PR2.plot: make a Parity rule 2 (PR2) plot, where the AT-bias [A3/(A3 +T3)] at the third codon position of the four-codon amino acids of entire genes is the ordinate and the GC-bias [G3/(G3 +C3)] is the abscissa. The center of the plot, where both coordinates are 0.5, is where A = U and G = C (PR2), with no bias between the influence of the mutation and selection rates.

Author(s)

Ali Mostafa Anwar <ali.mo.anwar@std.agr.cu.edu.eg> and Mohmed Soudy <MohmedSoudy2009@gmail.com>

Examples

```
# read DNA from fasta files
fasta <- fasta.read("virus.fasta", "host.fasta")
fasta.v <- fasta[[1]]
fasta.h <- fasta[[2]]
# calculate GC content
gc.df <- GC.content(fasta.v)
# measure of statistical dinucleotide over- and underrepresentation
syncodon <- dinuc.syncodon(fasta.v,permutations=10)
base <- dinuc.base(fasta.v,permutations=10)
codon <- dinuc.codon(fasta.v,permutations=10)
# calculate ENc
enc.df <- ENc.values(fasta.v)
enc.df.h <- ENc.values(fasta.h)
# calculate SCUO and CAI
SCUO.df <- SCUO.values(fasta.v)
cai.df <- CAI.values(fasta.v,enc.df.h, fasta.h)
# calculate RSCU
RSCU.H <- RSCU.values(fasta.h)
RSCU.V <- RSCU.values(fasta.v)
# calculate SiD
SiD <- SiD.value(RSCU.H,RSCU.V)
# calculate RCDI
rcdi.df <- RCDI.values(fasta.v,fasta.h, enc.df.h)
# plot ENc.GC3plot
ENc.GC3plot(enc.df,gc.df)
# plot PR2.plot
PR2.plot(fasta.v)
```

Index

CAI.values, 2

dinuc.base, 3
dinuc.codon, 4
dinuc.syncodon, 5

ENc.GC3plot, 6
ENc.values, 7

fasta.read, 8

GC.content, 9

PR2.plot, 10

RCDI.values, 11
RSCU.values, 12

SCUO.values, 13
SiD.value, 14

vhcub, 15
vhcub-package *(*vhcub*)*, 15