Package 'UITOTO'

May 14, 2024

Type Package

Version 1.0.0

Title User InTerface for Optimal molecular diagnoses in high-throughput TaxOnomy

Author Ambrosio Torres [aut, cre] <atorresgalvis@gmail.com>
Leshon Lee [ctb],
Amrita Srivathsan [ctb],
Rudolf Meier [ctb]

BugReports/Maintainer Ambrosio Torres <atorresgalvis@gmail.com>

Description UITOTO is an R package designed for deriving, testing, and visualizing Diagnostic Molecular Combinations (DMCs). The package features a shiny app (from which its name is derived) that can be executed either locally or online (please visit https://atorresgalvis.shinyapps.io/MolecularDiagnoses/). Additionally, users can utilize the functions included in UITOTO via various command-line interfaces available for the R Statistical Software (e.g., RStudio).

Depends R (>= 4.2)

Imports BiocManager (>= 1.30.22),
Biostrings (>= 2.70.2),
DECIPHER (>= 2.30.0),
dplyr (>= 1.1.4),
ggplot2 (>= 3.4.0),
readr (>= 2.1.5),
seqinr (>= 4.2-36),
shiny (>= 1.8.0),
shinyjs (>= 2.1.0),
shinyWidgets (>= 0.8.2)

License GPL (>=3)

URL https://github.com/atorresgalvis/UITOTO

Encoding UTF-8

R topics documented:

ALnID *Align and Identify unknown sequences using DMCs*

Description

Align a pool of unknown sequences against the alignment used for obtaining the provided DMCs. Susequently, the resulting aligned unknown sequences are identified based on their matching patterns with respect to the DMCs. This procedure could also be used to evaluate the performance of the available DMCs.

Usage

```
ALnID(
 DMC_Output,
 dataTarget,
 dataQuery,
 mismatches = 1,
 against = c("First", "All"),
 perfectMatch = 5,
 misMatch = 0,
 gapOpening = -14,
 gapExtension = -2,
 gapPower = -1,
 terminalGap = 0,
 OutName = "IdentificationOutput.csv",
 MissLogFile = "LogMissing.csv",
 AligName = "none"
)
```

Arguments

DMC_Output	mandatory: the name of the CSV file that contains the available DMCs.
dataTarget	mandatory: the name of the fasta file with the alingment used for obtaining the DMCs.
dataQuery	mandatory: the name of the fasta file with the sequences/specimens to be identified.
mismatches	optional integer: the maximum number of mismatches allowed for the identification step. The default is 1.
against	optional: align each unknown sequence against the First sequence or against All sequences available in the dataTarget file. The default is First (much faster than option All).
perfectMatch	optional numeric: giving the reward for aligning two matching nucleotides in the alignment. The default is 5.
misMatch	optional numeric: giving the cost for aligning two mismatched nucleotides in the alignment. The default is 0.
gapOpening	optional numeric: giving the cost for opening a gap in the alignment. The default is -14.

gapExtension	optional numeric: giving the cost for extending an open gap in the alignment. The default is -2.
gapPower	optional numeric: specifying the exponent to use in the gap cost function (see the function AlignProfiles of the DECIPHER package). The default is -1.
terminalGap	optional numeric: giving the cost for allowing leading and trailing gaps ("-" or "." characters) in the alignment. The default is 0.
OutName	optional: the name of the CSV output file containing the identification of the unknown sequences. The default is "IdentificationOutput.csv".
MissLogFile	optional: the name of the CSV output file containing the comparisons that were not possible to do due missing data. The default is "LogMissing.csv". If you use "none", the step of checking for missing data will not be performed (this saves a lot of time but comparisons are less precise. Thus, this option is not recommended for sequences that could result in gappy alignments).
AligName	optional: the name of the output fasta file containing the resulting alignment. The default is "ResultingAlignment.fasta".

Details

The alignment function AlignProfiles of the DECIPHER package is used for aligning the unknown sequences against the alignment used for obtaining the provided DMCs. The users can modify the different settings of the alignment step. In addition, the users can define the number of mismatches allowed for the identification step.

Author(s)

Ambrosio Torres (Researcher [Ctr. Integr. Biodivers. Discov. - Museum für Naturkunde, Berlin, Germany)

References

Charif, D., Lobry, J.R. (2007). SeqinR 1.0-2: A Contributed Package to the R Project for Statistical Computing Devoted to Biological Sequences Retrieval and Analysis. In: Bastolla, U., Porto, M., Roman, H.E., Vendruscolo, M. (eds) Structural Approaches to Sequence Evolution. Biological and Medical Physics, Biomedical Engineering. Springer, Berlin, Heidelberg.

Pagès, H., Aboyoun, P., Gentleman, R., DebRoy, S. (2024). Biostrings: Efficient manipulation of biological strings. R package https://bioconductor.org/packages/Biostrings.

Wright, E.S. (2016). Using DECIPHER v2.0 to Analyze Big Biological Sequence Data in R. The R Journal, 8(1), 352-359.

Examples

```
## You can find the files "XXX" and "XXX" in the data folder of the packages.
## If you want to run the example. Make sure you have these files in your current working directory.
ALnID(
 "OpDMC_output.csv",
 "DatasetAlignedTarget.fas",
 "DatasetUnalignedQuery.fas",
 mismatches = 1,
 against= "First",
 perfectMatch=5,
 misMatch=0,
 gapOpening = -14,
```

```
 gapExtension = -2,
 gapPower = -1,
 terminalGap = 0,
 OutName = "IdentificationOutput.csv",
 MissLogFile= "LogMissing.csv",
 AligName = "none"
)
```

`Identifier` *Identify unknown aligned sequences using DMCs*

Description

A set of unknown aligned sequences are identified based on their matching patterns with respect to the DMCs. This procedure could also be used to evaluate the performance of the available DMCs. Additionally, the users can define the number of mismatches allowed for the identification.

Usage

```
Identifier(
 DMC_Output,
 dataset,
 mismatches = 1,
 OutName = "IdentificationOutput.csv",
 MissLogFile = "LogMissing.csv"
)
```

Arguments

`DMC_Output`	mandatory: the name of the CSV file that contains the available DMCs.
`dataset`	mandatory: the name of the fasta file with the sequences/specimens to be identified.
`mismatches`	optional integer: the maximum number of mismatches allowed for the identification step. The default is `1`.
`OutName`	optional: the name of the CSV output file containing the identification of the unknown sequences. The default is `"IdentificationOutput.csv"`.
`MissLogFile`	optional: the name of the CSV output file containing the comparisons that could not be done due missing data. The default is `"LogMissing.csv"`. If you use `"none"`, the step of checking for missing data will not be performed (this saves a lot of time but comparisons are less precise. Thus, this option is not recommended for gappy alignments).

Author(s)

Ambrosio Torres (Researcher [Ctr. Integr. Biodivers. Discov. - Museum für Naturkunde, Berlin, Germany)

References

Charif, D., Lobry, J.R. (2007). SeqinR 1.0-2: A Contributed Package to the R Project for Statistical Computing Devoted to Biological Sequences Retrieval and Analysis. In: Bastolla, U., Porto, M., Roman, H.E., Vendruscolo, M. (eds) Structural Approaches to Sequence Evolution. Biological and Medical Physics, Biomedical Engineering. Springer, Berlin, Heidelberg.

Examples

```
## You can find the files "XXX" and "XXX" in the data folder of the packages.
## If you want to run the example. Make sure you have these files in your current working directory.
Identifier(
 "OpDMC_output.csv",
 "DatasetAlignedQuery.fas",
 mismatches = 1,
 OutName = "IdentificationOutput.csv",
 MissLogFile= "LogMissing.csv"
)
```

`IdentifierU` *Alignment-free identification of unknown sequences using DMCs*

Description

This approach follows an alignment-free methodology for identification of unknown sequences using DMCs.

Usage

```
IdentifierU(
 DMC_Output,
 dataset,
 mismatches = 1,
 WinWidth = 20,
 OutName = "IdentificationOutput.csv"
)
```

Arguments

`DMC_Output`	mandatory: the name of the CSV file that contains the available DMCs.
`dataset`	mandatory: the name of the fasta file with the (aligned or unaligned) sequences to be identified.
`mismatches`	optional integer: the maximum number of mismatches allowed for the identification step. The default is `1`.
`WinWidth`	optional integer: width of the sliding window expresed in number of sites. The default is `20`.
`OutName`	optional: the name of the CSV output file containing the identification of the unknown sequences. The default is `"IdentificationOutput.csv"`.

Details

Through iterative exploration utilizing a dynamic sliding window, it compares the extracted patterns with each provided DMC pattern. Users can also set a number of mismatches allowed. This procedure is the less precise among the methods included in UITOTO for Taxonomic verification and identification using DMCs. Thus, it is only recommended for special situations in which the alignment used for obtaining the provided DMCs is not available.

Author(s)

Ambrosio Torres (Researcher [Ctr. Integr. Biodivers. Discov. - Museum für Naturkunde, Berlin, Germany)

References

Charif, D., Lobry, J.R. (2007). SeqinR 1.0-2: A Contributed Package to the R Project for Statistical Computing Devoted to Biological Sequences Retrieval and Analysis. In: Bastolla, U., Porto, M., Roman, H.E., Vendruscolo, M. (eds) Structural Approaches to Sequence Evolution. Biological and Medical Physics, Biomedical Engineering. Springer, Berlin, Heidelberg.

Examples

```
## You can find the files "XXX" and "XXX" in the data folder of the packages.
## If you want to run the example. Make sure you have these files in your current working directory.
IdentifierU(
 "OpDMC_output.csv",
 "DatasetQuerySequences.fas",
 mismatches = 0,
 WinWidth = 20,
 OutName = "IdentificationOutput.csv"
)
```

`OpDMC` *Find Diagnostic Molecular Combinations (DMCs)*

Description

Find reliable Diagnostic Molecular Combinations (DMCs) to be used in high-throughput taxonomy.

Usage

```
OpDMC(
 FastaFile,
 species,
 iter = 20000,
 MnLen = 4,
 exclusive = 4,
 RefStrength = 0.25,
 OutName = "OpDMC_output.csv",
 GapsNew = FALSE
)
```

Arguments

`FastaFile`	mandatory: the name of the fasta file containing aligned sequences for obtaining the DMCs.
`species`	mandatory: the name of the CSV file with the query-taxa list (i.e., taxa for which you want to find the DMCs).
`iter`	optional integer: the number of iterations (i.e., the number of molecular combinations to be tested). The default is `20000`.

MnLen	optional integer: the minimum length that DMCs must have. The default is 4.
exclusive	optional integer: the minimum number of exclusive character states that DMCs must have. The default is 4.
RefStrength	optional double (decimal value): the refinement strength (i.e., the proportion of sub-combinations from each DMC to test. The higher the refinement strength, the more likely the program is to identify potential shorter DMCs. However, take into account that the refinement strength also increases the time comsumption). The default is 0.25.
OutName	optional: the name of the CSV output file containing the DMCs. The default is "OpDMC_output.csv".
GapsNew	optional logical (TRUE or TRUE): If TRUE, the gaps will be treated as a new state. The default is FALSE.

Details

This method assigns a weight to each site based on the Jaccard index (i.e., the number of sequences in which the site states differ from the site state in the query taxon). It then uses a Weighted Random Sampling approach to build the candidate combinations to become DMCs. At the same time, the method uses a stability measure of the specificity of the candidate combinations to assess their reliability. This measure relies on the minimum number of exclusive character states for each one of the candidate combinations. As a final step, with the most frequent sites in the combinations that meet the preceding criterion, the final DMC is built (which must also meet the same criterion).

Author(s)

Ambrosio Torres (Researcher [Ctr. Integr. Biodivers. Discov. - Museum für Naturkunde, Berlin, Germany)

References

Charif, D., Lobry, J.R. (2007). SeqinR 1.0-2: A Contributed Package to the R Project for Statistical Computing Devoted to Biological Sequences Retrieval and Analysis. In: Bastolla, U., Porto, M., Roman, H.E., Vendruscolo, M. (eds) Structural Approaches to Sequence Evolution. Biological and Medical Physics, Biomedical Engineering. Springer, Berlin, Heidelberg.

Examples

```
## You can find the files "XXX" and "XXX" in the data folder of the packages.
## If you want to run the example. Make sure you have these files in your current working directory.
OpDMC(
 "MegaseliaTraining.fasta",
 "SpeciesListMegaselia.csv",
 iter = 5000,
 MnLen = 3,
 exclusive = 2,
 RefStrength = 0.10,
 OutName = "OpDMC_Megaselia.csv",
 GapsNew = FALSE
)
```

`runUITOTO` *Run the UITOTO shiny app locally*

Description

Execute the UITOTO shiny app locally. You can also run UITOTO online by visiting https://atorresgalvis.shinyapps.io/MolecularDiagnoses/.

Usage

```
runUITOTO()
```

Details

IMPORTANT: By default, users of Shiny apps can only upload files up to 5 MB. You can increase this limit by setting the `shiny.maxRequestSize` option before executing UITOTO.

```
#For example, to allow up to 12 MB use:
options(shiny.maxRequestSize = 12 * 1024^2)
#And then run UITOTO normally:
runUITOTO()
```

Index