Package 'SCORER2'

January 16, 2014

Type Package

Version 0.99.0

Date 2014-01-08

Title SCORER 2.0: an algorithm for distinguishing parallel dimeric and trimeric coiled-coil sequences

Author Craig T. Armstrong, Thomas L. Vincent <tlfvincent@gmail.com>, Peter J. Green and Derek N. Woolfson <D.N.Woolfson@bristol.ac.uk>

Maintainer Thomas L. Vincent <tlfvincent@gmail.com>

Depends R (>= 2.12)

LazyData true

ZipData no

License GPL (>= 2)

Description This package contains the functions necessary to run the SCORER 2.0 algorithm. SCORER 2.0 can be used to differentiate between parallel dimeric and trimeric coiled-coil sequence, which are the two most more frequent coiled-coil structures observed naturally. As such, SCORER 2.0 is particularly useful for researchers looking to characterize novel coiled-coil sequences. It may also be used to assist in the structural characterization of synthetic coiled-coil sequences. Also included in this package are functions that allows the user to retrain the SCORER 2.0 algorithm using user-defined training data.

Repository CRAN

URL http://coiledcoils.chm.bris.ac.uk/Scorer/

R topics documented:

scorer2-package *Predict oligomerization state of coiled-coil sequences*

Description

Package for predicting the oligomeric state of a coiled-coil sequence

Details

Package:	scorer2
Type:	Package
Version:	1.0
Date:	2014-01-08
License:	GPL >= 2.0

Functions in the SCORER2 package allow the user to apply the SCORER 2.0 prediction algorithm to coiled-coil sequences. Also included are functions to retrain the SCORER 2.0 algorithm using user-defined training data

Author(s)

Thomas L. Vincent <tlfvincent@gmail.com>

References

Craig T. Armstrong, Thomas L. Vincent, Peter J. Green and Dek N. Woolfson. (2011) SCORER 2.0: an algortihm for distinguishing parallel dimeric and trimeric coiled-coil sequences. Bioinformatics. DOI: 10.1093/bioinformatics/btr299

Examples

```
# load pssm data
data(pssm)

# predict oligomerization of GCN4 wildtype
GCN4wt.score <- scorer2("GCN4wt", "MKQLEDKVEELLSKNYHLENEVARLKKLV",
"abcdefgabcdefgabcdefgabcdefga", pssm,  delta=1)
```

CreatePssm *Compute profile scoring matrices for training data*

Description

Function used to compute new profile scoring matrices in the event that the user wishes to retrain SCORER 2.0 with his own training data

Usage

```
CreatePssm(training.data, var)
```

Arguments

`training.data` A dataframe or matrix with three columns containing the information of n coiled-coil sequences. The three columns must be named "sequence", "register" and "type". The order of the columns in the dataframe does not matter

1. column "type": contains the known oligomeric state of the coiled-coil sequences in the training data. Acceptable oligomeric states are "DIMER" and "TRIMER" only.
2. column "sequence": contains the amino-acid sequences of the coiled coils in the training data. Valid characters are all uppercase letters except 'B', 'J', 'O', 'U', 'X', and 'Z'; invalid characters will not be tolerated and their use will result in a failure of the program.
3. Contains the register assignments specific to each coiled-coil sequence in the training data. As such, it must always have the same length as the matching amino-acid sequence in the "sequence" column. Valid characters are the lowercase letters 'a' to 'g' only. Register assignments are not required to be in proper order and may start with any of the seven letters.

`var` A list of two elements containing all valid amino-acid and register characters.

Value

returns a profile scoring matrix derived from inputted training data

Author(s)

Thomas L. Vincent <tlfvincent@gmail.com>

References

Craig T. Armstrong, Thomas L. Vincent, Peter J. Green and Dek N. Woolfson. (2011) SCORER 2.0: an algortihm for distinguishing parallel dimeric and trimeric coiled-coil sequences. Bioinformatics. DOI: 10.1093/bioinformatics/btr299

Examples

```
# load training data
data(training)

# define allowed amino and register characters
var <- list(
    amino = c("A","C","D","E","F","G","H","I","K","L",
    "M","N","P","Q","R","S","T","V","W","Y","X"),
    register = letters[1:7])

# create profile scoring matrix
pssm <- CreatePssm(training, var)
```

`EstimateProbability` *Estimate oligomeric state score of coiled-coil sequences*

Description

Sub-function used in scorer2.R in order to compute the oligomeric state score of input coiled-coil sequences.

Usage

```
EstimateProbability(id, seq, reg, pssm, var, delta=1)
```

Arguments

`id`	A string that represents the id name of the test sequence
`seq`	A character string of the amino-acid sequence to be predicted. Valid characters are all uppercase letters except 'B', 'J', 'O', 'U', 'X', and 'Z';
`reg`	A character string of register assignements. Valid characters are the lowercase letters 'a' to 'g'. Register characters are not required to be in proper order and may start with any of the seven letters. It must always have the same length as the matching amino-acid sequence.
`pssm`	A profile scoring matrix generated from the SCORER 2.0 training data. You can either use the default one or create your own PSSM using the pssm.R function
`var`	A list of two elements containing all valid amino-acid and register characters.
`delta`	The pseudocount parameter introduced in the PSSM used for the estimation of oligomeric state scores. This helps avoid cases with zero count. Empirical analysis has shown that a default delta score of 1 is optimal.

Value

It is used to apply the SCORER 2.0 prediction algorithm to a new coiled-coil sequence. By default the final classification is computed on the basis of the discriminant function value. If $f(x) >= 0$, x is predicted as a dimer, otherwise as a trimer.

Author(s)

Thomas L. Vincent <tlfvincent@gmail.com>

References

Craig T. Armstrong, Thomas L. Vincent, Peter J. Green and Dek N. Woolfson. (2011) SCORER 2.0: an algortihm for distinguishing parallel dimeric and trimeric coiled-coil sequences. Bioinformatics. DOI: 10.1093/bioinformatics/btr299

Examples

```
# load pssm data
data(pssm)

# define allowed amino and register characters
var <- list(
      amino = c("A","C","D","E","F","G","H","I","K","L",
      "M","N","P","Q","R","S","T","V","W","Y","X"),
      register = letters[1:7])

# run SCORER 2.0 on GCN4 wild-type
GCN4wt.score <- EstimateProbability("GCN4wt",
"MKQLEDKVEELLSKNYHLENEVARLKKLV",
"abcdefgabcdefgabcdefgabcdefga",
pssm,
var,
delta=1)
```

`pssm` *Profile scoring matrix derived from the original SCORER 2.0 training set.*

Description

This data set includes the Profile scoring matrix derived from the training set compiled from the CC+ coiled-coil database. It is used as the default PSSM used by SCORER 2.0 when predicting the oligomeric state of coiled-coil sequences. More details on the training set can be found in the reference below.

Usage

```
data(pssm)
```

Format

A multi-dimensional array with 7 element, each of dimension 2x21.

Source

DOI: 10.1093/bioinformatics/btr299.

References

Craig T. Armstrong, Thomas L. Vincent, Peter J. Green and Dek N. Woolfson. (2011) SCORER 2.0: an algortihm for distinguishing parallel dimeric and trimeric coiled-coil sequences. Bioinformatics. DOI: 10.1093/bioinformatics/btr299

Examples

```
data(pssm)
print(pssm)
```

RetrainScorer2 *Retrain the SCORER 2.0 algorithm with user-defined training data*

Description

Function used to train the SCORER 2.0 algorithm with user-defined training data. It is recommended that the training data contains at least 30 amino-sequence/register assignment pairs for each oligomeric state in the training set.

Usage

```
RetrainScorer2(seq, reg, type)
```

Arguments

seq	A vector of amino acid sequences; where each element of the vector is a character string of amino-acid sequence. Valid characters are all uppercase letters except 'B', 'J', 'O', 'U', 'X', and 'Z'; invalid characters will not be tolerated and their use will result in a failure of the program.
reg	A vector of heptad register assigned to each amino acid sequence; valid characters are the lowercase letters 'a' to 'g'. Register characters are not required to be in proper order. The register can start with any of the seven letters. It must always have the same length as the matching amino-acid sequence in "seq".
type	A vector containing the known oligomeric state of the coiled-coil sequences in the training data. Acceptable oligomeric states are "DIMER" and "TRIMER" only.

Value

Returns a profile scoring matrix derived from the user-defined training data

Author(s)

Thomas L. Vincent <tlfvincent@gmail.com>

References

Craig T. Armstrong, Thomas L. Vincent, Peter J. Green and Dek N. Woolfson. (2011) SCORER 2.0: an algortihm for distinguishing parallel dimeric and trimeric coiled-coil sequences. Bioinformatics. DOI: 10.1093/bioinformatics/btr299

Examples

```
# load training data
data(training)
seq <- training[, 1]
reg <- training[, 2]
type <- training[, 3]

# retrain SCORER 2.0 to obtain new pssm
pssm <- RetrainScorer2(seq, reg, type)
```

scorer2 *Predict oligomerization state of coiled-coil sequences*

Description

Function for predicting the oligomeric state of a coiled-coil sequence

Usage

```
scorer2(id, seq, reg, pssm, delta=1)
```

Arguments

id	A vector of id names for each sequence to be predicted
seq	A vector of amino acid sequences; where each element of the vector is a character string of amino-acid sequence. Valid characters are all uppercase letters except 'B', 'J', 'O', 'U', 'X', and 'Z'; invalid characters will not be tolerated and their use will result in a failure of the program.
reg	A character string denoting the heptad register of each amino acid sequence; valid characters are the lowercase letters 'a' to 'g'. Register characters are not required to be in proper order. The register can start with any of the seven letters. It must always have the same length as the seq argument.
pssm	A profile scoring matrix generated from the SCORER 2.0 training data. You can either use the default one or create your own PSSM using the pssm.R function
delta	A numeric value strictly superior to 0 that serves as pseudocount in the profile scoring matrix. Empirical analysis has shown that a value of delta=1 is optimal.

Value

The function scorer2 is the most important one in the **scorer2** package. It is used to apply the SCORER 2.0 prediction algorithm to predict the oligomeric state of a new coiled-coil sequence. By default the final classification is computed on the basis of the discriminant function value. If $f(x) >= 0$, x is predicted as a dimer, otherwise as trimer.

Author(s)

Thomas L. Vincent <tlfvincent@gmail.com>

References

Craig T. Armstrong, Thomas L. Vincent, Peter J. Green and Dek N. Woolfson. (2011) SCORER 2.0: an algortihm for distinguishing parallel dimeric and trimeric coiled-coil sequences. Bioinformatics. DOI: 10.1093/bioinformatics/btr299

Examples

```
# load pssm data
data(pssm)

# predict oligomerization of GCN4 wildtype
GCN4wt.score <- scorer2("GCN4wt", "MKQLEDKVEELLSKNYHLENEVARLKKLV",
"abcdefgabcdefgabcdefgabcdefga", pssm,  delta=1)
```

`training` *Training dataset used to contruct the profile scoring matrix in the SCORER 2.0 algorithm.*

Description

A dataframe containing three columns that must be named "type", "sequence" and "register". The order of the columns in the dataframe does not matter

1. column "type": contains the known oligomeric state of the coiled-coil sequences in the training data. Acceptable oligomeric states are "DIMER" and "TRIMER" only.
2. column "sequence": contains the amino-acid sequences of the coiled coils in the training data. Valid characters are all uppercase letters except 'B', 'J', 'O', 'U', 'X', and 'Z'; invalid characters will not be tolerated and their use will result in a failure of the program.
3. Contains the register assignments specific to each coiled-coil sequence in the training data. As such, it must always have the same length as the matching amino-acid sequence in the "sequence" column. Valid characters are the lowercase letters 'a' to 'g' only. Register assignments are not required to be in proper order and may start with any of the seven letters.

Usage

```
data(training)
```

Format

A multi-dimensional array with 7 element, each of dimension 2x21.

Source

DOI: 10.1093/bioinformatics/btr299.

References

Craig T. Armstrong, Thomas L. Vincent, Peter J. Green and Dek N. Woolfson. (2011) SCORER 2.0: an algortihm for distinguishing parallel dimeric and trimeric coiled-coil sequences. Bioinformatics. DOI: 10.1093/bioinformatics/btr299

Examples

```
data(training)
print(training)
```

Index