Package ‘PSSMCOOL’

October 25, 2020

Type Package
Title Features Extracted from Position Specific Scoring Matrix (PSSM)
Version 0.2.1
Imports utils, gtools, infotheo, phonTools, dtt
Maintainer Alireza mohammadi <alireza691111@gmail.com>
Depends R (>= 3.1.0)
License GPL-3
URL https://github.com/Alireza9651501005/PSSMCOOL
BugReports https://github.com/Alireza9651501005/PSSMCOOL/issues
Encoding UTF-8
LazyData true
RoxygenNote 7.1.1
Suggests testthat, spelling, waveslim,
Language en-US
NeedsCompilation no
Author  Javad zahiri [aut],  
   Alireza mohammadi [aut, cre]
Repository  CRAN
Date/Publication  2020-10-25 17:00:08 UTC

R topics documented:

PSSMCOOL-package .................................................. 3
AATP_TPCC .............................................................. 4
AB_PSSM ................................................................. 5
Average_Block ........................................................... 6
consensus_sequence ..................................................... 6
CS_PSe_PSSM ............................................................. 7
DFMCA_PSSM ............................................................. 8
Discrete_Cosine_Transform ........................................... 9
disulfid ................................................................. 9
DPC_PSSM ............................................................... 10
DP_PSSM ............................................................... 11
dwt_PSSM ............................................................... 12
EDP_MEDP ............................................................... 13
FPSSM ................................................................. 14
FPSSM2 ............................................................... 14
grey_pssm_pseAAC ................................................... 15
kiderafactor ............................................................ 16
k_mers ................................................................. 17
k_seperated_bigrame ................................................ 17
LPC_PSSM ............................................................... 18
MBMGACPSSM .......................................................... 19
pse_pssm .............................................................. 19
pssm400 .............................................................. 20
PSSMAC ............................................................... 21
PSSMBLOCK ............................................................ 22
pssm_ac ............................................................... 22
pssm_cc ............................................................... 23
PSSM_SD ............................................................... 24
pssm_seg .............................................................. 25
rpssm ................................................................. 25
scsh2 ................................................................. 26
single_Average ......................................................... 27
smoothed_PSSM ....................................................... 27
SOMA_PSSM ........................................................... 28
SVD_PSSM ............................................................. 29
three_mer ............................................................. 30
trigrame_pssm ........................................................ 30

Index  32
Description

This package is a Comprehensive toolkit for generating various numerical features from PSSM Matrix correspond to each protein sequence and covers all features introduced in POSSUM Package and some other new features. see PSSMCOOL_Rpubs for more information.

Author(s)

Maintainer: Alireza Mohammadi <alireza691111@gmail.com>
Authors:
  • Alireza Mohammadi <alireza691111@gmail.com>
  • Javad Zahiri <zahiri@modares.ac.ir>

See Also

All functions in this package:
  • AATP_TPCC
  • AB_PSSM
  • Averag_Block
  • consensus_sequence
  • CS_PSe_PSSM
  • DFMCA_PSSM
  • Discrete_Cosine_Transform
  • disulfid
  • DP_PSSM
  • DPC_PSSM
  • dwt_PSSM
  • EDP_MEDP
  • FPSSM
  • FPSSM2
  • grey_pssm_pseAAC
  • k_mers
  • k_seperated_bigrame
  • kiderafactor
  • LPC_PSSM
  • MBMGACPSSM
AATP_TPCC

AATP TPCC feature vector

**Description**

For getting this feature which was used to protein structural class prediction, at first mean of every column in PSSM Matrix is computed to achieve a 20-dimensional vector called AAC.then by fusing it with other vector of length 400 called TPC, which is similar to DPC_PSSM AATP feature vector of length 420 is obtained.

**Usage**

AATP_TPCC(pssm_name)

**Arguments**

- **pssm_name** is name of PSSM Matrix file

**Value**

- a feature vector of length 420

Useful links:

- [https://rpubs.com/alireza_69/640294](https://rpubs.com/alireza_69/640294)
- [https://github.com/Alireza9651501005/PSSMCOOL/](https://github.com/Alireza9651501005/PSSMCOOL/)
- [http://bioinf.modares.ac.ir/software/pssm/DownloadALL.html](http://bioinf.modares.ac.ir/software/pssm/DownloadALL.html)
- Report bugs at [https://github.com/Alireza9651501005/PSSMCOOL/issues](https://github.com/Alireza9651501005/PSSMCOOL/issues)
References


See Also

DPC_PSSM

Examples

as<-AATP_TPCC(paste0(system.file("extdata",package="PSSMCOOL"),"/C7GQS7.txt.pssm"))

---

**Description**

This feature consists of two types of feature vectors. At first, each protein sequence is divided into 20 equal parts, each of which is called a block, and in each block the row vectors of the PSSM matrix related to that block are added together and the resulting final vector is divided by the length of that block, which is 5. Finally, by placing these 20 vectors side by side, the first feature vector of length 400 is obtained. The second feature for each amino acid in each column is the average of the positive numbers in that column and for each block, and these 20 numbers, corresponding to 20 blocks, are placed next to each other, and therefore for each of the 20 types of amino acids, a vector of length 20 is obtained, and by placing these together, the vector of the second feature, length 400, is obtained.

Usage

```
AB_PSSM(pssm_name)
```

Arguments

- `pssm_name` name of PSSM Matrix file

Value

two feature vectors AB-PSSM and RPM-PSSM each of length 400

References


Examples

```
zz<- AB_PSSM(system.file("extdata","C7GRQ3.txt.pssm",package="PSSMCOOL"))
```
**Averag Block**

**Averag Block feature vector**

**Description**

In this feature at first PSSM Matrix is divided to 20 Blocks. Then for each Block mean of columns is computed to get 20-dimensional vector, eventually by appending these vectors to each other final feature vector is obtained which would be of length 400. This feature vector is similar to `PSSMBLOCK` for N=20.

**Usage**

```r
Averag_Block(pssm_name)
```

**Arguments**

- `pssm_name`: name of PSSM Matrix file

**Value**

feature vector of length 400

**References**


**Examples**

```r
v<-Averag_Block(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"))
```

---

**consenso_sequence**

**Description**

This feature vector is constructed from PSSM Matrix as: \( \alpha(i) = \text{argmax}(P_{i,j}) \) where \( i \) varies between 1 and \( L \) and \( j \) between 1 and 20, \( L \) indicates protein length and "arg" represents the argument of the maximum the \( i \)th base of the consensus sequence (CS) is then set to be the \( \alpha(i) \)th amino acid in the amino acid alphabet and a consensus sequence is constructed.

**Usage**

```r
consensus_sequence(pssm_name)
```
CS_PSe_PSSM

Arguments

  pssm_name is the name of PSSM Matrix file

Value

  consensus sequence which extracted from PSSM

References

  Y. Liang, S. Liu, S. J. C. Zhang, and m. m. i. medicine, "Prediction of protein structural classes for low-similarity sequences based on consensus sequence and segmented PSSM," vol. 2015, 2015.

Examples

  w<-consensus_sequence(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCool"))

Description

  This feature vector is constructed by fusing consensus sequence (CS), segmented PsePSSM, and segmented auto-covariance transformation (ACT) based on PSSM. by consensus sequence a 40-dimensional feature vector is obtained, in segmented PsePSSM group, by dividing PSSM Matrix to 2 and 3 segments a 380-dimensional feature vector is obtained and in ACT group, similar to the previous group at first PSSM Matrix is divided to 2 and 3 segments then a feature vector of length 280 is obtained.eventually by fusing these features a 700-dimensional feature vector is obtained.

Usage

  CS_PSe_PSSM(pssm_name, vec_name)

Arguments

  pssm_name name of PSSM Matrix file
  vec_name a character that user imports to specify kind of feature vector which it can be varied between four values

Details

  If vec_name equals to "segmented_psepssm" then a feature vector of length 380 is obtained. if vec_name equals to "segmented_acpssm" then a feature vector of length 280 is obtained, and if vec_name equals to "cspssm" the obtained feature vector would be of length 40 eventually if vec_name equals to "total" then feature vector would be of length 700.

Value

  feature vector that its length depends on the vec_name which user imports
DFMCA_PSSM

References

Y. Liang, S. Liu, S. J. C. Zhang, and m. m. i. medicine, "Prediction of protein structural classes for low-similarity sequences based on consensus sequence and segmented PSSM," vol. 2015, 2015.

Examples

A<-CS_PSe_PSSM(system.file("extdata", "C7GS16.txt.pssm", package="PSSMCOOL"),"total")

<table>
<thead>
<tr>
<th>DFMCA_PSSM</th>
<th>DMACA-PSSM feature</th>
</tr>
</thead>
</table>

Description

In this feature each column of PSSM Matrix, can be regarded as a time series. Each PSSM contains 20 columns. Hence, each PSSM can be considered as 20 time series. The detrended moving-average cross-correlation analysis (DMCA) is developed to measure the level of cross-correlation between two non-stationary time series by fusing the detrended cross-correlation analysis (DCCA) and the detrended moving average (DMA). This function utilizes this algorithm for each column and each pair of columns to produce a feature vector of length 290.

Usage

DFMCA_PSSM(pssm_name, n)

Arguments

- **pssm_name**: name of PSSM Matrix file
- **n**: A parameter called the window size that must be smaller than the length of the sequence

Value

feature vector of length 210

Note

parameter n must be equal or greater than 3 and equal or less than L which L is length of protein

References


Examples

as<-DFMCA_PSSM(system.file("extdata", "C7GQ57.txt.pssm", package="PSSMCOOL"),7)
Discrete Cosine Transform

Description
To construct this feature vector, Two-Dimensional DCT algorithm has been used by applying \texttt{dct} function from \texttt{dtt} package which DCT stands for Discrete Cosin Transform.

Usage
\texttt{Discrete.Cosine.Transform(pssm\_name)}

Arguments
\texttt{pssm\_name} \hspace{1cm} name of PSSM Matrix file

Value

feature vector of length 400

References

Examples
\begin{verbatim}
as<-Discrete.Cosine.Transform(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"))
\end{verbatim}

Disulfide connectivity feature

Description
This feature is used to predict the disulfide bond within a protein.

Usage
\texttt{disulfid(pssm\_name)}
Arguments

pssm_name  name of PSSM Matrix file

Details

For the purpose of predicting disulfide bond in protein at first, the total number of cysteine amino acids in the protein sequence is counted and their position in the protein sequence is identified. Then, using a sliding window with length of 13, moved on the PSSM matrix from top to bottom so that the middle of the window is on the amino acid cysteine, then the rows below the matrix obtained from the PSSM matrix with dimension of 13 x 20 are placed next to each other to get a feature vector with a length of 260 = 20 * 13 per cysteine, and if the position of the first and last cysteine in the protein sequence is such that the middle of sliding window is not on cysteine residue when moving on PSSM Matrix, then the required number of zero rows from top and bottom is added to the PSSM matrix to achieve this goal. Thus, for every cystine amino acid presented in protein sequence, a feature vector with a length of 260 is formed. Then all the pairwise combinations of these cysteines is wrote in the first column of a table, and in front of each of these pairwise combinations, the corresponding feature vectors are sticked together to get a feature vector of length 520 for each of these compounds. Finally, the table obtained in this way will have the number of rows equal to the number of all pairwise combinations of these cysteines and the number of columns will be equal to 521 (the first column includes the name of these pair combinations). And it is easy to divide this table into training and testing data and predict the desired disulfide bonds between cysteines.

Value

a table with number of all cyctein pairs in rows and 521 columns correspond to feature vector length.

References


Examples

aq<-disulfid(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"))

DPC_PSSM          DPC-PSSM, AAC-PSSM and AADP-PSSM feature vectors
**Description**

This feature is combination of amino acid composition and dipeptide composition feature vectors. DPC feature stands for dipeptide composition, which multiplies the values in two consecutive rows and two different columns, calculating this for each of the different columns and obtaining the sum of these and for each. And for both columns, the product divides their sum by L-1, and because the result depends on two different columns, length of this feature vector would be 400. AAC-PSSM is actually mean of PSSM Matrix columns which its length is 20. eventually AADP-PSSM is combination of these vectors and with length 420.

**Usage**

```r
DPC_PSSM(pssm_name)
```

**Arguments**

- `pssm_name` name of PSSM Matrix file

**Value**

feature vector of length 420

**References**


**Examples**

```r
ss<-DPC_PSSM(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"))
```

**Description**

This feature results from the connection of two vectors. The vector is the first feature of a vector with a length of 40, which calculates the average of positive and negative values for each column separately and puts them together. in the second feature vector, correspond to each column the difference between the numbers in the rows that have distance of k is calculated, and then the square average for the differences that are positive is calculated, and the same action for the differences that are negative is performed. since k varies between 1 and \( \alpha \), and because the value of \( \alpha \) in this function is equal to 2, the length of the second feature vector will be 80, which by merging with the first feature vector, the total feature vector of length 120 will be obtained.

**Usage**

```r
DP_PSSM(pssm_name, a = 2)
```
**Arguments**

- **pssm_name**: name of PSSM matrix file
- **a**: fixed parameter that user chooses which usually equals to 2

**Value**

- feature vector of length 120

**References**


**Examples**

```r
ss<-DP_PSSM(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"))
```

| dwt_PSSM | discrete wavelet transform feature vector |

**Description**

In construction of this feature vector, the `dwt.nondyadic` function is used from "waveslim", package to calculate the discrete wavelet transform for each column of the PSSM matrix, which considers it as a discrete signal. At last, 4 levels DWT is used to analysis of these discrete signals of PSSM (each column) and extracted the PSSM-DWT feature from PSSM of protein.

**Usage**

```r
dwt_PSSM(pssm_name)
```

**Arguments**

- **pssm_name**: name of psm Matrix file

**Value**

- feature vector of length 80

**References**


Examples

```r
as <- dwt_PSSM(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"))
```

<table>
<thead>
<tr>
<th>EDP_MEDP</th>
<th>EDP_EEDP_MEDP feature vector</th>
</tr>
</thead>
</table>

Description

This is a feature vector of length 420 which is used for prediction of protein structural class for low-similarity sequences. First, an ED-PSSM Matrix with 20x20 dimensions is constructed from the PSSM Matrix. Then, by using this Matrix, EDP and EEDP vectors are obtained. Eventually, the MEDP feature vector is obtained by fusing these vectors.

Usage

```r
EDP_MEDP(pssm_name)
```

Arguments

- `pssm_name` is the name of the PSSM Matrix file.

Value

- a feature vector of length 420

References


Examples

```r
as <- EDP_MEDP(paste0(system.file("extdata", package="PSSMCOOL"), "/C7GS61.txt.pssm"))
```
**FPSSM**

*D-FPSSM and SF-PSSM feature vectors*

**Description**

This function produces list of two feature vectors named D-FPSSM and S-FPSSM which then used by FPSSM2 function to construct feature vector of length 100 for each pair of proteins which then used for protein-protein interaction prediction in each dataset.

**Usage**

FPSSM(pssm_name, hk)

**Arguments**

- `pssm_name` name of PSSM Matrix file
- `hk` a parameter that indicates which amino acid alphabet must be used

**Value**

two feature vectors of different length which is used in later steps.

**References**


**Examples**

q<-FPSSM(system.file("extdata","C7GQS7.txt.pssm",package="PSSMCOOL"),8)

---

**FPSSM2**

*Mixture of Two FPSSM Features*

**Description**

This function takes two PSSM files as argument and uses FPSSM function for making feature vector of length 100 correspond to this pair of proteins.

**Usage**

FPSSM2(pssm_name1, pssm_name2, hk)
grey_pssm_pseAAC

Arguments

- **pssm_name1**: The name of first PSSM Matrix file
- **pssm_name2**: The name of second PSSM Matrix file
- **hk**: a parameter that indicates which amino acid alphabet must be used

Value

Feature vector of length 100

References


See Also

- entropy
- mutinformation

Examples

```r
s1<-system.file("extdata","C7GQS7.txt.pssm",package="PSSMCOOL")
s2<-system.file("extdata","C7GRQ3.txt.pssm",package="PSSMCOOL")
s<-FPSSM2(s1,s2,8)
```

---

grey_pssm_pseAAC  grey pssm feature vector

Description

This function produces a feature vector of length 100 which the first 20 components of this vector is the normalized occurrence frequency of the native amino acids in the protein. the next 20 components are mean of 20 PSSM columns and grey system model approach as elaborated in (Min et al. 2013) is used to define the next 60 components.

Usage

```r
grey_pssm_pseAAC(pssm_name)
```

Arguments

- **pssm_name**: name of PSSM matrix file

Value

feature vector of length 100
kiderafactor

References

Examples
as<-grey_pssm_pseAAC(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"))

<table>
<thead>
<tr>
<th>kiderafactor</th>
<th>kiderafactor feature</th>
</tr>
</thead>
</table>

Description
For product of this feature vector similar to smoothed_PSSM feature, firstly PSSM Matrix is smoothed by appending zero vectors to its head and tail and utilizing sliding window of size odd, then this smoothed PSSM Matrix is condensed by the Kidera factors to produce feature vector for each residue.

Usage
kiderafactor(pssm_name, v = NULL)

Arguments
- **pssm_name**: name of PSSM Matrix file
- **v**: vector of amino acids positions which we want to produce feature vector for them.

Value
matrix of feature vectors

References
C. Fang, T. Noguchi, H. J. I. j. o. d. m. Yamana, and bioinformatics, "Condensing position-specific scoring matrixs by the Kidera factors for ligand-binding site prediction," vol. 12, no. 1, pp. 70-84, 2015.

See Also
smoothed_PSSM
**k_mers**

**Examples**

```r
w<-kiderafactor(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"),c(2,3,8,9))
```

---

**k_mers**

*3-mer and 2-mer in dataframe*

---

**Description**

This function produces all possible 2-mers or 3-mers by counting paths of length 2 or 3 in a dataframe which is thought as a graph.

**Usage**

```r
k_mers(s, h)
```

**Arguments**

- `s` a dataframe with 2 columns
- `h` is length of k-mer

**Value**

all k-mers by counting paths of length h in dataframe which is considered as a graph

**Examples**

```r
s1<-LETTERS[1:4]
s2<-LETTERS[3:6]
s<-data.frame(s1,s2)
dc<-k_mers(s,3)
```

---

**k_separated_bigrame**

*k_separated_bigrame feature vector*

---

**Description**

This feature is almost identical to the **DPC_PSSM** feature, and in fact the DPC feature is part of this feature (for k=1) and for two different columns, considers rows that differ by the size of the unit k.

**Usage**

```r
k_separated_bigrame(pssm_name, k)
```

**Arguments**

- `pssm_name` is name of PSSM Matrix file
- `k` a parameter that specifies separated length between amino acids
**Value**

a feature vector of length 400

**References**


**Examples**

```r
w<-k_seperated_bigrame(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"),5)
```

---

**LPC_PSSM**

*Linear predictive coding feature*

**Description**

This function uses Linear predictive coding algorithm for each column of PSSM Matrix. So in this script `lpc` function is used which produces a 14-dimensional vector for each column, since PSSM has 20 column eventually it will be obtained a 20*14=280 dimensional feature vector for each PSSM Matrix by this function.

**Usage**

```r
LPC_PSSM(pssm_name)
```

**Arguments**

- `pssm_name` name of PSSM Matrix file

**Value**

feature vector of length 280

**References**

L. Li et al., "PSSP-RFE: accurate prediction of protein structural class by recursive feature extraction from PSI-BLAST profile, physical-chemical property and functional annotations," vol. 9, no. 3, 2014.

**Examples**

```r
w<-LPC_PSSM(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"))
```
**MBMGACPSSM**

**MBMGACPSSM feature**

**Description**
In this function three different autocorrelation descriptors based on PSSM are adopted, which include: normalized Moreau-Broto autocorrelation, Moran autocorrelation and Geary autocorrelation descriptors. Autocorrelation descriptor is a powerful statistical tool and defined based on the distribution of amino acid properties along the sequence, which measures the correlation between two residues separated by a distance of $d$ in terms of their evolution scores.

**Usage**

MBMGACPSSM(pssm_name)

**Arguments**

- **pssm_name**: name of PSSM Matrix file

**Value**

- feature vector of length 560

**References**


**Examples**

w <- MBMGACPSSM(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"))

**pse_pssm**

**pseudo position-specific scoring matrix feature**

**Description**

This feature vector is combination of $F_{\text{PSSM}}$ feature vector and vector of correlation factors correspond to 20 columns in PSSM Matrix. $F_{\text{PSSM}}$ actually is mean of PSSM Matrix columns of length 20.

**Usage**

pse_pssm(pssm_name, g = 15)
Arguments

- pssm_name: is the name of PSSM matrix file
- g: a parameter which its size corresponds to the database used.

Value

- feature vector of length 20+20\*g

References


Examples

v<-pse_pssm(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"))

Description

This function firstly normalizes PSSM Matrix by formula: \( P - \min(P) / \max(P) - \min(P) \) then for any standard amino acid specifies its position in protein sequence whereby a sub-matrix from PSSM corresponding to these positions will be extracted, then for this sub-matrix computes \( \text{colSums} \) of its columns to create a vector of length 20, eventually a feature vector of length 400 will be obtained.

Usage

pssm400(pssm_name)

Arguments

- pssm_name: name of PSSM Matrix file

Value

- feature vector of length 400

Note

- if a specific amino acid did not exist in protein then \( \text{colSums} \) of whole PSSM is computed.
Examples

q<-pssm400(system.file("extdata","C7GQS7.txt.pssm",package="PSSMCOOL"))

<table>
<thead>
<tr>
<th>PSSMAC</th>
<th>PSSMAC feature</th>
</tr>
</thead>
</table>

Description

This feature, which stands for auto covariance transformation, for jth column calculates the average of this column, and then subtracts the resulting number from the elements on the i and (i + g)th rows of this column, and finally multiplies them. By changing the variable i from 1 to L-g, it calculates the sum of these, since the variable j changes between 1 and 20, and the variable g between 1 and 10 eventually a feature vector of length 200 will be obtained.

Usage

PSSMAC(pssm_name)

Arguments

pssm_name name of PSSM Matrix files

Value

feature vector of length 200

References


Examples

w<-PSSMAC(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"))
PSSMBLOCK  

**PSSM BLOCK feature vector**

**Description**

In this feature at first PSSM Matrix is divided to Blocks based on Number N which user imports. Then for each Block mean of columns is computed to get 20-dimensional vector, eventually by appending these vectors to each other final feature vector is obtained.

**Usage**

`PSSMBLOCK(pssm_name, N)`

**Arguments**

- `pssm_name`: name of PSSM Matrix file
- `N`: number of blocks

**Value**

feature vector that it’s length depends on parameter N

**References**


**Examples**

```r
as<-PSSMBLOCK(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"),5)
```

---

**pssm_ac**

**auto covariance transformation feature vector**

**Description**

The AC variable measures the correlation of the same property between two residues separated by a distance of lg along the sequence

**Usage**

`pssm_ac(pssm_name, lg = 18)`
pssm_cc

Arguments

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>pssm_name</td>
<td>name of the PSSM Matrix file</td>
</tr>
<tr>
<td>lg</td>
<td>a parameter which indicates distance between two residues</td>
</tr>
</tbody>
</table>

Value

feature vector which its length depends on parameter lg

Note

in use of this function The lg parameter must be less than the length of the smallest sequence in the database.

References


Examples

```r
e<-pssm_ac(system.file("extdata", "C7GQ57.txt.pssm", package="PSSMCOOL"),17)```

---

pssm_cc  Cross covarianse feature vector

Description

The CC variable measures the correlation of two different properties between two residues separated by a distance of t along the sequence.

Usage

```r
pssm_cc(pssm_name, t)
```

Arguments

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>pssm_name</td>
<td>name of PSSM Matrix file</td>
</tr>
<tr>
<td>t</td>
<td>shortest protein length in dataset minus one</td>
</tr>
</tbody>
</table>

Value

feature vector of length 380

References

Examples

```r
aa<-pssm_cc(system.file("extdata","C7GQS7.txt.pssm",package="PSSMCOOL"),18)
```

### Description

In this feature, by considering a specific column, at first sum of all components in this column is denoted by "L", then starting from the first row in this column, the components are added together to reaching a value less than or equal to 25 is calculated and stored. In the next step, the same work is done starting from the first row to reaching a value less than or equal to 50 is done started from the last row. To reaching 25 By appending these saved numbers together for each column, a vector of length 4 is obtained. If this is done for all the columns and the obtained vectors are connected to each other, for each protein, a feature vector of length 80 is obtained which its name is PSSM-SD.

### Usage

```r
PSSM_SD(pssm_name)
```

### Arguments

- `pssm_name` name of PSSM Matrix file

### Value

feature vector of length 80

### References


### Examples

```r
ww<-PSSM_SD(system.file("extdata","C7GQS7.txt.pssm",package="PSSMCOOL"))
```
pssm_seg

**PSSM-Seg feature vector**

Description

This feature vector uses PSSM-SD to produce Segmented Auto Covariance Features.

Usage

```r
pssm_seg(pssm_name, m = 4)
```

Arguments

- `pssm_name`: name of PSSM Matrix file
- `m`: a parameter between 1 and 11

Value

feature vector of length 100

References


See Also

- PSSM_SD

Examples

```r
q <- pssm_seg(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"), 3)
```

rpssm

**RPSSM feature**

Description

To obtain this feature, first the columns of the PSSM matrix are merged to obtain an L*10 matrix. Then, with a relationship similar to the auto covariance transformation feature, this feature with a length of 110 is obtained from this matrix.

Usage

```r
rpssm(pssm_name)
```
Arguments

pssm_name  name of PSSM Matrix file

Value

feature vector of length 110

References


Examples

w<-rpssm(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"))

zz<- scsh2(system.file("extdata", "C7GRQ3.txt.pssm", package="PSSMCOOL"), 2)
**single_Average**

**Description**
This descriptor is a variant of the Average Block descriptor and is designed to group together rows related to the same amino acid, thus considering domains of a sequence with similar conservation rates.

**Usage**
```r
single_Average(pssm_name)
```

**Arguments**
- `pssm_name` name of PSSM Matrix file

**Value**
feature vector of length 400

**References**

**See Also**
- `Averag_Block`

**Examples**
```r
w<-single_Average(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"))
```

---

**smoothed_PSSM**

**Description**
In this function at first a Matrix called smoothed-PSSM is constructed from PSSM Matrix by applying "ws" parameter which called sliding window size and taken from user and usually is equals to 7. Then using other window size parameter "w" which usually equals to 11 at each position smoothed feature vector is constructed.

**Usage**
```r
smoothed_PSSM(pssm_name, ws, w, v = NULL)
```
Arguments

**pssm_name**  
name of PSSM Matrix file

**ws**  
window size for smoothing PSSM Matrix

**w**  
window size for extracting feature vector

**v**  
vector of desired positions to extract their features

Details

In the construction of a smoothed PSSM, each row vector of a residue $\alpha_i$ is represented and smoothed by the summation of $ws$ surrounding row vectors $(V_{smoothed_i} = V_i - (ws-1)/2 + \ldots + V_i + \ldots + V_i + (ws+1)/2)$ For the N-terminal and C-terminal of a protein, $(w-1)/2$ ZERO vectors, are appended to the head or tail of a smoothed PSSM profile. Using the smoothed PSSM encoding scheme the feature vector of a residue $\alpha_i$ is represented by $(V_{smoothed_i} - (ws-1)/2, \ldots, V_{smoothed_i}, \ldots, V_{smoothed_i} + (ws+1)/2)$ The feature values in each vector are normalized to a range between -1 and 1.

Value

a matrix of feature vectors

References


See Also

kiderafactor

Examples

```r
w<-smoothed_PSSM(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"),7,11,c(2,3,8,9))
```

<table>
<thead>
<tr>
<th>SOMA_PSSM</th>
<th>SOMA PSSM Feature</th>
</tr>
</thead>
</table>

Description

In this function each column can be viewed as a stochastic time series, and each PSSM contains 20 columns, in other words, each PSSM contains 20 stochastic time series and Second-order moving average (SOMA) algorithm is applied to these columns to extract SOMA PSSM feature vector.

Usage

```
SOMA_PSSM(pssm_name)
```

Arguments

**pssm_name**  
name of PSSM file
**Value**

feature vector of length 160

**References**


**Examples**

```r
w<-SOMA_PSSM(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"))
```

---

**SVD_PSSM**

**Singular Value Decomposition (SVD)**

**Description**

Singular value decomposition is a general purpose matrix factorization approach that has many useful applications in signal processing and statistics. In this function SVD is applied to a matrix representation of a protein with the aim of reducing its dimensionality. Given an input matrix $\text{Mat}$ with dimensions $N \times M$, SVD is used to calculate its factorization of the form: $\text{Mat} = U \Sigma V^T$, where $\Sigma$ is a diagonal matrix whose diagonal entries are known as the singular values of $\text{Mat}$. The resulting descriptor is the ordered set of singular values: $SV \in \mathbb{R}^L$, where $L = \min(M, N)$, and here `svd` function is used for this purpose.

**Usage**

```r
SVD_PSSM(pssm_name)
```

**Arguments**

- `pssm_name` name of PSSM Matrix file

**Value**

feature vector of length 20

**References**


**Examples**

```r
w<-SVD_PSSM(system.file("extdata", "C7GQS7.txt.pssm", package="PSSMCOOL"))
```
three_mer  3-Mer and 2-Mer

Description

This function produces all possible k-mers from 20 amino acids for use in other functions.

Usage

three_mer(k)

Arguments

k is length of k-mer which user imports

Value

a matrix which its first row includes all k-mers

Examples

ax<-three_mer(3)

trigrame_pssm  trigrame feature vector

Description

This feature vector is 8000-dimensioned feature vector which is computed from tri-gram probability matrix T obtained from PSSM Matrix. To achieve this purpose elements in three successive rows and arbitrary columns are multiplied together then these results are added together by changing variable i from 1 to L-1, which i is counter of row and L indicates protein length. Since there are 20 columns thus final feature vector would be of length 8000.

Usage

trigrame_pssm(pssm_name)

Arguments

pssm_name name of PSSM Matrix file

Value

feature vector of length 8000
References
Paliwal, K.K., et al. (2014) A tri-gram based feature extraction technique using linear probabilities of position specific scoring matrix for protein fold recognition, IEEE transactions on nanobioscience, 13, 44-50

Examples
as<-trigrame_pssm(paste0(system.file("extdata",package="PSSMCOOL"),"/C7GSI6.txt.pssm"))
Index

<table>
<thead>
<tr>
<th>Term</th>
<th>Page(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AATP_TPCC</td>
<td>3, 4</td>
</tr>
<tr>
<td>AB_PSSM</td>
<td>3, 5</td>
</tr>
<tr>
<td>Averag_Block</td>
<td>3, 6, 27</td>
</tr>
<tr>
<td>colSums</td>
<td>20</td>
</tr>
<tr>
<td>consensus_sequence</td>
<td>3, 6</td>
</tr>
<tr>
<td>CS_PSe_PSSM</td>
<td>3, 7</td>
</tr>
<tr>
<td>dct</td>
<td>9</td>
</tr>
<tr>
<td>DFMCA_PSSM</td>
<td>3, 8</td>
</tr>
<tr>
<td>Discrete_Cosine_Transform</td>
<td>3, 9</td>
</tr>
<tr>
<td>disulfdi</td>
<td>3, 9</td>
</tr>
<tr>
<td>DP_PSSM</td>
<td>3, 11</td>
</tr>
<tr>
<td>DPC_PSSM</td>
<td>3–5, 10, 17</td>
</tr>
<tr>
<td>dwt.nondyadic</td>
<td>12</td>
</tr>
<tr>
<td>dwt_PSSM</td>
<td>3, 12</td>
</tr>
<tr>
<td>EDP_MEDP</td>
<td>3, 13</td>
</tr>
<tr>
<td>entropy</td>
<td>15</td>
</tr>
<tr>
<td>FPSSM</td>
<td>3, 14</td>
</tr>
<tr>
<td>FPSSM2</td>
<td>3, 14</td>
</tr>
<tr>
<td>grey_pssm_pseAAC</td>
<td>3, 15</td>
</tr>
<tr>
<td>k_mers</td>
<td>3, 17</td>
</tr>
<tr>
<td>k_separated_bigrame</td>
<td>3, 17</td>
</tr>
<tr>
<td>kiderafactor</td>
<td>3, 16, 28</td>
</tr>
<tr>
<td>lpc</td>
<td>18</td>
</tr>
<tr>
<td>LPC_PSSM</td>
<td>3, 18</td>
</tr>
<tr>
<td>MBMGACPSSM</td>
<td>3, 19</td>
</tr>
<tr>
<td>mutinformation</td>
<td>15</td>
</tr>
<tr>
<td>pse_pssm</td>
<td>4, 19</td>
</tr>
<tr>
<td>pssm400</td>
<td>4, 20</td>
</tr>
<tr>
<td>pssm_ac</td>
<td>4, 22</td>
</tr>
<tr>
<td>pssm_cc</td>
<td>4, 23</td>
</tr>
<tr>
<td>PSSM_SD</td>
<td>4, 24, 25</td>
</tr>
<tr>
<td>pssm_seg</td>
<td>4, 25</td>
</tr>
<tr>
<td>PSSMAC</td>
<td>4, 21</td>
</tr>
<tr>
<td>PSSMBLOCK</td>
<td>4, 6, 22</td>
</tr>
<tr>
<td>PSSMCOOL-package</td>
<td>3</td>
</tr>
<tr>
<td>rpssm</td>
<td>4, 25</td>
</tr>
<tr>
<td>scsh2</td>
<td>4, 26</td>
</tr>
<tr>
<td>single_Average</td>
<td>4, 27</td>
</tr>
<tr>
<td>smoothed_PSSM</td>
<td>4, 16, 27</td>
</tr>
<tr>
<td>SOMA_PSSM</td>
<td>4, 28</td>
</tr>
<tr>
<td>svd</td>
<td>29</td>
</tr>
<tr>
<td>SVD_PSSM</td>
<td>4, 29</td>
</tr>
<tr>
<td>three_mer</td>
<td>30</td>
</tr>
<tr>
<td>trigrame_pssm</td>
<td>4, 30</td>
</tr>
</tbody>
</table>