Package ‘Peptides’

March 18, 2018

Version 2.3
Date 2018-03-18
Title Calculate Indices and Theoretical Physicochemical Properties of Protein Sequences
URL https://github.com/dosorio/Peptides/
Suggests testthat
Description Includes functions to calculate several physicochemical properties and indices for amino-acid sequences as well as to read and plot ‘XVG’ output files from the ‘GROMACS’ molecular dynamics package.
License GPL-2
LinkingTo Rcpp
Imports Rcpp
RoxygenNote 6.0.1
NeedsCompilation yes
Author Daniel Osorio [aut, cre], Paola Rondon-Villarreal [aut, ths], Rodrigo Torres [aut, ths], J. Sebastian Paez [ctb]
Maintainer Daniel Osorio <dcosorioh@unal.edu.co>
Repository CRAN
Date/Publication 2018-03-18 17:36:57 UTC

R topics documented:

  aaComp .............................................. 2
  AAd ata ........................................... 4
  aaDescriptors .................................. 9
  aIndex ............................................. 10
  autoCorrelation ................................ 11
  autoCovariance .................................. 12
  blosumIndices .................................. 13
Compute the amino acid composition of a protein sequence

This function calculates the amount of amino acids of a particular class and classified as: Tiny, Small, Aliphatic, Aromatic, Non-polar, Polar, Charged, Basic and Acidic based on their size and R-groups using same function implemented in EMBOSS 'pepstat'. The output is a matrix with the number and percentage of amino acids of a particular class.

Usage

aaComp(seq)

Arguments

seq An amino-acid sequence

Details

Amino acids are zwitterionic molecules with an amine and a carboxyl group present in their structure. Some amino acids possess side chains with specific properties that allow grouping them in different ways. The aaComp function classifies amino acids based on their size, side chains, hydrophobicity, charge and their response to pH 7.
The output is a matrix with the number and percentage of amino acids of a particular class:

- **Tiny** (A + C + G + S + T)
- **Small** (A + B + C + D + G + N + P + S + T + V)
- **Aliphatic** (A + I + L + V)
- **Aromatic** (F + H + W + Y)
- **Non-polar** (A + C + F + G + I + L + M + P + V + W + Y)
- **Polar** (D + E + H + K + N + Q + R + S + T + Z)
- **Charged** (B + D + E + H + K + R + Z)
- **Basic** (H + K + R)
- **Acidic** (B + D + E + Z)

**Note**

This function was originally written by Alan Bleasby (ajb@ebi.ac.uk) for the EMBOSS package. Further information: http://emboss.sourceforge.net/apps/cvs/emboss/apps/pepstats.html

**References**


**Examples**

```bash
# COMPARED TO PEPSTATS
# http://emboss.bioinformatics.nl/cgi-bin/emboss/pepstats
# Property  Residues  Number  Mole%
# Tiny     (A+C+G+S+T) 4 19.048
# Small    (A+B+C+D+G+N+P+S+T+V) 4 19.048
# Aliphatic (A+I+L+V) 5 23.810
# Aromatic (F+H+W+Y) 5 23.810
# Non-polar (A+C+F+G+I+L+M+P+V+W+Y) 11 52.381
# Polar    (D+E+H+K+N+Q+R+S+T+Z) 9 42.857
# Charged  (B+D+E+H+K+R+Z) 8 38.095
# Basic    (H+K+R) 8 38.095
# Acidic   (B+D+E+Z) 0 00.000

## AA composition of PDB: 1D9J Cecropin Peptide
aaComp(seq= "KWKLFKKGIGKFLHSAKFKX")

## Output
#     Number  Mole %
# Tiny 4 19.048
# Small 4 19.048
# Aliphatic 5 23.810
# Aromatic 5 23.810
# NonPolar 11 52.381
```
Description

A list with a collection of properties, scales and indices for the 20 naturally occurring amino acids from various sources.

Usage

data(AAdata)

Format

A list as follows:

- Hydrophobicity The hydrophobicity is an important stabilization force in protein folding; this force changes depending on the solvent in which the protein is found.


• crucianiProperties: The three Cruciani et. al (2004) properties, are the scaled principal component scores that summarize a broad set of descriptors calculated based on the interaction of each amino acid residue with several chemical groups (or "probes"), such as charged ions, methyl, hydroxyl groups, and so forth.
  - PP1: Polarity,
  - PP2: Hydrophobicity,
  - PP3: H-bonding

• kideraFactors: The Kidera Factors were originally derived by applying multivariate analysis to 188 physical properties of the 20 amino acids and using dimension reduction techniques. A 10-dimensional vector of orthogonal factors was then obtained for each amino acid. The first four factors are essentially pure physical properties; the remaining six factors are superpositions of several physical properties, and are labelled for convenience by the name of the most heavily weighted component
  - helix.bend.pref: Helix/bend preference
  - side.chain.size: Side-chain size
  - extended.str.pref: Extended structure preference
  - hydrophobicity: Hydrophobicity
  - double.bend.pref: Double-bend preference
- partial.spec.vol: Partial specific volume
- flat.ext.pref: Flat extended preference
- occurrence.alpha.reg: Occurrence in alpha region
- pK.C: pK-C
- surrounding.hydrop: Surrounding hydrophobicity

- pK
  - EMBOSS: EMBOSS data are from http://emboss.sourceforge.net/apps/release/5.0/emboss/apps/iep.html.
- zScales The five Sandberg et al. (1998) Z-scales describe each amino acid with numerical values, descriptors, which represent the physicochemical properties of the amino acids including NMR data and thin-layer chromatography (TLC) data.
  - Z1: Lipophilicity
  - Z2: Steric properties (Steric bulk/Polarizability)
  - Z3: Electronic properties (Polarity / Charge)
  - Z4: Related to electronegativity, heat of formation, electrophilicity and hardness.
  - Z5: Related to electronegativity, heat of formation, electrophilicity and hardness.
- FASGAI Factor Analysis Scale of Generalized Amino Acid Information (FASGAI) proposed by Liang and Li (2007), is a set of amino acid descriptors, that reflects hydrophobicity, alpha and turn propensities, bulky properties, compositional characteristics, local flexibility, and electronic properties, was derived from multi-dimensional properties of 20 naturally occurring amino acids.
  - F1: Hydrophobicity index
  - F2: Alpha and turn propensities
  - F3: Bulky properties
  - F4: Compositional characteristic index
  - F5: Local flexibility
  - F6: Electronic properties
The principal components score Vectors of Hydrophobic, Steric, and Electronic properties, is derived from principal components analysis (PCA) on independent families of 18 hydrophobic properties, 17 steric properties, and 15 electronic properties, respectively, which are included in total 50 physicochemical variables of 20 coded amino acids.

- VHSE1 and VHSE2: Hydrophobic properties
- VHSE3 and VHSE4: Steric properties
- VHSE5 to VHSE8: Electronic properties

Source

- Hydrophobicity
  - ExPASy-Protscale (http://web.expasy.org/protscale/)
  - AAIndex Database (http://www.genome.jp/aaindex/)

- pK

References

- Hydrophobicity

- crucianiProperties

- kideraFactors

- pK

**• zScales**

**• FASGAI**

**• VHSE**

<table>
<thead>
<tr>
<th>Descriptors</th>
<th>Compute 66 descriptors for each amino acid of a protein sequence.</th>
</tr>
</thead>
</table>

**Description**

The function return 66 amino acid descriptors for the 20 natural amino acids. Available descriptors are:

- **VHSE**: VHSE-scales (principal components score Vectors of Hydrophobic, Steric, and Electronic properties), is derived from principal components analysis (PCA) on independent families of 18 hydrophobic properties, 17 steric properties, and 15 electronic properties, respectively, which are included in total 50 physicochemical variables of 20 coded amino acids.


Usage

aaDescriptors(seq)

Arguments

seq An amino-acids sequence. If multiple sequences are given all of them must have the same length (gap symbols are allowed.)

Value

a matrix with 66 amino acid descriptors for each aminoacid in a protein sequence.

Examples

aaDescriptors(seq = "KLKLLLLKLK")

aIndex

Compute the aliphatic index of a protein sequence

Description

This function calculates the Ikai (1980) aliphatic index of a protein. The aindex is defined as the relative volume occupied by aliphatic side chains (Alanine, Valine, Isoleucine, and Leucine). It may be regarded as a positive factor for the increase of thermostability of globular proteins.

Usage

aIndex(seq)

Arguments

seq An amino-acids sequence
**autoCorrelation**

**Details**

Aliphatic amino acids (A, I, L and V) are responsible for the thermal stability of proteins. The aliphatic index was proposed by Ikai (1980) and evaluates the thermostability of proteins based on the percentage of each of the aliphatic amino acids that build up proteins.

**Value**

The computed aliphatic index for a given amino-acids sequence

**References**


**Examples**

```r
# COMPARED TO ExPASy ALIPHATIC INDEX
# http://web.expasy.org/protparam/
# SEQUENCE: SDKEVDEVAALSDELITLE
# Aliphatic index: 117.00

aIndex(seq = "SDKEVDEVAALSDELITLE")
# [1] 117
```

**autoCorrelation**  
Compute the auto-correlation index of a protein sequence

**Description**

This function computes the Cruciani et al (2004) auto-correlation index. The autoCorrelation index is calculated for a lag 'd' using a descriptor 'f' (centred) over a sequence of length 'L'.

**Usage**

```r
autoCorrelation(sequence, lag, property, center = TRUE)
```

**Arguments**

- `sequence`: An amino-acids sequence
- `lag`: A value for a lag, the max value is equal to the length of shortest peptide minus one.
- `property`: A property to use as value to be correlated.
- `center`: A logical value TRUE or FALSE if the property must be centered.

**Value**

The computed auto-correlation index for a given amino-acids sequence
autoCovariance

References


Examples

# Loading a property to evaluate its autocorrelation
data(AAdata)

# Calculate the auto-correlation index for a lag=1
autoCorrelation(
    sequence = "SDKEDEVDAALSLEITLE",
    lag = 1,
    property = AAdata$Hydrophobicity$KyteDoolittle,
    center = TRUE
)
# [1] -0.3519908

# Calculate the auto-correlation index for a lag=5
autoCorrelation(
    sequence = "SDKEDEVDAALSLEITLE",
    lag = 5,
    property = AAdata$Hydrophobicity$KyteDoolittle,
    center = TRUE
)
# [1] 0.001133553

autoCovariance  
Compute the auto-covariance index of a protein sequence

Description

This function computes the Cruciani et al (2004) auto-covariance index. The autoCovariance index is calculated for a lag 'd' using a descriptor 'f' (centred) over a sequence of length 'L'.

Usage

autoCovariance(sequence, lag, property, center = TRUE)

Arguments

sequence An amino-acids sequence
lag A value for a lag, the max value is equal to the length of the shortest peptide minus one.
property A property to use as value to evaluate the covariance.
center A logical value TRUE or FALSE if the property must be centered.
Value

The computed auto-covariance index for a given amino-acids sequence

References


Examples

# Loading a property to evaluate its autocorrelation
data(AAdata)

# Calculate the auto-covariance index for a lag=1
autoCovariance(
    sequence = "SDKEVDEVDAALSLEITLE",
    lag = 1,
    property = AAdata$Hydrophobicity$KyteDoolittle,
    center = TRUE
)
# [1] -0.4140053

# Calculate the auto-covariance index for a lag=5
autoCovariance(
    sequence = "SDKEVDEVDAALSLEITLE",
    lag = 5,
    property = AAdata$Hydrophobicity$KyteDoolittle,
    center = TRUE
)
# [1] 0.001000336

blosumIndices

Compute the BLOSUM62 derived indices of a protein sequence

Description

BLOSUM indices were derived of physicochemical properties that have been subjected to a VARI-MAX analyses and an alignment matrix of the 20 natural AAs using the BLOSUM62 matrix.

Usage

blosumIndices(seq)

Arguments

seq An amino-acids sequence
Value

The computed average of BLOSUM indices of all the amino acids in the corresponding peptide sequence.

References


Examples

```r
blosumIndices(seq = "KLKLKLLKLK")
# [[1]]
# [BLOSUM1 BLOSUM2 BLOSUM3 BLOSUM4 BLOSUM5
# -0.4827273 -0.5618182 -0.8509091 -0.4172727 0.3172727]
# [BLOSUM6 BLOSUM7 BLOSUM8 BLOSUM9 BLOSUM10
# 0.2527273 0.1463636 0.1427273 -0.2145455 -0.3218182]
```

---

**boman**

*Compute the Boman (Potential Protein Interaction) index*

Description

This function computes the potential protein interaction index proposed by Boman (2003) based in the amino acid sequence of a protein. The index is equal to the sum of the solubility values for all residues in a sequence, it might give an overall estimate of the potential of a peptide to bind to membranes or other proteins as receptors, to normalize it is divided by the number of residues. A protein have high binding potential if the index value is higher than 2.48.

Usage

```r
boman(seq)
```

Arguments

- `seq` An amino-acid sequence

Details

The potential protein interaction index was proposed by Boman (2003) as an easy way to differentiate the action mechanism of hormones (protein-protein) and antimicrobial peptides (protein-membrane) through this index. This function predicts the potential peptide interaction with another protein.

Value

The computed potential protein-protein interaction for a given amino-acids sequence
References


Examples

```r
# COMPARED TO YADAMP DATABASE
# http://yadamp.unisa.it/showItem.aspx?yadampid=8455&x=0.4373912
# SEQUENCE: FLPVLAGLTPSIVPKLVCLLTKKC
# BOMAN INDEX -1.24

boman(seq= "FLPVLAGLTPSIVPKLVCLLTKKC")
# [1] -1.235833
```

---

charge

Compute the theoretical net charge of a protein sequence

Description

This function computes the net charge of a protein sequence based on the Henderson-Hasselbalch equation described by Moore, D. S. (1985). The net charge can be calculated at defined pH using one of the 9 pKa scales availables: Bjellqvist, Dawson, EMBoss, Lehninger, Murray, Rodwell, Sillero, Solomon or Stryer.

Usage

```r
charge(seq, pH = 7, pKscale = "Lehninger")
```

Arguments

- `seq` An amino-acids sequence
- `pH` A pH value
- `pKscale` A character string specifying the pKa scale to be used; must be one of "Bjellqvist", "Dawson", "EMBoSS", "Lehninger", "Murray", "Rodwell", "Sillero", "Solomon" or "Stryer"

References


EMBOSS data are from http://emboss.sourceforge.net/apps/release/5.0/emboss/apps/iep.html.


Examples

# COMPARED TO EMBOSS PEPSTATS
# http://emboss.bioinformatics.nl/cgi-bin/emboss/pepstats
# SEQUENCE: FLPVLAGTPISVPIKLCLLTKKC
# Charge = 3.0

charge(seq= "FLPVLAGTPISVPIKLCLLTKKC",pH= 7, pKscale= "Bjellqvist")
  # [1] 2.737303
charge(seq= "FLPVLAGTPISVPIKLCLLTKKC",pH= 7, pKscale= "EMBOSS")
  # [1] 2.914112
charge(seq= "FLPVLAGTPISVPIKLCLLTKKC",pH= 7, pKscale= "Murray")
  # [1] 2.987541
charge(seq= "FLPVLAGTPISVPIKLCLLTKKC",pH= 7, pKscale= "Sillero")
  # [1] 2.919812
charge(seq= "FLPVLAGTPISVPIKLCLLTKKC",pH= 7, pKscale= "Solomon")
  # [1] 2.844406
charge(seq= "FLPVLAGTPISVPIKLCLLTKKC",pH= 7, pKscale= "Stryer")
  # [1] 2.876504
charge(seq= "FLPVLAGTPISVPIKLCLLTKKC",pH= 7, pKscale= "Lehninger")
  # [1] 2.87315
charge(seq= "FLPVLAGTPISVPIKLCLLTKKC",pH= 7, pKscale= "Dawson")
  # [1] 2.844406
charge(seq= "FLPVLAGTPISVPIKLCLLTKKC",pH= 7, pKscale= "Rodwell")
  # [1] 2.819755

# COMPARED TO YADAMP
# http://yadamp.unisa.it/showItem.aspx?yadampid=84508x=0,7055475
# SEQUENCE: FLPVLAGTPISVPIKLCLLTKKC
# CHARGE pH5: 3.00
# CHARGE pH7: 2.91
# CHARGE pH9: 1.09
charge(seq= "FLPVLAGTPISVPIKLCLLTKKC",pH= 5, pKscale= "EMBOSS")
  # [1] 3.037398
charge(seq= "FLPVLAGTPISVPIKLCLLTKKC",pH= 7, pKscale= "EMBOSS")
  # [1] 2.914112
charge(seq= "FLPVLAGTPISVPIKLCLLTKKC",pH= 9, pKscale= "EMBOSS")
  # [1] 0.7184524
crossCovariance

# JUST ONE COMMAND
charge(seq="FLPVLAGLTPSIVPKLYCLLLKKC", pH= seq(from = 5, to = 9, by = 2), pKscale="EMBOSS")
# [1] 3.0373984 2.9141123 0.7184524

crossCovariance

Compute the cross-covariance index of a protein sequence

Description

This function computes the Cruciani et al (2004) cross-covariance index. The lagged crossCovariance index is calculated for a lag 'd' using two descriptors 'f1' and 'f2' (centred) over a sequence of length 'L'.

Usage

crossCovariance(sequence, lag, property1, property2, center = TRUE)

Arguments

sequence An amino-acids sequence
lag A value for a lag, the max value is equal to the length of the shortest peptide minus one.
property1 A property to use as value to evaluate the cross-covariance.
property2 A property to use as value to evaluate the cross-covariance.
center A logical value TRUE or FALSE if the property must be centered.

Value

The computed cross-covariance index for a given amino-acids sequence

References


Examples

# Loading a property to evaluate its autocorrelation
data(AAdata)

# Calculate the cross-covariance index for a lag=1
crossCovariance(
    sequence = "SDKEKDEVDAALSDELITLE",
    lag = 1,
    property1 = AAdata$Hydrophobicity$KyteDoolittle,
    property2 = AAdata$Hydrophobicity$Eisenberg,
crucianiProperties

Description

This function calculates the Cruciani properties of an amino-acids sequence using the scaled principal component scores that summarize a broad set of descriptors calculated based on the interaction of each amino acid residue with several chemical groups (or "probes"), such as charged ions, methyl, hydroxyl groups, and so forth.

Usage

```r
crucianiProperties(seq)
```

Arguments

- **seq** An amino-acids sequence

Value

The computed average of Cruciani properties of all the amino acids in the corresponding peptide sequence. Each PP represent an amino-acid property as follows:

- PP1: Polarity,
- PP2: Hydrophobicity,
- PP3: H-bonding

References

\textbf{fasgaiVectors}  

\textit{Compute the FASGAI vectors of a protein sequence}

\textbf{Description}

The FASGAI vectors (Factor Analysis Scales of Generalized Amino Acid Information) is a set of amino acid descriptors, that reflects hydrophobicity, alpha and turn propensities, bulky properties, compositional characteristics, local flexibility, and electronic properties, that can be utilized to represent the sequence structural features of peptides or protein motifs.

\textbf{Usage}

\texttt{fasgaiVectors(seq)}

\textbf{Arguments}

\begin{itemize}
  \item \texttt{seq} An amino-acids sequence
\end{itemize}

\textbf{Value}

The computed average of FASGAI factors of all the amino acids in the corresponding peptide sequence. Each factor represent an amino-acid property as follows:

- F1: Hydrophobicity index,
- F2: Alpha and turn propensities,
- F3: Bulky properties,
- F4: Compositional characteristic index,
- F5: Local flexibility,
- F6: Electronic properties

\textbf{References}


\textbf{Examples}

\begin{verbatim}
fasgaiVectors(seq = "QWGRCCWGPGRYCVRWC")
#      PP1   PP2   PP3
# -0.1130 -0.0220  0.2735

fasgaiVectors(seq = "QWGRCCWGPGRYCVRWC")
# [[1]]
#   F1    F2    F3    F4    F5    F6
# -0.13675 -0.45485 -0.11695 -0.45800 -0.38015  0.52740
\end{verbatim}
**hmoment**

*Compute the hydrophobic moment of a protein sequence*

**Description**

This function computes the hmoment based on Eisenberg, D., Weiss, R. M., & Terwilliger, T. C. (1984). Hydrophobic moment is a quantitative measure of the amphiphilicity perpendicular to the axis of any periodic peptide structure, such as the a-helix or b-sheet. It can be calculated for an amino acid sequence of N residues and their associated hydrophobicities Hn.

**Usage**

```r
hmoment(seq, angle = 100, window = 11)
```

**Arguments**

- `seq` An amino-acids sequence
- `angle` A protein rotational angle (Suggested: a-helix = 100, b-sheet = 160)
- `window` A sequence fraction length

**Details**

The hydrophobic moment was proposed by Eisenberg et al. (1982), as a quantitative measure of the amphiphilicity perpendicular to the axis of any periodic peptide structure. It is computed using the standardized Eisenberg (1984) scale, windows (fragment of sequence) of eleven amino acids (by default) and specifying the rotational angle at which it should be calculated.

**Value**

The computed maximal hydrophobic moment (uH) for a given amino-acids sequence

**Note**

This function was written by an anonymous reviewer of the RJournal

**References**


**Examples**

```r
# COMPARED TO EMBOSS:HMOMENT
# http://emboss.bioinformatics.nl/cgi-bin/emboss/hmoment
# SEQUENCE: FLPLVLAGLPSIVPKLVCPLTKKC
# ALPHA-HELIX ANGLE=100 : 0.52
# BETA-SHEET ANGLE=160 : 0.271
```
# ALPHA HELIX VALUE
hmoment(seq = "FLPVLGLTPSIVPKLVLLETKK", angle = 100, window = 11)
# [1] 0.5199226

# BETA SHEET VALUE
hmoment(seq = "FLPVLGLTPSIVPKLVLLETKK", angle = 160, window = 11)
# [1] 0.2705906

---

## hydrophobicity

**Compute the hydrophobicity index of a protein sequence**

### Description

This function calculates the GRAVY hydrophobicity index of an amino acids sequence using one of the 38 scales from different sources.

### Usage

```r
hydrophobicity(seq, scale = "KyteDoolittle")
```

### Arguments

- **seq**: An amino-acids sequence

### Details

The hydrophobicity is an important stabilization force in protein folding; this force changes depending on the solvent in which the protein is found. The hydrophobicity index is calculated adding the hydrophobicity of individual amino acids and dividing this value by the length of the sequence.

### Value

The computed GRAVY index for a given amino-acid sequence
References


Examples

# COMPARED TO GRAVY Grand average of hydropathicity (GRAVY) ExPASy
# http://web.expasy.org/cgi-bin/protparam/protparam
# SEQUENCE: QWGRRCWPPGRRYYCRCVRC
# GRAVY: -0.950

hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "Aboderin") # [1] 3.84
hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "AbrahamLeo") # [1] 0.092
hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "Argos") # [1] 1.033
hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "BlackMould") # [1] 0.50125
hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "BullBreese") # [1] 0.1575
hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "Casari") # [1] 0.38
hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "Chothia") # [1] 0.262
hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "Cid") # [1] 0.198
hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "Cowan3.4") # [1] 0.0845
hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "Cowan7.5") # [1] 0.0605
hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "Eisenberg") # [1] -0.3265
hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "Engelman") # [1] 2.31
hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "Fasman") # [1] -1.2905
hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "Fauchere") # [1] 0.527
hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "Goldsack") # [1] 1.2245
hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "Guy") # [1] 0.193
hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "HoppWoods") # [1] -0.14
hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "Janin") # [1] -0.105
hydrophobicity(seq = "QWGRRCWPPGRRYYCRCVRC", scale = "Jones")
### instaIndex

**Compute the instability index of a protein sequence**

#### Description

This function calculates the instability index proposed by Guruprasad (1990). This index predicts the stability of a protein based on its amino acid composition, a protein whose instability index is smaller than 40 is predicted as stable, a value above 40 predicts that the protein may be unstable.
Usage

instaIndex(seq)

Arguments

seq An amino-acids sequence

Value

The computed instability index for a given amino-acids sequence

References


Examples

# COMPARED TO ExPaSy INSTAINDEX
# http://web.expasy.org/protparam/
# SEQUNECE: QWGRCCWPGRRYCVRWC
# The instability index (II) is computed to be 83.68

instaIndex(seq = "QWGRCCWPGRRYCVRWC")
# [1] 83.68

kideraFactors

Compute the Kidera factors of a protein sequence

Description

The Kidera Factors were originally derived by applying multivariate analysis to 188 physical properties of the 20 amino acids and using dimension reduction techniques. This function calculates the average of the ten Kidera factors for a protein sequence.

Usage

kideraFactors(seq)

Arguments

seq An amino-acids sequence
**Value**

A list with the average of the ten Kidera factors. The first four factors are essentially pure physical properties; the remaining six factors are superpositions of several physical properties, and are labelled for convenience by the name of the most heavily weighted component.

- KF1: Helix/bend preference,
- KF2: Side-chain size,
- KF3: Extended structure preference,
- KF4: Hydrophobicity,
- KF5: Double-bend preference,
- KF6: Partial specific volume,
- KF7: Flat extended preference,
- KF8: Occurrence in alpha region,
- KF9: pK-C,
- KF10: Surrounding hydrophobicity

**References**


**Examples**

```r
kideraFactors(seq = "KLKLKLLK")
# [[1]]
#   KF1  KF2  KF3  KF4  KF5
#  0.7854545 0.2981818 -0.2363636 -0.0818182 0.2100000
#  1.8936364 1.02909091 -0.51272727 0.1118182 0.8100000
```

```r
lengthpep(seq)
```

**Description**

This function counts the number of amino acids in a protein sequence

**Usage**

`lengthpep(seq)`

**Arguments**

- `seq` An amino-acids sequence
Details

All proteins are formed by linear chains of small residues known as amino acids attached to each other by peptide bonds. The function `lengthpep` counts the number of amino acids in a sequence and returns a vector with the count for each peptide used as argument.

Examples

```r
# COMPARED TO ExPASy ProtParam
# http://web.expasy.org/protparam
# SEQUENCE: QWGRCCGWPGRRYCVRWC
# Number of amino acids: 20

lengthpep(seq = "QWGRCCGWPGRRYCVRWC")
# [1] 20
```

---

### membpos

**Compute theoretically the class of a protein sequence**

Description

This function calculates the theoretical class of a protein sequence based on the relationship between the hydrophobic moment and hydrophobicity scale proposed by Eisenberg (1984).

Usage

```
membpos(seq, angle = 100)
```

Arguments

- `seq` An amino-acids sequence
- `angle` A protein rotational angle

Details

Eisenberg et al. (1982) found a correlation between hydrophobicity and hydrophobic moment that defines the protein section as globular, transmembrane or superficial. The function calculates the hydrophobicity (H) and hydrophobic moment (uH) based on the standardized scale of Eisenberg (1984) using windows of 11 amino acids for calculate the theoretical fragment type.

Value

A data frame for each sequence given with the calculated class for each window of eleven amino-acids
### References


### Examples

```r
membpos(seq = "ARQQNLFINFCLILIFLLLI", angle = 100)
# Pep  H  uH MembPos
# 1 ARQQNLFINFCL 0.083 0.353 Globular
# 2 RQQNLFINFCLI 0.147 0.317 Globular
# 3 QQNLFINFCLIL 0.446 0.274 Globular
# 4 QNLFINFCLILI 0.632 0.274 Transmembrane
# 5 NLFINFCLILIF 0.802 0.253 Surface
# 6 LFINFCLILIFFL 0.955 0.113 Transmembrane
# 7 FINFCLILILFL 0.955 0.113 Transmembrane
# 8 INFCLILIFFLL 0.944 0.108 Transmembrane
# 9 NFCLILILILLI 0.944 0.132 Transmembrane

membpos(seq = "ARQQNLFINFCLILIFLLLI", angle = 160)
# Pep  H  uH MembPos
# 1 ARQQNLFINFCL 0.083 0.467 Globular
# 2 RQQNLFINFCLI 0.147 0.467 Globular
# 3 QQNLFINFCLIL 0.446 0.285 Globular
# 4 QNLFINFCLILI 0.632 0.358 Surface
# 5 NLFINFCLILIF 0.802 0.358 Surface
# 6 LFINFCLILIFFL 0.955 0.269 Surface
# 7 FINFCLILILFL 0.955 0.269 Surface
# 8 INFCLILIFFLL 0.944 0.257 Surface
# 9 NFCLILILILLI 0.944 0.229 Surface
```

---

**mswhimScores**

**Compute the MS-WHIM scores of a protein sequence**

**Description**

MS-WHIM scores were derived from 36 electrostatic potential properties derived from the three-dimensional structure of the 20 natural amino acids

**Usage**

```r
mswhimScores(seq)
```

**Arguments**

- `seq` An amino-acids sequence
Value

The computed average of MS-WHIM scores of all the amino acids in the corresponding peptide sequence.

References


Examples

```
mswhimScores(seq = "KLKLKLLK")
# [1]
# MSWHIM1 MSWHIM2 MSWHIM3
# -0.6563636 0.4872727 0.1163636
```

mw

**Compute the molecular weight of a protein sequence**

Description

This function calculates the molecular weight of a protein sequence. It is calculated as the sum of the mass of each amino acid using the scale available on Compute pI/MW tool.

Usage

```
mw(seq, monoisotopic = FALSE)
```

Arguments

- `seq` An amino-acids sequence
- `monoisotopic` A logical 'TRUE' or 'FALSE' indicating if monoisotopic weights of amino-acids should be used

Details

The molecular weight is the sum of the masses of each atom constituting a molecule. The molecular weight is directly related to the length of the amino acid sequence and is expressed in units called daltons (Da). In Peptides the function mw computes the molecular weight using the same formulas and weights as ExPASy’s "compute pI/mw" tool (Gasteiger et al., 2005).

Source

The formula and amino acid scale are the same available on ExPASy Compute pI/Mw tool: http://web.expasy.org/compute_pia
pepdata

References

Examples
# COMPARED TO ExPASy Compute pI/Mw tool
# http://web.expasy.org/compute_pi/
# SEQUENCE: QWGRCCGWGPGRYCVRWC
# Theoretical pI/Mw: 9.88 / 2485.91

mw(seq = "QWGRCCGWGPGRYCVRWC",monoisotopic = FALSE)  
  # [1] 2485.911

mw(seq = "QWGRCCGWGPGRYCVRWC",monoisotopic = TRUE)  
  # [1] 2484.12

pepdata  
Physicochemical properties and indices from 100 amino acid sequences

Description
Physicochemical properties and indices from 100 amino acid sequences (50 antimicrobial and 50 non antimicrobial)

Usage
data(pepdata)

Format
A data frame with 100 observations on the following 23 variables.

sequence a character vector with the sequences of 100 peptides (50 antimicrobial and 50 non-antimicrobial)
group Integer vector with the group code "0" for non antimicrobial and "1" for antimicrobial
length a numeric vector with the length of the amino acid sequence
mw a numeric vector with the molecular weight of the amino acid sequence
tinyAA A numeric vector with the fraction (as percent) of tiny amino acids that make up the sequence
smallAA A numeric vector with the fraction (as percent) of small amino acids that make up the sequence
aliphaticAA A numeric vector with the fraction (as percent) of aliphatic amino acids that make up the sequence
aromaticAA A numeric vector with the fraction (as percent) of aromatic amino acids that make up the sequence
nonpolarAA A numeric vector with the fraction (as percent) of non-polar amino acids that make up the sequence
polarAA A numeric vector with the fraction (as percent) of polar amino acids that make up the sequence
chargedAA A numeric vector with the fraction (as percent) of charged amino acids that make up the sequence
basicAA A numeric vector with the fraction (as percent) of basic amino acids that make up the sequence
acidicAA A numeric vector with the fraction (as percent) of acid amino acids that make up the sequence
charge a numeric vector with the charge of the amino acid sequence
pI a numeric vector with the isoelectric point of the amino acid sequence
aindex a numeric vector with the aliphatic index of the amino acid sequence
instaindex a numeric vector with the instability index of the amino acid sequence
boman a numeric vector with the potential peptide-interaction index of the amino acid sequence
hydrophobicity a numeric vector with the hydrophobicity index of the amino acid sequence
hmoment a numeric vector with the hydrophobic moment of the amino acid sequence
transmembrane A numeric vector with the fraction of Transmembrane windows of 11 amino acids that make up the sequence
surface A numeric vector with the fraction of Surface windows of 11 amino acids that make up the sequence
globular A numeric vector with the fraction of Globular windows of 11 amino acids that make up the sequence

\[
\text{pI} \quad \text{Compute the isoelectric point (pI) of a protein sequence}
\]

Description

The isoelectric point (pI), is the pH at which a particular molecule or surface carries no net electrical charge.

Usage

\[ \text{pI(seq, pKscale = "EMBOSS")} \]

Arguments

- **seq** An amino-acids sequence
- **pKscale** A character string specifying the pK scale to be used; must be one of "Bjellqvist", "EMBOSS", "Murray", "Sillero", "Solomon", "Stryer", "Lehninger", "Dawson" or "Rodwell"
Details

The isoelectric point (pI) is the pH at which the net charge of the protein is equal to 0. It is a variable that affects the solubility of the peptides under certain conditions of pH. When the pH of the solvent is equal to the pI of the protein, it tends to precipitate and lose its biological function.

Examples

```cpp
# COMPARED TO ExPASy ProtParam
# http://web.expasy.org/cgi-bin/protparam/protparam
# SEQUENCE: QWGRCCGWGPGRRYCVRWC
# Theoretical pI: 9.88

pI(seq= "QWGRCCGWGPGRRYCVRWC", pKscale= "Bjellqvist")
# [1] 9.881

# COMPARED TO EMBoss PEPSTATS
# http://emboss.bioinformatics.nl/cgi-bin/emboss/pepstats
# SEQUENCE: QWGRCCGWGPGRRYCVRWC
# Isoelectric Point = 9.7158

pI(seq= "QWGRCCGWGPGRRYCVRWC", pKscale= "EMBOSS")
# [1] 9.716

# OTHER SCALES

pI(seq= "QWGRCCGWGPGRRYCVRWC", pKscale= "Murray")
# [1] 9.818
pI(seq= "QWGRCCGWGPGRRYCVRWC", pKscale= "Sillero")
# [1] 9.891
pI(seq= "QWGRCCGWGPGRRYCVRWC", pKscale= "Solomon")
# [1] 9.582
pI(seq= "QWGRCCGWGPGRRYCVRWC", pKscale= "Stryer")
# [1] 9.623
pI(seq= "QWGRCCGWGPGRRYCVRWC", pKscale= "Lehninger")
# [1] 9.931
pI(seq= "QWGRCCGWGPGRRYCVRWC", pKscale= "Dawson")
# [1] 9.568
pI(seq= "QWGRCCGWGPGRRYCVRWC", pKscale= "Rodwell")
# [1] 9.718
```

---

**plotXVG**

**Plot time series from GROMACS XVG files**

**Description**

Read and plot output data from a XVG format file.

**Usage**

```cpp```
plotXVG(XVGfile, ...)
```
protFP

**Arguments**

- **XVGfile**
  A .XVG output file of the GROMACS molecular dynamics package
- ... Arguments to be passed to methods, such as graphical parameters.

**Details**

GROMACS (GROningen MAchine for Chemical Simulations) is a molecular dynamics package designed for simulations of proteins, lipids and nucleic acids. It is free, open source software released under the GNU General Public License. The file format used by GROMACS is XVG. This format can be displayed in graphical form through the GRACE program on UNIX/LINUX systems and the GNUPlot program on Windows. XVG files are plain text files containing tabular data separated by tabulators and two types of comments which contain data labels. Although manual editing is possible, this is not a viable option when working with multiple files of this type. For ease of reading, information management and data plotting, the functions `read.xvg` and `plot.xvg` were incorporated.

**Author(s)**

Latest: J. Sebastian Paez <jpaezpae@purdue.edu>
Original: Daniel Osorio <dcosorioh@unal.edu.co>

**References**


**Examples**

```r
XVGfile <- system.file("xvg-files/epot.xvg",package="Peptides")
plotXVG(XVGfile)
```

---

protFP

Compute the protFP descriptors of a protein sequence

**Description**

The ProtFP descriptor set was constructed from a large initial selection of indices obtained from the AAindex database for all 20 naturally occurring amino acids.

**Usage**

protFP(seq)

**Arguments**

- **seq**
  An amino-acids sequence
Value

The computed average of protFP descriptors of all the amino acids in the corresponding peptide sequence.

References


Examples

\begin{verbatim}
protFP(seq = "QWGRCCGWPGRYRCVRWC")
# [[1]]
# ProtFP1 ProtFP2 ProtFP3 ProtFP4 ProtFP5 ProtFP6 ProtFP7 ProtFP8
# 0.2065 -0.0565 1.9930 -0.2845 0.7315 0.7000 0.1715 0.1135
\end{verbatim}

readXVG

Read output data from a XVG format file.

Description

XVG is the default format file of the GROMACS molecular dynamics package, contains data formatted to be imported into the Grace 2-D plotting program.

Usage

readXVG(file)

Arguments

file A .XVG output file of the GROMACS molecular dynamics package

Details

GROMACS (GROningen MAchine for Chemical Simulations) is a molecular dynamics package designed for simulations of proteins, lipids and nucleic acids. It is free, open source software released under the GNU General Public License. The file format used by GROMACS is XVG. This format can be displayed in graphical form through the GRACE program on UNIX/LINUX systems and the GNUPlot program on Windows. XVG files are plain text files containing tabular data separated by tabulators and two types of comments which contain data labels. Although manual editing is possible, this is not a viable option when working with multiple files of this type. For ease of reading, information management and data plotting, the functions \texttt{read.xvg} and \texttt{plot.xvg} were incorporated.
**Description**

ST-scales were proposed by Yang et al, taking 827 properties into account which are mainly constitutional, topological, geometrical, hydrophobic, electronic, and steric properties of a total set of 167 AAs.

**Usage**

```r
stscales(seq)
```

**Arguments**

- `seq` An amino-acids sequence

**Value**

The computed average of ST-scales of all the amino acids in the corresponding peptide sequence.

**References**

tScales

**Examples**

```r
tsScales(seq = "QWGRRCGWGPGRRYCVRWC")
# [[1]]
#   ST1   ST2   ST3   ST4   ST5   ST6   ST7   ST8
# -0.63760  0.07965  0.05150  0.87135 -0.27905 -0.80995  0.58020  0.54400
```

**tScales**

*Compute the T-scales of a protein sequence*

**Description**

T-scales are based on 67 common topological descriptors of 135 amino acids. These topological descriptors are based on the connectivity table of amino acids alone, and to not explicitly consider 3D properties of each structure.

**Usage**

```r
tsScales(seq)
```

**Arguments**

- `seq` An amino-acids sequence

**Value**

The computed average of T-scales of all the amino acids in the corresponding peptide sequence.

**References**


**Examples**

```r
tsScales(seq = "QWGRRCGWGPGRRYCVRWC")
# [[1]]
#   T1    T2    T3    T4    T5
# -3.2700 -0.0035 -0.3855 -0.1475  0.7585
```
VHSE-scales (principal components score Vectors of Hydrophobic, Steric, and Electronic properties), is derived from principal components analysis (PCA) on independent families of 18 hydrophobic properties, 17 steric properties, and 15 electronic properties, respectively, which are included in total 50 physicochemical variables of 20 coded amino acids.

Usage

vhseScales(seq)

Arguments

seq 
An amino-acids sequence

Value

The computed average of VHSE-scales of all the amino acids in the corresponding peptide sequence. Each VHSE-scale represent an amino-acid property as follows:

- VHSE1 and VHSE2: Hydrophobic properties
- VHSE3 and VHSE4: Steric properties
- VHSE5 to VHSE8: Electronic properties

References


Examples

vhseScales(seq = "QWGRRCGWPGRRCVRWC")
# [[1]]
# VHSE1  VHSE2  VHSE3  VHSE4  VHSE5  VHSE6  VHSE7  VHSE8
#-0.1150  0.0630  -0.0055  0.7955  0.4355  0.2485  0.1740  -0.0960
zScales

*Compute the Z-scales of a protein sequence*

**Description**

Z-scales are based on physicochemical properties of the AAs including NMR data and thin-layer chromatography (TLC) data.

**Usage**

`zScales(seq)`

**Arguments**

- **seq**: An amino-acids sequence

**Value**

The computed average of Z-scales of all the amino acids in the corresponding peptide sequence. Each Z scale represent an amino-acid property as follows:

- **Z1**: Lipophilicity
- **Z2**: Steric properties (Steric bulk/Polarizability)
- **Z3**: Electronic properties (Polarity / Charge)
- **Z4 and Z5**: They relate electronegativity, heat of formation, electrophilicity and hardness.

**References**


**Examples**

```r
zScales(seq = "QWGRCCGWPGRRYCVRWC")
# [1]
# Z1   Z2   Z3   Z4   Z5
# 0.6200 0.0865 0.0665 0.7280 -0.8740
```
Index

aaComp, 2
AAdat, 4
aaDescriptors, 9
aindex, 10
autoCorrelation, 11
autoCovariance, 12
blosumIndices, 13
boman, 14
charge, 15
crossCovariance, 17
crucianiProperties, 18
fagsaiVectors, 19
hmoment, 20
hydrophobicity, 21
instaindex, 25
kideraFactors, 26
lengthpep, 27
membpos, 28
mswhimScores, 29
mw, 30
pepdata, 31
pi, 32
plotXVG, 33
protFP, 34
readXVG, 35
stScales, 36
tScales, 37
vhseScales, 38
zScales, 39