Package ‘protein8k’

August 16, 2021

**Type**  Package

**Title**  Perform Analysis and Create Visualizations of Proteins

**Version**  0.0.1

**Author**  Simon Liles

**Maintainer**  Simon Liles <simon@quantknot.com>

**Description**  Read Protein Data Bank (PDB) files, performs its analysis, and presents the result using different visualization types including 3D. The package also has additional capability for handling Virus Report data from the National Center for Biotechnology Information (NCBI) database.


**Depends**  R (>= 3.1.2)

**Imports**  pryr, lattice, methods, magick, dplyr, grid, gridExtra, ggplot2, rjson, rlang, shiny

**License**  CC0

**Encoding**  UTF-8

**LazyData**  true

**RoxygenNote**  7.1.1

**Suggests**  knitr, rmarkdown

**VignetteBuilder**  knitr

**NeedsCompilation**  no

**Repository**  CRAN

**Date/Publication**  2021-08-16 08:30:07 UTC

**R topics documented:**

fromJSONL . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2
getAtomicRecord . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2
getTitleSection . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3
**Description**

Retrieve the Atomic Record from a Protein Object

**Usage**

getAtomicRecord(protein)

**Arguments**

- **filepath**
  A character string indicating the file path from the working directory to the desired file.

- **maxLines**
  An integer representing the max number of lines to read. Negative values indicate that one should read up to the end of input on the connection.

**Value**

a large list, each element containing the contents of a JSON file after being converted.

---

**Description**

Decode a JSON List into an R List Object.

**Usage**

fromJSONL(filepath, maxLines = -1)

**Arguments**

- **filepath**
  A character string indicating the file path from the working directory to the desired file.

- **maxLines**
  An integer representing the max number of lines to read. Negative values indicate that one should read up to the end of input on the connection.

**Value**

a large list, each element containing the contents of a JSON file after being converted.
getTitleSection

Arguments
protein        input for a protein object

Format
Dataframe with 16 columns:

1. record_type: Type of record in this section. Generally ATOM or HETATM
2. serial_num: The serial number for the position of the atom in the sequence
3. atom_name: A name to identify the atom in a structure
4. alt_location_id:
5. residue_name: 3 character identifier for a residue
6. chain_id:
7. residue_seq_num: Number representing where in the sequence a residue is.
8. insert_residue_code:
9. x_ortho_coord: X coordinate in Ångstroms on an orthogonal plane
10. y_ortho_coord: Y coordinate in Ångstroms on an orthogonal plane
11. z_ortho_coord: Z coordinate in Ångstroms on an orthogonal plane
12. occupancy:
13. temp_factor: The amount of overall error in the measurement of an atom.
14. segment_id:
15. element_symbol: Periodic symbol representing an atom.
16. charge: Charge of the given atom. Can be +, -, or none at all

Details
This is an accessor function for retrieving the Atomic Record from a Protein object.

Value
Returns a dataframe containing the atomic record. There are 16 variables in this data frame.

getTitleSection(protein)
Arguments

protein input for a a protein object

Details

This is an accessor function for retrieving the title section from a Protein object.

Value

Returns a list containing elements from the title section.

---

p53_tetramerization  P53 Tetramerization Domain Crystal Structure

Description

Formal class protein representing data from a PDB, code 1AIE, p53 tetramerization Domain Crystal Structure. This is a small and simple R object of example data for users to play with and is used in example vignettes.

Usage

p53_tetramerization

Format

A Protein S4 object. List comprised of several sublists and dataframes

- header: List of 2, Header Line and Title
  - header_line: List of 3, Classification, depDate, and idCode
    - classification: Classification of the Protein in the PDB
    - depDat: Date the PDB was deposited or created
    - idCode: 4 digit identifier for the PDB. Always unique.
  - title: The title of the PDB.
- structure: Dataframe of 16 variables
  1. record_type: Type of record in this section. Generally ATOM or HETATM
  2. serial_num: The serial number for the position of the atom in the sequence
  3. atom_name: A name to identify the atom in a structure
  4. alt_location_id:
  5. residue_name: 3 character identifier for a residue
  6. chain_id:
  7. residue_seq_num: Number representing where in the sequence a residue is.
  8. insert_residue_code:
  9. x_ortho_coord: X coordinate in Ångstroms on an orthogonal plane
10. y_ortho Coord: Y coordinate in Ångstroms on an orthogonal plane
11. z_ortho Coord: Z coordinate in Ångstroms on an orthogonal plane
12. occupancy:
13. temp factor: The amount of overall error in the measurement of an atom.
14. segment id:
15. element symbol: Periodic symbol representing an atom.
16. charge: Charge of the given atom. Can be +, -, or none at all

plot3D

Description

plot the protein structure in 3D

Usage

plot3D(
  protein,
  animated = FALSE,
  type = "p",
  groups = NULL,
  screen = list(x = -60, z = 0, y = 0),
  image_width = 480,
  image_height = 480
)

Arguments

protein Protein object to be plotted. Can be either of S3 or S4 Protein object type.
animated logical indicating whether the object is to be animated in the viewer. Will spin the plot on the Z axis.
type character vector indicating the type of cloud plot. Can include one or more of "p", "l", "h", or "b". "p" and "l" mean points and lines respectively, and "b" means both. "h" stands for histogram and draws lines from each point to the XY plane, either lower or upper bounding box face, whichever is closer.
groups the name of a column from the Atomic Record of the protein. Causes the points to be colored by the different values in that group.
screen A list determining the sequence of rotations to be applied to the data before plotting. Each component of the list should be one of "x", "y" or "z", repetitions are allowed with values indicating amount of rotation in degrees.
image_width width of the resulting image in pixels. Currently only applies when ‘animated = TRUE’. Defaults to 480 pixels.
image_height hieght of the resulting image in pixels. Currently only applies when ‘animated = TRUE’. Defaults to 480 pixels.
plotModels

Details

This function uses lattice and magick to create the 3D plot and animate it.
Currently this function is incomplete and will change dramatically as new features and documentation are added.

Value

An object to be plotted. If not assigned to a variable, it will plot directly in the viewer.

plotModels(protein, separate = FALSE)

Arguments

protein Protein object to be plotted
separate indicate whether to plot each plane separately or as one visual.

Details

Create a plot of each plane and model the shape of the protein.
This function uses ggplot and grid to create 3 plots, one for each plane, of the protein model, and then create a smoothing model.
Currently this function is incomplete and will change dramatically as new features and documentation are added.

Value

An object to be plotted. If not assigned to a variable, it will plot directly in the viewer.
Protein Class Definitions

Description

Protein Class used to Define Protein Objects of S3 and S4 Types. Currently still in development, Integrity checks still need to be added.

Format

Breakdown of a Protein Object’s structure:

- header: List of 2, Header Line and Title
  - header_line: List of 3, Classification, depDate, and idCode
    * classification: Classification of the Protein in the PDB
    * depDat: Date the PDB was deposited or created
    * idCode: 4 digit identifier for the PDB. Always unique.
  - title: The title of the PDB.
- structure: Dataframe of 16 variables
  1. record_type: Type of record in this section. Generally ATOM or HETATM
  2. serial_num: The serial number for the position of the atom in the sequence
  3. atom_name: A name to identify the atom in a structure
  4. alt_location_id:
  5. residue_name: 3 character identifier for a residue
  6. chain_id:
  7. residue_seq_num: Number representing where in the sequence a residue is.
  8. insert_residue_code:
  9. x_ortho_coord: X coordinate in Ångstroms on an orthogonal plane
  10. y_ortho_coord: Y coordinate in Ångstroms on an orthogonal plane
  11. z_ortho_coord: Z coordinate in Ångstroms on an orthogonal plane
  12. occupancy:
  13. temp_factor: The amount of overall error in the measurement of an atom.
  14. segment_id:
  15. element_symbol: Periodic symbol representing an atom.
  16. charge: Charge of the given atom. Can be +, -, or none at all
Protein3D

Description
Opens a viewer for exploratory and interactive analysis of a protein structure.

Usage
Protein3D(protein)

Arguments
protein Protein object to use in plotting

Value
Does not return a value.

read.pdb

Description
Read in a Protein Data Bank file

Usage
read.pdb(fileName, createAsS4 = TRUE)

Arguments
fileName character string for location and name of file to be read.
createAsS4 Logical indicating whether to create the new protein object as S4 or not. Defaults to TRUE if not specified. This argument is optional.

Format
A Protein object. List comprised of several sublists and dataframes
- header: List of 2, Header Line and Title
  - header_line: List of 3, Classification, depDate, and idCode
    * classification: Classification of the Protein in the PDB
    * depDat: Date the PDB was deposited or created
    * idCode: 4 digit identifier for the PDB. Always unique.
• title: The title of the PDB.

• structure: Dataframe of 16 variables

  1. record_type: Type of record in this section. Generally ATOM or HETATM
  2. serial_num: The serial number for the position of the atom in the sequence
  3. atom_name: A name to identify the atom in a structure
  4. alt_location_id:
  5. residue_name: 3 character identifier for a residue
  6. chain_id:
  7. residue_seq_num: Number representing where in the sequence a residue is.
  8. insert_residue_code:
  9. x_ortho_coord: X coordinate in Ångstroms on an orthogonal plane
  10. y_ortho_coord: Y coordinate in Ångstroms on an orthogonal plane
  11. z_ortho_coord: Z coordinate in Ångstroms on an orthogonal plane
  12. occupancy:
  13. temp_factor: The amount of overall error in the measurement of an atom.
  14. segment_id:
  15. element_symbol: Periodic symbol representing an atom.
  16. charge: Charge of the given atom. Can be +, -, or none at all

Details

Reads a Protein Data Bank file (PDB) from the given location. The function then parses the file and creates a new object of the Protein class. This object can be either defined as an S3 or S4 object if different capabilities are required.

Value

A new protein object as either an S3 or S4 object.

In general terms, the new object will be a list of two, a data frame containing the atomic record, and a list of header elements.

Description

Function to transform a list of NCBI Virus Report metadata into a table.

Usage

```r
report_as_dataframe(report, records = c(1:length(report)))
```
Arguments

- `report` a list derived a vaccine report from NCBI Datasets.
- `records` a vector of indices to pull from the report.

Value

A large dataframe with 23 variables containing metadata from NCBI Virus report.

### Description

`summary.Protein`

### Usage

```r
## S4 method for signature 'Protein'
summary(object)

## S4 method for signature 'Protein,ANY'
summary(object,...)
```

### Arguments

- `object` A Protein object of either S3 or S4 type.
- `...` other objects passed to `summary()`. Currently not supported.

### Details

Prints a description of the protein object to the console. The lines of output are as follows.

1. Prints if it is S3 or S4 object type.
2. ID Code of the PDB and the Data it was deposited in the Data Bank.
3. The Classification of the protein.
4. The title of the PDB.
5. The number of rows in the Atomic Record.

### Value

Does not return a value.
## Summary

### Protein

**Usage**

```r
## S3 method for class 'Protein'
summary(object, ...)
```

**Arguments**

- `object`: A Protein object of either S3 or S4 type.
- `...`: other objects passed to `summary()`. Currently not supported.

**Details**

Prints a description of the protein object to the console. The lines of output are as follows.

1. Prints if it is S3 or S4 object type.
2. ID Code of the PDB and the Data it was deposited in the Data Bank.
3. The Classification of the protein.
4. The title of the PDB.
5. The number of rows in the Atomic Record.

**Value**

Does not return a value.

---

### write_viz

**Description**

Wrapper function for writing images to the disk. This function comes from the magick package under the same name.

**Usage**

```r
write_viz(image, path = "my_image", format = "png")
```
Arguments

- **image**: magick image object or trellis object.
- **path**: a file path starting from the working directory
- **format**: file type to save the image as. Can be "png", "jpeg", "gif", "rgb", or "rgba".

Value

Does not return a value.
Index

* datasets
  p53_tetramerization, 4

fromJSONL, 2
getAtomicRecord, 2
getTitleSection, 3

p53_tetramerization, 4
plot3D, 5
plotModels, 6
Protein-class, 7
Protein3D, 8

read.pdb, 8
report_as_dataframe, 9

summary, 10
summary, Protein, ANY-method (summary), 10
summary, Protein-method (summary), 10
summary, Protein, 11

write_viz, 11