Protein Feature Extractor

Jesse Thomas, Alyssa Capeheart, Harica BNL

Mar 15, 2021

PROTEIN FEATURE EXTRACTOR:

CHAPTER

ONE

PROTEIN FEATURE EXTRACTOR REPORT

1.1 Introduction

Under construction.

CHAPTER

TWO

DATA HANDLER

2.1 Data Handler Class

class DataHandler(*filename: str*)
I'm used to read csv data in.

Parameters **filename** (*str*) – The filename, including the path to read from, used to pull desired data.

__slots__ = ['__data']
Reserve space for writable attributes and limits addition attribute creation.

__init__(*filename: str*) → None
Constructor Method.

2.2 Data Handler Properties

data = <property object>
I'm used to access the data that was read in.

Returns Dataframe

Return type pandas.DataFrame

2.3 Data Handler Module Global Variables

logger = <CustomLogger.CustomLogger object>
Local logger

CHAPTER

THREE

PROTEIN DATA HANDLER

3.1 Protein Data Handler Class

class ProteinDataHandler (*filename: str*)
I'm used to read in csv data pertaining to protein data.

Inherits from *Data Handler*

3.2 Protein Data Handler Public Methods

covert_to_count_vector (*attributes_values: numpy.ndarray*) → *Handlers.Processor.Processor*
I convert all sequences to a count vector based on attributes given.

Parameters **attributes_values** (*numpy.ndarray*) – List of attributes to use to convert file to.

Returns A processor object containing processed data.

Return type *Processor.Processor*

3.3 Protein Data Handler Module Global Variables

logger = <CustomLogger.CustomLogger object>
Local logger

3.4 Protein Data Handler Module Global Functions

count_sequence (*sequence: str*) → dict
I count the number of times a character is seen in a sequence.

Parameters **sequence** (*str*) – Protein sequence.

Returns Dictionary with the keys being a character found in the given sequence and the value being the number of time it was seen.

Return type dict

CHAPTER

FOUR

PROTEIN ATTRIBUTE DATA HANDLER

4.1 Protein Attribute Data Handler Class

class ProteinAttributeDataHandler (*filename: str*)
I'm used to read in csv data pertaining to protein attribute data.

Inherits from *Data Handler*.

4.2 Protein Attribute Data Handler Public Methods

get_attribute_headers () → numpy.ndarray
I get the attribute headers from the given data.

Returns An array of attributes.

Return type numpy.ndarry

get_attribute_values (*value: int*) → tuple
I get the attribute name and list of associated values associated with the given value - 1.

Parameters **value** (*int*) – Row number associated with desired attribute.

Returns Attribute name and attributes

Return type tuple

4.3 Protein Attribute Data Handler Module Global Variables

logger = <CustomLogger.CustomLogger object>
Local logger

CHAPTER

FIVE

PROCESSOR

5.1 Processor Class

class Processor (*file: pandas.core.frame.DataFrame*)
I'm used to process a given dataframe.

Parameters **file** (*pandas.DataFrame*) – Dataframe to work on.

__slots__ = ['__data']
Reserve space for writable attributes and limits addition attribute creation.

__init__ (*file: pandas.core.frame.DataFrame*) → None
Constructor Method.

5.2 Processor Properties

data = <property object>
I'm used to access the data that was read in.

Returns Dataframe

Return type pandas.DataFrame

5.3 Processor Public Methods

apply_attribute (*attribute_name: str*, *attribute: pandas.core.series.Series*) → None
I apply a given attribute series to the data currently held, sum the result, and adds it as a new column to the data.

Parameters

- **attribute_name** (*str*) – Name of the attribute being applied.
- **attribute** (*pandas.Series*) – Series of attribute to be applied.

Returns Nothing

Return type NoneType

normalize_via_length () → None
I normalize the data held by dividing each attribute by the length of the sequence.

Returns Nothing

Return type NoneType

save_processed_data(*path: str*) → None
I save the data to a given path name.

Parameters **path** (*str*) – The path name to save the data held.

Returns Nothing

Return type NoneType

5.4 Processor Private Methods

__get_attribute_header() → numpy.ndarray
I get the header from the held data associated with the attributes.

Returns Nothing

Return type numpy.ndarray

5.5 Processor Module Global Variables

any_uint
Typing that's a union of all the unsigned integers

alias of Union[numpy.uint8, numpy.uint16, numpy.uint32, numpy.uint64]

logger = <CustomLogger.CustomLogger object>
Local logger

CHAPTER

SIX

CUSTOM LOGGER

6.1 Custom Logger Class

class CustomLogger (*filename: str*, *level: numpy.uint8 = 4*)
I am used to log information to specific files associated with the module that called me.

Parameters

- **filename** (*str*) – The name of the module or file used for this logger.
- **level** (*numpy.uint8*) – The level of logging wanted

Note:

- 0: NOTSET - Doesn't log anything.
- 1: CRITICAL - logs only critical log calls.
- 2: ERROR - logs error log calls and everything before it.
- 3: INFO - logs info log calls and everything before it.
- 4: DEBUG - logs debug log calls and everything before it.

__slots__ = ['__level', '__logger']
Reserve space for writable attributes and limits addition attribute creation.

__init__ (*filename: str*, *level: numpy.uint8 = 4*) → None
Constructor Method.

6.2 Custom Logger Properties

level = <property object>
I'm used to access the current logging level.

Returns Unsigned 8 bit Integer.

Return type numpy.uint8

logger = <property object>
I'm used to access the logger object.

Returns Logger object.

Return type logging.Logger

6.3 Custom Logger Methods

flow(*message: str*) → None
I'm used to capture the flow of your application. Eg. 'Starting connection'.

Parameters **message** (*str*) – The string wanting to be logged.

Returns Nothing

Return type NoneType

sanity_check(*message: str*) → None
I'm used to capture debugging information from your application. Eg. the current value of a specific variable at a specific point in the application state.

Parameters **message** (*str*) – The string wanting to be logged.

Returns Nothing

Return type NoneType

CHAPTER

SEVEN

INDICES AND TABLES

- genindex
- modindex
- search

INDEX

Symbols

A

C

D

F

G

L

N

P

S