

Package: mop (via r-universe)

November 6, 2024

Title A 'broom'-like Interface to Lab Data

Version 0.0.0.9000

Description What the package does (one paragraph).

License MIT + file LICENSE

Encoding UTF-8

Language en

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Imports crayon, dplyr, fs, gplate, janitor, lubridate, purrr, readr,
readxl, rlang, snakecase, stringr, tibble, tidyr, utils, vctrs

Remotes KaiAragaki/gplate, KaiAragaki/amplify

URL <https://kaiaragaki.github.io/mop/>

Config/pak/sysreqs make libicu-dev libx11-dev

Repository <https://kaiaragaki.r-universe.dev>

RemoteUrl <https://github.com/KaiAragaki/mop>

RemoteRef HEAD

RemoteSha 131a0764d479f3671e332f3c24c132d615dee752

Contents

as_nanodrop	2
as_pcr	3
incucyte	3
is_incucyte	5
is_nanodrop	5
nanodrop	6
new_incucyte	7
new_nanodrop	8
new_pcr	9

new_spectramax	10
pad_zero	10
plate_data	11
read_incucyte	12
read_nanodrop	12
read_pcr	13
read_spectramax	14
scrub	14
scrub.nanodrop	15
scrub.pcr	15
scrub.spectramax	16
tidy_lab.incucyte_data	16
tidy_lab.nanodrop	17
tidy_lab.pcr	18
tidy_lab.spectramax	19

Index	20
--------------	-----------

as_nanodrop	<i>Coerce object to nanodrop</i>
-------------	----------------------------------

Description

Coerce object to nanodrop

Usage

```
as_nanodrop(x, ...)
```

```
## S3 method for class 'data.frame'
```

```
as_nanodrop(x, nucleotide = NULL, is_tidy = FALSE, date = NULL, ...)
```

Arguments

x	a data.frame
...	additional arguments passed to their respective functions
nucleotide	an optional character vector describing the measured nucleotide (DNA, RNA, dsDNA, dsRNA)
is_tidy	Logical. Is data already tidy?
date	Character coercible to a Date object via lubridate::as_date

Value

a nanodrop object

as_pcr	<i>Convert a scrubbed pcr object back to a pcr object</i>
--------	---

Description

Convert a scrubbed pcr object back to a pcr object

Usage

```
as_pcr(x)
```

Arguments

x	A tibble - usually a previously scrubbed pcr object.
---	--

Value

A pcr object

incucyte	<i>Make a nanodrop object</i>
----------	-------------------------------

Description

Make a nanodrop object

Usage

```
incucyte(data, platemap)
```

Arguments

data	a <code>data.frame</code> - or something coercible to one - containing nanodrop data. See details for requirements.
date	an optional character vector or something coercible to a Date object with <code>lubridate::as_date</code>
nucleotide	a character vector to specifying the analyte in the nanodrop experiment. Must match one of the following: RNA, DNA, dsDNA, or dsRNA.
is_tidy	logical. Is the supplied data tidy?

Details

Making a nanodrop object 'by hand' (that is, not using `read_nanodrop`) is not recommended, since it is challenging to ensure a given `data.frame` is truly a valid nanodrop file. To increase reliability, flexibility is reduced. As such, the supplied `data.frame` must *at least* have the following columns *if* `is_tidy = TRUE`:

- `date`
- `sample_name`
- `conc`
- `a260_280`
- `a230_280`

If `is_tidy = FALSE`, *at least* the following columns must be provided:

- `Date`
- `Sample.Name`
- `Nucleic.Acid.ng.uL.`
- `A260.A280`
- `A260.A230`

Note: technically, the given column names will be stripped of all non-alphanumerics and forced to lower, then compared against the following:

- `date`
- `samplename`
- `nucleicacidngul`
- `a260a280`
- `a260a230`

In both cases, other columns allowed are those that appear in Example A and Example B below.

If there are additional columns provided, they will be silently dropped.

Value

an `incucyte` object

is_incucyte *Test if an object is a incucyte object*

Description

Test if an object is a incucyte object

Usage

is_incucyte(incucyte)

Arguments

incucyte Object to be tested

Value

logical

is_nanodrop *Test if an object is a nanodrop object*

Description

Test if an object is a nanodrop object

Usage

is_nanodrop(nanodrop)

Arguments

nanodrop Object to be tested

Value

logical

nanodrop *Make a nanodrop object*

Description

Make a nanodrop object

Usage

```
nanodrop(
  data,
  date = NULL,
  nucleotide = c("RNA", "DNA", "dsDNA", "dsRNA"),
  is_tidy = FALSE
)
```

Arguments

data	a <code>data.frame</code> - or something coercible to one - containing nanodrop data. See details for requirements.
date	an optional character vector or something coercible to a Date object with <code>lubridate::as_date</code>
nucleotide	a character vector to specifying the analyte in the nanodrop experiment. Must match one of the following: RNA, DNA, dsDNA, or dsRNA.
is_tidy	logical. Is the supplied data tidy?

Details

Making a nanodrop object 'by hand' (that is, not using `read_nanodrop`) is not recommended, since it is challenging to ensure a given `data.frame` is truly a valid nanodrop file. To increase reliability, flexibility is reduced. As such, the supplied `data.frame` must *at least* have the following columns *if* `is_tidy = TRUE`:

- date
- sample_name
- conc
- a260_280
- a230_280

If `is_tidy = FALSE`, *at least* the following columns must be provided:

- Date
- Sample.Name
- Nucleic.Acid.ng.uL.
- A260.A280
- A260.A230

Note: technically, the given column names will be stripped of all non-alphanumerics and forced to lower, then compared against the following:

- date
- samplename
- nucleicacidngul
- a260a280
- a260a230

In both cases, other columns allowed are those that appear in Example A and Example B below.

If there are additional columns provided, they will be silently dropped.

Value

a nanodrop object

Examples

```
# Example A: colnames allowed when is_tidy = FALSE
a <- system.file("extdata", "nanodrop.csv", package = "mop") |>
read_nanodrop()

colnames(a$data)

# Technically, these are the names that are checked for after the given names
# have alphanumerics removed and are converted to lowercase:

colnames(a$data) |> stringr::str_remove_all("[^[:alnum:]]") |> tolower()

# Example B: colnames allowed when is_tidy = TRUE
b <- a |> tidy_lab()

colnames(b$data)
```

new_incucyte

Constructor for a Incucyte object

Description

Constructor for a Incucyte object

Usage

```
new_incucyte(
  data = data.frame(),
  raw_data = data.frame(),
  date = lubridate::Date(),
  nucleotide = character(),
  is_tidy = logical()
)
```

Arguments

data	data.frame, no restrictions on form.
raw_data	data.frame, no restrictions on form. Typically represents read-in data with no changes to it.
date	lubridate::Date object
is_tidy	logical. Is the data.frame provided tidy?

Value

a nanodrop object

new_nanodrop	<i>Constructor for a nanodrop object</i>
--------------	--

Description

As this is a low-level constructor, it will let you do some un-recommended things without error. You have been warned.

Usage

```
new_nanodrop(
  data = data.frame(),
  raw_data = data.frame(),
  date = lubridate::Date(),
  nucleotide = character(),
  is_tidy = logical()
)
```

Arguments

data	data.frame, no restrictions on form.
raw_data	data.frame, no restrictions on form. Typically represents read-in data with no changes to it.
date	lubridate::Date object
nucleotide	character. Should likely be DNA, RNA, dsDNA, or even dsRNA.
is_tidy	logical. Is the data.frame provided tidy?

Value

a nanodrop object

new_pcr	<i>Constructor for a pcr object</i>
---------	-------------------------------------

Description

Constructor for a pcr object

Usage

```
new_pcr(
  data = data.frame(),
  raw_data = raw(),
  header = character(),
  footer = character(),
  date = lubridate::Date(),
  experiment_type = character(),
  wells = integer(),
  is_tidy = logical()
)
```

Arguments

data	data.frame or gp, no restrictions on form.
raw_data	raw, read in from the provided file with readr::read_file_raw()
header	data.frame, the upper portion of the data before the body of the data. Usually contains run information.
footer	data.frame, the lower portion of the data after the body of the data. Usually contains analysis information. May not exist.
date	lubridate::Date object
experiment_type	character. Could
wells	Number of wells in plate
is_tidy	logical. Is the data.frame provided tidy?

Value

A pcr object

new_spectramax	<i>Constructor for a spectramax object</i>
----------------	--

Description

Constructor for a spectramax object

Usage

```
new_spectramax(  
  data = data.frame(),  
  raw_data = raw(),  
  date = lubridate::Date(),  
  wavelengths = numeric(),  
  is_tidy = logical()  
)
```

Arguments

data	a data.frame, no restrictions on form.
raw_data	raw, read in from the provided file with readr::read_file_raw()
date	lubridate::Date object
is_tidy	logical. Are the data provided tidy?
experiment_type	character, no restrictions on form

Value

a spectramax object

pad_zero	<i>Add leading 0 to sample names</i>
----------	--------------------------------------

Description

Add leading 0 to sample names

Usage

```
pad_zero(sample_names)
```

Arguments

sample_names	a vector of sample names
--------------	--------------------------

Value

A sample name with up to one zero padded

Examples

```
c("blueberry", "Sample 1", "Sample 10", "sample 2", "john", "larry", "toxic waste", "Sample 02") |>
  pad_zero()
```

```
# pad_zero will not pad if it doesn't need to:
c("Sample 1", "Sample 2", "Sample 9") |> pad_zero()
```

```
# pad_zero is case sensitive, since these will be coming off a machine in a standardized format
```

```
c("sample 10", "sample 1") |> pad_zero()
c("Sample 10", "Sample 1") |> pad_zero()
```

 plate_data

Get or set an object's plate data

Description

Get or set an object's plate data

Usage

```
plate_data(x, ...)
```

```
## S3 method for class 'spectramax'
plate_data(x, ...)
```

```
plate_data(x) <- value
```

```
## S3 replacement method for class 'spectramax'
plate_data(x) <- value
```

Arguments

x	Object to extract plate data from
...	Arguments to be passed to their respective methods
value	Value to set the plate to.

Value

A gp object

read_incucyte	<i>Read an Incucyte File</i>
---------------	------------------------------

Description

Read an Incucyte File

Usage

```
read_incucyte(data, platemap)
```

Arguments

data	Plotting data output from the Incucyte analysis
platemap	.PlateMap file output from the platemap maker

read_nanodrop	<i>Read in a NanoDrop file</i>
---------------	--------------------------------

Description

Read in a NanoDrop file

Usage

```
read_nanodrop(path, nucleotide = NULL, date = NULL, date_order = NULL)
```

Arguments

path	path to a NanoDrop .csv
nucleotide	An optional character vector describing the analyte detected. If NULL (the default), it will automatically try to extract one from the file path. Valid values include RNA, DNA, dsDNA, and dsRNA (or NULL).
date	An optional character vector coercible to a date object. If NULL (the default), it will automatically try to extract one from the file path. See details for detection methods.
date_order	An optional character vector to enforce an order of date to be read (if date is unspecified). Can be "ymd", "mdy", or "dmy"

Details

This function expects a file that has not been tampered with from the nanodrop machine itself, aside for the name.

This function calls `read.delim` and guesses its encoding (usually UTF-16LE, unless it's been re-exported). It's less mysterious than reading in an apparent `.csv` with `read.delim`, and it keeps you from having to remember the encoding (which R often fails to auto-detect). When possible, it attempts to extract the nucleotide type from the path provided.

if `date` is `NULL`, the function will attempt to extract a date from the file name (not the whole file path). It will look for `ymd`, `mdy`, and `dmy` (in order). If it detects the first format, it will not look for the second, etc. Note how a reading taken from April 4th but written as `4_3_2021` will be misinterpreted. This error can be remedied by specifying the argument `date_order`

Value

a nanodrop object

Examples

```
system.file("extdata", "nanodrop.csv", package = "mop") |>
  read_nanodrop()
```

read_pcr	<i>Read in a QuantStudio pcr file</i>
----------	---------------------------------------

Description

Read in a QuantStudio pcr file

Usage

```
read_pcr(path)
```

Arguments

path path to a QuantStudio .xls

Value

a pcr object

read_spectramax *Read and tidy a SPECTRAMax file*

Description

Read and tidy a SPECTRAMax file

Usage

```
read_spectramax(path, date = Sys.Date(), wavelengths = NULL)
```

Arguments

path Path to the SPECTRAMax .txt file
date Date of experiment. If not supplied, uses current date.
wavelengths What wavelengths were read in this experiment?

Details

SPECTRAMax files cannot be read in easily without tidying them simultaneously, due to their non-rectangular structure. Therefore, tidying is *required* to be read in and is not an option supplied.

Value

A list, with the following form:

data a tibble that contains two columns - the type, and the (tidy) data itself in a list-column.

raw_data a raw representation of the file, before tidying

date character. The date of the experiment

experiment_type character. The type of experiment (currently only pq (Protein Quantification) and mtt supported)

tidy logical. Should always return TRUE if read in using read_spectramax

scrub *Convert a lab object to a tidy tibble*

Description

Objects are a convenient way to represent lab data as it allows downstream functions to interact with them in specific ways depending on their type. Sometimes, however, it is more convenient to work with a flat tibble. scrub converts a lab object into a tibble

Usage

```
scrub(x, ...)
```

Arguments

x object to be converted into a tibble
 ... additional arguments to pass on to the respective object's class method

Value

a tibble

scrub.nanodrop *Convert a nanodrop object into a tibble*

Description

Convert a nanodrop object into a tibble

Usage

```
## S3 method for class 'nanodrop'
scrub(x, raw = FALSE, ...)
```

Arguments

x a nanodrop object
 raw Should the data be pulled from x\$raw_data?
 ... Unused

Value

a tibble

scrub.pcr *Convert a pcr object into a tibble*

Description

Convert a pcr object into a tibble

Usage

```
## S3 method for class 'pcr'
scrub(x, include_header = FALSE, ...)
```

Arguments

x a pcr object
 include_header Should the data from x\$header be included?
 ... Unused

Value

a tibble

scrub.spectramax *Convert a spectramax object into a tibble*

Description

Convert a spectramax object into a tibble

Usage

```
## S3 method for class 'spectramax'
scrub(x, n = 1, ...)
```

Arguments

x a spectramax object
 n List item number to select data from
 ... Unused

Value

a tibble

tidy_lab.incucyte_data
Tidy an incucyte object

Description

Tidy an incucyte object

Usage

```
## S3 method for class 'incucyte_data'
tidy_lab(x, force_tidy = FALSE, ...)
```

Arguments

x	An incucyte object to tidy
force_tidy	logical. Should the tidying take place, even if the is_tidy attribute is TRUE?
...	Unused

Details

This function:

Value

an incucyte object

tidy_lab.nanodrop	<i>Tidy a nanodrop object</i>
-------------------	-------------------------------

Description

Tidy a nanodrop object

Usage

```
## S3 method for class 'nanodrop'
tidy_lab(x, force_tidy = FALSE, ...)
```

Arguments

x	A nanodrop object to tidy
force_tidy	logical. Should the tidying take place, even if the is_tidy attribute is TRUE?
...	Unused

Details

This function:

- renames columns to sensible substitutes
- converts date-times to ISO 8601-esque date-time format (YYYY-MM-DD HH:MM:SS vs YYYY-MM-DDTHH:MM:SSZ)

It's recommended that you do any manipulations to these data after you tidy, rather than before, as tidy_nanodrop expects the output to be fairly similar to the output from read_nanodrop.

A tidy nanodrop object will (usually) contain the following columns:

- date The date-time of sample reading, as YYYY-MM-DD HH:MM:SS
- sample_name The name of the sample provided by the NanoDrop itself
- conc The concentration of nucleic acid in ng/uL

- `a260_280` The absorbance at 260nm / absorbance at 280nm. Typically a marker of protein contamination. Pure nucleic acid is typically around 2.
- `a260_230` The absorbance at 260nm / absorbance at 230nm. Typically a marker of guanadine salt contamination. Pure nucleic acid is typically around 2.
- `a260` The absorbance at 260nm, the wavelength nucleic acids absorb most strongly.
- `a280` The absorbance at 280nm, the wavelength proteins absorb most strongly.

The remaining columns are typically unused.

Value

a nanodrop object

tidy_lab.pcr	<i>Tidy a PCR object</i>
--------------	--------------------------

Description

Tidy a PCR object

Usage

```
## S3 method for class 'pcr'
tidy_lab(x, usr_standards = NULL, pad_zero = FALSE, ...)
```

Arguments

<code>x</code>	a pcr object
<code>usr_standards</code>	Custom supplied standards
<code>pad_zero</code>	Should, say, Sample 1 become Sample 01?
<code>...</code>	Unused

Value

a pcr object

tidy_lab.spectramax *Tidy a lab object*

Description

Tidy a lab object

Usage

```
## S3 method for class 'spectramax'  
tidy_lab(x, ...)  
  
tidy_lab(x, ...)
```

Arguments

x	An object to tidy
...	Arguments passed to their respective methods

Index

as_nanodrop, 2
as_pcr, 3

incucyte, 3
is_incucyte, 5
is_nanodrop, 5

nanodrop, 6
new_incucyte, 7
new_nanodrop, 8
new_pcr, 9
new_spectramax, 10

pad_zero, 10
plate_data, 11
plate_data<- (plate_data), 11

read_incucyte, 12
read_nanodrop, 12
read_pcr, 13
read_spectramax, 14

scrub, 14
scrub.nanodrop, 15
scrub.pcr, 15
scrub.spectramax, 16

tidy_lab (tidy_lab.spectramax), 19
tidy_lab.incucyte_data, 16
tidy_lab.nanodrop, 17
tidy_lab.pcr, 18
tidy_lab.spectramax, 19