Package ‘seahtrue’

July 24, 2025

Type Package

Title Seahtrue revives XF data for structured data analysis

Version 1.3.0

Description Seahtrue organizes oxygen consumption and extracellular acidification analysis data from experiments performed on an XF analyzer into structured nested tibbles.This allows for detailed processing of raw data and advanced data visualization and statistics. Seahtrue introduces an open and reproducible way to analyze these XF experiments. It uses file paths to .xlsx files. These .xlsx files are supplied by the userand are generated by the user in the Wave software from Agilent from the assay result files (.asyr). The .xlsx file contains different sheets of important data for the experiment;
1. Assay Information - Details about how the experiment was set up.
2. Rate Data - Information about the OCR and ECAR rates.
3. Raw Data - The original raw data collected during the experiment.
4. Calibration Data - Data related to calibrating the instrument.
Seahtrue focuses on getting the specific data needed for analysis. Once this data is extracted, it is prepared for calculations through preprocessing. To make sure everything is accurate, both the initial data and the preprocessed data go through thorough checks.

biocViews CellBasedAssays, FunctionalPrediction, DataRepresentation, DataImport, CellBiology, Cheminformatics, Metabolomics, MicrotitrePlateAssay, Visualization, QualityControl, BatchEffect, ExperimentalDesign, Preprocessing, GO

Encoding UTF-8

LazyData false

Suggests rmarkdown, knitr, testthat (>= 3.0.0), BiocStyle

Imports dplyr (>= 1.1.2), readxl (>= 1.4.1), logger (>= 0.2.2), tidyxl (>= 1.0.8), purrr (>= 0.3.5), tidyr (>= 1.3.0), lubridate (>= 1.8.0), stringr (>= 1.4.1), tibble (>= 3.1.8), validate (>= 1.1.1), rlang (>= 1.0.0), glue (>= 1.6.2), cli (>= 3.4.1), janitor (>= 2.2.0), ggplot2 (>= 3.5.0), RColorBrewer (>= 1.1.3), colorspace (>= 2.1.0), forcats (>= 1.0.0), ggridges (>= 0.5.6), readr (>= 2.1.5), scales (>= 1.3.0)

RoxygenNote 7.3.1

Depends R (>= 4.2.0)

VignetteBuilder knitr

URL https://vcjdeboer.github.io/seahtrue/

BugReports https://vcjdeboer.github.io/seahtrue/issues

git_url https://git.bioconductor.org/packages/seahtrue

git_branch devel

git_last_commit ab17164

git_last_commit_date 2025-04-15

Repository Bioconductor 3.22

Date/Publication 2025-07-23

Author Vincent de Boer [cre, aut] (ORCID:
<https://orcid.org/0000-0001-9928-1698>),
Gerwin Smits [aut],
Xiang Zhang [aut]

Maintainer Vincent de Boer <vincent.deboer@wur.nl>

Contents

`seahtrue-package` *seahtrue: Seahtrue revives XF data for structured data analysis*

Description

Seahtrue organizes oxygen consumption and extracellular acidification analysis data from experiments performed on an XF analyzer into structured nested tibbles.This allows for detailed processing of raw data and advanced data visualization and statistics. Seahtrue introduces an open and reproducible way to analyze these XF experiments. It uses file paths to .xlsx files. These .xlsx files are supplied by the userand are generated by the user in the Wave software from Agilent from the assay result files (.asyr). The .xlsx file contains different sheets of important data for the experiment; 1. Assay Information - Details about how the experiment was set up. 2. Rate Data - Information about the OCR and ECAR rates. 3. Raw Data - The original raw data collected during the experiment. 4. Calibration Data - Data related to calibrating the instrument. Seahtrue focuses on getting the specific data needed for analysis. Once this data is extracted, it is prepared for calculations through preprocessing. To make sure everything is accurate, both the initial data and the preprocessed data go through thorough checks.

Author(s)

Maintainer: Vincent de Boer `<vincent.deboer@wur.nl>` (ORCID)

Authors:

- Gerwin Smits
- Xiang Zhang

See Also

Useful links:

- `https://vcjdeboer.github.io/seahtrue/`
- Report bugs at `https://vcjdeboer.github.io/seahtrue/issues`

`glue_xfplates` *Glueing mulltiple plates from a folder*

Description

This function takes a folder path and on the available .xlsx files the revive_xfplate() function is run and output in one nested tibble.

Usage

```
glue_xfplates(folderpath_seahorse, arg_is_folder)
```

Arguments

`folderpath_seahorse`	the path to a folder where the .xlsx files are located or a vector of strings pointing to the path of each individual file
`arg_is_folder`	either TRUE or FALSE. When the input is a vector of path strings use FALSE, is it points to a folder use TRUE

Value

a nested tibble with all files organized in a row

Examples

```
c(
    system.file("extdata",
        "20191219_SciRep_PBMCs_donor_A.xlsx",
        package = "seahtrue"
    ),
    system.file("extdata",
        "20191219_SciRep_PBMCs_donor_A.xlsx",
        package = "seahtrue"
    )
) |>
    glue_xfplates(arg_is_folder = FALSE)
```

`revive_output_donor_A` *Dataset PBMC donor A*

Description

A dataset containing output for the "20191219_SciRep_PBMCs_donor_A.xlsx" file. This experiment was part of the study that was published in Janssen et al. (2021) Sci rep 11:1662. This is data from PBMCs that were isolated from buffy coats as described in the material and methods section of Janssen et al. (2021). The .xlsx file is the result of a single XF experiment on one plate. This .xlsx file was used as input for the revive_xfplate() function. The output of that function contains the read, preprocessed, and validated data in the form a a nested tibble. The file '20191219_SciRep_PBMCs_donor_A.xlsx' can be found in the inst/extdata directory.

Usage

```
data(revive_output_donor_A)
```

Format

A data frame with 1 row and 7 variables.

plate_id	Barcode plate id of the well plate containing the samples
filepath_seahorse	Path, and basename to .xlsx file
date_run	Date and time when the plate was run
date_processed	Date and time this output from revive_xfplate() was generated
assay_info	Meta information from 'Assay Configuration' sheet and 'Calibration' sheet
injection_info	Dataframe with information from the 'Operation log' sheet
raw_data	Preprocessed raw dataframe from 'Raw' sheet
rate_data	Preprocessed rate data from 'Rate' sheet
validation_output	A list of all validation information, output, and rules that are used

Source

Janssen et al. 2021 Sci Rep 11:1162 <https://doi.org/10.1038/s41598-021-81217-4>

`revive_xfplate` *Running the read, preprocess and validate*

Description

This function takes the Seahorse Wave .xlsx file and computes it through read, validate and preprocess

Usage

```
revive_xfplate(filepath_seahorse)
```

Arguments

`filepath_seahorse`	Absolute path to the Seahorse Excel file.

Value

A preprocessed seahorse dataset is returned as an output. This is a nested tibble with the following 7 columns:
* plate_id = Barcode plate id of the well plate containing the samples
* filepath_seahorse = Path, and basename to .xlsx input file
* date_run = Date and time when the plate was run
* date_processed = Date and time this output from revive_xfplate() was generated
* assay_info = Meta information from 'Assay Configuration' sheet and 'Calibration' sheet
* injection_info = Dataframe with information from the 'Operation log' sheet
* raw_data = Preprocessed raw dataframe from 'Raw' sheet
* rate_data = Preprocessed rate data from 'Rate' sheet
# validation_output = Output of the data checks. including rules

Examples

```
revive_xfplate(
    system.file("extdata",
        "20191219_SciRep_PBMCs_donor_A.xlsx",
        package = "seahtrue"
    )
)
```

`sketch_assimilate_rate` *Combine multiple revived xf plates into one plot for rate data*

Description

In this plot the OCR or ECAR is plotted per group for each plate in a faceted gpgplot

Usage

```
sketch_assimilate_rate(my_df, param = "OCR", my_measurements = c(3, 6, 7, 12))
```

Arguments

`my_df`	a tibble generated by glue_xfplates() with for each row representing a single xf experiment
`param`	either "OCR" or "ECAR"
`my_measurements`	the measurements that needs to be in the plot. For example, c(3,6,7,12) for a typical mito stress test.

Value

a ggplot object

Examples

```
suppressMessages(
    c(
        system.file("extdata",
            "20191219_SciRep_PBMCs_donor_A.xlsx",
            package = "seahtrue"
        ),
        system.file("extdata",
            "20191219_SciRep_PBMCs_donor_A.xlsx",
            package = "seahtrue"
        )
    ) |>
        glue_xfplates(arg_is_folder = FALSE) |>
        sketch_assimilate_rate(
            param = "OCR",
            my_measurements = c(3, 4, 9, 12)
        )
)
```

`sketch_assimilate_raw` *Combine multiple revived xf plates into one plot for raw data*

Description

In this plot the O2, pH, or its emission value at the very first measurement point plotted for all wells from all xfplates that are provided to the function.

Usage

```
sketch_assimilate_raw(my_df, param = "O2_mmHg")
```

Arguments

`my_df`	a tibble generated by glue_xfplates() with for each row representing a single xf experiment
`param`	either "O2_mmHg", "pH", "O2_em_corr" or "pH_em_corr

Value

a ggplot object

Examples

```
suppressMessages(
    c(
        system.file("extdata",
            "20191219_SciRep_PBMCs_donor_A.xlsx",
            package = "seahtrue"
        ),
        system.file("extdata",
            "20191219_SciRep_PBMCs_donor_A.xlsx",
            package = "seahtrue"
        )
```

```
    ) |>
        glue_xfplates(arg_is_folder = FALSE) |>
        sketch_assimilate_raw(param = "O2_mmHg")
)
```

`sketch_plate` *Make an overview of plate layout, with colored groups*

Description

A heatmap style ggplot figure with each well labeled with a color for each group

Usage

```
sketch_plate(xfplate, reorder_legend = FALSE)
```

Arguments

`xfplate`	This the ‘raw_data‘ or the ‘rate_data‘ tibble that is generated by the ‘revive_xfplate()‘ function
`reorder_legend`	either ‘TRUE‘ or ‘FALSE‘. When ‘TRUE‘ the groups are ordered based on the number in the character string of the group. It also adds a "__00" after each character string to make the forcats::refactor(group, parse_number(group)) work.

Value

a ggplot object of a 96 well plate with the group layout

Examples

```
system.file("extdata",
    "20191219_SciRep_PBMCs_donor_A.xlsx",
    package = "seahtrue"
) |>
    revive_xfplate() |>
    purrr::pluck("raw_data", 1) |>
    sketch_plate(reorder_legend = TRUE)
```

`sketch_rate` *Generate a plot for the rate data*

Description

The sketch_rate() function uses the rate_data from the generated output from the revive_xfplate() function. The injection info is annotated in the plot, using the information form the injections provided in the original experiment. Several options are available to plot either ECAR/OCR or normalize the data with the values from the normalization cells in the .xlsx file.

A number of validations are performed to check whether the data can be plotted and whether the layout of the plot will not be ruined...

Usage

```
sketch_rate(
  xf_rate,
  param = "OCR",
  normalize = FALSE,
  normalize_unit = "10000 cells",
  take_group_mean = TRUE,
  reorder_legend = FALSE
)
```

Arguments

`xf_rate`	The ‘rate_data‘ tibble as generated by ‘revive_plate‘
`param`	Either "OCR" or "ECAR"
`normalize`	Either TRUE or FALSE
`normalize_unit`	any string that will be pasted in the y-axis label when normalize = TRUE
`take_group_mean`	Either TRUE or FALSE
`reorder_legend`	Either TRUE or FALSE. When ‘TRUE‘ the groups are ordered based on the number in the character string of the group. It also adds a "__00" after each character string to make the forcats::refactor(group, parse_number(group)) work.

Value

a ggplot object

Examples

```
system.file("extdata",
    "20191219_SciRep_PBMCs_donor_A.xlsx",
    package = "seahtrue"
) |>
    revive_xfplate() |>
    purrr::pluck("rate_data", 1) |>
    sketch_rate(
        param = "OCR",
        reorder_legend = TRUE
    )

system.file("extdata",
    "20191219_SciRep_PBMCs_donor_A.xlsx",
    package = "seahtrue"
) |>
    revive_xfplate() |>
    purrr::pluck("rate_data", 1) |>
    sketch_rate(
        param = "OCR",
        take_group_mean = FALSE,
        reorder_legend = TRUE
    )

system.file("extdata",
    "20191219_SciRep_PBMCs_donor_A.xlsx",
```

```
        package = "seahtrue"
    ) |>
        revive_xfplate() |>
        purrr::pluck("rate_data", 1) |>
        sketch_rate(
            param = "ECAR",
            normalize = TRUE,
            take_group_mean = TRUE,
            reorder_legend = TRUE
        )

    system.file("extdata",
        "20191219_SciRep_PBMCs_donor_A.xlsx",
        package = "seahtrue"
    ) |>
        revive_xfplate() |>
        purrr::pluck("rate_data", 1) |>
        sketch_rate(
            param = "ECAR",
            normalize = TRUE,
            take_group_mean = FALSE,
            reorder_legend = TRUE
        )
```

Index