Package: amplify (via r-universe)

November 5, 2024

Title Automate PCR Tasks Reproducibly

Version 0.1.0

Description PCR tasks - like plate layout planning, dilution calculations, visualization, and analysis - are often repetitive, tedious, prone to error, and poorly documented. amplify seeks to automate these tasks, as well as documenting them (through both code and generated reports) as a bonus.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Imports dplyr (>= 1.1.0), forcats, ggplot2, gplate, gt, knitr, mop, purrr, readxl, rlang, rmarkdown, stats, stringr, tidyr, utils, vctrs

Remotes KaiAragaki/gplate, KaiAragaki/mop

Depends R (>= 4.1.0)

VignetteBuilder knitr

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

URL https://kaiaragaki.github.io/amplify/

Config/pak/sysreqs make libicu-dev libxml2-dev libssl-dev libnode-dev libx11-dev

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

RemoteUrl https://github.com/KaiAragaki/amplify

RemoteRef HEAD

RemoteSha d3bb78428ff1b437652b2289d2402a063840787b

Contents

`dummy_rna_conc` *Example data of RNA samples with concentrations*

Description

A dataset containing fabricated sample names and RNA concentrations

Usage

```
dummy_rna_conc
```

Format

A data frame with 8 rows and 2 columns

sample name of sample

conc concentration of RNA, in ng/uL

pcr_calc_slope *Recalcuate standard slope of quantity vs Ct*

Description

Recalcuate standard slope of quantity vs Ct

Usage

```
pcr_calc_slope(tidy_pcr)
```

Arguments

tidy_pcr	a object that has been tidied by tidy_pcr

Value

a tibble with an updated slope column

pcr_control *Calculate Delta Ct mean based on given control probe*

Description

Calculate Delta Ct mean based on given control probe

Usage

```
pcr_control(x, control_probe)

## S3 method for class 'pcr'
pcr_control(x, control_probe, ...)

## S3 method for class 'data.frame'
pcr_control(x, control_probe, ...)
```

Arguments

x	A pcr or data.frame object
control_probe	A probe to be used as an endogenous control (eg GAPDH)

Value

An object with class the same as input

Examples

```
system.file("extdata", "untidy-pcr-example.xls", package = "amplify") |>
  read_pcr() |>
  pcr_control("GAPDH")
```

`pcr_lib_calc` *Calculate library PCR concentrations*

Description

Calculate library PCR concentrations

Usage

```
pcr_lib_calc(pcr, dil_factor = 1000)
```

Arguments

`pcr`	a `pcr` object. Will be tidied if not already.
`dil_factor`	integer. The factor that the libraries were diluted for pcr

Value

a `pcr` object, containing the input columns as well as:

- standard_diff The differences between the `ct_mean` of a standard and one step up in the dilution (ie more concentrated, lower Ct). The most concentrated dilution has a value of 0
- dil 2^(standard_diff). The accuracy of this metric assumes that the efficiency of the PCR is 100%, which is likely good but not perfect! In the case of the first standard, `dil` = 0
- quant_actual For standards, the presumed quantity of standard, calculated from `dil`. For samples, `quantity`
- concentrationThe concentration of library, before dilution

Examples

```
system.file("extdata", "untidy-standard-curve.xlsx", package = "amplify") |>
  read_pcr() |>
  pcr_lib_calc()
```

pcr_lib_qc *Create library prep quality control data*

Description

Create library prep quality control data

Usage

```
pcr_lib_qc(lib_calc_pcr)
```

Arguments

lib_calc_pcr	A pcr object, output from pcr_lib_calc

Details

While the output of this function on its own is can theoretically be used to gauge library quality, it is best used in conjunction with a function like pcr_lib_calc_report

Value

a pcr object with list with:

- standards Data for individual standards, including calculated dilutions, given and calculated quantities, raw Ct, etc.
- samples Data for individual samples, including calculated concentrations, raw Ct, etc.
- sample_summary Summary statistics for samples grouped by replicates
- standard_summary Summary statistics for standards groupd by replicates
- outliers Data for individual samples and standards with and without their putative outliers (po) per replicate group

Examples

```
system.file("extdata", "untidy-standard-curve.xlsx", package = "amplify") |>
  pcr_tidy(pad_zero = TRUE) |>
  pcr_lib_calc() |>
  pcr_lib_qc()
```

pcr_lib_qc_plot_conc *Plot concentration of libraries across samples*

Description

Plot concentration of libraries across samples

Usage

```
pcr_lib_qc_plot_conc(lib_qc)
```

Arguments

lib_qc	Output of pcr_lib_qc

Value

a ggplot

Examples

```
system.file("extdata", "untidy-standard-curve.xlsx", package = "amplify") |>
  pcr_tidy(pad_zero = TRUE) |>
  pcr_lib_calc() |>
  pcr_lib_qc() |>
  pcr_lib_qc_plot_conc()
```

pcr_lib_qc_plot_dil *Plot standard dilutions compared to a perfect dilution*

Description

Plot standard dilutions compared to a perfect dilution

Usage

```
pcr_lib_qc_plot_dil(lib_qc)
```

Arguments

lib_qc	Output of pcr_lib_qc

Details

An optimal dilution will show blue and grey dots perfectly aligned. A plot with blue dots consistently lagging more behind the gray dots implies the dilutions are consistent, but less dilute than a 1:10 dilution. Likewise, a plot with blue dots that consistently outpace the gray dots more with each passing dot signifies consistently over-diluting the standards.

Samples are shown as red dots.

Value

a ggplot

Examples

```
system.file("extdata", "untidy-standard-curve.xlsx", package = "amplify") |>
  pcr_tidy() |>
  pcr_lib_calc() |>
  pcr_lib_qc() |>
  pcr_lib_qc_plot_dil()
```

---

pcr_lib_qc_plot_outliers

Plot mean centered samples without putative outliers

Description

Plot mean centered samples without putative outliers

Usage

```
pcr_lib_qc_plot_outliers(lib_qc)
```

Arguments

lib_qc	Output of pcr_lib_qc

Details

A sample is deemed an outlier if, upon its removal, it is more that 3Z from the mean of the remaining. This boundary of +/-3Z is demarcated by the shaded area. Gray samples are outliers. Samples |Z| > 10 away are denoted by arrows («<) pointing in their direction as well as with their Z

Value

a ggplot

Examples

```
system.file("extdata", "untidy-standard-curve.xlsx", package = "amplify") |>
  pcr_tidy(pad_zero = TRUE) |>
  pcr_lib_calc() |>
  pcr_lib_qc() |>
  pcr_lib_qc_plot_outliers()
```

---

pcr_lib_qc_plot_slope *Plot the log of library quantities vs Ct*

Description

Plot the log of library quantities vs Ct

Usage

```
pcr_lib_qc_plot_slope(lib_qc)
```

Arguments

lib_qc	Output of pcr_lib_qc

Details

An optimal plot will have a slope of -3.32. This is because we expect that a sample 10x more concentrated than another will reach the same abundance in 3.32 doublings FASTER (that is, 3.32 fewer doubles, or Cts). Therefore, for each 10x increase in concentration (one point left to right on the plot) we expect a decrease in CT of 3.32. A steeper slope (more negative) implies a poorer efficiency (more cycles are required to reach 10x than perfect doubling would imply)

Value

a ggplot

Examples

```
system.file("extdata", "untidy-standard-curve.xlsx", package = "amplify") |>
  pcr_tidy(pad_zero = TRUE) |>
  pcr_lib_calc() |>
  pcr_lib_qc() |>
  pcr_lib_qc_plot_slope()
```

pcr_lib_qc_report *Generate visual library prep pcr quality control report*

Description

Generate visual library prep pcr quality control report

Usage

```
pcr_lib_qc_report(pcr_lib_qc, report_path = NULL)
```

Arguments

pcr_lib_qc	output from pcr_lib_qc
report_path	the name of the report as well as where it should be output. If NULL, it will export to a temp directory

Value

The path to the report

Examples

```
system.file("extdata", "untidy-standard-curve.xlsx", package = "amplify") |>
  pcr_tidy() |>
  pcr_lib_calc() |>
  pcr_lib_qc() |>
  pcr_lib_qc_report()
```

pcr_plan *Plan PCR experiment*

Description

Plan PCR experiment

Usage

```
pcr_plan(
  data,
  n_primers,
  format = 384,
  exclude_border = TRUE,
  primer_names = NULL,
  headless = TRUE,
  has_names = TRUE
)
```

Arguments

data	a data.frame, with samples as the first column (if has_names = TRUE) and RNA concentrations as the second (or first, if has_names = FALSE)
n_primers	integer. Number of primers to be used in the experiment.
format	integer. 96 or 384 - the number of wells of the plate planned to be used
exclude_border	logical. Should the border be excluded to avoid edge effects? Default is TRUE.
primer_names	character vector. Names of primers.
headless	logical. If FALSE, return invisible and redirect to shiny application.
has_names	logical. Is the first column the names of the samples?

Value

a named list

Examples

```
dummy_rna_conc |>
  pcr_plan(n_primers = 3)
```

---

pcr_plan_report *Create a report from a PCR plan*

Description

Create a report from a PCR plan

Usage

```
pcr_plan_report(pcr_plan, file_path = NULL)
```

Arguments

pcr_plan	output from pcr_plan
file_path	Where the report should be written, as well as the file name. Defaults to temp file.

Value

a named list, like the output of pcr_plan, but with the output file path appended.

Examples

```
dummy_rna_conc |>
  pcr_plan(n_primers = 3) |>
  pcr_plan_report()
```

pcr_plate_view *View sample plating layout*

Description

View sample plating layout

Usage

```
pcr_plate_view(pcr, fill = target_name)
```

Arguments

fill	character. A column in tidy_pcr used to use to fill the geom_tiles
tidy_pcr	an output from the pcr_tidy function

Value

a ggplot

Examples

```
system.file("extdata", "untidy-pcr-example.xls", package = "amplify") |>
  read_pcr() |>
  pcr_plate_view()
```

pcr_plot *Plot qPCR results*

Description

Plot qPCR results

Usage

```
pcr_plot(x, ...)

## S3 method for class 'pcr'
pcr_plot(x, ...)

## S3 method for class 'data.frame'
pcr_plot(x, ...)
```

Arguments

x	a pcr object or data.frame

Value

a ggplot

Examples

```
system.file("extdata", "untidy-pcr-example.xls", package = "amplify") |>
  pcr_tidy() |>
  pcr_rq("RD1") |>
  pcr_plot()
```

---

pcr_rq *Recalculate relative quantities for a given experiment*

Description

Recalculate relative quantities for a given experiment

Usage

```
pcr_rq(x, relative_sample, control_probe = NULL, ...)

## S3 method for class 'pcr'
pcr_rq(x, relative_sample, control_probe = NULL, ...)

## S3 method for class 'data.frame'
pcr_rq(x, relative_sample, control_probe = NULL, ...)
```

Arguments

x	A pcr or data.frame
relative_sample	A sample to set others relative to (eg my_dmso_sample)
control_probe	Character. target_name to serve as endogenous control.
...	Arguments passed to respective method

Value

An object of same class as x

Examples

```
dat_path <- system.file("extdata", "untidy-pcr-example.xls", package = "amplify")

read_pcr(dat_path) |>
  pcr_rq("U6D1")

# Can also be run after using pcr_control:
```

```
read_pcr(dat_path) |>
  pcr_control("GAPDH") |>
  pcr_rq("U6D1")
```

Index