Package 'pcr'

April 1, 2020

Version 1.2.2

Title Analyzing Real-Time Quantitative PCR Data

Description Calculates the amplification efficiency and curves from real-time quantitative PCR (Polymerase Chain Reaction) data. Estimates the relative expression from PCR data using the double delta CT and the standard curve methods Livak & Schmittgen (2001) <doi:10.1006/meth.2001.1262>. Tests for statistical significance using two-group tests and linear regression Yuan et al. (2006) <doi: 10.1186/1471-2105-7-85>.

Maintainer Mahmoud Ahmed <mahmoud.s.fahmy@students.kasralainy.edu.eg>

URL https://github.com/MahShaaban/pcr

BugReports https://github.com/MahShaaban/pcr/issues

Depends R (>= 3.4.0)

Encoding UTF-8

LazyData true

RoxygenNote 7.0.2

Imports ggplot2

Suggests testthat, knitr, rmarkdown, covr, cowplot

VignetteBuilder knitr

License GPL-3

NeedsCompilation no

Author Mahmoud Ahmed [aut, cre] (<https://orcid.org/0000-0002-4377-6541>)

Repository CRAN

Date/Publication 2020-04-01 06:10:02 UTC

R topics documented:

`ct1` *C_T values from qPCR (separate tubes)*

Description

A dataset containing the C_T values of two genes from a qPCR experiment. Samples were prepared from human tissues; Brain and kidney (n = 6) each. Primers for each genes were run in separate reaction tubes.

Usage

```
ct1
```

Format

A data.frame with 12 rows and 2 variables:

c_myc C_T values of the target gene **c-myc**

GAPDH C_T values of the control gene **GAPDH**

Source

http://www3.appliedbiosystems.com/cms/groups/mcb_support/documents/generaldocuments/cms_040980.pdf

See Also

`ct2`

`ct3`

ct2 *C_T values from qPCR (same tubes)*

Description

A dataset containing the C_T values of two genes from a qPCR experiment. Samples were prepared from human tissues; Brain and kidney (n = 6) each. Primers for both genes were run in the same tubes with different reporting dyes.

Usage

```
ct2
```

Format

A data.frame with 12 rows and 2 variables:

c_myc C_T values of the target gene **c-myc**

GAPDH C_T values of the control gene **GAPDH**

Source

http://www3.appliedbiosystems.com/cms/groups/mcb_support/documents/generaldocuments/cms_040980.pdf

See Also

ct1

ct3

ct3 *C_T values from qPCR (Serial dilutions)*

Description

A dataset containing the C_T values of two genes from a serial dilution qPCR experiment. The original dataset shows only the averages and standard deviations of each of the 7 different dilutions (1, .5, .2, .1, .05, .02 and .01). These summaries were used to regenerate 3 replicates for each of the dilutions to be used in testing and examples of the different functions.

Usage

```
ct3
```

Format

A data.frame with 21 rows and 2 variables:

c_myc C_T values of the target gene **c-myc**

GAPDH C_T values of the control gene **GAPDH**

Source

http://www3.appliedbiosystems.com/cms/groups/mcb_support/documents/generaldocuments/cms_040980.pdf

See Also

ct1

ct2

---

ct4 *C_T values from qPCR (Serial dilutions)*

Description

A dataset containing the C_T values of two genes from a controlled serial dilution qPCR experiment. The data were prepared from four different dilutions (10, 2, 0.4 and 0.08) and two control groups; control and treatment (n = 12) each.

Usage

```
ct4
```

Format

A data.frame with 24 rows and 2 variables:

ref C_T values of the reference gene

target C_T values of the target gene

Source

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1395339/

pcr — pcr *package*

Description

Analyzing real-time quantitative PCR data

Details

Calculates the amplification efficiency and curves from real-time quantitative PCR (Polymerase Chain Reaction) data. Estimates the relative expression from PCR data using the double delta CT and the standard curve methods Livak & Schmittgen (2001) <doi:10.1006/meth.2001.1262>. Tests for statistical significance using two-group tests and linear regression Yuan et al. (2006) <doi:10.1186/1471-2105-7-85>.

pcr_analyze — *Apply qPCR analysis methods*

Description

A unified interface to invoke different analysis methods of qPCR data.

Usage

```
pcr_analyze(df, method = "delta_delta_ct", ...)
```

Arguments

df	A data.frame of C_T values with genes in the columns and samples in rows rows
method	A character string; 'delta_delta_ct' default, 'delta_ct' or 'relative_curve' for invoking a certain analysis model
...	Arguments passed to the methods

Details

The different analysis methods can be invoked using the argument method with 'delta_delta_ct' default, 'delta_ct' or 'relative_curve' for the double delta C_T, delta ct or the standard curve model respectively. Alternatively, the same methods can be applied by using the corresponding functions directly: pcr_ddct, pcr_dct or pcr_curve

Value

A data.frame by default, when plot is TRUE returns a plot. For details; pcr_ddct, pcr_dct and pcr_curve.

References

Livak, Kenneth J, and Thomas D Schmittgen. 2001. "Analysis of Relative Gene Expression Data Using Real-Time Quantitative PCR and the Double Delta CT Method." Methods 25 (4). ELSEVIER. doi:10.1006/meth.2001.1262.

Examples

```
# applying the delta delta ct method
## locate and read raw ct data
fl <- system.file('extdata', 'ct1.csv', package = 'pcr')
ct1 <- read.csv(fl)

# add grouping variable
group_var <- rep(c('brain', 'kidney'), each = 6)

# calculate all values and errors in one step
pcr_analyze(ct1,
            group_var = group_var,
            reference_gene = 'GAPDH',
            reference_group = 'brain',
            method = 'delta_delta_ct')

# return a plot
pcr_analyze(ct1,
            group_var = group_var,
            reference_gene = 'GAPDH',
            reference_group = 'brain',
            method = 'delta_delta_ct',
            plot = TRUE)

# applying the delta ct method
# make a data.frame of two identical columns
pcr_hk <- data.frame(
  GAPDH1 = ct1$GAPDH,
  GAPDH2 = ct1$GAPDH
  )

# calculate fold change
pcr_analyze(pcr_hk,
            group_var = group_var,
            reference_group = 'brain',
            method = 'delta_ct')

# return a plot
pcr_analyze(pcr_hk,
            group_var = group_var,
            reference_group = 'brain',
            method = 'delta_ct',
            plot = TRUE)

# applying the standard curve method
# locate and read file
```

```
fl <- system.file('extdata', 'ct3.csv', package = 'pcr')
ct3 <- read.csv(fl)

# make a vector of RNA amounts
amount <- rep(c(1, .5, .2, .1, .05, .02, .01), each = 3)

# calculate curve
standard_curve <- pcr_assess(ct3, amount = amount, method = 'standard_curve')
intercept <- standard_curve$intercept
slope <- standard_curve$slope

# apply the standard curve method
pcr_analyze(ct1,
            group_var = group_var,
            reference_gene = 'GAPDH',
            reference_group = 'brain',
            intercept = intercept,
            slope = slope,
            method = 'relative_curve')

# return a plot
pcr_analyze(ct1,
            group_var = group_var,
            reference_gene = 'GAPDH',
            reference_group = 'brain',
            intercept = intercept,
            slope = slope,
            method = 'relative_curve',
            plot = TRUE)
```

`pcr_assess` *Assess qPCR data quality*

Description

A unified interface to invoke different quality assessment methods of qPCR data.

Usage

```
pcr_assess(df, method = "standard_curve", ...)
```

Arguments

`df`	A data.frame of C_T values with genes in the columns and samples in rows rows. Each sample are replicates of a known input/dilution given by amount
`method`	A character string; 'standard_curve' (default) or 'efficiency' for invoking a certain quality assessment model
`...`	Arguments passed to the methods

Details

The different quality assessment methods can be invoked using the argument method with 'standard_curve' or 'efficiency'. Alternatively, the same methods can be applied by using the corresponding functions: pcr_standard or pcr_efficiency for calculating the amplification efficiency of a PCR reaction or the individual standard curves respectively. Unlike the amplification efficiency calculation when, using the double delta ct model, the standard curves are required in calculating the standard curve analysis model.

Value

A data.frame or a plot. For details; pcr_standard and pcr_efficiency

Examples

```
#' # locate and read file
fl <- system.file('extdata', 'ct3.csv', package = 'pcr')
ct3 <- read.csv(fl)

# make amount/dilution variable
amount <- rep(c(1, .5, .2, .1, .05, .02, .01), each = 3)

# calculate the standard curve
pcr_assess(ct3,
           amount = amount,
           method = 'standard_curve')

# retrun a plot
pcr_assess(ct3,
           amount = amount,
           method = 'standard_curve',
           plot = TRUE)

# calculate amplification efficiency
pcr_assess(ct3,
           amount = amount,
           reference_gene = 'GAPDH',
           method = 'efficiency')

# return a plot
pcr_assess(ct3,
           amount = amount,
           reference_gene = 'GAPDH',
           method = 'efficiency',
           plot = TRUE)
```

---

pcr_curve *Calculate the standard curve model*

Description

Uses the C_T values and a reference gene and a group, in addition to the intercept and slope of each gene form a serial dilution experiment, to calculate the standard curve model and estimate the normalized relative expression of the target genes.

Usage

```
pcr_curve(
  df,
  group_var,
  reference_gene,
  reference_group,
  mode = "separate_tube",
  intercept,
  slope,
  plot = FALSE,
  ...
)
```

Arguments

df	A data.frame of C_T values with genes in the columns and samples in rows rows
group_var	A character vector of a grouping variable. The length of this variable should equal the number of rows of df
reference_gene	A character string of the column name of a control gene
reference_group	A character string of the control group in group_var
mode	A character string of; 'separate_tube' (default) or 'same_tube'. This is to indicate whether the different genes were run in separate or the same PCR tube
intercept	A numeric vector of intercept and length equals the number of genes
slope	A numeric vector of slopes length equals the number of genes
plot	A logical (default is FALSE)
...	Arguments passed to customize plot

Details

this model doesn't assume perfect amplification but rather actively use the amplification in calculating the relative expression. So when the amplification efficiency of all genes are 100% both methods should give similar results. The standard curve method is applied using two steps. First, serial dilutions of the mRNAs from the samples of interest are used as input to the PCR reaction. The linear trend of the log input amount and the resulting C_T values for each gene are used to calculate an intercept and a slope. Secondly, these intercepts and slopes are used to calculate the amounts of mRNA of the genes of interest and the control/reference in the samples of interest and the control sample/reference. These amounts are finally used to calculate the relative expression.

Value

A data.frame of 7 columns

- group The unique entries in group_var
- gene The column names of df
- normalized The normalized expression of target genes relative to a reference_gene
- calibrated The calibrated expression of target genes relative to a reference_group
- error The standard deviation of normalized relative expression
- lower The lower interval of the normalized relative expression
- upper The upper interval of the normalized relative expression

When `plot` is TRUE, returns a bar graph of the calibrated expression of the genes in the column and the groups in the column group. Error bars are drawn using the columns lower and upper. When more one gene are plotted the default in dodge bars. When the argument facet is TRUE a separate panel is drawn for each gene.

References

Livak, Kenneth J, and Thomas D Schmittgen. 2001. "Analysis of Relative Gene Expression Data Using Real-Time Quantitative PCR and the Double Delta CT Method." Methods 25 (4). ELSEVIER. doi:10.1006/meth.2001.1262.

Examples

```
# locate and read file
fl <- system.file('extdata', 'ct3.csv', package = 'pcr')
ct3 <- read.csv(fl)

fl <- system.file('extdata', 'ct1.csv', package = 'pcr')
ct1 <- read.csv(fl)

# make a vector of RNA amounts
amount <- rep(c(1, .5, .2, .1, .05, .02, .01), each = 3)

# calculate curve
standard_curve <- pcr_assess(ct3, amount = amount, method = 'standard_curve')
intercept <- standard_curve$intercept
slope <- standard_curve$slope

# make grouping variable
group <- rep(c('brain', 'kidney'), each = 6)

# apply the standard curve method
pcr_curve(ct1,
          group_var = group,
          reference_gene = 'GAPDH',
          reference_group = 'brain',
          intercept = intercept,
          slope = slope)
```

```
# returns a plot
pcr_curve(ct1,
          group_var = group,
          reference_gene = 'GAPDH',
          reference_group = 'brain',
          intercept = intercept,
          slope = slope,
          plot = TRUE)
```

`pcr_dct` *Calculate the delta_ct model*

Description

Uses the C_T values and a reference group to calculate the delta C_T model to estimate the relative fold change of a gene between groups

Usage

```
pcr_dct(
  df,
  group_var,
  reference_group,
  mode = "separate_tube",
  plot = FALSE,
  ...
)
```

Arguments

`df`	A data.frame of C_T values with genes in the columns and samples in rows rows
`group_var`	A character vector of a grouping variable. The length of this variable should equal the number of rows of df
`reference_group`	A character string of the control group in group_var
`mode`	A character string of; 'separate_tube' (default) or 'same_tube'. This is to indicate whether the different genes were run in separate or the same PCR tube
`plot`	A logical (default is FALSE)
`...`	Arguments passed to customize plot

Details

This method is a variation of the double delta C_T model, `pcr_ddct`. It can be used to calculate the fold change of in one sample relative to the others. For example, it can be used to compare and choosing a control/reference genes.

Value

A data.frame of 7 columns

- group The unique entries in group_var
- gene The column names of df
- calibrated The average C_T value of target genes after subtracting that of the reference_group
- fold_change The fold change of genes relative to a reference_group
- error The standard deviation of the fold_change
- lower The lower interval of the fold_change
- upper The upper interval of the fold_change

When `plot` is TRUE, returns a bar graph of the fold change of the genes in the column and the groups in the column group. Error bars are drawn using the columns lower and upper. When more one gene are plotted the default in dodge bars. When the argument facet is TRUE a separate panel is drawn for each gene.

References

Livak, Kenneth J, and Thomas D Schmittgen. 2001. "Analysis of Relative Gene Expression Data Using Real-Time Quantitative PCR and the Double Delta CT Method." Methods 25 (4). ELSEVIER. doi:10.1006/meth.2001.1262.

Examples

```
# locate and read file
fl <- system.file('extdata', 'ct1.csv', package = 'pcr')
ct1 <- read.csv(fl)

# make a data.frame of two identical columns
pcr_hk <- data.frame(
  GAPDH1 = ct1$GAPDH,
  GAPDH2 = ct1$GAPDH
  )

# add grouping variable
group_var <- rep(c('brain', 'kidney'), each = 6)

# calculate caliberation
pcr_dct(pcr_hk,
        group_var = group_var,
        reference_group = 'brain')

# returns a plot
pcr_dct(pcr_hk,
        group_var = group_var,
        reference_group = 'brain',
        plot = TRUE)

# returns a plot with facets
```

```
pcr_dct(pcr_hk,
        group_var = group_var,
        reference_group = 'brain',
        plot = TRUE,
        facet = TRUE)
```

---

pcr_ddct *Calculate the delta_delta_ct model*

Description

Uses the C_T values and a reference gene and a group to calculate the delta delta C_T model to estimate the normalized relative expression of target genes.

Usage

```
pcr_ddct(
  df,
  group_var,
  reference_gene,
  reference_group,
  mode = "separate_tube",
  plot = FALSE,
  ...
)
```

Arguments

df	A data.frame of C_T values with genes in the columns and samples in rows rows
group_var	A character vector of a grouping variable. The length of this variable should equal the number of rows of df
reference_gene	A character string of the column name of a control gene
reference_group	A character string of the control group in group_var
mode	A character string of; 'separate_tube' (default) or 'same_tube'. This is to indicate whether the different genes were run in separate or the same PCR tube
plot	A logical (default is FALSE)
...	Arguments passed to customize plot

Details

The comparative C_T methods assume that the cDNA templates of the gene/s of interest as well as the control/reference gene have similar amplification efficiency. And that this amplification efficiency is near perfect. Meaning, at a certain threshold during the linear portion of the PCR reaction, the amount of the gene of the interest and the control double each cycle. Another assumptions is that,

the expression difference between two genes or two samples can be captured by subtracting one (gene or sample of interest) from another (reference). This final assumption requires also that these references don't change with the treatment or the course in question.

Value

A data.frame of 8 columns:

- group The unique entries in group_var
- gene The column names of df. reference_gene is dropped
- normalized The C_T value (or the average C_T value) of target genes after subtracting that of the reference_gene
- calibrated The normalized average C_T value of target genes after subtracting that of the reference_group
- relative_expression The expression of target genes normalized by a reference_gene and calibrated by a reference_group
- error The standard deviation of the relative_expression
- lower The lower interval of the relative_expression
- upper The upper interval of the relative_expression

When `plot` is TRUE, returns a bar graph of the relative expression of the genes in the column and the groups in the column group. Error bars are drawn using the columns lower and upper. When more one gene are plotted the default in dodge bars. When the argument facet is TRUE a separate panel is drawn for each gene.

Examples

```
## locate and read raw ct data
fl <- system.file('extdata', 'ct1.csv', package = 'pcr')
ct1 <- read.csv(fl)

# add grouping variable
group_var <- rep(c('brain', 'kidney'), each = 6)

# calculate all values and errors in one step
pcr_ddct(ct1,
         group_var = group_var,
         reference_gene = 'GAPDH',
         reference_group = 'brain')

# return a plot
pcr_ddct(ct1,
         group_var = group_var,
         reference_gene = 'GAPDH',
         reference_group = 'brain',
         plot = TRUE)
```

pcr_efficiency *Calculate amplification efficiency*

Description

Uses the C_T values from a serial dilution experiment to calculate the amplification efficiency of a PCR reaction.

Usage

```
pcr_efficiency(df, amount, reference_gene, plot = FALSE)
```

Arguments

df	A data.frame of C_T values with genes in the columns and samples in rows rows. Each sample are replicates of a known input/dilution given by amount
amount	A numeric vector of the input amounts or dilutions. The length of this vector should equal the row number of df
reference_gene	A character string of the column name of a control gene
plot	A logical (default FALSE) to indicate whether to return a data.frame or a plot

Details

Fortunately, regardless of the method used in the analysis of qPCR data, The quality assessment are done in a similar way. It requires an experiment similar to that of calculating the standard curve. Serial dilutions of the genes of interest and controls are used as input to the reaction and different calculations are made. The amplification efficiency is approximated be the linear trend between the difference between the C_T value of a gene of interest and a control/reference (ΔC_T) and the log input amount. This piece of information is required when using the $\Delta\Delta C_T$ model. Typically, the slope of the curve should be very small and the R^2 value should be very close to one. Other analysis methods are recommended when this is not the case.

Value

When plot is FALSE returns a data.frame of 4 columns describing the line between the ΔC_T of target genes and the log of amount

- gene The column names of df. reference_gene is dropped
- intercept The intercept of the line
- slope The slope of the line
- r_squared The squared correlation

When plot is TRUE returns a graph instead shows the average and standard deviation of of the ΔC_T at different input amounts. In addition, a linear trend line is drawn.

References

Livak, Kenneth J, and Thomas D Schmittgen. 2001. "Analysis of Relative Gene Expression Data Using Real-Time Quantitative PCR and the Double Delta CT Method." Methods 25 (4). ELSEVIER. doi:10.1006/meth.2001.1262.

Examples

```
# locate and read file
fl <- system.file('extdata', 'ct3.csv', package = 'pcr')
ct3 <- read.csv(fl)

# make amount/dilution variable
amount <- rep(c(1, .5, .2, .1, .05, .02, .01), each = 3)

# calculate amplification efficiency
pcr_efficiency(ct3,
               amount = amount,
               reference_gene = 'GAPDH')

# plot amplification efficiency
pcr_efficiency(ct3,
               amount = amount,
               reference_gene = 'GAPDH',
               plot = TRUE)
```

---

pcr_lm *Linear regression qPCR data*

Description

Linear regression qPCR data

Usage

```
pcr_lm(
  df,
  group_var,
  reference_gene,
  reference_group,
  model_matrix = NULL,
  mode = "subtract",
  tidy = TRUE,
  ...
)
```

Arguments

df	A data.frame of C_T values with genes in the columns and samples in rows rows
group_var	A character vector of a grouping variable. The length of this variable should equal the number of rows of df
reference_gene	A character string of the column name of a control gene
reference_group	A character string of the control group in group_var
model_matrix	A model matrix for advanced experimental design. for constructing such a matrix with different variables check model.matrix
mode	A character string for the normalization mode. Possible values are "subtract" (default) or "divide".
tidy	A logical whether to return a list of lm or a tidy data.frame. Default TRUE.
...	Other arguments to lm

Value

A data.frame of 6 columns

- term The term being tested
- gene The column names of df. reference_gene is dropped
- estimate The estimate for each term
- p_value The p-value for each term
- lower The low 95% confidence interval
- upper The high 95% confidence interval

When tidy is FALSE, returns a list of lm objects.

Examples

```
# locate and read data
fl <- system.file('extdata', 'ct4.csv', package = 'pcr')
ct4 <- read.csv(fl)

# make group variable
group <- rep(c('control', 'treatment'), each = 12)

# test
pcr_lm(ct4,
       group_var = group,
       reference_gene = 'ref',
       reference_group = 'control')

# testing using lm method
pcr_test(ct4,
         group_var = group,
         reference_gene = 'ref',
         reference_group = 'control',
         test = 'lm')
```

`pcr_standard` *Calculate the standard curve*

Description

Uses the C_T values from a serial dilution experiment to calculate the a curve for each gene and the log of the input amount

Usage

```
pcr_standard(df, amount, plot = FALSE)
```

Arguments

`df`	A data.frame of C_T values with genes in the columns and samples in rows rows. Each sample are replicates of a known input/dilution given by amount
`amount`	A numeric vector of the input amounts or dilutions. The length of this vector should equal the row number of df
`plot`	A logical (default FALSE) to indicate whether to return a data.frame or a plot

Details

Fortunately, regardless of the method used in the analysis of qPCR data, The quality assessment are done in a similar way. It requires an experiment similar to that of calculating the standard curve. Serial dilutions of the genes of interest and controls are used as input to the reaction and different calculations are made. Curves are required for each gene using the C_T value and the log of the input amount. In this case, a separate slope and intercept are required for the calculation of the relative expression when applying the standard curve model.

Value

When plot is FALSE returns a data.frame of 4 columns describing the line between the C_T of each gene and the log of amount

- gene The column names of df
- intercept The intercept of the line
- slope The slope of the line
- r_squared The squared correlation

When plot is TRUE returns a graph instead shows the average and standard deviation of of the C_T at different input amounts.

References

Livak, Kenneth J, and Thomas D Schmittgen. 2001. “Analysis of Relative Gene Expression Data Using Real-Time Quantitative PCR and the Double Delta CT Method.” Methods 25 (4). ELSEVIER. doi:10.1006/meth.2001.1262.

Examples

```
# locate and read file
fl <- system.file('extdata', 'ct3.csv', package = 'pcr')
ct3 <- read.csv(fl)

# make amount/dilution variable
amount <- rep(c(1, .5, .2, .1, .05, .02, .01), each = 3)

# calculate the standard curve
pcr_standard(ct3,
             amount = amount)

# plot the standard curve
pcr_standard(ct3,
             amount = amount,
             plot = TRUE)
```

---

pcr_test *Statistical testing of PCR data*

Description

A unified interface to different statistical significance tests for qPCR data

Usage

```
pcr_test(df, test = "t.test", ...)
```

Arguments

df	A data.frame of C_T values with genes in the columns and samples in rows rows
test	A character string; 't.test' default, 'wilcox.test' or 'lm'
...	Other arguments for the testing methods

Details

The simple t-test can be used to test the significance of the difference between two conditions ΔC_T. t-test assumes in addition, that the input C_T values are normally distributed and the variance between conditions are comparable. Wilcoxon test can be used when sample size is small and those two last assumptions are hard to achieve.

Two use the linear regression here. A null hypothesis is formulated as following,

$$C_{T,target,treatment} - C_{T,control,treatment} = C_{T,target,control} - C_{T,control,control} \quad \text{or} \quad \Delta\Delta C_T$$

This is exactly the $\Delta\Delta C_T$ as explained earlier. So the $\Delta\Delta C_T$ is estimated and the null is rejected when $\Delta\Delta C_T \neq 0$.

Value

A data.frame of 5 columns in addition to term when test == 'lm'

- term The linear regression comparison terms
- gene The column names of df. reference_gene is dropped
- estimate The estimate for each term
- p_value The p-value for each term
- lower The low 95% confidence interval
- upper The high 95% confidence interval

For details about the test methods themselves and different parameters, consult `t.test`, `wilcox.test` and `lm`

References

Yuan, Joshua S, Ann Reed, Feng Chen, and Neal Stewart. 2006. "Statistical Analysis of Real-Time PCR Data." BMC Bioinformatics 7 (85). BioMed Central. doi:10.1186/1471-2105-7-85.

Examples

```
# locate and read data
fl <- system.file('extdata', 'ct4.csv', package = 'pcr')
ct4 <- read.csv(fl)

# make group variable
group <- rep(c('control', 'treatment'), each = 12)

# test using t-test
pcr_test(ct4,
         group_var = group,
         reference_gene = 'ref',
         reference_group = 'control',
         test = 't.test')

# test using wilcox.test
pcr_test(ct4,
         group_var = group,
         reference_gene = 'ref',
         reference_group = 'control',
         test = 'wilcox.test')

# testing using lm
pcr_test(ct4,
         group_var = group,
         reference_gene = 'ref',
         reference_group = 'control',
         test = 'lm')

# testing advanced designs using a model matrix
# make a model matrix
```

```
group <- relevel(factor(group), ref = 'control')
dose <- rep(c(100, 80, 60, 40), each = 3, times = 2)
mm <- model.matrix(~group:dose, data = data.frame(group, dose))

# test using lm
pcr_test(ct4,
         reference_gene = 'ref',
         model_matrix = mm,
         test = 'lm')

# using linear models to check the effect of RNA quality
# make a model matrix
group <- relevel(factor(group), ref = 'control')
set.seed(1234)
quality <- scale(rnorm(n = 24, mean = 1.9, sd = .1))
mm <- model.matrix(~group + group:quality, data = data.frame(group, quality))

# testing using lm
pcr_test(ct4,
         reference_gene = 'ref',
         model_matrix = mm,
         test = 'lm')

# using linear model to check the effects of mixing separate runs
# make a model matrix
group <- relevel(factor(group), ref = 'control')
run <- factor(rep(c(1:3), 8))
mm <- model.matrix(~group + group:run, data = data.frame(group, run))

# test using lm
pcr_test(ct4,
         reference_gene = 'ref',
         model_matrix = mm,
         test = 'lm')
```

`pcr_ttest` *t-test qPCR data*

Description

t-test qPCR data

Usage

```
pcr_ttest(df, group_var, reference_gene, reference_group, tidy = TRUE, ...)
```

Arguments

df	A data.frame of C_T values with genes in the columns and samples in rows rows
group_var	A character vector of a grouping variable. The length of this variable should equal the number of rows of df
reference_gene	A character string of the column name of a control gene
reference_group	A character string of the control group in group_var
tidy	A logical whether to return a list of htest or a tidy data.frame. Default TRUE.
...	Other arguments to t.test

Value

A data.frame of 5 columns

- gene The column names of df. reference_gene is dropped
- estimate The estimate for each term
- p_value The p-value for each term
- lower The low 95% confidence interval
- upper The high 95% confidence interval

When tidy is FALSE, returns a list of htest objects.

Examples

```
# locate and read data
fl <- system.file('extdata', 'ct4.csv', package = 'pcr')
ct4 <- read.csv(fl)

# make group variable
group <- rep(c('control', 'treatment'), each = 12)

# test
pcr_ttest(ct4,
          group_var = group,
          reference_gene = 'ref',
          reference_group = 'control')

# test using t.test method
pcr_test(ct4,
         group_var = group,
         reference_gene = 'ref',
         reference_group = 'control',
         test = 't.test')
```

`pcr_wilcox` *Wilcoxon test qPCR data*

Description

Wilcoxon test qPCR data

Usage

```
pcr_wilcox(df, group_var, reference_gene, reference_group, tidy = TRUE, ...)
```

Arguments

`df`	A data.frame of C_T values with genes in the columns and samples in rows rows
`group_var`	A character vector of a grouping variable. The length of this variable should equal the number of rows of df
`reference_gene`	A character string of the column name of a control gene
`reference_group`	A character string of the control group in group_var
`tidy`	A `logical` whether to return a `list` of `htest` or a tidy `data.frame`. Default TRUE.
`...`	Other arguments to `wilcox.test`

Value

A data.frame of 5 columns

- gene The column names of df. reference_gene is dropped
- estimate The estimate for each term
- p_value The p-value for each term
- lower The low 95% confidence interval
- upper The high 95% confidence interval

When `tidy` is FALSE, returns a `list` of `htest` objects.

Examples

```
# locate and read data
fl <- system.file('extdata', 'ct4.csv', package = 'pcr')
ct4 <- read.csv(fl)

# make group variable
group <- rep(c('control', 'treatment'), each = 12)

# test
pcr_wilcox(ct4,
```

```
            group_var = group,
            reference_gene = 'ref',
            reference_group = 'control')

# test using wilcox.test method
pcr_test(ct4,
         group_var = group,
         reference_gene = 'ref',
         reference_group = 'control',
         test = 'wilcox.test')
```

Index