Package: qPCRtools (via r-universe)

October 29, 2026

Version 1.0.2

Title Tools for qPCR

Description Provides tools for quantitative PCR (qPCR) data analysis, including standard curve calculation with amplification efficiency, relative expression calculation by the 2-'dCt', 2-'ddCt', standard curve and 'RqPCR' methods, and RNA volume calculation for reverse transcription.

URL https://github.com/lixiang117423/qPCRtools

BugReports https://github.com/lixiang117423/qPCRtools/issues

License MIT + file LICENSE

Encoding UTF-8

Imports broom, dplyr, ggplot2, ggpmisc, ggthemes, kableExtra, magrittr, multcomp, rstatix, stats, tibble, tidyr

Config/roxygen2/version 8.1.0

NeedsCompilation no

Author Xiang LI [cre, aut]

Maintainer Xiang LI <lixiang117423@gmail.com>

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

Date/Publication 2026-10-29 05:50:02 UTC

RemoteUrl https://github.com/cran/qPCRtools

RemoteRef HEAD

RemoteSha 7d482647d4953ab738664bcef04e1b02536d9346

Contents

CalCurve 2
CalExp2dCt 3
CalExp2ddCt 4
CalExpCurve 5
CalExpRqPCR 7
CalRTable 8

Index 9

`CalCurve` *Standard Curve Calculation.*

Description

The function can calculate the standard curve. At the same time, it can get the amplification efficiency of primer(s). Based on the amplification efficiency, we can know which method can be used to calculate the expression level.

Usage

```
CalCurve(
  cq.table,
  concen.table,
  highest.concen,
  lowest.concen,
  dilution = 4,
  by.mean = TRUE
)
```

Arguments

`cq.table`	The data frame of the position and Cq value.
`concen.table`	The data frame of the position and concentration.
`highest.concen`	The highest concentration for calculation.
`lowest.concen`	The lowest concentration for calculation.
`dilution`	Dilution factor of cDNA template. The default value is 4.
`by.mean`	Calculation by mean Cq value or not. The default value is TRUE.

Value

A list.

Author(s)

Xiang LI <lixiang117423@gmail.com>

Examples

```
df.1.path <- system.file("examples", "calsc.cq.txt", package = "qPCRtools")
df.2.path <- system.file("examples", "calsc.info.txt", package = "qPCRtools")
df.1 <- read.table(df.1.path, header = TRUE)
df.2 <- read.table(df.2.path, header = TRUE)
CalCurve(
  cq.table = df.1,
  concen.table = df.2,
```

```
  lowest.concen = 4,
  highest.concen = 4096,
  dilu = 4,
  by = "mean"
) -> p

p[["table"]]
p[["figure"]]
```

CalExp2dCt *Calculate expression using standard curve.*

Description

Calculate expression using standard curve.

Usage

```
CalExp2dCt(cq.table, design.table, ref.gene = "Actin")
```

Arguments

cq.table	The data frame of the position and cq value.
design.table	The data frame of the position and corresponding information.
ref.gene	The name of reference gene.

Value

A list contain a table and a figure.

Author(s)

Xiang LI <lixiang117423@gmail.com>

Examples

```
df1.path <- system.file("examples", "dct.cq.txt", package = "qPCRtools")
df2.path <- system.file("examples", "dct.design.txt", package = "qPCRtools")
cq.table <- read.table(df1.path, sep = ",", header = TRUE)
design.table <- read.table(df2.path, sep = ",", header = TRUE)
CalExp2dCt(cq.table,
           design.table,
           ref.gene = "Actin"
) -> res
```

CalExp2ddCt *Calculate expression using standard curve.*

Description

Calculate expression using standard curve.

Usage

```
CalExp2ddCt(
  cq.table,
  design.table,
  ref.gene = "OsUBQ",
  ref.group = "CK",
  stat.method = "t.test",
  remove.outliers = TRUE,
  fig.type = "box",
  fig.ncol = NULL
)
```

Arguments

cq.table	The data frame of the position and cq value.
design.table	The data frame of the position and corresponding information.
ref.gene	The name of reference gene.
ref.group	The name of reference group.
stat.method	Statistical method.
remove.outliers	Remove the outliers of each group and gene, or not.
fig.type	Output image type, 'box' represents 'boxplot', 'bar' represents 'barplot'.
fig.ncol	Number of columes of figure.

Value

A list contain a table and a figure.

Author(s)

Xiang LI <lixiang117423@gmail.com>

Examples

```
df1.path = system.file("examples", "ddct.cq.txt", package = "qPCRtools")
df2.path = system.file("examples", "ddct.design.txt", package = "qPCRtools")

cq.table = read.table(df1.path, header = TRUE)
design.table = read.table(df2.path, header = TRUE)

CalExp2ddCt(cq.table,
            design.table,
            ref.gene = "OsUBQ",
            ref.group = "CK",
            stat.method = "t.test",
            remove.outliers = TRUE,
            fig.type = "box",
            fig.ncol = NULL) -> res

res[["table"]]
res[["figure"]]
```

CalExpCurve *Calculate expression using standard curve.*

Description

Calculate expression using standard curve.

Usage

```
CalExpCurve(
  cq.table,
  curve.table,
  design.table,
  correction = TRUE,
  ref.gene = "OsUBQ",
  stat.method = "t.test",
  ref.group = "CK",
  fig.type = "box",
  fig.ncol = NULL,
  dilution = 4
)
```

Arguments

cq.table	The data frame of the position and Cq value.
curve.table	The data frame of standard curve parameters by gene.
design.table	The data frame of the position and corresponding information.

correction	Correct expression value by reference gene.
ref.gene	The name of reference gene.
stat.method	Statistical method.
ref.group	The name of reference group.
fig.type	Output image type, ‘box‘ represents ‘boxplot‘, ‘bar‘ represents ‘barplot‘.
fig.ncol	Number of columes of figure.
dilution	Dilution factor used when the standard curve was fitted; it is the base of the log transform in ‘CalCurve()‘ (default 4). Expression is reported as concentration, i.e. ‘dilution^((Cq - Intercept) / Slope)‘.

Value

A list contain a table and a figure.

Author(s)

Xiang LI <lixiang117423@gmail.com>

Examples

```
df1.path = system.file("examples", "cal.exp.curve.cq.txt", package = "qPCRtools")
df2.path = system.file("examples", "cal.expre.curve.sdc.txt", package = "qPCRtools")
df3.path = system.file("examples", "cal.exp.curve.design.txt", package = "qPCRtools")

cq.table = read.table(df1.path, header = TRUE)
curve.table = read.table(df2.path, sep = "\t", header = TRUE)
design.table = read.table(df3.path, header = TRUE)

CalExpCurve(
  cq.table,
  curve.table,
  design.table,
  correction = TRUE,
  ref.gene = "OsUBQ",
  stat.method = "t.test",
  ref.group = "CK",
  fig.type = "box",
  fig.ncol = NULL) -> res

res[["table"]]
res[["figure"]]
```

CalExpRqPCR *Calculate expression using standard curve.*

Description

Calculate expression using standard curve.

Usage

```
CalExpRqPCR(
  cq.table,
  design.table,
  ref.gene = NULL,
  ref.group = "CK",
  stat.method = "t.test",
  fig.type = "box",
  fig.ncol = NULL
)
```

Arguments

cq.table	The data frame of the position and cq value.
design.table	The data frame of the position and corresponding information.
ref.gene	The name of reference gene.
ref.group	The name of reference group.
stat.method	Statistical method.
fig.type	Output image type, ‘box‘ represents ‘boxplot‘, ‘bar‘ represents ‘barplot‘.
fig.ncol	Number of columes of figure.

Value

A list contain a table and a figure.

Author(s)

Xiang LI <lixiang117423@gmail.com>

Examples

```
df1.path <- system.file("examples", "cal.expre.rqpcr.cq.txt", package = "qPCRtools")
df2.path <- system.file("examples", "cal.expre.rqpcr.design.txt", package = "qPCRtools")

cq.table <- read.table(df1.path, header = TRUE)
design.table <- read.table(df2.path, header = TRUE)

CalExpRqPCR(cq.table,
```

```
            design.table,
            ref.gene = NULL,
            ref.group = "CK",
            stat.method = "t.test",
            fig.type = "box",
            fig.ncol = NULL
            ) -> res

res[["table"]]
res[["figure"]]
```

`CalRTable` *Calculate RNA volume for reverse transcription.*

Description

The first step of qPCR is usually the preparation of cDNA. We need to calculate the column of RNA for reverse transcription to cDNA. So, if we have the concentration of RNA, we can use the function 'CalRTable' to do that.

Usage

```
CalRTable(data, template, RNA.weight = 1)
```

Arguments

`data`	A data.frame contained the sample names and the concentration value. The default unit of concentration is ng/uL.
`template`	A data.frame contained the information of reverse transcription. In this data.frame there must be a column called 'all'.
`RNA.weight`	RNA weight required for reverse transcription. Default is 1 ug.

Value

A list contain a table and a figure.

Author(s)

Xiang LI <lixiang117423@gmail.com>

Examples

```
df.1.path <- system.file("examples", "crtv.data.txt", package = "qPCRtools")
df.2.path <- system.file("examples", "crtv.template.txt", package = "qPCRtools")
df.1 <- read.table(df.1.path, sep = "\t", header = TRUE)
df.2 <- read.table(df.2.path, sep = "\t", header = TRUE)
result <- CalRTable(data = df.1, template = df.2, RNA.weight = 2)
head(result)
```

Index

CalCurve, 2
CalExp2dCt, 3
CalExp2ddCt, 4
CalExpCurve, 5
CalExpRqPCR, 7
CalRTable, 8