Package 'NORMAgene'

February 28, 2026

Type Package

Title Data-Driven qRT-PCR Normalization Using NORMAgene

Version 0.1.1

Description Enables correction for technical variance in raw quantitative reverse
transcription polymerase chain reaction (qRT-PCR) data using the least
squares-based NORMAgene data-driven normalization algorithm originally
described by Heckmann et al. (2011) <doi:10.1186/1471-2105-12-250>.
Performs normalization of raw crossing threshold values (CT) and also
calculates relative variability metrics that can be used to assess the impact
of normalization on variance.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.3.3

Suggests testthat (>= 3.0.0)

Author Grant C. O'Connell [aut, cre] (ORCID:
<https://orcid.org/0000-0002-2642-6670>)

Maintainer Grant C. O'Connell <goconnell.phd@gmail.com>

Repository CRAN

Depends R (>= 3.5)

Config/testthat/edition 3

NeedsCompilation no

Date/Publication 2026-02-28 20:40:13 UTC

Contents

NORMAgene-package	*Data-Driven qRT-PCR Normalization Using NORMAgene*

Description

The 'NORMAgene' package enables normalization of raw quantitative reverse transcription poly-merase chain reaction (qRT-PCR) crossing threshold (CT) values using the NORMAgene data-driven normalization algorithm originally described by Heckmann et al. (2011). The NORMAgene algorithm uses within experimental condition least squares fits to estimate per-replicate technical variance and generate corresponding scaling factors that are ultimately applied for normalization. Normalization is reference gene agnostic, and can be carried out on data from as few as five target genes.

Details

The primary user-facing function is [norma_gene()](#), which is suitable for most standalone single experiment normalization workflows. [norma_gene()](#) applies the NORMAgene normalization algorithm to raw CT values provided via an input data frame appended with requisite experimental metadata. In addition to generating normalized CT values, the ratio of standard deviations-based relative variability metric originally described by Heckmann et al. (2011) is also calculated, both within target genes and experimental conditions, which users can employ to evaluate the effect of normalization on CT variability. This metric represents the proportional change in CT-value standard deviation pre to post normalization, with values of less than one indicating a reduction in variance as a result of normalization, and values of greater than one indicating an increase in variance as a result of normalization. The scaling factors applied for normalization can be accessed using [correction_factors()](#), while relative variability metrics can be accessed using [relative_variability()](#).

Note: [norma_gene_core()](#) provides matrix-based execution of the NORMAgene algorithm and is internally called by [norma_gene()](#). While use of [norma_gene()](#) is recommended in a majority of situations, directly calling [norma_gene_core()](#) may afford advanced users a lightweight option for cleaner integration into larger post-analytical pipelines. [norma_gene_core()](#) is not exported and can only be called using the internal namespace operator.

Two real-world qPCR datasets are also included, which are used in the documentation examples. The dataset [multi_cond_data](#) contains raw CT values and experimental meta-data from a case-control comparison of whole blood gene expression in human ischemic stroke published by O'Connell et al. (2017). It can be used to demonstrate or evaluate normalization workflows for use-cases involving data from multiple experimental condition. The dataset [single_cond_data](#) contains raw CT values and experimental meta-data from a single cohort study of whole blood gene expression in human ischemic stroke published by O'Connell et al. (2017). It can be used to demonstrate or evaluate normalization workflows for use-cases involving data from a single experimental condition.

Main functions

[norma_gene()](#) Normalize CT values stored in a data frame.

relative_variability() Extract relative variability metrics.

correction_factors() Extract per-replicate correction factors.

Datasets

multi_cond_data Example dataset from a real-world multi-condition experiment.

single_cond_data Example dataset from a real-world single condition experiment.

Citation

If you use the NORMAgene package in published work, please cite:

O'Connell, GC. (2024). *NORMAgene*. R package version 0.1.1. Available from `https://CRAN.R-project.org/package=NORMAgene`.

References

Heckmann, LH., Sørensen, PB., Krogh, PH., & Sørensen, JG. (2011). NORMA-Gene: a simple and robust method for qPCR normalization based on target gene data. *BMC Bioinformatics*, 12, 250. doi:10.1186/1471210512250

O'Connell, GC., Treadway, MB., Petrone, AB., Tennant, CS., Lucke-Wold, N, Chantler, PD., & Barr, TL. (2017). Leukocyte dynamics influence reference gene stability in whole blood: Data-driven qRT-PCR normalization is a robust alternative for measurement of transcriptional biomarkers. *Laboratory Medicine*, 3, 48. doi:10.1093/labmed/lmx035

O'Connell, GC., Treadway, MB., Petrone, AB., Tennant, CS., Lucke-Wold, N, Chantler, PD., & Barr, TL. (2017). Peripheral blood AKAP7 expression as an early marker for lymphocyte-mediated post-stroke blood brain barrier disruption. *Scientific Reports*, 1, 7. doi:10.1038/s41598017011785

See Also

norma_gene()
correction_factors()
relative_variability()
multi_cond_data
single_cond_data

---

correction_factors   *Retrieve scaling factors from NORMAgene output*

Description

Retrieves the per-replicate scaling factors used for normalization.

Usage

```
correction_factors(x)
```

Arguments

x   An object returned by [norma_gene()](#).

Value

A numeric vector of correction factors. If replicate identifiers were passed to [norma_gene()](#), the vector is named accordingly.

See Also

[norma_gene()](#)

Examples

```
# load example dataset containing raw CT values and
# metadata from a multi-condition experiment

data(multi_cond_data)
raw_data<-multi_cond_data

#normalize CT values via NORMAgene

norm_data<-norma_gene(
  data = raw_data,
  conditions = "Diagnosis",
  replicates= "Sample_id"
)

# retrieve scaling factors

correction_factors(norm_data)
```

---

multi_cond_data   *Example dataset from a multi-condition qPCR experiment.*

Description

A real-world qRT-PCR dataset containing raw CT values for 12 genes measured in whole blood total RNA originating from 29 human subjects diagnosed with ischemic stroke and 33 neurologically normal controls, as described in O'Connell et al. (2017).

Format

A data frame structured with biological replicates in rows, replicate identifiers in a single column, stroke diagnoses in a single column, and raw CT values for each of the 12 target genes in the remaining columns.

Details

This dataset is suitable for demonstrating or evaluating normalization workflows for use-cases involving data from multiple experimental conditions.

References

O'Connell, GC., Treadway, MB., Petrone, AB., Tennant, CS., Lucke-Wold, N, Chantler, PD., & Barr, TL. (2017). Leukocyte dynamics influence reference gene stability in whole blood: Data-driven qRT-PCR normalization is a robust alternative for measurement of transcriptional biomarkers. *Laboratory Medicine*, 3, 48. doi:10.1093/labmed/lmx035

Examples

```
#load example dataset

data(multi_cond_data)

#return dataset structure

str(multi_cond_data)
```

---

norma_gene   *Normalize CT values using NORMAgene*

Description

Applies the least squares fit-based NORMAgene data-driven normalization algorithm originally described by Heckmann et al. (2011) to raw CT values provided via an input data frame appended with experimental meta-data, and returns a data frame containing normalized CT values with scaling factors and relative variability metrics attached as attributes.

Usage

```
norma_gene(data, conditions = NULL, replicates = NULL, ct_values = NULL)
```

Arguments

data	A data frame structured with biological replicates in rows, and experimental metadata and gene-wise raw CT values in columns.
conditions	A single column name in data specifying experimental condition membership in the case of a multi-condition experiment, or NULL in the case of a single condition experiment. Normalization is applied within experimental conditions when specified, or across all replicates when NULL. This argument must be explicitly provided.

replicates   A single column name in `data` containing replicate identifiers, or `NULL` if repli-
                    cate identifiers are not present. If provided, replicate identifiers are used for
                    naming of outputs only, and are not used in normalization calculations. This
                    argument must be explicitly provided.

ct_values   Optional character vector specifying column names in `data` containing CT val-
                    ues to be normalized. If `NULL`, all numeric columns except `conditions` and
                    `replicates` are used.

Details

Users must explicitly specify how experimental conditions and replicate identifiers are handled to
avoid accidental normalization of numeric metadata. Scaling factors can be retrieved from the
output object using [correction_factors()](). Relative variability metrics can be retrieved from
the output object using [relative_variability()](). For more information on the NORMAgene
algorithm or relative variability metrics, see [NORMAgene-package]().

Value

A data frame with the same organization as `data` containing normalized CT values and any provided
experimental metadata. The per-replicate scaling factors used for normalization, as well as within
gene and within experimental condition relative variability metrics, are attached as attributes.

References

Heckmann, LH., Sørensen, PB., Krogh, PH., & Sørensen, JG. (2011). NORMA-Gene: a simple
and robust method for qPCR normalization based on target gene data. *BMC Bioinformatics*, 12,
250. [doi:10.1186/1471210512250]()

See Also

[correction_factors()]()
[relative_variability()]()
[NORMAgene-package]()

Examples

```
# USE-CASE WITH MULTIPLE EXPERIMENTAL CONDITIONS

# load example dataset containing raw CT values and
# metadata from a multi-condition experiment

data(multi_cond_data)
raw_data<-multi_cond_data

#normalize CT values via NORMAgene

norm_data<-norma_gene(
  data = raw_data,
  conditions = "Diagnosis",
  replicates= "Sample_id"
```

```
)

# retrieve relative variability metrics

relative_variability(norm_data, type = "by_gene")
relative_variability(norm_data, type = "by_condition")

# USE-CASE WITH a SINGLE EXPERIMENTAL CONDITION

# load example dataset containing raw CT values and
# metadata from a single-condition experiment

data(single_cond_data)
raw_data<-single_cond_data

#normalize CT values via NORMAgene

norm_data<-norma_gene(
  data = raw_data,
  conditions = NULL,
  replicates= "Sample_id"
)

# retrieve relative variability metrics

relative_variability(norm_data, type = "by_gene")
relative_variability(norm_data, type = "by_condition")
```

---

relative_variability   *Retrieve relative variability metrics from NORMAgene output*

Description

Retrieves relative variability metrics calculated during normalization.

Usage

```
relative_variability(x, type = c("by_gene", "by_condition"))
```

Arguments

x	An object returned by norma_gene().
type	Character string specifying which relative variability metric to return. One of "by_gene" or "by_condition".

Details

For more information on relative variability metrics, see NORMAgene-package.

Value

Depending on `type`:

by_gene A named numeric matrix of gene-level relative variability values, calculated both within experimental conditions and cumulatively across all experimental conditions.

by_condition A named numeric vector of relative variability calculated within experimental conditions, as well as cumulatively across all experimental conditions, regardless of gene.

See Also

[norma_gene()](#)
[NORMAgene-package](#)

Examples

```
# load example dataset containing raw CT values and
# metadata from a multi-condition experiment

data(multi_cond_data)
raw_data<-multi_cond_data

#normalize CT values via NORMAgene

norm_data<-norma_gene(
  data = raw_data,
  conditions = "Diagnosis",
  replicates= "Sample_id"
)

# retrieve relative variability metrics

relative_variability(norm_data, type = "by_gene")
relative_variability(norm_data, type = "by_condition")
```

---

single_cond_data   *Example dataset from a single condition qPCR experiment.*

Description

A real-world qRT-PCR dataset containing raw crossing threshold (CT) values for 12 genes measured in whole blood total RNA originating a single cohort of 26 human subjects diagnosed with ischemic stroke as described in O'Connell et al. (2017).

Format

A data frame structured with biological replicates in rows, replicate identifiers in a single column, and raw CT values for each of the 12 target genes in the remaining columns.

Details

This dataset is suitable for demonstrating or evaluating normalization workflows for use-cases involving data from a single experimental condition.

References

O'Connell, GC., Treadway, MB., Petrone, AB., Tennant, CS., Lucke-Wold, N, Chantler, PD., & Barr, TL. (2017). Peripheral blood AKAP7 expression as an early marker for lymphocyte-mediated post-stroke blood brain barrier disruption. *Scientific Reports*, 1, 7. doi:10.1038/s41598017011785

Examples

```
#load example dataset

data(single_cond_data)

#return dataset structure

str(single_cond_data)
```

Index