Package: SIEVEseq (via r-universe)

June 25, 2026

Title Unified Differential Expression, Variability and Skewness Analysis for RNA-Seq Data

Version 0.0.0

Date 2026-06-07

Maintainer Hongxiang Li <`hxli@ynnu.edu.cn`>

Description Provides a unified framework for the simultaneous testing of differential expression, variability, and skewness of genes using RNA-Seq data. The framework adopts a compositional data analysis approach for modelling RNA-Seq count data, applies the centered log-ratio transformation to obtain continuous variables, and uses the skew-normal distribution for statistical inference. Methods are described in Li and Khang (bioRxiv preprint, 2024, version 3) <doi:10.1101/2024.04.09.588804>.

Depends R (>= 4.1)

Imports sn, stats, ggplot2

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

URL `https://github.com/Divo-Lee/SIEVEseq`

License GPL (>= 2)

Encoding UTF-8

LazyData TRUE

Config/roxygen2/version 8.0.0

Config/testthat/edition 3

NeedsCompilation no

Author Hongxiang Li [aut, cre], Tsung Fei Khang [aut]

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

Date/Publication 2026-06-25 18:28:45 UTC

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

RemoteRef HEAD

RemoteSha c610429b4510de96c06dafb247cbf88841564a92

Contents

`clr.SN.fit` *Fit the skew-normal distribution to CLR-transformed RNA-Seq data*

Description

Estimate the mean, standard deviation, and skewness parameters of the skew-normal distribution using centered log-ratio (CLR) transformed RNA-Seq data.

Usage

```
clr.SN.fit(data)
```

Arguments

`data`	A table of CLR-transformed count data, with genes/transcripts on the rows and samples on columns.

Value

`mu`	The maximum likelihood estimate of the mean parameter.
`se.mu`	The standard error of the maximum likelihood estimate of `mu`.
`z.mu`	The Wald statistic for `mu`.
`p.mu`	The p-value of the Wald statistic for `mu`.
`sigma`	The maximum likelihood estimate of the standard deviation parameter.
`se.sigma`	The standard error of the maximum likelihood estimate of `sigma`.
`z.sigma`	The Wald statistic for `sigma`.
`p.sigma`	The p-value of the Wald statistic for `sigma`.
`gamma`	The maximum likelihood estimate of the skewness parameter.
`se.gamma`	The standard error of the maximum likelihood estimate of `gamma`.
`z.gamma`	The Wald statistic for `gamma`.
`p.gamma`	The p-value of the Wald statistic for `gamma`.

References

Azzalini, A. (1985). *A class of distributions which includes the normal ones*. *Scandinavian Journal of Statistics* **12**(2), 171–178, JSTOR

Azzalini, A. and Capitanio, A. (2014). *The Skew-Normal and Related Families*. Cambridge University Press, IMS Monographs series.

Azzalini, A. and Arellano-Valle, R. B. (2013). Maximum penalized likelihood estimation for skew-normal and skew-*t* distributions. *Journal of Statistical Planning and Inference* **143**, 419–433.

Azzalini, A. (2022). *The R package **sn**: The skew-normal and related distribution such as the skew-t and the SUN (version 2.0.2).* Universit\'a degli Studi di Padova, Italia. Home page: https://cran.r-project.org/package=sn.

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. Chapman & Hall, London.

Examples

```
library(SIEVEseq)
data(clrCounts1)
clr.SN.fit(clrCounts1[1:2, ])
clr.SN.fit(clrCounts1[1, ])
```

`clrCounts1` *Simulated CLR-transformed count table*

Description

A simulated CLR-transformed count table with genes on the rows and samples on the columns for one condition.

Usage

```
data(clrCounts1)
```

Format

A data frame containing CLR-transformed counts, with 200 genes and 500 samples.

`clrCounts2` *Simulated DV genes CLR-transformed count table*

Description

A simulated CLR-transformed count table with genes on the rows and samples on the columns for two groups. Ten percent of the genes are DV genes (gene1 to gene50).

Usage

```
data(clrCounts2)
```

Format

A data frame containing CLR-transformed counts, with 500 genes and 200 samples per group.

`clrCounts3` *Simulated DE genes CLR-transformed count table*

Description

A simulated CLR-transformed count table with genes on the rows and samples on the columns for two groups. Ten percent of the genes are DE genes (gene1 to gene50).

Usage

```
data(clrCounts3)
```

Format

A data frame containing CLR-transformed counts, with 500 genes and 200 samples per group.

`clrDE` *Differential Expression (DE) Test of CLR-transformed RNA-Seq Data*

Description

Model CLR-transformed RNA-Seq data using the skew-normal distribution, and then conduct a statistical test for finding genes/transcripts with differential expression using the Wald test.

Usage

```
clrDE(data = NULL, group = NULL)
```

Arguments

`data`	CLR-transformed counts matrix with genes/transcripts on the rows and samples on the columns
`group`	2 groups (control vs. treatment)

Value

An object of ‘`clrDE`’ class that contains the results of the DV test and associated information:

`DE`	The difference of `mu` between group 2 and group 1 (`mu2 - mu1`).
`se`	The standard error of `DE`.
`z`	The observed Wald statistic.
`pval`	The unadjusted p-value of the Wald test.
`adj_pval`	The p-value of the Wald test adjusted using the Benjamini-Yekutieli procedure.
`mu1`	The maximum likelihood estimate of the mean parameter of group 1.
`se.mu1`	The standard error of the maximum likelihood estimate of `mu1`.
`z.mu1`	The Wald statistic for `mu1`.
`p.mu1`	The p-value of the Wald test for `mu1`.
`mu2`	The maximum likelihood estimate of the mean parameter of group 2.
`se.mu2`	The standard error of the maximum likelihood estimate of `mu2`.
`z.mu2`	The Wald statistic for `mu2`.
`p.mu2`	The p-value of the Wald test for `mu2`.

See Also

`clrSIEVE()` for DE, DV and DS tests.

Examples

```
library(SIEVEseq)
data(clrCounts3) # The first 50 genes (gene1 to gene50) are DE genes
groups <- c(rep(0, 200), rep(1, 200))
clrDE_test <- clrDE(clrCounts3[46:100, ], group = groups)
sum(is.na(clrDE_test))  # check NA values
head(clrDE_test, 5) # adj_pval < 0.05, DE genes
tail(clrDE_test, 5) # adj_pval > 0.05, non-DE genes
```

`clrDV` *Differential variability (DV) test of CLR-transformed RNA-Seq data*

Description

Model CLR-transformed RNA-Seq data using the skew-normal distribution, and then conduct a statistical test for finding genes/transcripts with differential variability using the Wald test.

Usage

```
clrDV(data = NULL, group = NULL)
```

Arguments

`data`	A CLR-transformed count matrix with genes/transcripts on the rows and samples on the columns.
`group`	A vector specifying the group labels of the data.

Value

An object of '`clrDV`' class that contains the results of the DV test and associated information:

`DV`	The difference of standard deviation (`sigma`) between group 2 and group 1 (`sigma2` - `sigma1`).
`se`	The standard error of `DV`.
`z`	The observed Wald statistic.
`pval`	The unadjusted p-value of Wald test.
`adj_pval`	The p-value of the Wald test adjusted using the Benjamini-Yekutieli procedure.
`sigma1`	The maximum likelihood estimate of the standard deviation parameter for group 1.
`se.sigma1`	The standard error of the maximum likelihood estimate of `sigma1`.
`z.sigma1`	The Wald statistic for `sigma1`.
`p.sigma1`	The p-value of the Wald test for `sigma1`.
`sigma2`	The maximum likelihood estimate of the standard deviation parameter for group 2.

`se.sigma2`	The standard error of the maximum likelihood estimate of `gamma2`.
`z.sigma2`	The Wald statistic for `sigma2`.
`p.sigma2`	The p-value of the Wald test for `sigma2`.

See Also

`clrSIEVE()` for DE, DV and DS tests.

Examples

```
library(SIEVEseq)
data(clrCounts2) # first 50 genes (gene1 to gene50) are DV genes
groups <- c(rep(0, 200), rep(1, 200))
clrDV_test <- clrDV(clrCounts2[46:100, ], group = groups)
sum(is.na(clrDV_test))  # check NA values
head(clrDV_test, 5)  # adj_pval < 0.05, DV genes
tail(clrDV_test, 5)  # adj_pval > 0.05, non-DV genes
```

`clrSeq` *Fit the skew-normal model to CLR-transformed RNA-Seq data for 2 groups*

Description

Estimate the mean, standard deviation, and skewness parameters of the skew-normal distribution using centered log-ratio (CLR) transformed RNA-Seq data for 2 groups. The related computational works (standard error, Wald statistic, and p value) are also provided.

Usage

```
clrSeq(data = NULL, group = NULL)
```

Arguments

`data`	A table of CLR-transformed count data, with genes/transcripts on the rows and samples on the columns for 2 groups.
`group`	A vector specifying the group labels of the data.

Value

`mu1`	The maximum likelihood estimate of the mean parameter of group 1.
`se.mu1`	The standard error of the maximum likelihood estimate of `mu1`.
`z.mu1`	The Wald statistic for `mu1`.
`p.mu1`	The p-value of the Wald test for `mu1`.

`sigma1`	The maximum likelihood estimate of the standard deviation parameter of group 1.
`se.sigma1`	The standard error of the maximum likelihood estimate of `sigma1`.
`z.sigma1`	The Wald statistic for `sigma1`.
`p.sigma1`	The p-value of the Wald test for `sigma1`.
`gamma1`	The maximum likelihood estimate of the skewness parameter of group 1.
`se.gamma1`	The standard error of the maximum likelihood estimate of `gamma1`.
`z.gamma1`	The Wald statistic for `gamma1`.
`p.gamma1`	The p-value of the Wald test for `gamma1`.
`mu2`	The maximum likelihood estimate of mean parameter of group 2.
`se.mu2`	The standard error of the maximum likelihood estimate of `mu2`.
`z.mu2`	The Wald statistic for `mu2`.
`p.mu2`	The p-value of the Wald test for `mu2`.
`sigma2`	The maximum likelihood estimate of the standard deviation parameter of group 2.
`se.sigma2`	The standard error of the maximum likelihood estimate of `sigma2`.
`z.sigma2`	The Wald statistic for `sigma2`.
`p.sigma2`	The p-value of the Wald test for `sigma2`.
`gamma2`	The maximum likelihood estimate of the skewness parameter of group 2.
`se.gamma2`	The standard error of the maximum likelihood estimate of `gamma2`.
`z.gamma2`	The Wald statistic for `gamma2`.
`p.gamma2`	The p-value of the Wald test for `gamma2`.

References

Azzalini, A. (1985). *A class of distributions which includes the normal ones*. *Scandinavian Journal of Statistics* **12**(2), 171–178, JSTOR

Azzalini, A. and Capitanio, A. (2014). *The Skew-Normal and Related Families*. Cambridge University Press, IMS Monographs series.

Azzalini, A. and Arellano-Valle, R. B. (2013). Maximum penalized likelihood estimation for skew-normal and skew-*t* distributions. *Journal of Statistical Planning and Inference* **143**, 419–433.

Azzalini, A. (2022). *The R package* ***sn****: The skew-normal and related distribution such as the skew-t and the SUN (version 2.0.2)*. Universit\‘a degli Studi di Padova, Italia. Home page: `https://cran.r-project.org/package=sn`.

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. Chapman & Hall, London.

See Also

`clrSIEVE()` for DE, DV and DS tests.

Examples

```
library(SIEVEseq)
data("clrCounts2")
groups <- c(rep(0, 200), rep(1, 200))
clrSeq_result <- clrSeq(clrCounts2[46:100, ], group = groups)
tail(clrSeq_result, 5)
SN.plot(clrCounts2[1, 1:200])
clrSeq_result[1, c("mu1", "sigma1", "gamma1")]
```

`clrSIEVE` *Simultaneous Differential Expression, Variability and Skewness Analysis Using RNA-Seq Data*

Description

Model CLR-transformed RNA-Seq data using the skew-normal distribution, and then conduct statistical tests for finding genes/transcripts with differential expression, variability and skewness using the Wald test.

Usage

```
clrSIEVE(
  clrSeq_result = NULL,
  alpha_level = 0.05,
  order_DE = FALSE,
  order_LFC = FALSE,
  order_DS = FALSE,
  order_sieve = FALSE
)
```

Arguments

`clrSeq_result`	The result of clrSeq() function.
`alpha_level`	The adjusted p-value cutoff used for flagging genes that show significant differential expression, variability and skewness.
`order_DE`	Logical string. "FALSE" for no ordering; "TRUE" for ordering by the value of `DE`.
`order_LFC`	Logical string. "FALSE" for no ordering; "TRUE" for ordering by the value of `LFC`.
`order_DS`	Logical string. "FALSE" for no ordering; "TRUE" for ordering by the value of `DS`.
`order_sieve`	Character/logical string specifying the order method. Possibilities are "DE" for the value of `DE`, "LFC" for log2 fold change of variability, "DS" for the value of `DS` or "FALSE" for no ordering.

Value

`clrDE_test`, `clrDV_test` and `clrDS_test` contain the result of the DE, DV and DS tests, respectively. `clrSIEVE_tests` contains the result of all three tests.

`clrDE_test`	A data.frame contating the results of differential expression test.
`clrDV_test`	A data.frame contating the results of differential variability test.
`clrDS_test`	A data.frame contating the results of differential skewness test.
`clrSIEVE_tests`	A data.frame contating the results of differential expression, variability and skewness tests.
`DE`	The difference of mean (`mu`) between group 2 and group 1 (`mu2 - mu1`).
`se_DE`	The standard error of `DE`.
`z_DE`	The observed Wald statistic of `DE`.
`pval_DE`	The unadjusted p-value of Wald test of `DE`.
`adj_pval_DE`	The p-value of the Wald test of `DE`, adjusted using the Benjamini-Yekutieli procedure.
`mu1`	The maximum likelihood estimate of the mean parameter for group 1.
`mu2`	The maximum likelihood estimate of the mean parameter for group 2.
`de_indicator`	1: DE gene; 0: non-DE gene.
`SD_ratio`	The ratio of standard deviation (`sigma`) between group 2 and group 1 (`sigma2/sigma1`).
`LFC`	`log2` fold change: `log2(sigma2/sigma1)`.
`DV`	The difference of standard deviation (`sigma`) between group 2 and group 1 (`sigma2 - sigma1`).
`se_DV`	The standard error of `DV`.
`z_DV`	The observed Wald statistic of `DV`.
`pval_DV`	The unadjusted p-value of Wald test of `DV`.
`adj_pval_DV`	The p-value of the Wald test of `DV`, adjusted using the Benjamini-Yekutieli procedure.
`sigma1`	The maximum likelihood estimate of the standard deviation parameter for group 1.
`sigma2`	The maximum likelihood estimate of the standard deviation parameter for group 2.
`dv_indicator`	1: DV gene; 0: non-DV gene.
`DS`	The difference of skewness parameter (`gamma`) between group 2 and group 1 (`gamma2 - gamma1`.).
`se_DS`	The standard error of `DS`.
`z_DS`	The observed Wald statistic of `DS`.
`pval_DS`	The unadjusted p-value of Wald test of `DS`.
`adj_pval_DS`	The p-value of the Wald test of `DS` adjusted using the Benjamini-Yekutieli procedure.
`gamma1`	The maxiimum likelihood estimate of the skewness parameter for group 1.
`gamma2`	The maxiimum likelihood estimate of the skewness parameter for group 2.
`ds_indicator`	1: DS gene; 0: non-DS gene.

References

Azzalini, A. (1985). *A class of distributions which includes the normal ones*. *Scandinavian Journal of Statistics* **12**(2), 171–178, JSTOR

Azzalini, A. and Capitanio, A. (2014). *The Skew-Normal and Related Families*. Cambridge University Press, IMS Monographs series.

Azzalini, A. and Arellano-Valle, R. B. (2013). Maximum penalized likelihood estimation for skew-normal and skew-*t* distributions. *Journal of Statistical Planning and Inference* **143**, 419–433.

Azzalini, A. (2022). *The R package **sn**: The skew-normal and related distribution such as the skew-t and the SUN (version 2.0.2)*. Universit\‘a degli Studi di Padova, Italia. Home page: `https://cran.r-project.org/package=sn`.

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. Chapman & Hall, London.

See Also

`clrSeq()` for the results of parameters estimation; alternatively, `clrDE()` only provides DE test and `clrDV()` only provides DV test.

Examples

```
library(SIEVEseq)
data(clrCounts3) # first 50 genes (gene1 to gene50) are DE genes
groups <- c(rep(0, 200), rep(1, 200))
clrSeq_result3 <- clrSeq(clrCounts3[46:100, ], group = groups) # DE dataset
clrSIEVE_result3 <- clrSIEVE(clrSeq_result = clrSeq_result3,
                             alpha_level = 0.05,
                             order_DE = FALSE,
                             order_LFC = FALSE,
                             order_DS = FALSE,
                             order_sieve = FALSE)
clrDE_test3 <- clrSIEVE_result3$clrDE_test # DE test
head(clrDE_test3, 5)
clrDS_test3 <- clrSIEVE_result3$clrDS_test # DS test
clrDS_test3[clrDS_test3$adj_pval_DS < 0.05, ]
clrSIEVE_tests3 <- clrSIEVE_result3$clrSIEVE_tests # Sieve DE, DV and DS genes
head(clrSIEVE_tests3, 5)
tail(clrSIEVE_tests3, 5)
```

`SN.plot` *Distribution of CLR-transformed counts*

Description

Produce a histogram of observed CLR-transformed counts, with the fitted skew-normal probability density function for a particular gene/transcript.

Usage

```
SN.plot(data)
```

Arguments

`data`	CLR-transformed counts for a particular gene/transcript.

Value

No return value. This function is called for its side effect of generating a plot showing the observed CLR-transformed counts and the fitted skew-normal density.

Examples

```
library(SIEVEseq)
data("clrCounts1")
SN.plot(clrCounts1[1,])
SN.plot(clrCounts1[2,])
SN.plot(clrCounts1[3,])
```

`violin.plot.SIEVE` *Violin plots*

Description

Produce violin plots of CLR-transformed count data for two or three groups.

Usage

```
violin.plot.SIEVE(
  data = NULL,
  name.gene = NULL,
  group = NULL,
  group.names = NULL,
  xlab = "CLR-transformed count",
  ylab = "Condition"
)
```

Arguments

`data`	A CLR-transformed count table.
`name.gene`	Gene/transcript name.
`group`	A vector specifying the group labels of the data.
`group.names`	A vector specifying the group names.
`xlab`	Name of the x-axis.
`ylab`	Name of the y-axis.

Value

A ggplot object displaying the distribution of CLR-transformed expression values for the specified gene across groups.

Examples

```
library(SIEVEseq)
data(clrCounts2)  # first 50 genes (gene1 to gene50) are DV genes
data(clrCounts3)  # first 50 genes (gene1 to gene50) are DE genes
group0 <- c(rep(0, 200), rep(1, 200))
group1 <- c(rep(0, 200), rep(1, 100), rep(2, 100))
violin.plot.SIEVE(data = clrCounts2, "gene1", group = group0,
              group.names = c("control", "case")) # DV
violin.plot.SIEVE(data = clrCounts2, "gene1", group = group1,
              group.names = c("control", "case1", "case2")) # DV
violin.plot.SIEVE(data = clrCounts3, "gene1", group = group0,
              group.names = c("control", "case")) # DE
violin.plot.SIEVE(data = clrCounts3, "gene2", group = group0,
              group.names = c("control", "case")) # DE
violin.plot.SIEVE(data = clrCounts3, "gene200", group = group0,
              group.names = c("control", "case")) # non-DE
```

Index