# Package 'spqn'

February 2, 2026

**Title** Spatial quantile normalization

**Version** 1.23.0

**Description** The spqn package implements spatial quantile normalization (SpQN). This method was developed to remove a mean-correlation relationship in correlation matrices built from gene expression data. It can serve as pre-processing step prior to a co-expression analysis.

**License** Artistic-2.0

**Imports** graphics, stats, utils, matrixStats

**Depends** R (>= 4.0), ggplot2, ggridges, SummarizedExperiment, BiocGenerics

**Suggests** BiocStyle, knitr, rmarkdown, tools, spqnData (>= 0.99.3), RUnit

**VignetteBuilder** knitr

**URL** <https://github.com/hansenlab/spqn>

**BugReports** <https://github.com/hansenlab/spqn/issues>

**biocViews** NetworkInference, GraphAndNetwork, Normalization

**git_url** https://git.bioconductor.org/packages/spqn

**git_branch** devel

**git_last_commit** 86a8920

**git_last_commit_date** 2025-10-29

**Repository** Bioconductor 3.23

**Date/Publication** 2026-02-01

**Author** Yi Wang [cre, aut],
Kasper Daniel Hansen [aut]

**Maintainer** Yi Wang <yiwangthu5@gmail.com>

# Contents

---

spqn-package                    *Spatial quantile normalization*

---

## Description

The spqn package implements spatial quantile normalization (SpQN). This method was developed to remove a mean-correlation relationship in correlation matrices built from gene expression data. It can serve as pre-processing step prior to a co-expression analysis.

## Details

See references for details on spatial quantile normalization.

The main function is normalize_correlation. We include a number of plotting functions for examining the mean-correlation relationship, see the vignette for examples.

## References

Y Wang, SC Hicks, KD Hansen (2020). *Co-expression analysis is biased by a mean-correlation relationship.* bioRxiv 2020.02.13.944777. doi:10.1101/2020.02.13.944777

---

normalize_correlation   *Spatial quantile normalization (SpQN)*

---

## Description

This method was developed to remove a mean-correlation relationship in correlation matrices built from gene expression data. It can serve as pre-processing step prior to a co-expression analysis.

## Usage

```
normalize_correlation(cor_mat, ave_exp, ngrp, size_grp, ref_grp)
```

## Arguments

| | |
|---|---|
| cor_mat | A (square and symmetrix) correlation matrix. |
| ave_exp | A vector of expression levels, same length as the number of rows of the correlation matrix in cor_mat. For other types of data, ave_exp can be the vector corresponding to the row/column of the correlation matrix, whose dependency with the distribution of correlations need to be removed. |
| ngrp | Number of bins in each row/column to be used to partition the correlation matrix, integer. |
| size_grp | Size of the outer bins to be used to appriximate the distribution of the inner bins, in order to smooth the normalization. Note that the product of size_grp and ngrp must be equal or larger than than the row/column number of cor_mat, and there is no smoothness in the normalization when they are equal. |
| ref_grp | Location of the reference bin on the diagonal, whose distribution will be used as target distribution in the normalization, an integer. |

## Value

A normalized correlation matrix.

## Examples

```
if(require(spqnData)){
  data(gtex.4k)
  cor_ori <- cor(t(assay(gtex.4k)))
  ave_logrpkm <- rowData(gtex.4k)$ave_logrpkm
  normalize_correlation(cor_ori, ave_exp = ave_logrpkm,
                        ngrp=10, size_grp=15, ref_grp=9)}
```

---

plot_IQR_condition_exp

*Get and plot the IQRs of submatrices of the correlation matrix.*

---

## Description

The get_IQR_condition_exp function computes the IQRs of a set of 10 by 10 same-size bins that partition the correlation matrix, ordered according to expression level.

The plot_IQR_condition_exp function plots the IQR for each bin among a set of 10 by 10 same-size bins that partition the correlation matrix, with IQR denoted by the width of boxes in the plot.

## Usage

```
get_IQR_condition_exp(cor_mat, ave_exp)
plot_IQR_condition_exp(IQR_list)
```

## Arguments

| | |
|---|---|
| `cor_mat` | correlation matrix, generated by gene expression matrix, with genes sorted by average expression levels. |
| `ave_exp` | vector, average expression level of each gene for the normalized gene expression matrix. |
| `IQR_list` | List, output of `get_IQR_condition_exp`. |

## Value

A plot with boxes that shows the IQR of each bin

## Note

The mnemonic for `condition_exp` is 'conditional on expression'.

## Examples

```
if(require(spqnData)) {
    data(gtex.4k)
    cor_mat <- cor(t(assay(gtex.4k)))
    ave_logrpkm <- rowData(gtex.4k)$ave_logrpkm
    IQR_list <- get_IQR_condition_exp(cor_mat, ave_exp = ave_logrpkm)
    plot_IQR_condition_exp(IQR_list)
    }
```

---

plot_signal_condition_exp

*Plot the signal and background distribution of a correlation matrix.*

---

## Description

This function allows users to visualize the distributions of (assumed) signal and background, conditional on expression levels. The predicted signals are defined by the 0.1% highest correlations in each bin.

## Usage

```
plot_signal_condition_exp(cor_mat, ave_exp, signal)
```

## Arguments

| | |
|---|---|
| `cor_mat` | Matrix, correlation matrix, generated by gene expression matrix |
| `ave_exp` | Vector, average expression level of each gene for the normalized expression matrix |
| `signal` | a value between 0 and 1 giving the fraction of correlations which should be considered signal. We often use a value of `0.001`. |

## Value

Invoked for the side effect of producing a plot.

## Note

The mnemonic for `condition_exp` is 'conditional on expression'.

## Examples

```
if(require(spqnData)) {
  data(gtex.4k)
  cor_mat <- cor(t(assay(gtex.4k)))
  ave_logrpkm <- rowData(gtex.4k)$ave_logrpkm
  plot_signal_condition_exp(cor_mat, ave_exp=ave_logrpkm, signal=0.05)}
```

---

| qqplot_condition_exp | *Q-Q plot for examing the distributions across submatrices of a correlation matrix.* |
|---|---|

---

## Description

We partition the correlation matrix into 10x10 bins of equal size, with genes ordered according to expression level. As reference bin, we choose the (9,9) bin (ie. the almost-highest expressed genes). We then make a QQ-plot of the (i,j)'th submatrix vs. the (9,9) submatrix. See the SpQN paper for detail on these choices.

## Usage

```
qqplot_condition_exp(cor_mat,ave_exp, i,j)
```

## Arguments

| | |
|---|---|
| cor_mat | Matrix, correlation matrix, generated by gene expression matrix. |
| ave_exp | Vector, average expression level of each gene for the normalized expression matrix. |
| i | Integer, row number of the submatrix (see details). |
| j | Integer, column number of the submatrix (see details). |

## Value

Invoked for the side effect of producing a plot.

## Note

The mnemonic for `condition_exp` is 'conditional on expression'.

**Examples**

```
if(require(spqnData)) {
  data(gtex.4k)
  cor_mat <- cor(t(assay(gtex.4k)))
  ave_logrpkm <- rowData(gtex.4k)$ave_logrpkm
  qqplot_condition_exp(cor_mat, ave_exp=ave_logrpkm, 1, 1)
}
```

# Index