

# Package ‘SQN’

October 12, 2022

**Type** Package

**Title** Subset Quantile Normalization

**Version** 1.0.6

**Date** 2022-06-10

**Author** Zhijin(Jean) Wu, Martin Aryee

**Maintainer** Martin Aryee <martin.aryee@gmail.com>

**Depends** R (>= 2.6.0), mclust(>= 3.2), nor1mix(>= 1.0-7)

**Description** Normalization based a subset of negative control probes as described in 'Subset quantile normalization using negative control features'. Wu Z, Aryee MJ, J Comput Biol. 2010 Oct;17(10):1385-95 [PMID 20976876].

**License** LGPL (>= 2.0)

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2022-06-10 16:00:02 UTC

## R topics documented:

|                    |   |
|--------------------|---|
| SQN . . . . .      | 1 |
| sqnData0 . . . . . | 2 |

|              |          |
|--------------|----------|
| <b>Index</b> | <b>4</b> |
|--------------|----------|

---

|     |                                      |
|-----|--------------------------------------|
| SQN | <i>subset quantile normalization</i> |
|-----|--------------------------------------|

---

## Description

This function performs normalization based on a subset of negative controls whose distribution is expected to be unchanged in various samples. There is no restriction on the behavior of the rest of the measurements.

**Usage**

```
SQN(y, N.mix = 5, ctrl.id, model.weight = 0.9)
```

**Arguments**

|                           |   |
|---------------------------|---|
| <code>y</code>            | A matrix of unnormalized data.  |
| <code>N.mix</code>        | Number of normal distributions in the mixture approximation.          |
| <code>ctrl.id</code>      | index of controls. Must be a vector smaller than <code>nrow(y)</code> |
| <code>model.weight</code> | weight given to the parametric normal mixture model                   |

**Value**

A matrix of normalized data

**Author(s)**

Zhijin Wu

**References**

Wu Z and Aryee M. Subset Quantile Normalization using Negative Control Features (2010) Journal of Computational Biology, 17(10)

**Examples**

```
require(mclust)
require(nor1mix)
data(sqnData0)
Ynorm=SQN(sqnData0,ctrl.id=1:1000) #after normalization
par(mfrow=c(1,2))
  boxplot(sqnData0,main="before normalization")
  boxplot(sqnData0[1:1000,],add=TRUE,col=3,boxwex=.4)

  boxplot(Ynorm,main="after normalization")
  boxplot(Ynorm[1:1000,],add=TRUE,col=3,boxwex=.4)
  legend(.5,11,legend=c("probes for signal","negative control probes"),text.col=c(1,3),bg="white")
```

---

`sqnData0`

*example data*

---

**Description**

Simulated data with two samples, each with 1000 negative controls and 5000 signal bearing probes

**Usage**

```
data(sqnData0)
```

*sqnData0*

3

**Format**

A matrix with two columns

# Index

\* **datasets**

sqnData0, [2](#)

mix.qn (SQN), [1](#)

SQN, [1](#)

sqnData0, [2](#)