

# Package ‘PreProcessing’

July 21, 2025

**Type** Package

**Title** Various Preprocessing Transformations of Numeric Data Matrices

**Date** 2021-06-16

**Version** 0.1.0

**Author** Swamiji Pravedson

**Maintainer** Swamiji Pravedson <swamiji.pravedson@gmail.com>

**Description** Preprocess numeric data matrices into desired transformed representations.  
Standardization, Unitization, Cubitization and adaptive intervals are offered.

**License** GPL-3

**Encoding** UTF-8

**RoxygenNote** 7.1.1

**Imports** stats (>= 1.0.1), ggplot2 (>= 1.0.1)

**Suggests** knitr

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2021-06-24 12:00:02 UTC

## Contents

|                       |          |
|-----------------------|----------|
| cubitize . . . . .    | 2        |
| intervalize . . . . . | 2        |
| standardize . . . . . | 3        |
| unitize . . . . .     | 4        |
| <b>Index</b>          | <b>6</b> |

---

|          |                                            |
|----------|--------------------------------------------|
| cubitize | <i>Cubitizes the matrix given as input</i> |
|----------|--------------------------------------------|

---

**Description**

Cubitizes the matrix given as input

**Usage**

```
cubitize(xx)
```

**Arguments**

xx                      Matrix or a data frame of numeric entries

**Value**

Matrix with columns that have minimum zero and maximum one

**Examples**

```
## Not run:
# I don't want you to run this

## End(Not run)
n<-450; x <- data.frame(cbind(rnorm(n, 162, 4), rnorm(n, 108, 2),
rnorm(n, 117, 3), rnorm(n, 36, 2), rnorm(n, 45, 2)))
p <- ncol(x)
x.cube <- cubitize(x)
round(head(x),2)
round(head(x.cube),2)
round(rbind(apply(x, 2, min), apply(x.cube, 2, min)),2)
round(rbind(apply(x, 2, max),apply(x.cube, 2, max)),2)
oldpar<-par(mfrow=c(1,2))
boxplot(x[,1:min(5,p)], main='Original Data', col=rainbow(9))
boxplot(x.cube[,1:min(5,p)], main='PreProcessed Data', col=rainbow(7))
par(oldpar)
```

---

|             |                                               |
|-------------|-----------------------------------------------|
| intervalize | <i>Intervalizes the matrix given as input</i> |
|-------------|-----------------------------------------------|

---

**Description**

Intervalizes the matrix given as input

**Usage**

```
intervalize(xx, a = -1, b = 1)
```

**Arguments**

|    |                                           |
|----|-------------------------------------------|
| xx | Matrix or a data frame of numeric entries |
| a  | lower bound of the target interval        |
| b  | upper bound of the target interval        |

**Value**

Matrix with columns that have minimum zero and maximum one

**Examples**

```
## Not run:
# I don't want you to run this

## End(Not run)
n<-450; x <- data.frame(cbind(rnorm(n, 162, 4), rnorm(n, 108, 2),
rnorm(n, 117, 3), rnorm(n, 36, 2), rnorm(n, 45, 2)))
p <- ncol(x)
x.inter <- intervalize(x,a=-1,b=1)
round(head(x),2)
round(head(x.inter),2)
round(rbind(apply(x, 2, min), apply(x.inter, 2, min)),2)
round(rbind(apply(x, 2, max), apply(x.inter, 2, max)),2)
oldpar<-par(mfrow=c(1,2))
boxplot(x[,1:min(5,p)], main='Original Data', col=rainbow(9))
boxplot(x.inter[,1:min(5,p)], main='PreProcessed Data', col=rainbow(7))
par(oldpar)
```

---

standardize

*Standardizes the matrix given as input*


---

**Description**

This function takes as input a matrix of numeric values and then transforms it so that each column has a mean of zero and a variance of one

**Usage**

```
standardize(xx)
```

**Arguments**

|    |                                           |
|----|-------------------------------------------|
| xx | Matrix or a data frame of numeric entries |
|----|-------------------------------------------|

**Value**

Matrix with columns that have mean zero and variance one

**Examples**

```
## Not run:
# I don't want you to run this

## End(Not run)
n<-450; x <- data.frame(cbind(rnorm(n, 162, 4), rnorm(n, 108, 2),
rnorm(n, 117, 3), rnorm(n, 36, 2), rnorm(n, 45, 2)))
p <- ncol(x)
x.stan <- standardize(x)
round(head(x),2)
round(head(x.stan),2)
round(rbind(apply(x, 2, mean), apply(x.stan, 2, mean)),2)
round(rbind(apply(x, 2, sd),apply(x.stan, 2, sd)),2)

oldpar <- par(mfrow=c(1,2))
boxplot(x[,1:min(5,p)], main='Original Data', col=rainbow(9))
boxplot(x.stan[,1:min(5,p)], main='PreProcessed Data', col=rainbow(7))
par(oldpar)
```

unitize

*Unitizes the matrix given as input***Description**

Unitizes the matrix given as input

**Usage**

```
unitize(xx)
```

**Arguments**

xx                      Matrix or a data frame of numeric entries

**Value**

Matrix with columns that have mean zero and length one

**Examples**

```
## Not run:
# I don't want you to run this

## End(Not run)
n<-450; x <- data.frame(cbind(rnorm(n, 162, 4), rnorm(n, 108, 2),
rnorm(n, 117, 3), rnorm(n, 36, 2), rnorm(n, 45, 2)))
p <- ncol(x)
x.unit <- unitize(x)
round(head(x),2)
```

```
round(head(x.unit),2)
round(rbind(apply(x, 2, mean), apply(x.unit, 2, mean)),2)
round(rbind(apply(x, 2, sd),apply(x.unit, 2, sd)),2)
oldpar<-par(mfrow=c(1,2))
boxplot(x[,1:min(5,p)], main='Original Data', col=rainbow(9))
boxplot(x.unit[,1:min(5,p)], main='PreProcessed Data', col=rainbow(7))
par(oldpar)
```

# Index

cubitize, [2](#)

intervalize, [2](#)

standardize, [3](#)

unitize, [4](#)