

# Package ‘normalr’

October 13, 2022

**Title** Normalisation of Multiple Variables in Large-Scale Datasets

**Version** 1.0.0

**Description** The robustness of many of the statistical techniques, such as factor analysis, applied in the social sciences rests upon the assumption of item-level normality. However, when dealing with real data, these assumptions are often not met. The Box-Cox transformation (Box & Cox, 1964)  [<http://www.jstor.org/stable/2984418>](http://www.jstor.org/stable/2984418) provides an optimal transformation for non-normal variables. Yet, for large datasets of continuous variables, its application in current software programs is cumbersome with analysts having to take several steps to normalise each variable. We present an R package 'normalr' that enables researchers to make convenient optimal transformations of multiple variables in datasets. This R package enables users to quickly and accurately: (1) anchor all of their variables at 1.00, (2) select the desired precision with which the optimal lambda is estimated, (3) apply each unique exponent to its variable, (4) rescale resultant values to within their original X1 and X(n) ranges, and (5) provide original and transformed estimates of skewness, kurtosis, and other inferential assessments of normality.

**Depends** R (>= 3.3.0)

**License** GPL

**Encoding** UTF-8

**LazyData** true

**Imports** MASS, parallel, purrr, magrittr, rlang, shiny

**Suggests** testthat, covr

**RoxygenNote** 6.0.1

**URL** <https://github.com/kcha193/normalr>

**BugReports** <https://github.com/kcha193/normalr/issues>

**NeedsCompilation** no

**Author** Kevin Chang [aut, cre],  
Matthew Courtney [aut]

**Maintainer** Kevin Chang <k.chang@auckland.ac.nz>

**Repository** CRAN

**Date/Publication** 2018-03-30 03:20:03 UTC

## R topics documented:

getLambda	2
normalise	3
normaliseData	3
normalrShiny	4
testData	4
<b>Index</b>	<b>5</b>

---

getLambda	<i>Get Optimal Lambda value</i>
-----------	---------------------------------

---

### Description

Computes optimal lambda value using [boxcox](#) function from the provided data.

### Usage

```
getLambda(dat, lambda = seq(-10, 10, 1/100), parallel = TRUE)
```

### Arguments

dat	a data frame containing the variables of numeric or integer vectors.
lambda	a vector of values of lambda – default (-10, 10) in steps of 0.01.
parallel	perform the computation in parallel, default setting is TRUE.

### Value

a numeric vector

### References

Courtney, M. G. R., and Chang, K. C. (2018) Dealing with non-normality: an introduction and step-by-step guide using R. TEST, doi: 10.1111/test.12154.

### Examples

```
## Not run: getLambda(mtcars)
getLambda(mtcars, parallel = FALSE)
```

---

normalise	<i>Apply normalisation on a numeric vector using a specific Lambda value</i>
-----------	--

---

**Description**

Apply normalisation on a numeric vector using a specific Lambda value

**Usage**

```
normalise(x, lambda = 3)
```

**Arguments**

x	a numeric vector to be normalised.
lambda	a numeric vector from the <a href="#">boxcox</a> function

**Value**

a numeric vector

**Examples**

```
x <- c(1, 5, 9, 9, 9, 9, 10, 10, 10, 11, 11, 12)
normalise(x, lambda = 3)
```

---

normaliseData	<i>Apply normalisation on a data frame using specific Lambda value</i>
---------------	--

---

**Description**

Apply normalisation on a data frame using specific Lambda value

**Usage**

```
normaliseData(dat, lambdas)
```

**Arguments**

dat	a data frame containing the variables.
lambdas	a numeric vector from the <a href="#">boxcox</a> function

**Value**

a data frame

**Examples**

```
## Not run: normaliseData(mtcars, getLambda(mtcars, parallel = FALSE))
normaliseData(mtcars, getLambda(mtcars, parallel = FALSE))
```

---

normalrShiny	<i>Shiny application of the normalr</i>
--------------	---

---

**Description**

Shiny application of the normalr

**Usage**

```
normalrShiny(example = "normalr")
```

**Arguments**

example            name of the shiny apps

**Examples**

```
## Not run: normalrShiny()
```

---

testData	<i>Test dataset for the paper</i>
----------	-----------------------------------

---

**Description**

Test dataset for the paper

**Usage**

```
testData
```

**Format**

An object of class `data.frame` with 957 rows and 9 columns.

# Index

\* **datasets**

testData, 4

boxcox, 2, 3

getLambda, 2

normalise, 3

normaliseData, 3

normalrShiny, 4

testData, 4