

# Package ‘distrTeach’

April 23, 2016

**Version** 2.6.1

**Date** 2016-04-23

**Title** Extensions of Package 'distr' for Teaching  
Stochastics/Statistics in Secondary School

**Description** Provides flexible examples of LLN and CLT for teaching purposes in secondary school.

**Depends** R(>= 2.2.0), methods, distr(>= 2.2), distrEx(>= 2.2)

**Suggests** tcltk

**Imports** startupmsg, grDevices, graphics, stats

**ByteCompile** yes

**License** LGPL-3

**Encoding** latin1

**URL** <http://distr.r-forge.r-project.org/>

**LastChangedDate** {`$LastChangedDate`: 2016-04-23 14:53:23 +0200 (Sa, 23 Apr 2016) `$`}

**LastChangedRevision** {`$LastChangedRevision`: 1094 `$`}

**SVNRevision** 1094

**NeedsCompilation** no

**Author** Eleonora Feist [ctb] (contributed as student in the initial phase --2008),  
Matthias Kohl [aut, cph],  
Peter Ruckdeschel [cre, cph],  
Anja Hueller [ctb] (contributed as student in the initial phase --2008)

**Maintainer** Peter Ruckdeschel <peter.ruckdeschel@uni-oldenburg.de>

**Repository** CRAN

**Date/Publication** 2016-04-23 16:40:18

## R topics documented:

distrTeach-package	2
illustrateCLT	4
illustrateLLN	5

<b>Index</b>	<b>8</b>
--------------	----------

---

distrTeach-package      *distrTeach – Teaching Extensions of Package distr*

---

### Description

**distrTeach** provides some illustrations based on package **distr** for teaching Stochastics / Statistics in secondary school; so far the following has been implemented

- `illustrateLLT`: function for the generation of LLN - visualizations
- `illustrateCLT`: function for the generation of CLT - visualizations
- `plotCLT`: Generic function for the plotting of CLT-approximations

as well as a Tcl/Tk based demo for `illustrateCLT`

### Details

Package:	distrTeach
Version:	2.6
Date:	2016-04-23
Depends:	R(>= 2.2.0), methods, distr(>= 2.2), distrEx(>= 2.2)
Suggests:	tcltk
Imports:	startupmsg, grDevices, graphics, stats
LazyLoad:	yes
License:	LGPL-3
URL:	<a href="http://distr.r-forge.r-project.org/">http://distr.r-forge.r-project.org/</a>
SVNRevision:	1091

### Classes

Teaching Classes

### Methods

illustration:	
illustrateLLT	function for the generation of LLN - visualizations
illustrateCLT	function for the generation of CLT - visualizations
plotCLT	Generic function for the plotting of CLT-approximations

## Demos

Demos are available — see `demo(package="distrTeach")`.

## Start-up-Banner

You may suppress the start-up banner/message completely by setting `options("StartupBanner"="off")` somewhere before loading this package by `library` or `require` in your R-code / R-session.

If option "StartupBanner" is not defined (default) or setting `options("StartupBanner"=NULL)` or `options("StartupBanner"="complete")` the complete start-up banner is displayed.

For any other value of option "StartupBanner" (i.e., not in `c(NULL, "off", "complete")`) only the version information is displayed.

The same can be achieved by wrapping the `library` or `require` call into either `suppressStartupMessages()` or `onlytypeStartupMessages(., atypes="version")`.

As for general packageStartupMessage's, you may also suppress all the start-up banner by wrapping the `library` or `require` call into `suppressPackageStartupMessages()` from **startupmsg**-version 0.5 on.

## Package versions

Note: The first two numbers of package versions do not necessarily reflect package-individual development, but rather are chosen for the `distrXXX` family as a whole in order to ease updating "depends" information.

## Author(s)

Matthias Kohl <Matthias.Kohl@stamats.de> and  
 Peter Ruckdeschel <peter.ruckdeschel@uni-oldenburg.de>,  
 Eleonara Feist <eleonoragerber@gmx.de>, and,  
 Anja Hueller <anja\\_h86@web.de>

*Maintainer:* Peter Ruckdeschel <peter.ruckdeschel@uni-oldenburg.de>

## References

P. Ruckdeschel, M. Kohl, T. Stabla, F. Camphausen (2006): S4 Classes for Distributions, *R News*, 6(2), 2-6. [http://CRAN.R-project.org/doc/Rnews/Rnews\\_2006-2.pdf](http://CRAN.R-project.org/doc/Rnews/Rnews_2006-2.pdf)

a vignette for packages **distr**, **distrSim**, **distrTEst**, and **distrTeach** is included into the mere documentation package **distrDoc** and may be called by `require("distrDoc");vignette("distr")`

a homepage to this package is available under <http://distr.r-forge.r-project.org/> and the pages ...

M. Kohl (2005): *Numerical Contributions to the Asymptotic Theory of Robustness*. PhD Thesis. Bayreuth. Available as <http://www.stamats.de/ThesisMKohl.pdf>

### See Also

[distr-package](#) [distrEx-package](#)

---

illustrateCLT

*Functions for Illustrating the CLT*

---

### Description

Functions for generating a sequence of plots of the density and cdf of the consecutive standardized and centered sums of iid r.v. distributed according to a prescribed discrete or absolutely continuous distribution compared to the standard normal — uses the generic function plotCLT.

### Usage

```
illustrateCLT(Distr, len, sleep = 0)
illustrateCLT.tcl(Distr, k, Distrname)
```

### Arguments

Distr	object of class "AbscontDistribution", "LatticeDistribution" or "DiscreteDistribution": distribution of the summands
len	integer: up to which number of summands plots are generated
k	integer: number of summands for which a plot is to be generated
Distrname	character: name of the summand distribution to be used as title in the plot
sleep	numeric: pause in seconds between subsequent plots

### Details

illustrateCLT generates a sequence of plots, while illustrateCLT.tcl may be used with Tcl/Tk-widgets as in demo illustCLT\_tcl.R.

### Value

void

### Author(s)

Matthias Kohl <Matthias.Kohl@stamats.de>  
Peter Ruckdeschel <peter.ruckdeschel@uni-oldenburg.de>

### References

Kohl, M., Ruckdeschel, P., (2014): General purpose convolution algorithm for distributions in S4-Classes by means of FFT. *J. Statist. Softw.* **59**(4): 1-25.

**See Also**[plotCLT](#)**Examples**

```

distributions("DefaultNrFFTGridPointsExponent" = 13)
illustrateCLT(Distr = Unif(), len = 20)
distributions("DefaultNrFFTGridPointsExponent" = 12)
illustrateCLT(Distr = Pois(lambda = 2), len = 20)
distributions("DefaultNrFFTGridPointsExponent" = 13)
illustrateCLT(Distr = Pois(lambda = 2)+Unif(), len = 20)
illustrateCLT.tcl(Distr = Unif(), k = 4, "Unif()")

```

illustrateLLN

*Functions for Illustrating the LLN***Description**

Functions for generating a sequence of plots of randomly generated replicates of  $\bar{X}_n = \frac{1}{n} \sum_{i=1}^n X_i$  for sums of iid r.v. distributed according to a prescribed discrete or absolutely continuous distribution. A line for the expectation and CLT based (pointwise) 95%-confidence bands are also plotted and the empirical coverage of this band by the replicated plotted so far is indicated.

**Usage**

```

illustrateLLN(Distr = Norm(), n = c(1,3,5,10,25,50,100,500,1000,10000),
             m = 50, step = 1, sleep = 0, withConf = TRUE,
             withCover = (length(n)<=12), withEline = TRUE, withLegend = TRUE,
             CLTorCheb = "CLT", coverage = 0.95, ..., col.Eline = "blue",
             lwd.Eline = par("lwd"), lty.Eline = par("lty"), col.Conf = "red",
             lwd.Conf = par("lwd"), lty.Conf = 2, cex.Cover = 0.7,
             cex.legend = 0.8)

```

**Arguments**

Distr	object of class "UnivariateDistribution": distribution of the summands
n	vector of integers: sample sizes to be considered
m	integer: (total) number of replicates to be plotted subsequently
step	integer: number of replicates to be drawn at once
sleep	numeric: pause in seconds between subsequent plots
withEline	logical: shall a line for the limiting expectation (in case of class Cauchy instead: median) be drawn?
withConf	logical: shall (CLT-based) confidence bands be plotted?
withCover	logical: shall empirical coverage of (CLT-based) confidence bands be printed?
withLegend	logical: shall a legend be included?

CLTorCheb	character: type of confidence interval —"CLT" or "Chebyshev"; partial matching is used; if this fails "CLT" is used.
coverage	numerical: nominal coverage of the confidence bands —to be in (0,1)
col.Eline	character or integer code; color for confidence bands
lwd.Eline	integer code (see <a href="#">par</a> ); line width of the confidence bands
lty.Eline	integer code (see <a href="#">par</a> ); line type of the confidence bands
col.Conf	character or integer code; color for confidence bands
lwd.Conf	integer code (see <a href="#">par</a> ); line width of the confidence bands
lty.Conf	integer code (see <a href="#">par</a> ); line type of the confidence bands
cex.Cover	magnification w.r.t. the current setting of cex to be used for empirical coverages; as in <a href="#">par</a>
cex.legend	magnification w.r.t. the current setting of cex to be used for the legend as in <a href="#">par</a>
...	further arguments to be passed to <code>matplot</code> , <code>matlines</code> , <code>abline</code>

### Details

`illustrateLLN` generates a sequence of plots. Any parameters of `plot.default` may be passed on to this particular plot method.

There are default main titles as well as `xlab` and `ylab` annotations.

In all title arguments, the following patterns are substituted:

"%C" class of argument `x`

"%P" parameters of `x` in form of a comma-separated list of `<value>`'s coerced to character

"%Q" parameters of `x` in form of a comma-separated list of `<value>`'s coerced to character and in parenthesis — unless empty; then ""

"%N" parameters of `x` in form of a comma-separated list `<name> = <value>` coerced to character

"%A" deparsed argument `x`

"%D" time/date-string when the plot was generated

"%X" the expression  $\bar{X}_n = \sum_{i=1}^n X_i/n$

If not explicitly set, `col.Eline`, `col.Conf` are set to `col` if this arg is given and else to their default values as given above. Similarly for `cex`, `lwd` and `lty`.

### Value

void

### Author(s)

Peter Ruckdeschel <[peter.ruckdeschel@uni-oldenburg.de](mailto:peter.ruckdeschel@uni-oldenburg.de)>

**Examples**

```
illustrateLLN(Distr = Unif())
illustrateLLN(Distr = Pois(lambda = 2))
illustrateLLN(Distr = Pois(lambda = 2)+Unif())
illustrateLLN(Td(3), m = 50, col.Eline = "green", lwd = 2, cex = 0.6, main =
  "My LLN %C%Q", sub = "generated %D")
illustrateLLN(Td(3), m = 50, CLTorCheb = "Chebyshev")
illustrateLLN(Td(3), m = 50, CLTorCheb = "Chebyshev", coverage = 0.75)
```

# Index

\*Topic **distribution**

illustrateCLT, [4](#)

illustrateLLN, [5](#)

\*Topic **dynamic**

illustrateCLT, [4](#)

illustrateLLN, [5](#)

\*Topic **methods**

illustrateCLT, [4](#)

illustrateLLN, [5](#)

\*Topic **package**

distrTeach-package, [2](#)

distrTeach (distrTeach-package), [2](#)

distrTeach-package, [2](#)

illustrateCLT, [4](#)

illustrateLLN, [5](#)

par, [6](#)

plotCLT, [5](#)