Package: clttools (via r-universe)

September 17, 2024

Type Package

Title Central Limit Theorem Experiments (Theoretical and Simulation)

Version 1.3

Date 2016-02-18

Author Simiao Ye, Jingning Mei

Maintainer Simiao Ye <simiao.ye@gmail.com>

Description Central limit theorem experiments presented by data frames or plots. Functions include generating theoretical sample space, corresponding probability, and simulated results as well.

License GPL-2

LazyData true

NeedsCompilation no

Date/Publication 2016-02-19 08:49:57

Repository https://yesimiao.r-universe.dev

RemoteUrl https://github.com/cran/clttools

RemoteRef HEAD

RemoteSha 1c57ad403228168c8257c0de33f219afc98cc6c3

Contents

`beta.simu.plot` *Histogram and Q-Q plot of simulated Beta distribution*

Description

Histogram and Q-Q plot of simulated Beta distribution

Usage

```
beta.simu.plot(n, shape1, shape2, times, ylim = NULL, qqplot = FALSE)
```

Arguments

`n`	number of trials in one simulation
`shape1`	non-negative parameters of the Beta distribution
`shape2`	non-negative parameters of the Beta distribution
`times`	number of simulations
`ylim`	range of y-axis
`qqplot`	an argument to output Q-Q plot or not, can be TRUE or FALSE

Value

Histogram and Q-Q plot of simulated Beta distribution, red curve represents theoretical density

Examples

```
beta.simu.plot(n = 5, shape1 = 3, shape2 = 1, times = 100)
```

binom.simu.plot *Histogram and Q-Q plot of simulated Binomial distribution*

Description

Histogram and Q-Q plot of simulated Binomial distribution

Usage

```
binom.simu.plot(n, size, prob, times, ylim = NULL, qqplot = FALSE)
```

Arguments

n	number of observations
size	number of trials (zero or more)
prob	probability of success on each trial
times	number of simulations
ylim	range of y-axis
qqplot	an argument to output Q-Q plot or not, can be TRUE or FALSE

Value

Histogram and Q-Q plot of simulated Binomial distribution, red curve represents theoretical density

Examples

```
binom.simu.plot(n = 10, size = 5, prob = 0.2, times = 100)
```

chisq.simu.plot *Histogram and Q-Q plot of simulated Chi-Squared distribution*

Description

Histogram and Q-Q plot of simulated Chi-Squared distribution

Usage

```
chisq.simu.plot(n, df, times, ylim = NULL, qqplot = FALSE)
```

Arguments

n	number of trials in one simulation
df	degrees of freedom (non-negative, but can be non-integer)
times	number of simulations
ylim	range of y-axis
qqplot	an argument to output Q-Q plot or not, can be TRUE or FALSE

Value

Histogram and Q-Q plot of simulated Chi-Squared distribution, red curve represents theoretical density

Examples

```
chisq.simu.plot(n = 5, df = 4, times = 100)
```

`coin` *Theoretical Probability Distribution of Flipping Coins*

Description

Mean and probability of flipping fair or loaded coin

Usage

```
coin(n, prob = NULL)
```

Arguments

`n`	number of trials
`prob`	probability assigned to each possible outcome

Details

The default probabilty equals to 1/n. All the assigned probabilites must between 0 and 1.

Value

Mean value and corresponding probabilities for all possible outcomes.

Examples

```
coin(n = 4)
coin(6, c(0.1, 0.9))
```

coin.plot *Theoretical Probability Distribution Plot of Flipping Coins*

Description

Probability plot of flipping fair or loaded coin

Usage

```
coin.plot(n, prob = NULL, col = "black", type = NULL,
main = NULL, sub = NULL)
```

Arguments

n	number of trials
prob	probability assigned to each possible outcome
col	color of the plot
type	type of plot
main	an overall title for the plot
sub	a sub title for the plot

Details

The default probabilty equals to 1/n. All the assigned probabilites must between 0 and 1.

Value

Plot of mean value and corresponding probabilities for all possible outcomes.

Examples

```
coin.plot(n = 4, col ='red', type = 'p')
coin.plot(3, prob = c(0.3, 0.7))
```

coin.simu *Probability Distribution of Simulated Coins Flipping*

Description

Mean and probability plot of flipping fair or loaded coin

Usage

```
coin.simu(n, times, prob = NULL)
```

Arguments

n	number of trials in one simulation
times	number of simulations
prob	probability assigned to each possible outcome

Details

The default probabilty equals to 1/n. All the assigned probabilites must between 0 and 1.

Value

Mean value and corresponding probabilities for all simulated outcomes.

Examples

```
coin.simu(n = 4, times = 1000)
coin.simu(4, 1000, prob = c(0.3, 0.7))
```

---

coin.simu.plot *Probability Distribution Plot of Simulated Coins Flipping*

Description

Probability plot of simulated experiments on flipping coins

Usage

```
coin.simu.plot(n, times, prob = NULL, qqplot = FALSE, col = "black", type = NULL,
main = NULL, sub = NULL)
```

Arguments

n	number of trials in one simulation
times	number of simulations
prob	probability assigned to each possible outcome
qqplot	an argument to output Q-Q plot or not, can be TRUE or FALSE
col	color of the plot
type	type of plot
main	an overall title for the plot
sub	a sub title for the plot

Details

The default probabilty equals to 1/n. All the assigned probabilites must between 0 and 1.

Value

Plot of mean value and corresponding probabilities for all simulated outcomes.

Examples

```
coin.simu.plot(n = 4, times = 1000, col = 'red')
coin.simu.plot(4, 1000, prob = c(0.3, 0.7), type = 'p')
```

---

dice *Theoretical Probability Distribution of Rolling Dice*

Description

Mean and probability of rolling fair or loaded dice

Usage

```
dice(n, prob = NULL)
```

Arguments

n	number of trials
prob	probability assigned to each possible outcome

Details

The default probabilty equals to 1/n. All the assigned probabilites must between 0 and 1.

Value

Mean value and corresponding probabilities for all possible outcomes.

Examples

```
dice(n = 4)
dice(2, c(0.1, 0.2, 0.2, 0.1, 0.3, 0.1))
```

dice.plot *Theoretical Probability Distribution Plot of Rolling Dice*

Description

Probability plot of rolling fair or loaded dice

Usage

```
dice.plot(n, prob = NULL, col = "black", type = NULL,
main = NULL, sub = NULL)
```

Arguments

n	number of trials
prob	probability assigned to each possible outcome
col	color of the plot
type	type of plot
main	an overall title for the plot
sub	a sub title for the plot

Details

The default probabilty equals to 1/n. All the assigned probabilites must between 0 and 1.

Value

Plot of mean value and corresponding probabilities for all possible outcomes.

Examples

```
dice.plot(n = 4, col ='red', type = 'p')
dice.plot(3, prob = c(0.3, 0.1, 0.2, 0.1, 0.1, 0.2))
```

dice.simu *Probability Distribution of Simulated Dice Rolling*

Description

Mean and probabilityf of flipping fair or loaded dice

Usage

```
dice.simu(n, times, prob = NULL)
```

Arguments

n	number of trials in one simulation
times	number of simulations
prob	probability assigned to each possible outcome

Details

The default probabilty equals to 1/n. All the assigned probabilites must between 0 and 1.

Value

Mean value and corresponding probabilities for all simulated outcomes.

Examples

```
dice.simu(n = 4, times = 1000)
dice.simu(4, 1000, prob = c(0.3, 0.1, 0.1, 0.1, 0.3, 0.1))
```

---

dice.simu.plot *Probability Distribution Plot of Simulated Dice Rolling*

Description

Probability plot of dice simulated experiments

Usage

```
dice.simu.plot(n, times, prob = NULL, qqplot = FALSE, col = "black", type = NULL,
main = NULL, sub = NULL)
```

Arguments

n	number of trials in one simulation
times	number of simulations
prob	probability assigned to each possible outcome
qqplot	an argument to output Q-Q plot or not, can be TRUE or FALSE
col	color of the plot
type	type of plot
main	an overall title for the plot
sub	a sub title for the plot

Details

The default probabilty equals to 1/n. All the assigned probabilites must between 0 and 1.

Value

Plot of mean value and corresponding probabilities for all simulated outcomes.

Examples

```
dice.simu.plot(n = 4, times = 1000, col = 'red')
dice.simu.plot(4, 1000, prob = c(0.3, 0.1, 0.1, 0.1, 0.1, 0.3), type = 'p')
```

`distr.simu.plot` *Histogram and Q-Q plot of any given continuous distribution*

Description

Histogram and Q-Q plot of any given continuous distribution

Usage

```
distr.simu.plot(distr, n, times, prob = NULL, qqplot = FALSE, col = "black", type = NULL,
main = NULL, sub = NULL)
```

Arguments

`distr`	vector, all possible outcomes, population distribution
`n`	number of trials in one simulation
`times`	number of simulations
`prob`	probability assigned to each possible outcome
`qqplot`	an argument to output Q-Q plot or not, can be TRUE or FALSE
`col`	color of the plot
`type`	type of plot
`main`	an overall title for the plot
`sub`	a sub title for the plot

Details

The default probabilty equals to 1/n. All the assigned probabilites must between 0 and 1.

Value

Plot of mean value and corresponding probabilities for all simulated outcomes.

Examples

```
distr.simu.plot(distr = c(1,0.2,3.4,5,6.6,1.1,5,4.7,2.33,3), n = 4, times = 1000, col = 'red')
```

expo.simu.plot *Histogram and Q-Q plot of simulated Exponential distribution*

Description

Histogram and Q-Q plot of simulated Exponential distribution

Usage

```
expo.simu.plot(n, rate = 1, times, ylim = NULL, qqplot = FALSE)
```

Arguments

n	number of trials in one simulation
rate	vector of rates
times	number of simulations
ylim	range of y-axis
qqplot	an argument to output Q-Q plot or not, can be TRUE or FALSE

Value

Histogram and Q-Q plot of simulated Exponential distribution, red curve represents theoretical density

Examples

```
expo.simu.plot(n = 5, rate = 2, times = 100)
```

expt *Theoretical Probability Distribution of General Experiment*

Description

General experiment with basic probability

Usage

```
expt(x, n, prob = NULL)
```

Arguments

x	vector, possible outcomes in one trial of experiment
n	number of trials
prob	probability assigned to each possible outcome

Details

The default probabilty equals to 1/n. All the assigned probabilites must between 0 and 1.

Value

Mean value and corresponding probabilities for all possible outcomes.

Examples

```
expt(x = c(1:3), n = 4)
expt(c(2:4), 3, prob = c(0.3, 0.5, 0.2))
```

`expt.mse` *Mean square error of simulated experiments*

Description

Mean square error of simulated experiments

Usage

```
expt.mse(x, n, times, prob = NULL)
```

Arguments

`x`	vector, possible outcomes in one trial of experiment
`n`	number of trials
`times`	number of simulations
`prob`	probability assigned to each possible outcome

Details

The default probabilty equals to 1/n. All the assigned probabilites must between 0 and 1.

Value

Mean square error of simulated experiments

Examples

```
expt.mse(x = c(1:3), n = 4, times = 100)
expt.mse(c(0.1, 4, 2), 3, times = 50, prob = c(0.3, 0.5, 0.2))
```

`expt.plot` *Theoretical Probability Distribution Plot of General Experiment*

Description

General experiment plot with basic probability

Usage

```
expt.plot(x, n, prob = NULL, col = "black", type = NULL,
main = NULL, sub = NULL)
```

Arguments

`x`	vector, possible outcomes in one trial of experiment
`n`	number of trials
`prob`	probability assigned to each possible outcome
`col`	color of the plot
`type`	type of plot
`main`	an overall title for the plot
`sub`	a sub title for the plot

Details

The default probabilty equals to 1/n. All the assigned probabilites must between 0 and 1.

Value

Plot of mean value and corresponding probabilities for all possible outcomes.

Examples

```
expt.plot(x = c(1:3), n = 4, col ='red', type = 'p')
expt.plot(c(2:4), 3, prob = c(0.3, 0.5, 0.2))
```

expt.simu *Probability Distribution of Simulated General Experiments*

Description

Mean and probability of general simulated experiments

Usage

```
expt.simu(x, n, times, prob = NULL)
```

Arguments

x	vector, possible outcomes in one trial of experiment
n	number of trials in one simulation
times	number of simulations
prob	probability assigned to each possible outcome

Details

The default probabilty equals to 1/n. All the assigned probabilites must between 0 and 1.

Value

Mean value and corresponding probabilities for all simulated outcomes.

Examples

```
expt.simu(x = c(1:3), n = 4, times = 1000)
expt.simu(c(1:3), 4, 1000, prob = c(0.3, 0.1, 0.6))
```

expt.simu.plot *Probability Distribution Plot of Simulated General Experiments*

Description

Probability plot of general simulated experiments

Usage

```
expt.simu.plot(x, n, times, prob = NULL, qqplot = FALSE, col = "black", type = NULL,
main = NULL, sub = NULL)
```

Arguments

x	vector, possible outcomes in one trial of experiment
n	number of trials in one simulation
times	number of simulations
prob	probability assigned to each possible outcome
qqplot	an argument to output Q-Q plot or not, can be TRUE or FALSE
col	color of the plot
type	type of plot
main	an overall title for the plot
sub	a sub title for the plot

Details

The default probabilty equals to 1/n. All the assigned probabilites must between 0 and 1.

Value

Plot of mean value and corresponding probabilities for all simulated outcomes.

Examples

```
expt.simu.plot(x = c(1:3), n = 4, times = 1000, col = 'red')
expt.simu.plot(c(1:3), 4, 1000, prob = c(0.3, 0.1, 0.6), type = 'p')
```

---

gamm.simu.plot *Histogram and Q-Q plot of simulated Gamma distribution*

Description

Histogram and Q-Q plot of simulated Gamma distribution

Usage

```
gamm.simu.plot(n, shape, rate = 1, scale = 1/rate, times, ylim = NULL, qqplot = FALSE)
```

Arguments

n	number of trials in one simulation
shape	shape parameter
rate	vector of rates
scale	scale parameter
times	number of simulations
ylim	range of y-axis
qqplot	an argument to output Q-Q plot or not, can be TRUE or FALSE

Value

Histogram and Q-Q plot of simulated Gamma distribution, red curve represents theoretical density

Examples

```
gamm.simu.plot(n = 5, shape = 3, rate = 1, times = 100)
```

`geom.simu.plot` *Histogram and Q-Q plot of simulated Geometric distribution*

Description

Histogram and Q-Q plot of simulated Geometric distribution

Usage

```
geom.simu.plot(n, prob, times, ylim = NULL, qqplot = FALSE)
```

Arguments

`n`	number of observations
`prob`	probability of success on each trial
`times`	number of simulations
`ylim`	range of y-axis
`qqplot`	an argument to output Q-Q plot or not, can be TRUE or FALSE

Value

Histogram and Q-Q plot of simulated Geometric distribution, red curve represents theoretical density

Examples

```
geom.simu.plot(n = 10, prob = 0.2, times = 100)
```

hyper.simu.plot *Histogram and Q-Q plot of simulated Hypergeometric distribution*

Description

Histogram and Q-Q plot of simulated Hypergeometric distribution

Usage

```
hyper.simu.plot(n, a, b, k, times, ylim = NULL, qqplot = FALSE)
```

Arguments

n	number of observations
a	the number of white balls in the urn
b	the number of black balls in the urn
k	the number of balls drawn from the urn
times	number of simulations
ylim	range of y-axis
qqplot	an argument to output Q-Q plot or not, can be TRUE or FALSE

Value

Histogram and Q-Q plot of simulated Hypergeometric distribution, red curve represents theoretical density

Examples

```
hyper.simu.plot(n = 10, a = 10, b = 10, k = 5, times = 100)
```

nbinom.simu.plot *Histogram and Q-Q plot of simulated Negative Binomial distribution*

Description

Histogram and Q-Q plot of simulated Negative Binomial distribution

Usage

```
nbinom.simu.plot(n, size, prob, times, ylim = NULL, qqplot = FALSE)
```

Arguments

n	number of observations
size	number of trials (zero or more)
prob	probability of success on each trial
times	number of simulations
ylim	range of y-axis
qqplot	an argument to output Q-Q plot or not, can be TRUE or FALSE

Value

Histogram and Q-Q plot of simulated Negative Binomial distribution, red curve represents theoretical density

Examples

```
nbinom.simu.plot(n = 10, size = 5, prob = 0.2, times = 100)
```

---

normal.simu.plot *Histogram and Q-Q plot of simulated Normal distribution*

Description

Histogram and Q-Q plot of simulated Normal distribution

Usage

```
normal.simu.plot(n, mean=0, sd=1, times, ylim = NULL, qqplot = FALSE)
```

Arguments

n	number of trials in one simulation
mean	vector of means
sd	vector of standard deviations
times	number of simulations
ylim	range of y-axis
qqplot	an argument to output Q-Q plot or not, can be TRUE or FALSE

Value

Histogram and Q-Q plot of simulated Normal distribution, red curve represents theoretical density

Examples

```
normal.simu.plot(n = 5, mean = 3, sd =2,  times = 100)
```

pois.simu.plot *Histogram and Q-Q plot of simulated Poisson distribution*

Description

Histogram and Q-Q plot of simulated Poisson distribution

Usage

```
pois.simu.plot(n, lambda, times, ylim = NULL, qqplot = FALSE)
```

Arguments

n	number of trials in one simulation
lambda	parameter of Poisson distribution
times	number of simulations
ylim	range of y-axis
qqplot	an argument to output Q-Q plot or not, can be TRUE or FALSE

Value

Histogram and Q-Q plot of simulated Poisson distribution, red curve represents theoretical density

Examples

```
pois.simu.plot(n = 5, lambda = 3, times = 100)
```

unif.simu.plot *Histogram and Q-Q plot of simulated Uniform distribution*

Description

Histogram and Q-Q plot of simulated Uniform distribution

Usage

```
unif.simu.plot(n, min = 0, max = 1, times, ylim = NULL, qqplot = FALSE)
```

Arguments

n	number of trials in one simulation
min	possible minimum value of Uniform distribution. Must be finite
max	possible maximum value of Uniform distribution. Must be finite
times	number of simulations
ylim	range of y-axis
qqplot	an argument to output Q-Q plot or not, can be TRUE or FALSE

Value

Histogram and Q-Q plot of simulated Uniform distribution, red curve represents theoretical density

Examples

```
unif.simu.plot(n = 5, min = 3, max = 5, times = 100)
```

Index