Package 'polstat'

November 15, 2020

Type Package

Title Useful Functions for Northwestern University Political Science Methods Classes

Version 0.1.0

URL https://github.com/lin-jennifer/polstat

Description
A collection of helpful functions to use along with core concepts learned in PS 403 and others

License MIT

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Suggests testthat

Depends R (>= 2.10)

R topics documented:

`tscoreCI` *One Sample Confidence Interval from a t Distribution*

Description

Helpful functions to calculate a one-sample confidence interval using a t-distribution

Usage

```
tCIupper(x, se, ci, n)

tCIlower(x, se, ci, n)

tCI(x, se, ci, n)

tMOE(se, ci, n)
```

Arguments

x	x-bar for observed mean
se	Standard Error – see se() from zscoreCI section
ci	Confidence Interval Level
n	Number of participants

Details

NOTE: This is not NOT intended from a z-distribution. See relevant zscoreCI file for those calculations

See Also

zCI(), zCIupper(), zCIlower()

Examples

```
x = 3.5
sd = 2
n = 25

se <- se(sd, n)  # 0.4

ci = 0.95  # for 95% Confidence Interval

tCIupper(x, se, ci, n) # 4.325559
tCIlower(x, se, ci, n) # 2.674441
tMOE(se, ci, n)        # 0.8255594
```

Tstats *Two Sample T Test and Confidence Interval*

Description

For PS 403 purposes. In real life, you should always just use `t.test()`

Usage

```
tEqvar(x1, s1, n1, x2, s2, n2)

tUneqvar(x1, s1, n1, x2, s2, n2)
```

Arguments

x1	mean for sample 1
s1	sd for sample 1
n1	number on sample 1
x2	mean for sample 2
s2	sd for sample 2
n2	number on sample 2

Examples

```
x1 = 5
x2 = 8
s1 = 0.6
s2 = 0.4
n1 = 30
n2 = 25

tEqvar(x1, s1, n1, x2, s2, n2)
```

Zprop *Two Sample Proportions Test and Confidence Interval*

Description

For use with proportions and z-scores.

Usage

```
prop.se(p, n)

p.hat(p1, n1, p2, n2)

z.prop(p1, n1, p2, n2)
```

Arguments

p	pi for a proportion, usually denoted as pi_0
n	number of responses
p1	probability for sample 1
n1	number on sample 1
p2	probability for sample 2
n2	number on sample 2

Details

When working with a one-sample proportion, it is suffucient to use `prop.se()` from here and the `zscore()` function, replacing the notions of `x` and `means` with the proper values for proportions

When working with these functions, it is CRUCIAL that you keep in mind what vales are for what. I would recommend setting up code similiar to the example to keep values in order.

Examples

```
pi = 0.5
n = 100

p1 = 0.25
n1 = 200
p2 = 0.45
n2 = 350

prop.se(pi, n)
p.hat(p1, n1, p2, n2)
```

`zscore` *z-score Calculations*

Description

Calculating a standard score in Base R can be hard.

Usage

```
zscore(x, mean, sd)
```

Arguments

`x`	the observation
`mean`	mean of interest – can be sample or population depending on zscore interest
`sd`	standard deviation or standard error, depending on context

Examples

```
zscore(10, 15, 2)
```

`zscoreCI` *One Sample Confidence Interval from a Normal Distribution*

Description

Helpful functions to calculate a one-sample confidence interval using a z-distribution

Usage

```
se(sd, n)

zCIupper(x, se, ci)

zCIlower(x, se, ci)

zCI(x, se, ci)

zMOE(se, ci)
```

Arguments

sd	Standard Deviation – reference sdNA()
n	Number of participants
x	x-bar for observed mean
se	Standard Error – see se()
ci	Confidence Interval Level

Details

NOTE: This is not NOT intended from a t-distribution. See relavant tscoreCI file for those calculations

Examples

```
x = 3.5
sd = 2
n = 25

se <- se(sd, n)  # 0.4

ci = 0.95  # for 95% Confidence Interval

zCIupper(x, se, ci) # 4.283986
zCIlower(x, se, ci) # 2.716014
zMOE(se, ci)        # 0.7839856
```

Index