Package: NPS (via r-universe)

November 13, 2026

Type Package

Title Convenience Functions and Tests for Working with the Net Promoter Score (NPS)

Version 1.1.9001

Author Brendan Rocks [aut, cre]

Maintainer Brendan Rocks <rocks.brendan@gmail.com>

Description Small functions to make working with survey data in the context of a Net Promoter programme easier. Specifically, data transformation methods, some methods for examining the statistical properties of the NPS, such as its variance and standard errors, and some simple inferential testing procedures. Net Promoter and NPS are registered trademarks of Bain & Company, Satmetrix Systems and Fred Reichheld. Compatibility updates by Renova.

License GPL (>= 2)

Encoding UTF-8

NeedsCompilation no

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

Date/Publication 2026-11-13 16:11:33 UTC

RemoteUrl https://github.com/renova-r/NPS

RemoteRef renova-1.1.9001

RemoteSha eb93b068ca1260d7b4bbe6b3abea5d606ab8cd6e

Contents

NPS-package 2
npc 2
nps 4
nps.se 5
nps.test 6
nps.var 7

npvar 8
scalestrip 8
testdata 9

Index **10**

`NPS-package` *Convenience Functions and Tests for Working With the Net Promoter Score (NPS)*

Description

Package:	NPS
Type:	Package
Version:	1.1
Date:	2014-12-03
License:	GPL (>= 2)
LazyLoad:	yes
Encoding:	UTF-8

Details

Small functions to make working with survey data in the context of a Net Promoter programme easier. Specifically, data transformation methods, some methods for examining the statistical properties of the NPS, such as its variance and standard errors, and some simple inferential testing procedures. Net Promoter and NPS are registered trademarks of Bain & Company, Satmetrix Systems and Fred Reichheld.

Author(s)

Brendan Rocks <rocks.brendan@gmail.com>

`npc` *Create Net Promoter Categories from Likelihood to Recommend Scores*

Description

This function produces Net Promoter Categories for `numeric` or `integer` *Recommend* data

Usage

```
npc(x, breaks = list(0:6, 7:8, 9:10))
```

Arguments

x	A vector of *Recommend* scores
breaks	A `list` of length three, giving the integer Likert scale points for *Detractors*, *Passives*, and *Promoters*, respectively. The default is `list(0:6, 7:8, 9:10)`

Value

Net Promoter categories

Author(s)

Brendan Rocks <rocks.brendan@gmail.com>

See Also

nps

Examples

```
# The command below will generate Net Promoter categories for each point
# on a standard 0:10 Likelihood to Recommend scale
npc(0:10)

 # Here's how scores and categories map out. Notice that scores which are
 # 'off the scale' drop out as missing/invalid
data.frame(score = -2:12, category = npc(-2:12))

# When you have lots of data, summaries are useful
rec <- sample(0:10, prob=c(0.02, 0.01, 0.01, 0.01, 0.01, 0.03, 0.03, 0.09,
    0.22, 0.22, 0.35), 1000, replace=TRUE)

# A Histrogram of the Likelihood to Recommend scores we just generated
hist(rec, breaks=-1:10)

# A look at the by nps category using summary
summary(npc(rec))

# As above
table(npc(rec))

# As a crosstabulation
table(rec, npc(rec))

nps(rec)
```

nps *Calculate a Net Promoter Score*

Description

This function calculates a Net Promoter Score from a vector of *Recommend* scores, ideally `numeric` ones. An attempt will be made to coerce `factor`, or `character` data. `NA` values, either in the data, or generated by type coercion, are automatically omitted from the calculation. No warning is given in the former case. Net Promoter Scores generated are on a [-1,1] scale; you may want to multiply them by 100 (and perhaps round them!) prior to presentation.

Usage

```
nps(x, breaks = list(0:6, 7:8, 9:10))
```

Arguments

`x`	A vector of *Recommend* scores
`breaks`	A `list` of length three, giving the integer Likert scale points for *Detractors*, *Passives*, and *Promoters*, respectively. The default is `list(0:6, 7:8, 9:10)`

Value

a Net Promoter Score. Unrounded.

Author(s)

Brendan Rocks <rocks.brendan@gmail.com>

See Also

`npc`

Examples

```
# This will generate 1000 dummy Likelihood to Recommend reponses
x <- sample(0:10, prob=c(0.02, 0.01, 0.01, 0.01, 0.01, 0.03, 0.03, 0.09,
    0.22, 0.22, 0.35), 1000, replace=TRUE)

# Here are the proportions of respondents giving each Likelihood to
# Recommend response
prop.table(table(x))

# Here's a histrogram of the scores
hist(x, breaks=-1:10, col=c(rep("red",7), rep("yellow",2), rep("green", 2)))

# Here's a barplot. It's very similar, though for categorical responses
# it's often slightly easier to interpret.
barplot(
```

```
    prop.table(table(x)),
     col=c(rep("red",7), rep("yellow",2), rep("green", 2))
     )

# Here's the nps
nps(x)

#You can round it if you like
round(nps(x)) ; round(nps(x),1)
```

`nps.se` *Calculate the standard error of a Net Promoter Score*

Description

This function calculates the standard error (see below) of a Net Promoter Score, taking a `vector` of raw *Recommend* data

Usage

```
nps.se(x, breaks = list(0:6, 7:8, 9:10))
```

Arguments

`x`	A vector of *Recommend* scores
`breaks`	A `list` of length three, giving the integer Likert scale points for *Detractors*, *Passives*, and *Promoters*, respectively. The default is `list(0:6, 7:8, 9:10)`

Value

`numeric`. The variance of the distribution, ranging from 0 to 1.

Author(s)

Brendan Rocks <rocks.brendan@gmail.com>

See Also

`npvar`, a version which works on counts or proportions of responses

`nps.test` *Significance tests and confidence intervals for Net Promoter Scores*

Description

This function performs one and two sample tests for the Net Promoter score(s) of *Recommend* data distributions. Currently, only a Wald type test is supported.

Usage

```
nps.test(x, y = NULL, test = "wald", conf = 0.95, breaks = list(0:6,
  7:8, 9:10))

## S3 method for class 'nps.test'
print(x, ...)
```

Arguments

`x`	A vector of *Recommend* scores
`y`	A vector of *Recommend* scores, to compare to `x`. If not specified, a one sample test on `x` is performed. Defaults to `NULL`
`test`	The type of test to perform. Currently only the Wald/Z-test ('`wald`') is supported
`conf`	the confidence level of the test and intervals. Defaults to 0.95
`breaks`	A `list` of length three, giving the integer Likert scale points for *Detractors*, *Passives*, and *Promoters*, respectively. The default is `list(0:6, 7:8, 9:10)`
`...`	Not used.

Value

A `list` of class `nps.test` containing:

`nps.x`, `nps.y`	The Net Promoter score(s)
`delta`	Where both `x` and `y` have been specified, the absolute difference between the two scores
`int`	The confidence interval generated. For a one sample test, this will be a confidence interval around `nps.x`. For a two sample test, this will be a confidence interval around the difference between `nps.x` and `nps.y`
`conf`	The confidence level used when performing the test, as specificed by `conf` in the function parameters
`p.value`	The p value of the significance test
`sig`	`logical`. Whether or not the `p.value` of the test exceeded 1-`conf`
`se.hat`	The estimated standard error of `delta` for a two sample test, otherwise of `x`
`type`	`character` string indicating whether a one or two sample test was performed
`n.x`, `n.y`	Counts for `x` and `y`

`NULL`

Author(s)

Brendan Rocks <rocks.brendan@gmail.com>

See Also

nps.var, nps.se, nps

---

nps.var *Calculate the variance of a Net Promoter Score*

Description

This function calculates the Net Promoter Score variance, taking a vector of raw *Recommend* data

Usage

```
nps.var(x, breaks = list(0:6, 7:8, 9:10))
```

Arguments

x	A vector of *Recommend* scores
breaks	A list of length three, giving the integer Likert scale points for *Detractors*, *Passives*, and *Promoters*, respectively. The default is list(0:6, 7:8, 9:10)

Value

numeric. The variance of the distribution, ranging from 0 to 1.

Author(s)

Brendan Rocks <rocks.brendan@gmail.com>

See Also

npvar, a version which works on counts or proportions of responses

npvar *Calculate the variance of a Net Promoter Score*

Description

This function calculates the Net Promoter Score variance, taking a vector of length three, with numbers or proportions of *Detractors*, *Passives*, and *Promoters*, respectively.

Usage

```
npvar(x)
```

Arguments

x	A vector of length three, with numbers or proportions of *Detractors*, *Passives*, and *Promoters*, respectively

Value

numeric. The variance of the distribution, ranging from 0 to 1.

Author(s)

Brendan Rocks <rocks.brendan@gmail.com>

See Also

nps.var, a version which works on raw Recommend responses

scalestrip *Strips Likert scale point labels, returns numeric or ordinal data*

Description

Survey systems export responses to Likhert scales with the scale labels on, meaning that R as factors or text as opposed to numeric data. This function takes labelled scales and returns unlabelled numeric data (by default), or an unlabelled ordered factor (if requested).

Usage

```
scalestrip(x, ordinal = FALSE)
```

Arguments

x	a vector, matrix, or data.frame, containing Likert data labelled in the format "Integer - some text", e.g. "10 - Extremely Likely"
ordinal	logical (TRUEFALSE). Should the data returned be an ordered factor? Otherwise the data returned is numeric. Defaults to FALSE.

Value

Unlabelled numeric data (by defualt), or an unlabelled ordered factor (if requested).

Author(s)

Brendan Rocks <rocks.brendan@gmail.com>

`testdata` *This is data to be included in my package*

Description

An idealised test data set, for demonstrating some of the NPS functions

Author(s)

Brendan Rocks <rocks.brendan@gmail.com>

Index

∗ **data**
testdata, 9
∗ **package**
NPS-package, 2

character, *4*, *6*

data.frame, *8*

factor, *4*

integer, *2*

list, *6*
logical, *6*, *8*

matrix, *8*

npc, 2, *4*
NPS *(*NPS-package*)*, 2
nps, *3*, 4, *7*
NPS-package, 2
nps.se, 5, *7*
nps.test, 6
nps.var, 7, *7*, *8*
npvar, *5*, *7*, 8
numeric, *2*, *4*, *5*, *7*, *8*

print.nps.test *(*nps.test*)*, 6

scalestrip, 8

testdata, 9

vector, *5*, *7*, *8*