Package 'digitTests'

October 13, 2022

Title Tests for Detecting Irregular Digit Patterns

Version 0.1.2

Date 2022-06-16

Description Provides statistical tests and support functions for detecting irregular digit patterns in numerical data. The package includes tools for extracting digits at various locations in a number, tests for repeated values, and (Bayesian) tests of digit distributions.

BugReports https://github.com/koenderks/digitTests/issues

URL https://koenderks.github.io/digitTests/, https://github.com/koenderks/digitTests

Imports graphics, stats

Suggests benford.analysis, BenfordTests, BeyondBenford, knitr, rmarkdown, testthat

Language en-US

License GPL (>= 3)

Encoding UTF-8

LazyData true

RoxygenNote 7.2.0

VignetteBuilder knitr

NeedsCompilation no

Author Koen Derks [aut, cre] (<https://orcid.org/0000-0002-5533-9349>)

Maintainer Koen Derks <k.derks@nyenrode.nl>

Repository CRAN

Date/Publication 2022-06-16 16:10:12 UTC

R topics documented:

`digitTests-package` *digitTests: Tests for Detecting Irregular Data Patterns*

Description

`digitTests` is an R package providing tests for detecting irregular data patterns.

The package and its analyses are also implemented with a graphical user interface in the Audit module of JASP, a free and open-source statistical software program.

Author(s)

Koen Derks (maintainer, author) <k.derks@nyenrode.nl>

Please use the citation provided by R when citing this package. A BibTex entry is available from `citation("digitTests")`.

See Also

Useful links:

- The issue page to submit a bug report or feature request.

Examples

```
# Load the digitTests package
library(digitTests)

##########################################
### Example 1: Benford's Law ####
##########################################

data('sinoForest')
distr.test(sinoForest$value, check = 'first', reference = 'benford')

##################################
### Example 2: Repeated Values ####
##################################

data('sanitizer')
```

```
rv.test(sanitizer$value, check = 'lasttwo', method = 'af', B = 1000)
```

`distr.btest` *Bayesian Test of Digits against a Reference Distribution*

Description

This function extracts and performs a Bayesian test of the distribution of (leading) digits in a vector against a reference distribution. By default, the distribution of leading digits is checked against Benford's law.

Usage

```
distr.btest(x, check = 'first', reference = 'benford',
            alpha = NULL, BF10 = TRUE, log = FALSE)
```

Arguments

`x`	a numeric vector.
`check`	location of the digits to analyze. Can be `first`, `firsttwo`, or `last`.
`reference`	which character string given the reference distribution for the digits, or a vector of probabilities for each digit. Can be `benford` for Benford's law, `uniform` for the uniform distribution. An error is given if any entry of `reference` is negative. Probabilities that do not sum to one are normalized.
`alpha`	a numeric vector containing the prior parameters for the Dirichlet distribution on the digit categories.
`BF10`	logical. Whether to compute the Bayes factor in favor of the alternative hypothesis (BF10) or the null hypothesis (BF01).
`log`	logical. Whether to return the logarithm of the Bayes factor.

Details

Benford's law is defined as $p(d) = log10(1/d)$. The uniform distribution is defined as $p(d) = 1/d$.

The Bayes Factor BF_{10} quantifies how much more likely the data are to be observed under H_1: the digits are not distributed according to the reference distribution than under H_0: the digits are distributed according to the reference distribution. Therefore, BF_{10} can be interpreted as the relative support in the observed data for H_1 versus H_0. If BF_{10} is 1, there is no preference for either H_1 or H_0. If BF_{10} is larger than 1, H_1 is preferred. If BF_{10} is between 0 and 1, H_0 is preferred. The Bayes factor is calculated using the Savage-Dickey density ratio.

Value

An object of class `dt.distr` containing:

`observed`	the observed counts.
`expected`	the expected counts under the null hypothesis.
`n`	the number of observations in `x`.
`statistic`	the value the chi-squared test statistic.
`parameter`	the degrees of freedom of the approximate chi-squared distribution of the test statistic.
`p.value`	the p-value for the test.
`check`	checked digits.
`digits`	vector of digits.
`reference`	reference distribution
`data.name`	a character string giving the name(s) of the data.

Author(s)

Koen Derks, <k.derks@nyenrode.nl>

References

Benford, F. (1938). The law of anomalous numbers. *In Proceedings of the American Philosophical Society*, 551-572.

See Also

`distr.test` `rv.test`

Examples

```
set.seed(1)
x <- rnorm(100)

# Bayesian digit analysis against Benford's law
distr.btest(x, check = 'first', reference = 'benford')

# Bayesian digit analysis against Benford's law, custom prior
distr.btest(x, check = 'first', reference = 'benford', alpha = 9:1)

# Bayesian digit analysis against custom distribution
distr.btest(x, check = 'last', reference = rep(1/9, 9))
```

`distr.test` *Test of Digits against a Reference Distribution*

Description

This function extracts and performs a test of the distribution of (leading) digits in a vector against a reference distribution. By default, the distribution of leading digits is checked against Benford's law.

Usage

```
distr.test(x, check = 'first', reference = 'benford')
```

Arguments

`x`	a numeric vector.
`check`	location of the digits to analyze. Can be `first`, `firsttwo`, or `last`.
`reference`	which character string given the reference distribution for the digits, or a vector of probabilities for each digit. Can be `benford` for Benford's law, `uniform` for the uniform distribution. An error is given if any entry of `reference` is negative. Probabilities that do not sum to one are normalized.

Details

Benford's law is defined as $p(d) = log10(1/d)$. The uniform distribution is defined as $p(d) = 1/d$.

Value

An object of class `dt.distr` containing:

`observed`	the observed counts.
`expected`	the expected counts under the null hypothesis.
`n`	the number of observations in `x`.
`statistic`	the value the chi-squared test statistic.
`parameter`	the degrees of freedom of the approximate chi-squared distribution of the test statistic.
`p.value`	the p-value for the test.
`check`	checked digits.
`digits`	vector of digits.
`reference`	reference distribution
`data.name`	a character string giving the name(s) of the data.

Author(s)

Koen Derks, <k.derks@nyenrode.nl>

References

Benford, F. (1938). The law of anomalous numbers. *In Proceedings of the American Philosophical Society*, 551-572.

See Also

`distr.btest` `rv.test`

Examples

```
set.seed(1)
x <- rnorm(100)

# Digit analysis against Benford's law
distr.test(x, check = 'first', reference = 'benford')

# Digit analysis against custom distribution
distr.test(x, check = 'last', reference = rep(1/9, 9))
```

`dt-methods` *Methods for da objects*

Description

Methods defined for objects returned from the `distr.test`, `distr.btest`, and `rv.test` functions.

Usage

```
## S3 method for class 'dt.distr'
print(x, digits = getOption("digits"), ...)

## S3 method for class 'dt.rv'
print(x, digits = getOption("digits"), ...)

## S3 method for class 'dt.distr'
plot(x, ...)

## S3 method for class 'dt.rv'
plot(x, ...)
```

Arguments

`x`	an object of class da as returned by one of the package functions.
`digits`	the number of digits to round to.
`...`	further arguments, currently ignored.

Value

The `print` methods simply print and return nothing.

`extract_digits` *Extraction of First or Last Digits*

Description

This function extracts the first (and optionally second) or last digits in a vector.

Usage

```
extract_digits(x, check = 'first', include.zero = FALSE)
```

Arguments

`x`	a numeric vector.
`check`	location of the digits to extract. Can be `first`, `firsttwo`, or `last`.
`include.zero`	logical. Whether to include the digit zero in the output.

Value

A vector of first (and optionally second) or last digits.

Author(s)

Koen Derks, <k.derks@nyenrode.nl>

Examples

```
set.seed(1)
x <- rnorm(100)

# Extract first digits (without zero)
extract_digits(x, check = 'first')

# Extract last digits (including zero)
extract_digits(x, check = 'last', include.zero = TRUE)
```

`rv.test` *Test of Repeated Values*

Description

This function analyzes the frequency with which values get repeated within a set of numbers. Unlike Benford's law, and its generalizations, this approach examines the entire number at once, not only the first or last digit.

Usage

```
rv.test(x, check = 'last', method = 'af', B = 2000)
```

Arguments

`x`	a numeric vector of values from which the digits should be analyzed.
`check`	which digits to shuffle during the procedure. Can be `last` or `lasttwo`.
`method`	which property of the data is calculated. Defaults to `af` for average frequency, but can also be `entropy` for entropy.
`B`	how many samples to use in the bootstraping procedure.

Details

To determine whether the data show an excessive amount of bunching, the null hypothesis that `x` does not contain an unexpected amount of repeated values is tested against the alternative hypothesis that `x` has more repeated values than expected. The statistic can either be the average frequency ($AF = sum(f_i^2)/sum(f_i)$) of the data or the entropy ($E = -sum(p_i * log(p_i))$, with $p_i = f_i/n$) of the data. Average frequency and entropy are highly correlated, but the average frequency is often more interpretable. For example, an average frequency of 2.5 means that, on average, your observations contain a value that appears 2.5 times in the data set.To quantify what is expected, this test requires the assumption that the integer portions of the numbers are not associated with their decimal portions.

Value

An object of class `dt.rv` containing:

`x`	input data.
`frequencies`	frequencies of observations in `x`.
`samples`	vector of simulated samples.
`integers`	counts for extracted integers.
`decimals`	counts for extracted decimals.
`n`	the number of observations in `x`.
`statistic`	the value the average frequency or entropy statistic.

p.value	the p-value for the test.
cor.test	correlation test for the integer portions of the number versus the decimals portions of the number.
method	method used.
check	checked digits.
data.name	a character string giving the name(s) of the data.

Author(s)

Koen Derks, <k.derks@nyenrode.nl>

References

Simohnsohn, U. (2019, May 25). Number-Bunching: A New Tool for Forensic Data Analysis. Retrieved from https://datacolada.org/77.

See Also

distr.test distr.btest

Examples

```
set.seed(1)
x <- rnorm(50)

# Repeated values analysis shuffling last digit
rv.test(x, check = 'last', method = 'af', B = 2000)
```

---

sanitizer *Factory Workers' use of Hand Sanitizer*

Description

Data from a study on factory workers' use of hand sanitizer. Sanitizer use was measured to a 100th of a gram.

Usage

```
data(sanitizer)
```

Format

A data frame with 1600 rows and 1 variable.

References

[Retracted] Li, M., Sun, Y., & Chen, H. (2019). The decoy effect as a nudge: Boosting hand hygiene with a worse option. Psychological Science, 30, 139–149.

Examples

```
data(sanitizer)
```

`sinoForest` *Financial Statemens of Sino Forest Corporation's 2010 Report*

Description

Financial Statemens numbers of Sino Forest Corporation's 2010 Report.

Usage

```
data(sinoForest)
```

Format

A data frame with 772 rows and 1 variable.

References

Nigrini, M. J. (2012). Benford's Law: Application for Forensic Accounting, Auditing and Fraud Detection. Wiley and Sons: New Jersey.

Examples

```
data(sinoForest)
```

Index