

# Package ‘BenfordTests’

August 4, 2015

**Type** Package

**Title** Statistical Tests for Evaluating Conformity to Benford's Law

**Version** 1.2.0

**Date** 2015-08-04

**Depends** R (>= 3.0.0), grDevices, graphics, stats

**Maintainer** Dieter William Joenssen <Dieter.Joenssen@gmail.com>

**Description** Several specialized statistical tests and support functions  
for determining if numerical data could conform to Benford's law.

**License** GPL-3

**URL** <https://cran.r-project.org/package=BenfordTests>,  
[https://www.researchgate.net/profile/Dieter\\_Joenssen](https://www.researchgate.net/profile/Dieter_Joenssen)

**Author** Dieter William Joenssen [aut, cre, cph],  
Thomas Muellerleile [ctb]

**NeedsCompilation** yes

**Repository** CRAN

**Date/Publication** 2015-08-04 18:25:11

## R topics documented:

BenfordTests-package . . . . .	2
chisq.benftest . . . . .	4
edist.benftest . . . . .	5
jointdigit.benftest . . . . .	7
jpsq.benftest . . . . .	8
ks.benftest . . . . .	10
mdist.benftest . . . . .	11
meandigit.benftest . . . . .	13
pbenf . . . . .	14
qbenf . . . . .	16
rbenf . . . . .	17
signifd . . . . .	18

signifd.analysis . . . . .	19
signifd.seq . . . . .	21
simulateH0 . . . . .	21
usq.benftest . . . . .	23

<b>Index</b>	<b>25</b>
--------------	-----------

---

BenfordTests-package    *Statistical Tests for Benford's Law*

---

## Description

This package contains several specialized statistical tests and support functions for determining if numerical data could conform to Benford's law.

## Details

Package: BenfordTests  
 Type: Package  
 Version: 1.2.0  
 Date: 2015-07-18  
 License: GPL-3

BenfordTests is the implementation of eight goodness-of-fit (GOF) tests to assess if data conforms to Benford's law.

Tests include:

Pearson  $\chi^2$  statistic (Pearson, 1900)

Kolmogorov-Smirnov  $D$  statistic (Kolmogorov, 1933)

Freedman's modification of Watson's  $U^2$  statistic (Freedman, 1981; Watson, 1961)

Chebyshev distance  $m$  statistic (Leemis, 2000)

Euclidean distance  $d$  statistic (Cho and Gaines, 2007)

Judge-Schechter mean deviation  $a^*$  statistic (Judge and Schechter, 2009)

Joenssen's  $J_P^2$  statistic, a Shapiro-Francia type correlation test (Shapiro and Francia, 1972)

Joint Digit Test  $T^2$  statistic, a Hotelling type test (Hotelling, 1931)

All tests may be performed using more than one leading digit. All tests simulate the specific p-values required for statistical inference, while p-values for the  $\chi^2$ ,  $D$ ,  $a^*$ , and  $T^2$  statistics may also be determined using their asymptotic distributions.

## Author(s)

Dieter William Joenssen

Maintainer: Dieter William Joenssen <Dieter.Joenssen@googlemail.com>

## References

- Benford, F. (1938) The Law of Anomalous Numbers. *Proceedings of the American Philosophical Society*. **78**, 551–572.
- Cho, W.K.T. and Gaines, B.J. (2007) Breaking the (Benford) Law: Statistical Fraud Detection in Campaign Finance. *The American Statistician*. **61**, 218–223.
- Freedman, L.S. (1981) Watson's Un2 Statistic for a Discrete Distribution. *Biometrika*. **68**, 708–711.
- Joenssen, D.W. (2013) Two Digit Testing for Benford's Law. *Proceedings of the ISI World Statistics Congress, 59th Session in Hong Kong*. [available under <http://www.statistics.gov.hk/wsc/CPS021-P2-S.pdf>]
- Judge, G. and Schechter, L. (2009) Detecting Problems in Survey Data using Benford's Law. *Journal of Human Resources*. **44**, 1–24.
- Kolmogorov, A.N. (1933) Sulla determinazione empirica di una legge di distribuzione. *Giornale dell'Istituto Italiano degli Attuari*. **4**, 83–91.
- Leemis, L.M., Schmeiser, B.W. and Evans, D.L. (2000) Survival Distributions Satisfying Benford's law. *The American Statistician*. **54**, 236–241.
- Newcomb, S. (1881) Note on the Frequency of Use of the Different Digits in Natural Numbers. *American Journal of Mathematics*. **4**, 39–40.
- Pearson, K. (1900) On the Criterion that a Given System of Deviations from the Probable in the Case of a Correlated System of Variables is Such that it can be Reasonably Supposed to have Arisen from Random Sampling. *Philosophical Magazine Series 5*. **50**, 157–175.
- Shapiro, S.S. and Francia, R.S. (1972) An Approximate Analysis of Variance Test for Normality. *Journal of the American Statistical Association*. **67**, 215–216.
- Watson, G.S. (1961) Goodness-of-Fit Tests on a Circle. *Biometrika*. **48**, 109–114.
- Hotelling, H. (1931). The generalization of Student's ratio. *Annals of Mathematical Statistics*. **2**, 360–378.

## Examples

```
#Set the random seed to an arbitrary number
set.seed(421)
#Create a sample satisfying Benford's law
X<-rbenf(n=20)
#Look at sample
X
#Look at the first digits of the sample
signifd(X)

#Perform a Chi-squared Test on the sample's first digits using defaults
chisq.benftest(X)
#p-value = 0.648
```

chisq.benftest

*Pearson's Chi-squared Goodness-of-Fit Test for Benford's Law***Description**

chisq.benftest takes any numerical vector reduces the sample to the specified number of significant digits and performs Pearson's chi-square goodness-of-fit test to assert if the data conforms to Benford's law.

**Usage**

```
chisq.benftest(x = NULL, digits = 1, pvalmethod = "asymptotic", pvalsims = 10000)
```

**Arguments**

x	A numeric vector.
digits	An integer determining the number of first digits to use for testing, i.e. 1 for only the first, 2 for the first two etc.
pvalmethod	Method used for calculating the p-value. Either "asymptotic" or "simulate".
pvalsims	An integer specifying the number of replicates to use if pvalmethod = "simulate".

**Details**

A  $\chi^2$  goodness-of-fit test is performed on `signifd(x,digits)` versus `pbenf(digits)`. Specifically:

$$\chi^2 = n \cdot \sum_{i=10^{k-1}}^{10^k-1} \frac{(f_i^o - f_i^e)^2}{f_i^e}$$

where  $f_i^o$  denotes the observed frequency of digits  $i$ , and  $f_i^e$  denotes the expected frequency of digits  $i$ . `x` is a numeric vector of arbitrary length. Values of `x` should be continuous, as dictated by theory, but may also be integers. `digits` should be chosen so that `signifd(x,digits)` is not influenced by previous rounding.

**Value**

A list with class "htest" containing the following components:

statistic	the value of the $\chi^2$ test statistic
p.value	the p-value for the test
method	a character string indicating the type of test performed
data.name	a character string giving the name of the data

**Author(s)**

Dieter William Joenssen <Dieter.Joenssen@googlemail.com>

## References

- Benford, F. (1938) The Law of Anomalous Numbers. *Proceedings of the American Philosophical Society*. **78**, 551–572.
- Joenssen, D.W. (2013) Two Digit Testing for Benford's Law. *Proceedings of the ISI World Statistics Congress, 59th Session in Hong Kong*. [available under <http://www.statistics.gov.hk/wsc/CPS021-P2-S.pdf>]
- Pearson, K. (1900) On the Criterion that a Given System of Deviations from the Probable in the Case of a Correlated System of Variables is Such that it can be Reasonably Supposed to have Arisen from Random Sampling. *Philosophical Magazine Series 5*. **50**, 157–175.

## See Also

[pbenf](#), [simulateH0](#)

## Examples

```
#Set the random seed to an arbitrary number
set.seed(421)
#Create a sample satisfying Benford's law
X<-rbenf(n=20)
#Perform a Chi-squared Test on the sample's
#first digits using defaults but determine
#the p-value by simulation
chisq.benftest(X,pvalmethod="simulate")
#p-value = 0.6401
```

---

edist.benftest

*Euclidean Distance Test for Benford's Law*

---

## Description

edist.benftest takes any numerical vector reduces the sample to the specified number of significant digits and performs a goodness-of-fit test based on the Euclidean distance between the first digits' distribution and Benford's distribution to assert if the data conforms to Benford's law.

## Usage

```
edist.benftest(x = NULL, digits = 1, pvalmethod = "simulate", pvalsims = 10000)
```

## Arguments

x	A numeric vector.
digits	An integer determining the number of first digits to use for testing, i.e. 1 for only the first, 2 for the first two etc.
pvalmethod	Method used for calculating the p-value. Currently only "simulate" is available.
pvalsims	An integer specifying the number of replicates used if pvalmethod = "simulate".

## Details

A statistical test is performed utilizing the Euclidean distance between `signifd(x,digits)` and `pbenf(digits)`. Specifically:

$$d = \sqrt{n} \cdot \sqrt{\sum_{i=10^{k-1}}^{10^k-1} (f_i^o - f_i^e)^2}$$

where  $f_i^o$  denotes the observed frequency of digits  $i$ , and  $f_i^e$  denotes the expected frequency of digits  $i$ .  $x$  is a numeric vector of arbitrary length. Values of  $x$  should be continuous, as dictated by theory, but may also be integers. `digits` should be chosen so that `signifd(x,digits)` is not influenced by previous rounding.

## Value

A list with class "htest" containing the following components:

<code>statistic</code>	the value of the Euclidean distance test statistic
<code>p.value</code>	the p-value for the test
<code>method</code>	a character string indicating the type of test performed
<code>data.name</code>	a character string giving the name of the data

## Author(s)

Dieter William Joenssen <Dieter.Joenssen@googlemail.com>

## References

Benford, F. (1938) The Law of Anomalous Numbers. *Proceedings of the American Philosophical Society*. **78**, 551–572.

Cho, W.K.T. and Gaines, B.J. (2007) Breaking the (Benford) Law: Statistical Fraud Detection in Campaign Finance. *The American Statistician*. **61**, 218–223.

Morrow, J. (2010) *Benford's Law, Families of Distributions and a Test Basis*. [available under <http://www.johnmorrow.info/projects/benford/benfordMain.pdf>]

## See Also

[pbenf](#), [simulateH0](#)

## Examples

```
#Set the random seed to an arbitrary number
set.seed(421)
#Create a sample satisfying Benford's law
X<-rbenf(n=20)
#Perform a Euclidean Distance Test on the
#sample's first digits using defaults
edist.benftest(X,pvalmethod ="simulate")
#p-value = 0.6085
```

---

jointdigit.benftest    *A Hotelling T-square Type Test for Benford's Law*

---

### Description

jointdigit.benftest takes any numerical vector reduces the sample to the specified number of significant digits and performs a Hotelling T-square type goodness-of-fit test to assert if the data conforms to Benford's law.

### Usage

```
jointdigit.benftest(x = NULL, digits = 1, eigenvalues="all", tol = 1e-15,
pvalmethod = "asymptotic", pvalsims = 10000)
```

### Arguments

x	A numeric vector.
digits	An integer determining the number of first digits to use for testing, i.e. 1 for only the first, 2 for the first two etc.
eigenvalues	How are the eigenvalues, which are used in testing, selected.
tol	Tolerance in detecting values that are essentially zero.
pvalmethod	Method used for calculating the p-value. Currently only "asymptotic" is available.
pvalsims	An integer specifying the number of replicates used if pvalmethod = "simulate".

### Details

A Hotelling  $T^2$  type goodness-of-fit test is performed on `signifd(x, digits)` versus `pbenf(digits)`. `x` is a numeric vector of arbitrary length. **argument:** `eigenvalues` can be defined as:

- *numeric*, a vector containing which eigenvalues should be used
- *string length = 1*, eigenvalue selection scheme:
  - "all", use all non-zero eigenvalues
  - "kaiser", use all eigenvalues larger than the mean of all non-zero eigenvalues

Values of `x` should be continuous, as dictated by theory, but may also be integers. `digits` should be chosen so that `signifd(x, digits)` is not influenced by previous rounding.

### Value

A list with class "htest" containing the following components:

statistic	the value of the $T^2$ test statistic
p.value	the p-value for the test
method	a character string indicating the type of test performed

data.name        a character string giving the name of the data  
 eigenvalues\_tested        a vector containing the index numbers of the eigenvalues used in testing.  
 eigen\_val\_vect        the eigen values and vectors of the null distribution. computed using eigen.

**Author(s)**

Dieter William Joenssen <Dieter.Joenssen@googlemail.com>

**References**

Benford, F. (1938) The Law of Anomalous Numbers. *Proceedings of the American Philosophical Society*. **78**, 551–572.  
 Hotelling, H. (1931). The generalization of Student's ratio. *Annals of Mathematical Statistics*. **2**, 360–378.

**See Also**

[pbenf](#)

**Examples**

```
#Set the random seed to an arbitrary number
set.seed(421)
#Create a sample satisfying Benford's law
X<-rbenf(n=20)
#Perform Test
#on the sample's first digits using defaults
jointdigit.benftest(X)
#p-value = 0.648
#Perform Test
#using only the two largest eigenvalues
jointdigit.benftest(x=X,eigenvalues=1:2)
#p-value = 0.5176
#Perform Test
#using the kaiser selection criterion
jointdigit.benftest(x=X,eigenvalues="kaiser")
#p-value = 0.682
```

---

jpsq.benftest

*Joenssen's JP-square Test for Benford's Law*

---

**Description**

jpsq.benftest takes any numerical vector reduces the sample to the specified number of significant digits and performs a goodness-of-fit test based on the correlation between the first digits' distribution and Benford's distribution to assert if the data conforms to Benford's law.



**Usage**

```
jpsq.benftest(x = NULL, digits = 1, pvalmethod = "simulate", pvalsims = 10000)
```

**Arguments**

x	A numeric vector.
digits	An integer determining the number of first digits to use for testing, i.e. 1 for only the first, 2 for the first two etc.
pvalmethod	Method used for calculating the p-value. Currently only "simulate" is available.
pvalsims	An integer specifying the number of replicates used if pvalmethod = "simulate".

**Details**

A statistical test is performed utilizing the sign-preserved squared correlation between `signifd(x, digits)` and `pbenf(digits)`. Specifically:

$$J_P^2 = \text{sgn}(\text{cor}(f^o, f^e)) \cdot \text{cor}(f^o, f^e)^2$$

where  $f^o$  denotes the observed frequencies and  $f^e$  denotes the expected frequency of digits  $10^{k-1}, 10^{k-1} + 1, \dots, 10^k - 1$ .  $x$  is a numeric vector of arbitrary length. Values of  $x$  should be continuous, as dictated by theory, but may also be integers. `digits` should be chosen so that `signifd(x, digits)` is not influenced by previous rounding.

**Value**

A list with class "htest" containing the following components:

statistic	the value of the $J_P^2$ test statistic
p.value	the p-value for the test
method	a character string indicating the type of test performed
data.name	a character string giving the name of the data

**Author(s)**

Dieter William Joenssen <Dieter.Joenssen@googlemail.com>

**References**

- Benford, F. (1938) The Law of Anomalous Numbers. *Proceedings of the American Philosophical Society*. **78**, 551–572.
- Joenssen, D.W. (2013) A New Test for Benford's Distribution. In: *Abstract-Proceedings of the 3rd Joint Statistical Meeting DAGStat, March 18-22, 2013*; Freiburg, Germany.
- Joenssen, D.W. (2013) Two Digit Testing for Benford's Law. *Proceedings of the ISI World Statistics Congress, 59th Session in Hong Kong*. [available under <http://www.statistics.gov.hk/wsc/CPS021-P2-S.pdf>]
- Shapiro, S.S. and Francia, R.S. (1972) An Approximate Analysis of Variance Test for Normality. *Journal of the American Statistical Association*. **67**, 215–216.

**See Also**

[pbenf](#), [simulateH0](#)

**Examples**

```
#Set the random seed to an arbitrary number
set.seed(421)
#Create a sample satisfying Benford's law
X<-rbenf(n=20)
#Perform Joenssen's \emph{JP-square} Test
#on the sample's first digits using defaults
jpsq.benftest(X)
#p-value = 0.3241
```

---

ks.benftest

*Kolmogorov-Smirnov Test for Benford's Law*

---

**Description**

ks.benftest takes any numerical vector reduces the sample to the specified number of significant digits and performs the Kolmogorov-Smirnov goodness-of-fit test to assert if the data conforms to Benford's law.

**Usage**

```
ks.benftest(x = NULL, digits = 1, pvalmethod = "simulate", pvalsims = 10000)
```

**Arguments**

x	A numeric vector.
digits	An integer determining the number of first digits to use for testing, i.e. 1 for only the first, 2 for the first two etc.
pvalmethod	Method used for calculating the p-value. Currently only "simulate" is available.
pvalsims	An integer specifying the number of replicates used if pvalmethod = "simulate".

**Details**

A Kolmogorov-Smirnov test is performed between `signifd(x, digits)` and `pbenf(digits)`. Specifically:

$$D = \sup_{i=10^{k-1}, \dots, 10^k-1} \left| \sum_{j=1}^i (f_j^o - f_j^e) \right| \cdot \sqrt{n}$$

where  $f_i^o$  denotes the observed frequency of digits  $i$ , and  $f_i^e$  denotes the expected frequency of digits  $i$ .  $x$  is a numeric vector of arbitrary length. Values of  $x$  should be continuous, as dictated by theory, but may also be integers. `digits` should be chosen so that `signifd(x, digits)` is not influenced by previous rounding.

**Value**

A list with class "htest" containing the following components:

statistic	the value of the Kolmogorov-Smirnov $D$ test statistic
p.value	the p-value for the test
method	a character string indicating the type of test performed
data.name	a character string giving the name of the data

**Author(s)**

Dieter William Joenssen <Dieter.Joenssen@googlemail.com>

**References**

Benford, F. (1938) The Law of Anomalous Numbers. *Proceedings of the American Philosophical Society*. **78**, 551–572.

Joenssen, D.W. (2013) Two Digit Testing for Benford's Law. *Proceedings of the ISI World Statistics Congress, 59th Session in Hong Kong*. [available under <http://www.statistics.gov.hk/wsc/CPS021-P2-S.pdf>]

Kolmogorov, A.N. (1933) Sulla determinazione empirica di una legge di distribuzione. *Giornale dell'Istituto Italiano degli Attuari*. **4**, 83–91.

**See Also**

[pbenf](#), [simulateH0](#)

**Examples**

```
#Set the random seed to an arbitrary number
set.seed(421)
#Create a sample satisfying Benford's law
X<-rbenf(n=20)
#Perform a Kolmogorov-Smirnov Test on the
#sample's first digits using defaults
ks.benftest(X)
#0.7483
```

---

mdist.benftest

*Chebyshev Distance Test (maximum norm) for Benford's Law*

---

**Description**

mdist.benftest takes any numerical vector reduces the sample to the specified number of significant digits and performs a goodness-of-fit test based on the Chebyshev distance between the first digits' distribution and Benford's distribution to assert if the data conforms to Benford's law.

**Usage**

```
mdist.benftest(x = NULL, digits = 1, pvalmethod = "simulate", pvalsims = 10000)
```

**Arguments**

x	A numeric vector.
digits	An integer determining the number of first digits to use for testing, i.e. 1 for only the first, 2 for the first two etc.
pvalmethod	Method used for calculating the p-value. Currently only "simulate" is available.
pvalsims	An integer specifying the number of replicates used if pvalmethod = "simulate".

**Details**

A statistical test is performed utilizing the Chebyshev distance between `signifd(x,digits)` and `pbenf(digits)`. Specifically:

$$m = \max_{i=10^{k-1}, \dots, 10^k-1} |f_i^o - f_i^e| \cdot \sqrt{n}$$

where  $f_i^o$  denotes the observed frequency of digits  $i$ , and  $f_i^e$  denotes the expected frequency of digits  $i$ .  $x$  is a numeric vector of arbitrary length. Values of  $x$  should be continuous, as dictated by theory, but may also be integers. `digits` should be chosen so that `signifd(x,digits)` is not influenced by previous rounding.

**Value**

A list with class "htest" containing the following components:

statistic	the value of the Chebyshev distance (maximum norm) test statistic
p.value	the p-value for the test
method	a character string indicating the type of test performed
data.name	a character string giving the name of the data

**Author(s)**

Dieter William Joenssen <Dieter.Joenssen@googlemail.com>

**References**

- Benford, F. (1938) The Law of Anomalous Numbers. *Proceedings of the American Philosophical Society*. **78**, 551–572.
- Leemis, L.M., Schmeiser, B.W. and Evans, D.L. (2000) Survival Distributions Satisfying Benford's law. *The American Statistician*. **54**, 236–241.
- Morrow, J. (2010) *Benford's Law, Families of Distributions and a Test Basis*. [available under <http://www.johnmorrow.info/projects/benford/benfordMain.pdf>]

**See Also**[pbenf](#), [simulateH0](#)**Examples**

```
#Set the random seed to an arbitrary number
set.seed(421)
#Create a sample satisfying Benford's law
X<-rbenf(n=20)
#Perform a Chebyshev Distance Test on the
#sample's first digits using defaults
mdist.benftest(X)
#p-value = 0.6421
```

---

meandigit.benftest      *Judge-Schechter Mean Deviation Test for Benford's Law*

---

**Description**

meandigit.benftest takes any numerical vector reduces the sample to the specified number of significant digits and performs a goodness-of-fit test based on the deviation in means of the first digits' distribution and Benford's distribution to assert if the data conforms to Benford's law.

**Usage**

```
meandigit.benftest(x = NULL, digits = 1, pvalmethod = "asymptotic", pvalsims = 10000)
```

**Arguments**

x	A numeric vector.
digits	An integer determining the number of first digits to use for testing, i.e. 1 for only the first, 2 for the first two etc.
pvalmethod	Method used for calculating the p-value. Either "asymptotic" or "simulate".
pvalsims	An integer specifying the number of replicates used if pvalmethod = "simulate".

**Details**

A statistical test is performed utilizing the deviation between the mean digit of signifd(x, digits) and pbenf(digits). Specifically:

$$a^* = \frac{|\mu_k^o - \mu_k^e|}{(9 \cdot 10^{k-1}) - \mu_k^e}$$

where  $\mu_k^o$  is the observed mean of the chosen  $k$  number of digits, and  $\mu_k^e$  is the expected/true mean value for Benford's predictions.  $a^*$  conforms asymptotically to a truncated normal distribution under the null-hypothesis, i.e.,

$$a^* \sim \text{truncnorm}(\mu = 0, \sigma = \sigma_B, a = 0, b = \infty)$$

$x$  is a numeric vector of arbitrary length. Values of  $x$  should be continuous, as dictated by theory, but may also be integers. `digits` should be chosen so that `signifd(x,digits)` is not influenced by previous rounding.

### Value

A list with class "htest" containing the following components:

statistic	the value of the $a^*$ test statistic
p.value	the p-value for the test
method	a character string indicating the type of test performed
data.name	a character string giving the name of the data

### Author(s)

Dieter William Joenssen <Dieter.Joenssen@googlemail.com>

### References

Benford, F. (1938) The Law of Anomalous Numbers. *Proceedings of the American Philosophical Society*. **78**, 551–572.

Judge, G. and Schechter, L. (2009) Detecting Problems in Survey Data using Benford's Law. *Journal of Human Resources*. **44**, 1–24.

### See Also

[pbenf](#), [simulateH0](#)

### Examples

```
#Set the random seed to an arbitrary number
set.seed(421)
#Create a sample satisfying Benford's law
X<-rbenf(n=20)
#Perform a Judge-Schechter Mean Deviation Test
#on the sample's first digits using defaults
meandigit.benftest(X)
#p-value = 0.1458
```

---

pbenf

*Probability Mass Function for Benford's Distribution*

---

### Description

Returns the complete probability mass function for Benford's distribution for a given number of first digits.

**Usage**

```
pbenf(digits = 1)
```

**Arguments**

**digits** An integer determining the number of first digits for which the pdf is returned, i.e. 1 for 1:9, 2 for 10:99 etc.

**Details**

Benford's distribution has the following probability mass function:

$$P(d_k) = \log_{10} (1 + d_k^{-1})$$

where  $d_k \in (10^{k-1}, 10^{k-1} + 1, \dots, 10^k - 1)$  for any chosen  $k$  number of digits.

**Value**

Returns an object of class "table" containing the expected density of Benford's distribution for the given number of digits.

**Author(s)**

Dieter William Joenssen <Dieter.Joenssen@googlemail.com>

**References**

Benford, F. (1938) The Law of Anomalous Numbers. *Proceedings of the American Philosophical Society*. **78**, 551–572.

Joenssen, D.W. (2013) Two Digit Testing for Benford's Law. *Proceedings of the ISI World Statistics Congress, 59th Session in Hong Kong*. [available under <http://www.statistics.gov.hk/wsc/CPS021-P2-S.pdf>]

**See Also**

[qbenf](#); [rbenf](#)

**Examples**

```
#show Benford's predictions for the frequencies of the first digit values  
pbenf(1)
```

---

qbenf

*Quantile Function for Benford's Distribution*

---

### Description

Returns the complete quantile function for Benford's distribution with a given number of first digits.

### Usage

```
qbenf(digits = 1)
```

### Arguments

`digits` An integer determining the number of first digits for which the qdf is returned, i.e. 1 for 1:9, 2 for 10:99 etc.

### Value

Returns an object of class "table" containing the expected quantile function of Benford's distribution with a given number of digits.

### Author(s)

Dieter William Joenssen <Dieter.Joenssen@googlemail.com>

### References

Benford, F. (1938) The Law of Anomalous Numbers. *Proceedings of the American Philosophical Society*. **78**, 551–572.

### See Also

[pbenf](#); [rbenf](#)

### Examples

```
qbenf(1)
```

```
qbenf(1)==cumsum(pbenf(1))
```



---

rbenf	<i>Random Sample Satisfying Benford's Law</i>
-------	---

---

**Description**

Returns a random sample with length n satisfying Benford's law.

**Usage**

```
rbenf(n)
```

**Arguments**

n                      Number of observations.

**Details**

This distribution has the density:

$$f(x) = \frac{1}{x \cdot \ln(10)} \forall x \in [1, 10]$$

**Value**

Returns a random sample with length n satisfying Benford's law.

**Author(s)**

Dieter William Joenssen <Dieter.Joenssen@googlemail.com>

**References**

Benford, F. (1938) The Law of Anomalous Numbers. *Proceedings of the American Philosophical Society*. **78**, 551–572.

**See Also**

[qbenf](#); [pbenf](#)

**Examples**

```
#Set the random seed to an arbitrary number
set.seed(421)
#Create a sample satisfying Benford's law
X<-rbenf(n=20)
#Look at sample
X
#should be
# [1] 6.159420 1.396476 5.193371 2.064033 7.001284 5.006184
#7.950332 4.822725 3.386809 1.619609 2.080063 2.242473 1.944697 5.460581
#[15] 6.443031 2.662821 2.079283 3.703353 1.364175 3.354136
```

---

`signifd`*First Digits Function*

---

**Description**

Applies the first digits function to each element of a given vector.

**Usage**

```
signifd(x = NULL, digits = 1)
```

**Arguments**

<code>x</code>	A numeric vector.
<code>digits</code>	An integer determining the number of first digits to use for testing, i.e. 1 for only the first, 2 for the first two etc.

**Details**

The first digits function can be written as:

$$D_k(x) = \lfloor |x| \cdot 10^{(-1 \cdot \lfloor \log_{10}|x| \rfloor + k - 1)} \rfloor$$

with  $k$  being the number of first digits that should be extracted.  $x$  is a numeric vector of arbitrary length. Unlike other solutions, this function will work reliably with all real numbers.

**Value**

Returns a vector of integers the same length as the input vector  $x$ .

**Author(s)**

Dieter William Joenssen <Dieter.Joenssen@googlemail.com>

**References**

Joenssen, D.W. (2013) Two Digit Testing for Benford's Law. *Proceedings of the ISI World Statistics Congress, 59th Session in Hong Kong*. [available under <http://www.statistics.gov.hk/wsc/CPS021-P2-S.pdf>]

**See Also**

[chisq.benftest](#); [ks.benftest](#); [usq.benftest](#); [mdist.benftest](#); [edist.benftest](#); [meandigit.benftest](#); [jpsq.benftest](#)

**Examples**

```
#Set the random seed to an arbitrary number
set.seed(421)
#Create a sample satisfying Benford's law
X<-rbenf(n=20)
#Look at the first digits of the sample
signifd(X)
#should be:
#[1] 6 1 5 2 7 5 7 4 3 1 2 2 1 5 6 2 2 3 1 3
```

signifd.analysis

*Graphical Analysis of First Significant Digits***Description**

signifd.analysis takes any numerical vector reduces the sample to the specified number of significant digits. The (relative) frequencies are then plotted so that a subjective analysis may be performed.

**Usage**

```
signifd.analysis(x = NULL, digits = 1, graphical_analysis = TRUE, freq = FALSE,
alphas = 20, tick_col = "red", ci_col = "darkgreen", ci_lines = c(.05))
```

**Arguments**

x	A numeric vector.
digits	An integer determining the number of first digits to use for testing, i.e. 1 for only the first, 2 for the first two etc.
graphical_analysis	Boolean value indicating if results should be plotted.
freq	Boolean value indicating if absolute frequencies should be used.
alphas	Either a vector containing the significance levels([0,1]) that will be shaded, or an integer defining the number of evenly spaced confidence intervals.
tick_col	Color code or name that will be passed to "points" for plotting.
ci_col	Color code or name that will be passed to "polygon" for shading the different confidence intervals. May be more than one color.
ci_lines	Boolean or fractional value(s) indicating significance levels where lines are drawn

**Details**

Confidence intervals are calculated from the normal distribution with  $\mu_i = np_i$  and  $\sigma^2 = np_i(1 - p_i)$ , where  $i$  represents the considered digit. Be aware that the normal approximation only holds for "large"  $n$ .

**Value**

A list containing the following components:

summary	the summary printed below the graph, a matrix of digits, their (relative) frequencies and individual p-values
CIs	confidence intervals used for plotting as defined by parameter "ci_lines" or "alphas" if ci_lines==FALSE

**Author(s)**

Dieter William Joenssen <Dieter.Joenssen@googlemail.com>

**References**

- Benford, F. (1938) The Law of Anomalous Numbers. *Proceedings of the American Philosophical Society*. **78**, 551–572.
- Freedman, L.S. (1981) Watson's Un2 Statistic for a Discrete Distribution. *Biometrika*. **68**, 708–711.

**See Also**

[pbenf](#)

**Examples**

```
#Set the random seed to an arbitrary number
set.seed(421)
#Create a sample satisfying Benford's law
X<-rbenf(n=20)
#Analyze the first digits using the the defaults
signifd.analysis(X)
#Turn off plot
signifd.analysis(X,graphical_analysis=FALSE)
#Use absolute frequencies
signifd.analysis(X,graphical_analysis=FALSE,freq=TRUE)
#Use five evenly spaced confidence intervals, no lines
#alphas is used for shadeing
signifd.analysis(X,graphical_analysis=TRUE,alphas=5,freq=TRUE,ci_lines=FALSE)
#Use fifty evenly spaced, gray confidence intervals, blue ticks, and lines at
#the 1 and 5 percent confidence intervals
signifd.analysis(X,graphical_analysis=TRUE,alphas=50,freq=TRUE,tick_col="blue",
ci_col="gray",ci_lines=c(.01,.05))
```

---

`signifd.seq`*Sequence of Possible Leading Digits*

---

**Description**

Returns a vector containing all possible significant digits for a given number of places.

**Usage**

```
signifd.seq(digits = 1)
```

**Arguments**

`digits` An integer determining the number of first digits to be returned, i.e. 1 for 1:9, 2 for 10:99 etc.

**Value**

Returns an integer vector.

**Author(s)**

Dieter William Joenssen <Dieter.Joenssen@googlemail.com>

**Examples**

```
signifd.seq(1)
seq(from=1, to=9)==signifd.seq(1)

signifd.seq(2)
seq(from=10, to=99)==signifd.seq(2)
```

---

`simulateH0`*Function for Simulating the H0-Distributions needed for BenfordTests*

---

**Description**

`simulateH0` is a wrapper function that calculates the specified test statistic under the null hypothesis a certain number of times.

**Usage**

```
simulateH0(teststatistic="chisq", n=10, digits=1, pvalsims=10)
```

**Arguments**

teststatistic	Which test statistic should be used: "chisq", "edist", "jpsq", "ks", "mdist", "meandigit", or "usq".
n	Sample size of interest.
digits	An integer determining the number of first digits to use for testing, i.e. 1 for only the first, 2 for the first two etc.
pvalsims	An integer specifying the number of replicates to be used in simulation.

**Details**

Wrapper function that directly outputs the distributions of the specified test statistic under the null hypothesis.

**Value**

A vector of length equal to "pvalsims".

**Author(s)**

Dieter William Joenssen <Dieter.Joenssen@googlemail.com>

**References**

Benford, F. (1938) The Law of Anomalous Numbers. *Proceedings of the American Philosophical Society*. **78**, 551–572.

Joenssen, D.W. (2013) Two Digit Testing for Benford's Law. *Proceedings of the ISI World Statistics Congress, 59th Session in Hong Kong*. [available under <http://www.statistics.gov.hk/wsc/CPS021-P2-S.pdf>]

**See Also**

[pbenf](#), [chisq.benfctest](#), [edist.benfctest](#), [jpsq.benfctest](#), [ks.benfctest](#), [mdist.benfctest](#), [meandigit.benfctest](#), [usq.benfctest](#)

**Examples**

```
#Set the random seed to an arbitrary number
set.seed(421)

#calculate critical value for chisquare test via simulation
quantile(simulateH0(teststatistic="chisq", n=100,digits=1,pvalsims=100000),probs=.95)

#calculate the "real" critical value
qchisq(.95,df=8)

#alternatively look at critical values for the jpsq statistic
#for different sample sizes (notice the low value for pvalsims)
set.seed(421)
apply(sapply((1:9)*10,FUN=simulateH0,teststatistic="jpsq", digits=1, pvalsims=100),
MARGIN=2,FUN=quantile,probs=.05)
```

usq.benftest

*Freedman-Watson U-square Test for Benford's Law***Description**

usq.benftest takes any numerical vector reduces the sample to the specified number of significant digits and performs the Freedman-Watson test for discrete distributions between the first digits' distribution and Benford's distribution to assert if the data conforms to Benford's law.

**Usage**

```
usq.benftest(x = NULL, digits = 1, pvalmethod = "simulate", pvalsims = 10000)
```

**Arguments**

x	A numeric vector.
digits	An integer determining the number of first digits to use for testing, i.e. 1 for only the first, 2 for the first two etc.
pvalmethod	Method used for calculating the p-value. Currently only "simulate" is available.
pvalsims	An integer specifying the number of replicates used if pvalmethod = "simulate".

**Details**

A Freedman-Watson test for discrete distributions is performed between `signifd(x,digits)` and `pbenf(digits)`. Specifically:

$$U^2 = \frac{n}{9 \cdot 10^{k-1}} \cdot \left[ \sum_{i=10^{k-1}}^{10^k-2} \left( \sum_{j=1}^i (f_j^o - f_j^e) \right)^2 - \frac{1}{9 \cdot 10^{k-1}} \cdot \left( \sum_{i=10^{k-1}}^{10^k-2} \sum_{j=1}^i (f_i^o - f_i^e) \right)^2 \right]$$

where  $f_i^o$  denotes the observed frequency of digits  $i$ , and  $f_i^e$  denotes the expected frequency of digits  $i$ .  $x$  is a numeric vector of arbitrary length. Values of  $x$  should be continuous, as dictated by theory, but may also be integers. `digits` should be chosen so that `signifd(x,digits)` is not influenced by previous rounding.

**Value**

A list with class "htest" containing the following components:

statistic	the value of the $U^2$ test statistic
p.value	the p-value for the test
method	a character string indicating the type of test performed
data.name	a character string giving the name of the data

**Author(s)**

Dieter William Joenssen <Dieter.Joenssen@googlemail.com>

**References**

Benford, F. (1938) The Law of Anomalous Numbers. *Proceedings of the American Philosophical Society*. **78**, 551–572.

Freedman, L.S. (1981) Watson's Un<sup>2</sup> Statistic for a Discrete Distribution. *Biometrika*. **68**, 708–711.

Joenssen, D.W. (2013) Two Digit Testing for Benford's Law. *Proceedings of the ISI World Statistics Congress, 59th Session in Hong Kong*. [available under <http://www.statistics.gov.hk/wsc/CPS021-P2-S.pdf>]

Watson, G.S. (1961) Goodness-of-Fit Tests on a Circle. *Biometrika*. **48**, 109–114.

**See Also**

[pbenf](#), [simulateH0](#)

**Examples**

```
#Set the random seed to an arbitrary number
set.seed(421)
#Create a sample satisfying Benford's law
X<-rbenf(n=20)
#Perform Freedman-Watson U-squared Test on
#the sample's first digits using defaults
usq.benftest(X)
#p-value = 0.4847
```



# Index

## \*Topic **datagen**

BenfordTests-package, 2  
rbenf, 17

## \*Topic **distribution**

BenfordTests-package, 2  
pbenf, 14  
qbenf, 16  
rbenf, 17  
simulateH0, 21

## \*Topic **hplot**

signifd.analysis, 19

## \*Topic **htest**

BenfordTests-package, 2  
chisq.benftest, 4  
edist.benftest, 5  
jointdigit.benftest, 7  
jpsq.benftest, 8  
ks.benftest, 10  
mdist.benftest, 11  
meandigit.benftest, 13  
simulateH0, 21  
usq.benftest, 23

## \*Topic **manip**

BenfordTests-package, 2  
signifd, 18

## \*Topic **package**

BenfordTests-package, 2

BenfordTests (BenfordTests-package), 2

BenfordTests-package, 2

chisq.benftest, 4, 18, 22

edist.benftest, 5, 18, 22

jointdigit.benftest, 7

jpsq.benftest, 8, 18, 22

ks.benftest, 10, 18, 22

mdist.benftest, 11, 18, 22

meandigit.benftest, 13, 18, 22

pbenf, 5, 6, 8, 10, 11, 13, 14, 14, 16, 17, 20,  
22, 24

qbenf, 15, 16, 17

rbenf, 15, 16, 17

signifd, 18

signifd.analysis, 19

signifd.seq, 21

simulateH0, 5, 6, 10, 11, 13, 14, 21, 24

usq.benftest, 18, 22, 23