Package: benford.analysis (via r-universe)

October 31, 2026

Type Package

Title Benford Analysis for Data Validation and Forensic Analytics

Version 0.1.5

Author Carlos Cinelli

Maintainer Carlos Cinelli <carloscinelli@hotmail.com>

Description Provides tools that make it easier to validate data using Benford's Law.

Depends R (>= 3.0.0)

Imports data.table

License GPL-3

Suggests testthat

URL http://github.com/carloscinelli/benford.analysis

BugReports http://github.com/carloscinelli/benford.analysis/issues

RoxygenNote 6.1.1

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

Date/Publication 2019-08-24 05:04:03 UTC

RemoteUrl https://github.com/carloscinelli/benford.analysis

RemoteRef HEAD

RemoteSha 31a98c4fafaf801322a933a64700a9cf697f5ec1

Contents

benford 2
benford.analysis 4
census.2000_2010 6
census.2009 6
chisq 7
corporate.payment 7
dfactor 8

duplicatesTable . . . 8
extract.digits . . . 9
fibonacci.500 . . . 10
getBfd . . . 10
getData . . . 11
getDigits . . . 11
getDuplicates . . . 12
getSuspects . . . 13
gm.payments . . . 13
journal.entry . . . 14
ks . . . 14
lakes.perimeter . . . 15
last.two.digits . . . 15
MAD . . . 16
MAD.conformity . . . 16
madoff.returns . . . 17
mantissa . . . 17
marc . . . 18
p.these.digits . . . 18
p.this.digit.at.n . . . 19
plot.Benford . . . 19
print.Benford . . . 20
purchasing.cards.2010 . . . 21
sino.forest . . . 21
streamflow . . . 22
suspectsTable . . . 22
taxable.incomes.1978 . . . 23

Index **24**

`benford` *Benford Analysis of a dataset*

Description

This function validates a dataset using Benford's Law. Its main purposes are to find out where the dataset deviates from Benford's Law and to identify suspicious data that need further verification.

For a more complete example, see the package help at benford.analysis.

Usage

```
benford(data, number.of.digits = 2, sign = "positive",
  discrete = TRUE, round = 3, data.name = NULL)
```

Arguments

`data`	a numeric vector.
`number.of.digits`	how many first digits to analyze.
`sign`	The default value for sign is "positive" and it analyzes only data greater than zero. There are also the options "negative" and "both" that will analyze only negative values or both positive and negative values of the data, respectively. For large datasets with both positive and negative numbers, it is usually recommended to perform a separate analysis for each group, for the incentives to manipulate the numbers are usually different.
`discrete`	most real data - like population numbers or accounting data - are discrete, so the default is TRUE. This parameter sets rounding to the differences of the ordered data to avoid floating point number errors in the second order distribution, that usually occurs when data is discrete and the ordered numbers are very close to each other. If your data is continuous (like a simulated lognormal) you should run with discrete = FALSE.
`round`	it defines the number of digits that the rounding will use if discrete = TRUE.
`data.name`	the name of your data to show. If `NULL` (the default), the name of the passed object will be used.

Value

An object of class Benford containing the results of the analysis. It is a list of eight objects, namely:

`info`	general information, including • data.name: the name of the data used. • n: the number of observations used. • n.second.order: the number of observations used for second order analysis. • number.of.digits: the number of first digits analyzed.
`data`	a data frame with: • lines.used: the original lines of the dataset. • data.used: the data used. • data.mantissa: the log data's mantissa. • data.digits: the first digits of the data.
`s.o.data`	a data frame with: • data.second.order: the differences of the ordered data. • data.second.order.digits: the first digits of the second order analysis.
`bfd`	a data frame with: • digits: the groups of digits analyzed. • data.dist: the distribution of the first digits of the data. • data.second.order.dist: the distribution of the first digits of the second order analysis. • benford.dist: the theoretical benford distribution.

- data.second.order.dist.freq: the frequency distribution of the first digits of the second order analysis.
- data.dist.freq: the frequency distribution of the first digits of the data.
- benford.dist.freq: the theoretical benford frequency distribution.
- benford.so.dist.freq: the theoretical benford frequency distribution of the second order analysis.
- data.summation: the summation of the data values grouped by first digits.
- abs.excess.summation: the absolute excess summation of the data values grouped by first digits.
- difference: the difference between the data and benford frequencies.
- squared.diff: the chi-squared difference between data and benford frequencies.
- absolute.diff: the absolute difference between data and benford frequencies.
- z.statistic: the z-statistic difference between data and benford relative frequencies.

`mantissa` a data frame with:

- mean.mantissa: the mean of the mantissa.
- var.mantissa: the variance of the mantissa.
- ek.mantissa: the excess kurtosis of the mantissa.
- sk.mantissa: the skewness of the mantissa.

`MAD` the mean absolute deviation.

`distortion.factor`
the distortion factor

`stats` list of "htest" class statistics:

- chisq: Pearson's Chi-squared test.
- mantissa.arc.test: Mantissa Arc Test.
- ks.test: Kolmogorov-Smirnov Test.

Examples

```
data(corporate.payment) #loads data
bfd.cp <- benford(corporate.payment$Amount) #generates benford object
bfd.cp #prints
plot(bfd.cp) #plots
```

`benford.analysis` *Benford Analysis for data validation and forensic analytics*

Description

The Benford Analysis package provides tools that make it easier to validate data using Benford's Law. The main purpose of the package is to identify suspicious data that need further verification.

Details

More information can be found on its help documentation.

The main function is benford. It generates a Benford S3 object.

The package defines S3 methods for plotting and printing Benford type objects.

After running benford you can easily get the "suspicious" data by using the functions: suspectsTable, getSuspects, duplicatesTable and getDuplicates. See help documentation and examples for further details.

The package also includes 6 real datasets for illustration purposes.

References

Alexander, J. (2009). Remarks on the use of Benford's Law. Working Paper, Case Western Reserve University, Department of Mathematics and Cognitive Science.

Berger, A. and Hill, T. (2011). A basic theory of Benford's Law. Probability Surveys, 8, 1-126.

Hill, T. (1995). A statistical derivation of the significant-digit law. Statistical Science, 10(4), 354-363.

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

Nigrini, M. J. (2011). Forensic Analytics: Methods and Techniques for Forensic Accounting Investigations.Wiley and Sons: New Jersey.

Examples

```
data(corporate.payment) #gets data
cp <- benford(corporate.payment$Amount, 2, sign="both") #generates benford object
cp #prints
plot(cp) #plots

head(suspectsTable(cp),10) #prints the digits by decreasing order of discrepancies

#gets observations of the 2 most suspicious groups
suspects <- getSuspects(cp, corporate.payment, how.many=2)

duplicatesTable(cp) #prints the duplicates by decreasing order

#gets the observations of the 2 values with most duplicates
duplicates <- getDuplicates(cp, corporate.payment,how.many=2)

MAD(cp) #gets the Mean Absolute Deviation

chisq(cp) #gets the Chi-squared test

#gets observations starting with 50 or 99
```

```
digits_50_and_99 <- getDigits(cp, corporate.payment, digits=c(50, 99))
```

`census.2000_2010` *Population data - US - 2000 and 2010*

Description

A dataset containing population data of the United States - 2000 and 2010.

Format

A data frame with 3143 rows and 5 variables

References

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

`census.2009` *Population data of Towns and Cities of the US - 2009*

Description

A dataset containing the population of towns and cities of the United States, as of july of 2009.

Format

A data frame with 19509 rows and 3 variables

References

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

chisq *Gets the Chi-squared test of a Benford object*

Description

It gets the Chi-squared test for a Benford object. See the section value of benford.

Usage

```
chisq(bfd)
```

Arguments

bfd	an object of class "Benford". See benford.

Value

A list with class "htest" containing the results of the Chi-squared test.

Examples

```
data(census.2009) #gets data
c2009 <- benford(census.2009$pop.2009) #generates benford object
chisq(c2009) # equivalent to c2009$stats$chisq
```

corporate.payment *Corporate payments of a West Coast utility company - 2010*

Description

A dataset of the 2010's payments data of a division of a West Coast utility company.

Format

A data frame with 189470 rows and 4 variables

References

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

dfactor *Gets the Distortion Factor of a Benford object*

Description

It gets the Distortion Factor of a Benford object. See the section value of benford.

Usage

```
dfactor(bfd)
```

Arguments

bfd	an object of class "Benford". See benford.

Value

The distortion factor.

Examples

```
data(corporate.payment) #gets data
cp <- benford(corporate.payment$Amount) #generates benford object
dfactor(cp) # equivalent to cp$distortion.factor
```

duplicatesTable *Shows the duplicates of the data*

Description

It creates a data frame with the duplicates in decreasing order.

Usage

```
duplicatesTable(bfd)
```

Arguments

bfd	an object of class "Benford". See benford.

Value

A data frame with 2 variables: number and duplicates.

Examples

```
data(census.2009) #gets data
c2009 <- benford(census.2009$pop.2009) #generates benford object
duplicatesTable(c2009)
```

`extract.digits` *Extracts the leading digits from the data*

Description

It extracts the leading digits from the data.

This function is used by the main function of the package `benford` to extract the leading digits of the data.

Usage

```
extract.digits(data, number.of.digits = 2,
               sign="positive", second.order = FALSE, discrete=TRUE, round=3)
```

Arguments

`data`	a numeric vector.
`number.of.digits`	how many first digits to analyze .
`sign`	The default value for sign is "positive" and it analyzes only data greater than zero. There are also the options "negative" and "both" that will analyze only negative values or both positive and negative values of the data, respectively. For large datasets with both positive and negative numbers, it is usually recommended to perform a separate analysis for each group, for the incentives to manipulate the numbers are usually different.
`second.order`	If TRUE, the function will extract the first digits of the second order distribution.
`discrete`	Most real data - like population numbers or accounting data - are discrete, so the default is TRUE. This paramater sets rounding to the differences of the ordered data to avoid floating point number errors in the second order distribution, that usually occurs when data is discrete and the ordered numbers are very close to each other. If your data is continuous (like a simulated lognormal) you should run with discrete = FALSE.
`round`	it defines the number of digits that the rounding will use if discrete = TRUE and second.order = TRUE.

Value

A data.frame with the data and the first digits.

fibonacci.500 *The first 500 Fibonacci numbers.*

Description

A dataset containing the first 500 Fibonacci numbers.

Format

A data frame with 500 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, pp. 51-53.

getBfd *Gets the the statistics of the first Digits of a benford object*

Description

It gets the statistics of the first digits (Frequencies, Squared Differences, Absolute Differences etc). See the section value of benford.

Usage

```
getBfd(bfd)
```

Arguments

bfd	an object of class "Benford". See benford.

Value

A data.frame with first digits and their statistics.

Examples

```
data(corporate.payment)
cp <- benford(corporate.payment$Amount) #generates benford object
getBfd(cp) # equivalent to cp$bfd
```

getData *Gets the data used of a Benford object*

Description

It gets the lines, values, mantissa and first digits of the data used of a Benford object . See the section value of benford.

Usage

```
getData(bfd)
```

Arguments

bfd	an object of class "Benford". See benford.

Value

A data.frame with the lines, values, mantissa and first digits of the data.

Examples

```
data(corporate.payment)
cp <- benford(corporate.payment$Amount) #generates benford object
getData(cp) # equivalent to cp$data
```

getDigits *Gets the data starting with some specific digits*

Description

It subsets the original data according to the leading digits.

Usage

```
getDigits(bfd, data, digits)
```

Arguments

bfd	an object of class "Benford". See benford.
data	the original data of the analysis.
digits	the first digits to get.

Value

The the original data starting only with the leading digits.

Examples

```
data(census.2000_2010) #gets data

#generates benford object
c2010 <- benford(census.2000_2010$pop.2010)

#subsets data starting with digits 10 and 25
digits.10.25 <- getDigits(c2010, census.2000_2010, c(10,25))
```

`getDuplicates` *Gets the duplicates from data*

Description

It gets the duplicates from the original data.

Usage

```
getDuplicates(bfd, data, how.many=2)
```

Arguments

`bfd`	an object of class "Benford". See `benford`.
`data`	the original data used for the benford analysis.
`how.many`	how many groups of duplicates to get.

Value

The duplicates from the original data.

Examples

```
data(census.2000_2010) #gets data
c2010 <- benford(census.2000_2010$pop.2010) #generates benford object
duplicates <- getDuplicates(c2010, census.2000_2010)
```

getSuspects *Gets the 'suspicious' observations according to Benford's Law*

Description

It gets the original data from the 'suspicious' digits groups according to benford analysis.

Usage

```
getSuspects(bfd, data, by="absolute.diff", how.many=2)
```

Arguments

bfd	an object of class "Benford". See benford.
data	the original data used for the benford analysis.
by	a character string selecting how to order the digits.It can be 'abs.excess.summation','difference','squared.d or 'absolute.diff'.
how.many	how many groups of digits to get.

Value

The 'suspicious' observations from the original data.

Examples

```
data(lakes.perimeter) #gets data
lk <- benford(lakes.perimeter[,1]) #generates benford object
suspects <- getSuspects(lk, lakes.perimeter)
```

gm.payments *General Motors payments.*

Description

A dataset containing the amounts paid to vendors for the 90 days preceding liquidation - 2009.

Format

A data frame with 15300 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, pp. 253-255.

journal.entry *Journal entries in a company's accounting system.*

Description

A dataset related to the 2005 journal entries in a company's accounting system.

Format

A data frame with 154935 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, pp. 104-107.

ks *Gets the Kolmogorov-Smirnov test of a Benford object*

Description

It gets the Kolmogorov-Smirnov test for a Benford object. See the section value of benford.

Usage

```
ks(bfd)
```

Arguments

bfd	an object of class "Benford". See benford.

Value

A list with class "htest" containing the results of the Kolmogorov-Smirnov test.

Examples

```
data(census.2009) #gets data
c2009 <- benford(census.2009$pop.2009) #generates benford object
ks(c2009) # equivalent to c2009$stats$ks.test
```

lakes.perimeter *Perimeter of lakes around the world*

Description

A dataset of the perimeter of the lakes arround the water from the global lakes and wetlands database (GLWD) <http://www.worldwildlife.org/pages/global-lakes-and-wetlands-database>.

Format

A data frame with 248607 rows and 1 variable.

References

Lehner, B. and Doll, P. (2004). Development and validation of a global database of lakes, reservoirs and wetlands. Journal of Hydrology, 296(1), pp.1-22.

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

last.two.digits *Extracts the last two digits from the data*

Description

It extracts the last two digits from the data.

This function is used by the main function of the package benford to extract the ast two digits of the data.

Usage

```
last.two.digits(data, sign = "positive")
```

Arguments

data	a numeric vector.
sign	The default value for sign is "positive" and it analyzes only data greater than zero. There are also the options "negative" and "both" that will analyze only negative values or both positive and negative values of the data, respectively. For large datasets with both positive and negative numbers, it is usually recommended to perform a separate analysis for each group, for the incentives to manipulate the numbers are usually different.

Value

A data.frame with the data and the last digits.

MAD *Gets the MAD of a Benford object*

Description

It gets the Mean Absolute Deviation (MAD) of a Benford object. See the section value of benford.

Usage

```
MAD(bfd)
```

Arguments

bfd	an object of class "Benford". See benford.

Value

The MAD.

Examples

```
data(census.2000_2010) #gets data
c2010 <- benford(census.2000_2010$pop.2010) #generates benford object
MAD(c2010) #equivalent to c2010$MAD
```

MAD.conformity *MAD conformity to Benford's Law using the MAD*

Description

This function checks the MAD against the conformity criteria proposed by Nigrini (2012).

Usage

```
MAD.conformity(MAD = NULL, digits.used = c("First Digit",
  "Second Digit", "First-Two Digits", "First-Three Digits"))
```

Arguments

MAD	The mean absolute deviation, as computed in the function benford
digits.used	How many digits used in the analysis.

Value

A list with the MAD, digits.used and the conformity level.

References

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

`madoff.returns` *The potential amount of the exposure of a partial list released by Wall Street Journal (WSJ) of the Madoff victims - 2009.*

Description

A dataset containing the potential amount of the exposure of a partial list of the Madoff victims.

Format

A data frame with 215 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, pp. 259-262.

`mantissa` *Gets the main stats of the Mantissa of a Benford object*

Description

It gets the Mean, Variance, Excess Kurtosis and Skewness of the Mantissa. See the section value of `benford`.

Usage

```
mantissa(bfd)
```

Arguments

`bfd`	an object of class "Benford". See `benford`.

Value

A data.frame with the main stats of the Mantissa.

Examples

```
data(corporate.payment) #gets data
cp <- benford(corporate.payment$Amount) #generates benford object
mantissa(cp) # equivalent to cp$mantissa
```

marc *Gets the Mantissa Arc test of a Benford object*

Description

It gets the Mantissa Arc Test of a Benford object. See the section value of benford.

Usage

```
marc(bfd)
```

Arguments

bfd	an object of class "Benford". See benford.

Value

A list with class "htest" containing the results of the Mantissa Arc test.

Examples

```
data(corporate.payment) #gets data
cp <- benford(corporate.payment$Amount) #generates benford object
marc(cp) # equivalent to cp$stats$mantissa.arc.test
```

p.these.digits *Probability of a digit sequence*

Description

It calculates the probability of a digit sequence "d".

Usage

```
p.these.digits(d)
```

Arguments

d	a digit sequence, like 1234 ou 999999.

Value

The probability of the sequence d.

Examples

```
p.these.digits(1) # 0.30103
p.these.digits(11) # 0.03778856
p.these.digits(999999) # 4.342947e-07
```

p.this.digit.at.n *Probability of a digit at the nth position*

Description

It calculates the probability of digit "d" at the "n"th position.

Usage

```
p.this.digit.at.n(d,n)
```

Arguments

d	a digit from 0 to 9 (except at position n=1, where d cannot be 0, it will give you NA).
n	the nth position.

Value

The probability of d at position n.

Examples

```
p.this.digit.at.n(1,1) # 0.30103
p.this.digit.at.n(1,2) # 0.1138901
p.this.digit.at.n(9,3) # 0.09826716
matrix <- as.data.frame(round(sapply(1:4, function(x) sapply(0:9,p.this.digit.at.n,n=x)),5))
names(matrix) <- paste0("n=",1:4)
rownames(matrix) <- paste0("d=",0:9)
matrix # a table with the probabilities of digits 0 to 9 in positions 1 to 4.
```

plot.Benford *Plot method for Benford Analysis*

Description

The plot method for "Benford" objects.

Usage

```
## S3 method for class 'Benford'
plot(x, select = c("digits", "second order",
  "summation", "chi squared", "ex summation"), except = NULL,
  multiple = TRUE, col.bar = "lightblue", err.bounds = FALSE,
  alpha = 0.05, grid = TRUE, ...)
```

Arguments

x	a "Benford" object
select	it specifies the order and which plots are going to be plotted. If NULL, the parameter except is used.
except	it specifies which plots are not going to be plotted. If NULL, the parameter select is used. Currently, you can choose from 9 plots: "digits", "rootogram digits", "second order", "rootogram second order", "summation", "mantissa", "chi square", "abs diff", "ex summation". If you want to plot all, just put except = "none". The default is not to plot the "mantissa" and "abs diff". If you want to plot all, just put except = "all"
multiple	if TRUE, all plots are grouped in the same window.
col.bar	a color to be used to fill the bars. The default is lightblue.
err.bounds	if TRUE, the upper and lower error bounds are draw. The error bounds indicate the binomial root mean square error.
alpha	it specifies level of confidence interval. The defaults to 95 percent confidence interval,i.e., the error bounds will represent 1.96 standard error from the expected count by Benford's Law.
grid	if TRUE, adds an rectangular grid to plot.
...	arguments to be passed to generic plot functions,

Value

Plots the Benford object.

---

print.Benford *Print method for Benford Analysis*

Description

The print method for "Benford" objects.

Usage

```
## S3 method for class 'Benford'
print(x, how.many=5, ...)
```

Arguments

x	a "Benford" object.
how.many	a number that defines how many of the biggest absolute differences to show.
...	arguments to be passed to generic print functions.

Value

Prints the Benford object.

purchasing.cards.2010 *Purchasing cards transactions - 2010.*

Description

A dataset containing the card transactions for a government entity - 2010.

Format

A data frame with 95076 rows and 8 variables.

References

Nigrini, M. J. (2011). Forensic Analytics: Methods and Techniques for Forensic Accounting Investigations. Wiley and Sons: New Jersey, pp. 435-438.

sino.forest *Financial Statements of Sino Forest Corporation's 2010 Report*

Description

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

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.

streamflow *Annual average flows.*

Description

A dataset containing all the annual average flows for any site (a gage) that had an annual average recorded in any of the years from 1950 to 2005.

Format

A data frame with 457440 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, pp. 268-274.

suspectsTable *Shows the first digits ordered by the mains discrepancies from Benford's Law*

Description

It creates a data frame with the first digits and the differences from Benford's Law in decreasing order.

Usage

```
suspectsTable(bfd, by="absolute.diff")
```

Arguments

bfd	an object of class "Benford". See benford.
by	a character string selecting how to order the digits.It can be 'abs.excess.summation','difference','squared.d or 'absolute.diff'.

Value

A data frame with 2 variables: digits and the group chosen in by.

Examples

```
data(corporate.payment) #gets data
cp <- benford(corporate.payment$Amount) #generates benford object
suspectsTable(cp)
```

`taxable.incomes.1978` *Taxable Income 1978*

Description

Taxable Incomes of the 1978 Individual Tax Model File (ITMF).

Format

A data frame with 157518 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.

Index

∗ **dataset**
census.2000_2010, 6
census.2009, 6
corporate.payment, 7
fibonacci.500, 10
gm.payments, 13
journal.entry, 14
lakes.perimeter, 15
madoff.returns, 17
purchasing.cards.2010, 21
sino.forest, 21
streamflow, 22
taxable.incomes.1978, 23

benford, 2, *5*, *7–18*, *22*
benford.analysis, 2, 4
benford.analysis-package *(*benford.analysis*)*, 4

census.2000_2010, 6
census.2009, 6
chisq, 7
corporate.payment, 7

dfactor, 8
duplicatesTable, *5*, 8

extract.digits, 9

fibonacci.500, 10

getBfd, 10
getData, 11
getDigits, 11
getDuplicates, *5*, 12
getSuspects, *5*, 13
gm.payments, 13

journal.entry, 14

ks, 14

lakes.perimeter, 15
last.two.digits, 15

MAD, 16
MAD.conformity, 16
madoff.returns, 17
mantissa, 17
marc, 18

p.these.digits, 18
p.this.digit.at.n, 19
plot.Benford, 19
print.Benford, 20
purchasing.cards.2010, 21

sino.forest, 21
streamflow, 22
suspectsTable, *5*, 22

taxable.incomes.1978, 23