# Package 'MorseGen'

February 19, 2015

**Type** Package

**Title** Simple raw data generator based on user-specified summary statistics.

**Version** 1.2

**Date** 2012-06-04

**Author** Brendan Morse

**Maintainer** Brendan Morse <bmorse@bridgew.edu>

**Description** MorseGen is a program for generating raw data based on user-specified summary (descriptive) statistics. Samples based on the supplied statistics are drawn from a normal distribution (or, in some cases, an exponential distribution) and scaled to match the desired descriptive statistics. Intended uses include creating raw data that fits desired characteristics or to replicate the results in a published study.

**License** GPL-2

**Repository** CRAN

**Date/Publication** 2012-06-04 17:20:21

**NeedsCompilation** no

## R topics documented:

## MorseGen-package *Simple raw data generator based on user-specified summary statistics*

### Description

MorseGen is a program for generating raw data based on user-specified summary (descriptive) statistics. Samples based on the supplied statistics are drawn from an appropriate distribution (normal or exponential) and scaled to match the desired descriptive statistics. Intended uses include creating raw data that fits desired characteristics or to replicate the results in a published study.

### Details

| | |
|---|---|
| Package: | MorseGen |
| Type: | Package |
| Version: | 1.2 |
| Date: | 2012-06-04 |
| License: | GPL-2 |

MorseGen includes three different functions for generating data based on user-specified summary (descriptive) statistics. The stat.gen function will create a single vector of N scores with a specified mean and standard deviation. The cor.gen function will create two vectors of N scores with the specified means and standard deviations as well as a specified correlation.

### Author(s)

Brendan Morse

Maintainer: Brendan Morse <bmorse@bridgew.edu>

### Examples

```
# Generate a sample of 10 scores
# with a mean equal to 18.4 and a
# standard deviation of 3.2.

stat.gen(10,18.4,3.2)

# Generate a sample of 20 scores with
# a mean equal to 9.1, and a standard
# deviation of 3.2. Additionally, the
# raw data must be in positive, whole-
# unit integers.

stat.gen(10,18.4,3.2,data.dec=0,neg.data=FALSE)

# Generate two variables (with means 5.65
# & 100.2 and standard deviations 2.13 & 16.8
# respectively) from 15 individuals that are
# correlated at r = -.68

cor.gen(15,5.65,2.13,100.2,16.8,-.68)
```

## `cor.gen` *cor.gen*

### Description

cor.gen is a function to generate two vectors of N scores drawn from a normal distribution with the user-specified means, standard deviations, and correlation. The raw data will be written to a text file in the user's working directory.

### Usage

```
cor.gen(num.subj, x.mean, x.sd, y.mean, y.sd, correlation, data.dec = 2, x.name = "Variable X", y.name
```

### Arguments

| | |
|---|---|
| `num.subj` | Specifies the intended sample size (N) |
| `x.mean` | Specifies the intended mean for variable X |
| `x.sd` | Specifies the intended standard deviation for variable X |
| `y.mean` | Specifies the intended mean for variable Y |
| `y.sd` | Specifies the intended standard deviation for variable Y |
| `correlation` | Specifies the indended correlation between X & Y |
| `data.dec` | Specifies the number of decimal places in the sample data |
| `x.name` | Specifies the name of variable X |
| `y.name` | Specifies the name of variable Y |

### Value

The values returned are the raw data which is printed to the R console as well as saved to a text file to be imported to other programs or back into R. The R console will also display performance data including the descriptive statistics.

### Author(s)

Brendan Morse

## Examples

```
# In a study published in Body Image on the effects
# of massage on state (temporary) body image, Dunigan,
# King, & Morse (2011) reported that their observed
# correlation between body mass index scores (BMI) and
# body image state scores (BISS) to be -.56 for their
# 49 participants. We can use cor.gen to generate a sample
# of raw scores to match these statistics and use this
# data as a teaching example for Pearson correlations in
# an introductory statistics course.

# To generate the data

cor.gen(49, 24.38, 6.22, 5.16, 1.47, -.56, data.dec=2, x.name="BMI", y.name="BISS")
```

# stat.gen *stat.gen*

## Description

stat.gen is a function to generate raw data based on user-specified descriptive statistics (n, mean, & standard deviation). A sample of N scores will be drawn from an appropriate distribution (normal or exponential) and scaled to match the desired mean and standard deviation. The raw data will be written to a text file in the user's working directory.

## Usage

```
stat.gen(num.subj, target.mean, target.sd, data.dec = 2, neg.data = TRUE, x.name = "Variable X")
```

## Arguments

| | |
|---|---|
| `num.subj` | Specifies the intended sample size (n) |
| `target.mean` | Specifies the intended sample mean |
| `target.sd` | Specifies the intended sample standard deviation |
| `data.dec` | Specifies the number of decimal places in the sample data |
| `neg.data` | Specifies whether the sample data can include negative values |
| `x.name` | Specifies the name of the variable |

## Value

The values returned are the raw data which is printed to the R console as well as saved to a text file to be imported to other programs or back into R. The R console will also display performance data including the descriptive statistics and the number of iterations needed to find a solution.

## Note

For most applications, stat.gen will be able to generate a succesful dataset. However, recognize that the raw data may look more or less like you would expect the original raw data to look given that an infinite combination of values could emulate any of the supplied descriptive statistics. Modifying the data.dec command and/or the neg.data command can help the generated raw data look more realistic.

If neg.data=FALSE (i.e., the raw data cannot be negative), and the target mean is near zero (i.e., the sample is positively skewed with a floor effect at 0), stat.gen will draw scores from an exponential distribution to facilitate finding a solution.

## Author(s)

Brendan Morse

## Examples

```
# In a study published in Body Image on the
# effects of massage on state (temporary) body
# image, Dunigan, King, & Morse (2011) that there
# were no significant differences in body mass
# index scores (BMI) between the control (n=26,
# M=25.26, SD=6.51) and experimental (n=23, M=23.30,
# SD=5.83) conditions as a check for potential confounds.
# We can use MorseGen to generate a sample of raw scores
# to match these statistics and use this data as a
# teaching example for an independent samples t-test in an
# introductory statistics course.

# Generate the control group data.
# I set non.neg=TRUE because the supplied descriptive
# statistics could return a negative value (0 is within 4 standard
# deviations from each mean) and BMI cannot be negative.
# Note that the output file "MorseGen Sample Results.txt" will
# overwrite the data with each run of the stat.gen function so
# the scores must be copied out or saved as a different file name
# after each run.

stat.gen(26, 25.26, 6.51, data.dec=2, neg.data=FALSE, x.name="Control")

#To generate the experimental group data

stat.gen(23, 23.30, 5.83, data.dec=2, neg.data=FALSE, x.name="Massage")
```

# Index