uDescriptives Function Descriptions
Version 20121017

This document contains help files for the descriptive statistics functions:

- `descrip( )`: general descriptive statistics for an arbitrary number of variables (numeric, survival, dates), possibly stratified.
- `print.uDescriptives( )`: the print method for `uDescriptives` objects returned by `descrip()`.
- `tableStat( )`: stratified descriptive statistics for a single variable returned in tabular form.
- `print.tableStat( )`: the print method for `tableStat` objects returned by `tableStat()`.
- `extract.tableStat( )`: an extraction method for `tableStat` objects returned by `tableStat()`.
- `clusterStats( )`: generates vectors containing summary statistics within clusters.
- `scatter( )`: scatter plots with lowess smooths and optional least squares fits, possibly stratified.
- `correlate( )`: estimated correlation matrices (and inference), possibly stratified.
- `print.uCorrelate( )`: the print method for `uCorrelate` objects returned by `correlate()`.

Function Interface: descrip

Description

Produces descriptive statistics for an arbitrary number of variables of class `integer, numeric, Surv, Date`, or `factor`. Descriptive statistics can be obtained within strata, and the user can specify that only a subset of the data be used. Descriptive statistics include the count of observations, the count of cases with missing values, the mean, standard deviation, geometric mean, minimum, and maximum. The user can specify arbitrary quantiles to be estimated, as well as specifying the estimation of proportions of observations within specified ranges.

Usage

```
descrip  (..., strata=NULL, subset=NULL, probs= c(.25,.50,.75), replaceZeroes=F,
         restriction=Inf, above=NULL, below=NULL, labove=NULL, rbelow=NULL,
         lbetween=NULL, rbetween=NULL,  interval=NULL, linterval=NULL,
         rinterval=NULL, lrinterval=NULL, version=F)
```

Arguments

`...`	an arbitrary number of variables for which descriptive statistics are desired. The arguments can be vectors, matrices, or lists. Individual columns of a matrix or elements of a list may be of class `numeric`, `factor`, `Surv`, or `Date`. Factor variables are converted to integers. Character vectors will be coerced to numeric. Variables may be of different lengths, unless `strata` or `subset` are non-NULL.
`strata`	vector, matrix, or list of stratification variables. Descriptive statistics will be computed within strata defined by each unique combination of the stratification variables, as well as in the combined sample. If `strata` is supplied, all variables must be of that same length.
`subset`	vector indicating a subset to be used for all descriptive statistics. If `subset` is supplied, all variables must be of that same length.
`probs`	a vector of probabilities between 0 and 1 indicating quantile estimates to be included in the descriptive statistics. Default is to compute the 25th, 50th (median), and 75th percentiles.
`replaceZeroes`	if not FALSE, this indicates a value to be used in place of zeroes when computing a geometric mean. If TRUE, a value equal to one-half the lowest nonzero value is used. If a numeric value is supplied, that value is used. Note that the same value is used for all variables.
`restriction`	a value used for computing restricted means, standard deviations, and geometric means with censored time to event data. The default value of `Inf` will cause restrictions at the highest observation. Note that the same value is used for all variables of class `Surv`.

`above` a vector of values used to dichotomize variables. The descriptive statistics will include an estimate for each variable of the proportion of measurements with values greater than each element of `above`.

`below` a vector of values used to dichotomize variables. The descriptive statistics will include an estimate for each variable of the proportion of measurements with values less than each element of `below`.

`labove` a vector of values used to dichotomize variables. The descriptive statistics will include an estimate for each variable of the proportion of measurements with values greater than or equal to each element of `above`.

`rbelow` a vector of values used to dichotomize variables. The descriptive statistics will include an estimate for each variable of the proportion of measurements with values less than or equal to each element of `below`.

`lbetween` a vector of values which with `-Inf` and `Inf` appended is used as cutpoints to categorize variables. The descriptive statistics will include an estimate for each variable of the proportion of measurements with values between successive elements of `lbetween`, with the left hand endpoint being included in each interval.

`rbetween` a vector of values which with `-Inf` and `Inf` appended is used as cutpoints to categorize variables. The descriptive statistics will include an estimate for each variable of the proportion of measurements with values between successive elements of `lbetween`, with the right hand endpoint being included in each interval.

`interval` a two column matrix of values in which each row is used to define intervals of interest to categorize variables. The descriptive statistics will include an estimate for each variable of the proportion of measurements with values between the two values in a row, with neither endpoint being included in each interval.

`linterval` a two column matrix of values in which each row is used to define intervals of interest to categorize variables. The descriptive statistics will include an estimate for each variable of the proportion of measurements with values between the two values in a row, with the left hand endpoint being included in each interval.

`rinterval` a two column matrix of values in which each row is used to define intervals of interest to categorize variables. The descriptive statistics will include an estimate for each variable of the proportion of measurements with values between the two values in a row, with the right hand endpoint being included in each interval.

`lrinterval` a two column matrix of values in which each row is used to define intervals of interest to categorize variables. The descriptive statistics will include an estimate for each variable of the proportion of measurements with values between the two values in a row, with both endpoints being included in each interval.

`version` if TRUE, the version of the function will be returned. No other computations will be performed.

Value

An object of class `uDescriptives` is returned. Descriptive statistics for each variable in the entire subsetted sample, as well as within each stratum if any is defined, are contained in a matrix with rows corresponding to variables and strata and columns corresponding to descriptive statistics that include

- `N`: the number of observations.
- `Msng`: the number of observations with missing values.
- `Mean`: the mean of the nonmissing observations (this is potentially a restricted mean for right censored time to event data).
- `Std Dev`: the standard deviation of the nonmissing observations (this is potentially a restricted standard deviation for right censored time to event data).
- `Geom Mn`: the geometric mean of the nonmissing observations (this is potentially a restricted geometric mean for right censored time to event data). Nonpositive values in the variable will generate NA, unless `replaceZeroes` was specified.
- `Min`: the minimum value of the nonmissing observations (this is potentially censored for right censored time to event data).
- columns corresponding to the quantiles as specified by `probs`.
- `Max`: the maximum value of the nonmissing observations (this is potentially censored for right censored time to event data).
- columns corresponding to the proportions as specified by `above, below, labove, rbelow, lbetween, rbetween, interval, linterval, rinterval, lrinterval`.
- `restriction`: the threshold for restricted means, standard deviations, geometric means.
- `FirstEvent`: the time of the first event for censored time to event variables.
- `LastEvent`: the time of the last event for censored time to event variables.
- `isDate`: an indicator that the variable is a `Date` object.

There is a print method that will format the descriptive statistics for the `Date` and `Surv` objects.

Details

This function depends on the `survival` R package. You should execute `library(survival)` if that library has not been previously installed.

Quantiles are computed for uncensored data using the default method in `quantile()`.

For variables of class `factor`, descriptive statistics will be computed using the integer coding for the factors.

For variables of class `Surv`, estimated proportions and quantiles will be computed from Kaplan-Meier estimates, as will be restricted means, restricted standard deviations, and restricted geometric means.

For variables of class `Date`, estimated proportions will be labeled using the Julian date since January 1, 1970.

Examples

```
# Reading in a dataset
mri <- read.table("http://www.emersonstatistics.com/datasets/mri.txt",header=T)

# Creating a Surv object to reflect time to death
mri$ttodth <- Surv(mri$obstime,mri$death)

# Reformatting an integer MMDDYY representation of date to be a Date object
mri$mridate <- as.Date(paste(trunc(mri$mridate/10000),trunc((mri$mridate %% 10000)/100),
               mri$mridate %% 100,sep="/"),"%m/%d/%y")

# Description of the entire data frame
descrip(mri)

# Description of time to death with more appropriate quantiles
# (Note indication of restricted mean, standard deviation, geometric mean and
# censored observation of maximum survival time)
descrip(mri$ttodth,probs=c(0.05,0.1,0.15,0.2))

# Stratified descriptive statistics
with (mri, descrip(age,dsst,strata=male))
with (mri, descrip(age,dsst,strata=cbind(male,chd)))

# Descriptive statistics on a subset comprised of males
with (mri, descrip(age,dsst,subset=male==1))

# Alternative methods for estimating proportions of subjects in specific age ranges
descrip(mri$age,above=c(75,85),probs=NULL)
descrip(mri$age,labove=c(75,85),probs=NULL)
descrip(mri$age,below=c(75,85),probs=NULL)
descrip(mri$age,rbelow=c(75,85),probs=NULL)
descrip(mri$age,lbetween=c(75,85),probs=NULL)
descrip(mri$age,rbetween=c(75,85),probs=NULL)
descrip(mri$age,interval=cbind(75,85),lrinterval=cbind(75,85),probs=NULL)
descrip(mri$age,linterval=cbind(75,85),rinterval=cbind(75,85),probs=NULL)
```

Function Interface: print.uDescriptives

Description

The print method for the `uDescriptives` object returned by function `descrip()`.

Usage

```
print.uDescriptives (x, sigfigs=max(3,getOption("digits")-3), width=9, nonsci.limit=5,
                     version=F)
```

Arguments

`x`	a `uDescriptives` object as returned by `descrip()`.
`sigfigs`	the desired number of significant figures used for printing.
`width`	column width used for formatting.
`nonsci.limit`	the number of digits to print before using scientific notation.
`version`	if TRUE, the version of the function will be returned. No other computations will be performed.

Value

The formatted table is printed and invisibly returned.

Function Interface: tableStat

Description

Produces a table of stratified descriptive statistics for a single variable of class `integer, numeric, Surv, Date`, or `factor`. Descriptive statistics are those that can be estimated using the `descrip()` function.

Usage

```
tableStat <- function (variable, ..., stat="count", na.rm=T,
             subset=NULL, probs= c(.25,.50,.75), replaceZeroes=F, restriction=Inf,
             above=NULL, below=NULL, labove=NULL, rbelow=NULL, lbetween=NULL,
             rbetween=NULL, interval=NULL, linterval=NULL, rinterval=NULL,
             lrinterval=NULL, version=F)
```

Arguments

`variable`	a vector or `Surv` object suitable for use as an argument to `descrip()`. If a `NULL` value is supplied for `variable`, the valid statistics returned by the function is only the cross-tabulation of counts and percentages within strata.
`...`	an arbitrary number of stratification variables. The arguments can be vectors, matrices, or lists. Individual columns of a matrix or elements of a list may be of class `numeric`, `factor`, or `character`. Stratification variables must all be the same length as each other and (if it is supplied) `variable`.
`stat`	a vector of character strings indicating the descriptive statistic(s) to be tabulated within strata. Possibilities include any statistic returned by `descrip()` as specified by one or more of "count", "missing", "mean", "geometric mean", "median", "sd", "variance", "minimum", "maximum", "quantiles", "probabilities", "mn(sd)", "range", "iqr", "all", "row%", "col%", or "tot%". Only enough of the string needs to be specified to disambiguate the choice. Alternatively (and more usefully), a single special format character string can be specified as described in the Details below.
`na.rm`	an indicator that missing data is to be removed prior to computation of the descriptive statistics.
`subset`	vector indicating a subset to be used for all descriptive statistics. If `subset` is supplied, it must be of the same length as `variable` and all stratification variables.
`probs`	a vector of probabilities between 0 and 1 indicating quantile estimates to be included in the descriptive statistics. Default is to compute the 25th, 50th (median), and 75th percentiles.
`replaceZeroes`	if not FALSE, this indicates a value to be used in place of zeroes when computing a geometric mean. If TRUE, a value equal to one-half the lowest nonzero value is used. If a numeric value is supplied, that value is used.

`restriction` a value used for computing restricted means, standard deviations, and geometric means with censored time to event data. The default value of `Inf` will cause restrictions at the highest observation.

`above` a vector of values used to dichotomize variables. The descriptive statistics will include an estimate of the proportion of measurements with values greater than each element of `above`.

`below` a vector of values used to dichotomize variables. The descriptive statistics will include an estimate of the proportion of measurements with values less than each element of `below`.

`labove` a vector of values used to dichotomize variables. The descriptive statistics will include an estimate of the proportion of measurements with values greater than or equal to each element of `above`.

`rbelow` a vector of values used to dichotomize variables. The descriptive statistics will include an estimate of the proportion of measurements with values less than or equal to each element of `below`.

`lbetween` a vector of values which with `-Inf` and `Inf` appended is used as cutpoints to categorize variables. The descriptive statistics will include an estimate of the proportion of measurements with values between successive elements of `lbetween`, with the left hand endpoint being included in each interval.

`rbetween` a vector of values which with `-Inf` and `Inf` appended is used as cutpoints to categorize variables. The descriptive statistics will include an estimate of the proportion of measurements with values between successive elements of `lbetween`, with the right hand endpoint being included in each interval.

`interval` a two column matrix of values in which each row is used to define intervals of interest to categorize variables. The descriptive statistics will include an estimate of the proportion of measurements with values between the two values in a row, with neither endpoint being included in each interval.

`linterval` a two column matrix of values in which each row is used to define intervals of interest to categorize variables. The descriptive statistics will include an estimate of the proportion of measurements with values between the two values in a row, with the left hand endpoint being included in each interval.

`rinterval` a two column matrix of values in which each row is used to define intervals of interest to categorize variables. The descriptive statistics will include an estimate of the proportion of measurements with values between the two values in a row, with the right hand endpoint being included in each interval.

`lrinterval` a two column matrix of values in which each row is used to define intervals of interest to categorize variables. The descriptive statistics will include an estimate of the proportion of measurements with values between the two values in a row, with both endpoints being included in each interval.

`version` if TRUE, the version of the function will be returned. No other computations will be performed.

Value

An object of class `tableStat` is returned, which consists of a list of arrays. Each array corresponds to a table of stratified statistics for one of the possible basic choices of `stat`. The print method provides the formatted output for the choice specified in `stat`.

Details

This function uses `descrip()` to compute the descriptive statistics.

In addition to the basic choices specified above for `stat`, the user can supply a special format character string. Arbitrary text can be specified to label any of the descriptive statistics, which are indicated by bracketing with "@". All text bracketed by "@" must refer to a descriptive statistic, and all other text is printed verbatim. For instance, a display of the mean, standard deviation, minimum, maximum, and sample size might be specified by "@mean@ (@sd@; @min@ - @max@; n=@count@)". Similarly, a cross tabulation displaying counts, row percentages, column percentages, and percentages of the total might be specified by "@count@ (r @row%@; c @col%@; t @tot%@)". See examples.

Examples

```
# Reading in a dataset
mri <- read.table("http://www.emersonstatistics.com/datasets/mri.txt",header=T)
# Creating a Surv object to reflect time to death
mri$ttodth <- Surv(mri$obstime,mri$death)
# Reformatting an integer MMDDYY representation of date to be a Date object
mri$mridate <- as.Date(paste(trunc(mri$mridate/10000),trunc((mri$mridate %% 10000)/100),
               mri$mridate %% 100,sep="/"),"%m/%d/%y")

# Cross tabulation of counts with sex and race strata
with (mri, tableStat (NULL, race, male, stat= "@count@ (r @row%@; c @col%@; t @tot%@)"))

# Cross tabulation of counts with sex, race, and coronary disease strata
# (Note row and column percentages are defined within the first two strata, while overall
# percentage considers all strata)
with (mri, tableStat (NULL, race, male, chd,
    stat= "@count@ (r @row%@; c @col%@; t @tot%@)"))

# Description of time to death with appropriate quantiles
with (mri, tableStat(ttodth,probs=c(0.05,0.1,0.15,0.2),
    stat="mean @mean@ (q05: @q@; q10: @q@; q15: @q@; q20: @q@; max: @max@)"))

# Description of mridate with mean, range stratified by race and sex
with (mri, tableStat(mridate, race, male,
    stat="mean @mean@ (range @min@ - @max@)"))

# Stratified descriptive statistics with proportions
with (mri, descrip(age,above=c(75,85),lbetween=c(75,85),
    stat=">75: @p@; >85: @p@; [-Inf,75): @p@; [75,85): @p@; [85,Inf): @p@" ))

# Descriptive statistics on a subset comprised of males
with (mri, tableStat(dsst,age,stroke,subset=male==1,
    stat="@mean@ (@sd@; n= @count@/@missing@)"))
```

Function Interface: print.tableStat

Description

The print method for the `tableStat` object returned by function `tableStat()`.

Usage

```
print.tableStat <- function (x, stat=attr(x,"stat"), na.rm=attr(x,"na.rm"),
       sigfigs=max(3,getOption("digits")-3), width=9, nonsci.limit=5, version=F)
```

Arguments

`x`	a `tableStat` object as returned by `tableStat()`.
`stat`	a vector of character strings indicating the descriptive statistic(s) to be printed within strata. See the documentation for `tableStat()` for a full description.
`na.rm`	an indicator that missing data is to be removed prior to computation of the descriptive statistics.
`sigfigs`	the desired number of significant figures used for printing.
`width`	column width used for formatting.
`nonsci.limit`	the number of digits to print before using scientific notation.
`version`	if TRUE, the version of the function will be returned. No other computations will be performed.

Value

The formatted table is printed and invisibly returned.

Function Interface: extract.tableStat

Description

An extraction method for the `tableStat` object returned by function `tableStat()`. This function presumes that only a single stratification variable was specified in the call toe `tableStat()`. The function is mainly used by the function `clusterStats()`.

Usage

```
extract.tableStat <- function (x, stat=attr(x,"stat"), na.rm=attr(x,"na.rm"),
        version=F)
```

Arguments

`x`	a `tableStat` object as returned by `tableStat()`.
`stat`	a character string indicating the descriptive statistic(s) to be printed within strata. See the documentation for `tableStat()` for a full description, although only single statistics can be specified in this function. If either `"probabilities"` or `"quantiles"` are specified, only the first such quantity is returned.
`na.rm`	an indicator that missing data is to be removed prior to computation of the descriptive statistics.
`version`	if TRUE, the version of the function will be returned. No other computations will be performed.

Value

A vector is returned that contains the stratified statistics (but not the overall statistics).

Function Interface: clusterStats

Description

Produces a vector containing summary measures computed within clusters.

Usage

```
clusterStats <- function (y, cluster=NULL, stat="count", subset=NULL, x=NULL, ...,
        version=F)
```

Arguments

`y`	a vector, `"Date"` object, or `"Surv"` object for which within cluster summary statistics are desired.
`cluster`	vector, matrix, or list of variables defining clusters. Descriptive statistics will be computed within strata defined by each unique combination of the cluster variables.
`stat`	a character string indicating the descriptive statistic(s) to be returned for each cluster. See the documentation for `tableStat()` for a full description, although only single statistics can be specified in this function. If either `"probabilities"` or `"quantiles"` are specified, only the first such quantity is returned. In addition to the summary statistics allowed by `tableStat()`, a user can also specify within cluster least squares slopes (`stat="slope"`) of `y` on `x`.
`subset`	a logical vector indicating a subset to be used for all descriptive statistics.
`x`	a numeric vector to be used as regression predictor for least squares slopes.
`...`	optional arguments specifying quantiles or thresholds for probabilities to be used in calculating summary statistics. See arguments for `descrip()`.
`version`	if TRUE, the version of the function will be returned. No other computations will be performed.

Value

A vector is returned that contains the summary statistic relevant for the cluster to which each observation in `y` belongs. Although only the cases indicated by `subset` are used to calculate the summary statistics, values are expanded out to cases beyond those indicated by `subset`.

Details

This function uses `tableStat()` to compute stratified statistics for each cluster. However, only single summary measures can be used in this function. See examples.

Examples

```
# Reading in a dataset
mtx <- read.table("http://www.emersonstatistics.com/datasets/mtxlabs.txt",header=T)

# Generating average bilirubin for each subject
mbili <- clusterStats (mtx$bili, mtx$ptid, "mean")
descrip(mbili,strata=mtx$tx)

# Generating average bilirubin for each subject while taking study drug
mdrugbili <- clusterStats (mtx$bill, mtx$ptid, "mean", subset=mtx$ondrug==1)
descrip(mdrugbili,strata=mtx$tx)

# Reading in a dataset
audio <- read.csv("http://www.emersonstatistics.com/datasets/audio.csv",header=T)

# Generating counts for each subject
counts <- clusterStats (audio$R4000, audio$Subject,  "count")
table(counts,strata=audio$Dose)

# Generating average R4000 for each subject
mR4000 <- clusterStats (audio$R4000, audio$Subject,  "mean")
descrip(mR4000,strata=audio$Dose)

# Generating average R4000 for each subject after visit 0
mtxR4000 <- clusterStats (audio$R4000, audio$Subject,  "mean", subset=audio$Visit>0)
descrip(mtxR4000,strata=audio$Dose)
```

Function Interface: scatter

Produces a scatterplot of two variables with (possibly stratified) superimposed lowess smooths and least squares fitted lines.

Usage

```
scatter <- function (y, x, strata=rep(1,length(y)), subset= rep(T,length(y)),
             reference=sort(unique(strata)), plotPoints=T, plotLowess=T, plotLSfit=F,
             legend=0.05, colors=c("black", "blue", "orange", "pink", "green", "red",
             "cornflowerblue", "darkolivegreen", "magenta"), xJitter=T, yJitter=F,
             newplot=T, ..., version=F)
```

Arguments

`y`	a numeric vector containing the values to be plotted on the y-axis.
`x`	a numeric vector containing the values to be plotted on the x-axis.
`strata`	vector, matrix, or list of stratification variables. Descriptive statistics will be computed within strata defined by each unique combination of the stratification variables, as well as in the combined sample. If `strata` is supplied, all variables must be of that same length.
`subset`	vector indicating a subset to be used for all descriptive statistics. If `subset` is supplied, all variables must be of that same length.
`reference`	a list of the strata in the order they are to be plotted.
`plotPoints`	an indicator that points are to be plotted. A different color and line type will be used for each stratum. Default is TRUE.
`plotLowess`	an indicator that lowess smooths are to be plotted. A different color and line type will be used for each stratum. Default is TRUE.

`plotLSfit`	an indicator that least squares fitted lines are to be plotted. A different color and line type will be used for each stratum. Default is FALSE.
`legend`	if 0, no legend is plotted. Otherwise, the x-dimensions of the plot are expanded by 25%, and a legend is plotted with `legend` defining the spacing of lines as a proportion of the y-dimensions of the plot. The default value of 0.05 indicates that lines in the legend are separated by 5% of the vertical dimensions of the plot.
`colors`	a vector of colors to be used in plotting strata.
`xJitter`	the proportion of the minimal difference between adjacent x-values divided by 8 by which plotted points are to be jittered in the x-dimension. A value of 0 implies no jittering.
`yJitter`	the proportion of the minimal difference between adjacent y-values divided by 8 by which plotted points are to be jittered in the y-dimension. A value of 0 implies no jittering.
`...`	optional arguments for plotting parameters (e.g., `xlab, ylab, main, xlim, ylim`) that will be passed to `plot()`.
`version`	if TRUE, the version of the function will be returned. No other computations will be performed.

Value

This function only produces a plot. No value is returned.

Examples

```
# Reading in a dataset
mri <- read.table("http://www.emersonstatistics.com/datasets/mri.txt",header=T)

# Scatterplot of DSST by age with jittering of age values and superimposed lowess smooth
with (mri, scatter (dsst, age, main="DSST by Age (years)"))

# Scatterplot with superimposed lowess smooth and LS fit
# (Note same color and line type used for both lowess smooth and LS fit)
with (mri, scatter (dsst, age, main="DSST by Age (years)", plotLSfit=T))

# Scatterplot with strata by sex and coronary heart disease
with (mri, scatter (dsst, age, strata=cbind(male,chd),
       main="DSST by Age (years) in Sex and CHD strata"))

# Scatterplot showing only the lowess smooths
with (mri, scatter (dsst, age, strata=cbind(male,chd), plotPoints=F,
       main="DSST by Age (years) in Sex and CHD strata"))
```

Function Interface: correlate

Description

Computes correlation matrix for an arbitrary number of numeric variables, optionally within strata.

Usage

```
correlate <- function (..., strata=NULL, subset=NULL, conf.level= 0.95,
          use="pairwise.complete.obs", method="pearson", stat="cor", byStratum=F,
          version=F)
```

Arguments

`...`	an arbitrary number of variables for which a correlation matrix is desired. The arguments can be vectors, matrices, or lists. Individual columns of a matrix or elements of a list that are not of class `numeric`, `factor`, or `Date` will be omitted. Factor and Date variables are converted to integers. Character vectors will be coerced to numeric. Variables must all be of the same lengths.
`strata`	vector, matrix, or list of stratification variables. Descriptive statistics will be computed within strata defined by each unique combination of the stratification variables, as well as in the combined sample. If `strata` is supplied, all variables must be of that same length.
`subset`	vector indicating a subset to be used for all descriptive statistics. If `subset` is supplied, all variables must be of that same length.
`conf.level`	a numeric scalar between 0 and 1 denoting the confidence level to be used in constructing confidence intervals for the correlation.
`use`	character string denoting the cases to use: "everything" uses all cases (and causes NA when any needed variable is missing), "complete.obs" uses only those rows with no missing data for any variable, and "pairwise.complete.obs" computes pairwise correlations using all cases that are not missing data for the relevant variables.
`method`	character string denoting the correlation method to use: "pearson" denotes Pearson's correlation coefficient and "spearman" denotes Spearman's rank correlation.)
`stat`	a vector of character strings indicating the descriptive statistic(s) to be tabulated. Possibilities include any statistic as specified by one or more of "cor", "n", "t.stat", "pval", "loCI", or "hiCI". Only enough of the string needs to be specified to disambiguate the choice. Alternatively (and more usefully), a single special format character string can be specified as described in the Details below.

`byStratum` a logical scalar indicating whether statistics should be grouped by pair of variables. If TRUE, the results will be displayed in a series of tables where each table correspond to a single variable, with rows corresponding to different strata and columns reflecting all other variables. If FALSE, the results will be displayed in a series of tables where each table corresponds to a single stratum and rows and columns reflect the variables.

`version` if TRUE, the version of the function will be returned. No other computations will be performed.

Value

An object of class `uCorrelate` is returned, which consists of a list of correlation estimates and inference for each specified stratum and for the combined dataset. Each element of the list has arrays `$cormtx` (containing the correlation estimates), `$n` (containing the sample sizes used to compute each correlation estimate), `$t.stat` (containing the t statistic testing a correlation of 0), `$pval` (containing a two sided p value for a test of the null hypothesis of 0 correlation), `$lo##%CI` (containing the lower bound of a confidence interval for the true correlation), and `$lo##%CI` (containing the lower bound of a confidence interval for the true correlation), where "##" denotes the confidence level.

Details

In addition to the basic choices specified above for `stat`, the user can supply a special format character string. Arbitrary text can be specified to label any of the descriptive statistics, which are indicated by bracketing with "@". All text bracketed by "@" must refer to one of the statistics, and all other text is printed verbatim. For instance, a display of the estimated correlation coefficient, confidence interval, p value, and sample size might be specified by "@cor@ (CI @lo@ - @hi@; P= @pval@; n=@n@)". See examples.

Examples

```
# Reading in a dataset
mri <- read.table("http://www.emersonstatistics.com/datasets/mri.txt",header=T)

# Estimated correlation matrix using all data, complete cases, or pairwise complete (the default)
with (mri, correlate(age,weight,ldl,use="everything"))
with (mri, correlate(age,weight,ldl,use="complete"))
with (mri, correlate(age,weight,ldl))

# Correlation matrices for each stratum
with (mri, correlate(age,weight,ldl,strata=male))

# Correlations grouped by variable
with (mri, correlate(age,weight,ldl,strata=male,byStratum=F))

# Special formatting of inference for correlations within strata
with (mri, correlate(age,weight,ldl,strata=male,stat="@cor@ (@lo@, @hi@); P @p@; n= @n@"))

# Special formatting of inference for correlations grouped by variable
with (mri, correlate(age,weight,ldl,strata=male,stat="@cor@ (@lo@, @hi@); P @p@; n= @n@",byStratum=F))
```

Function Interface: print.uCorrelate

Description

The print method for the `uCorrelate` object returned by function `correlate()`.

Usage

```
print.uCorrelate <- function (x, stat=attr(x,"stat"), byStratum=attr(x,"byStratum"),
       sigfigs=max(5,getOption("digits")-2), width=9, nonsci.limit=5, version=F)
```

Arguments

`x`	a `uCorrelate` object as returned by `correlate()`.
`stat`	a vector of character strings indicating the statistic(s) to be printed. See the documentation for `correlate()` for a full description.
`byStratum`	a logical scalar indicating whether statistics should be grouped by pair of variables. If TRUE, the results will be displayed in a series of tables where each table correspond to a single variable, with rows corresponding to different strata and columns reflecting all other variables. If FALSE, the results will be displayed in a series of tables where each table corresponds to a single stratum and rows and columns reflect the variables.
`sigfigs`	the desired number of significant figures used for printing.
`width`	column width used for formatting.
`nonsci.limit`	the number of digits to print before using scientific notation.
`version`	if TRUE, the version of the function will be returned. No other computations will be performed.

Value

The formatted tables are printed and invisibly returned.