BloomR main functions

R topics documented:

br.hist.csv
br.hist
br.idx
br.sample

TODO

Complete help for functions br.try.date, br.is.same.class, br.bdh Fix XXXX paragraphs

br.hist.csv

Historical data from grouped tickers in a CSV file
Reads a CSV file containing a group of tickers in each column and returns the historical data in xts or list format. The CSV file is assumed to have headers denoting group labels. It replaces 'br.bulk.csv"

Usage

```
br.hist.csv(con, file, field="PX_LAST", start=Sys.Date()-5, end.date=Sys.Date(),

                  cols=NULL, comma=TRUE,
                  addtype=FALSE, showtype=FALSE,
                  use.xts=TRUE, merge.xts=TRUE,

                  option.names = NULL, option.values = NULL,
                  only.trading.days = TRUE,

                  price=TRUE,
                  mean=ifelse(price, 10, 0.1), sd=1, jitter=0,
                  same.dates=FALSE, empty.sec=0,
                  weekend=TRUE, holidays=NULL)
```

Arguments

con the connection token returned from br.open(). If `NULL` simulated values are generated.
file path to CSV file.
field case insensitive string denoting the Bloomberg field queried. Defaults to "PX_LAST". If the field is wrong or not accessible, data will be empty but no error will be raised.
start start date. Can be a Date object or an ISO string without separators (YYYYMMDD). Defaults to 5 days before current date.
end.date end date. Same format as `start`. Defaults to current date.
cols Logical or integer vector for selecting CSV columns (ticker groups). Defaults to all columns.
comma to be set to FALSE for (non-English) CSV, using semicolon as separator.
addtype If a string denoting the security type, it will be added to all tickers; if TRUE "Equity", will be added; if FALSE (the default), nothing will be added.
showtype if TRUE, security types will be removed from names of list or xts output. It defaults to FALSE.

use.xts if TRUE (the default) time series are formatted as xts objects else as a data frame.
merge.xts if TRUE (the default) xts objects in the same group are merged using all rows and using NAs for missing observations.
option.names list of Bloomberg options names. Require `option.values` too.
option.values list of Bloomberg options values related to `option.names`.
only.trading.days if TRUE (the default) use only trading days, add non trading days to output with NA values.
price, mean, sd, jitter, same.dates, empty.sec, weekend, holidays arguments passed to `br.sample()` if `con=NULL`.

Details

Empty CSV cells or cells interpreted as NAs will be ignored.
If `con=NULL`, values are simulated by means of `br.sample()`. This function is used with default values, except for `start, end.date, price, mean, sd, jitter, same.dates, empty.sec, weekend, holidays`, which can be explicitly passed as arguments, and `sec.names` depending on tickers found in the CSV file. These arguments are ignored if `con!=NULL`. See `br.sample()` help for more.

Value

a list where each element is the historical data of a CSV group.
If `use.xts=TRUE` and `merge.xts=FAlSE`, each group is a sub-list, whose elements are the the securoty time series as an xts object. If `use.xts=TRUE` and `merge.xts=TRUE`, each group is the merged xts object, obtained merging historical data of all securities of that group. If `use.xts=FALSE`, each group is a sub-list, where each element is the historical data of a security as a data frame.
If there is only one group, the first (and unique) element of the list will be returned (XXXXto check).

br.hist

Historical data for vector of tickers
Returns the historical data for a vector of tickers in xts or list format. It replaces 'br.bulk.tiks"

Usage

```
br.hist(con, tiks, field="PX_LAST", start=Sys.Date()-7, end.date=Sys.Date(),

              addtype=FALSE, showtype=FALSE,
              use.xts=TRUE, merge.xts=TRUE,

              option.names = NULL, option.values = NULL,
              only.trading.days = TRUE,

              price=TRUE,
              mean=ifelse(price, 10, 0.1), sd=1, jitter=0,
              same.dates=FALSE, empty.sec=0,
              weekend=TRUE, holidays=NULL)
```

Arguments

tiks character vector of the tickers queried for data
use.xts if TRUE (the default) time series are formatted as xts objects else as a data frame.
merge.xts if TRUE (the default) xts objects are merged using all rows and using NAs for missing observations.

For other arguments see the function `br.hist.csv`

tiks character vector of the tickers queried for data
use.xts if TRUE (the default) time series are formatted as xts objects else as a data frame. stack test
use.xts inner test
only.trading.days inner test 2
fakearg fake arg

Details

If an element of `tiks` is `NA` or empty (`""`) it is ignored. This is intended to avoid errors when the character vector are read from a CSV file with empty cells.
If `con=NULL`, values are simulated by means of `br.sample()`. Sampled values are based on default values of `br.sample()`, but it is possible to set explicitly `start, end.date, price, mean, sd, jitter, same.dates, empty.sec, weekend, holidays`; `sec.names` depends on `tiks` argument. These arguments are ignored if `con!=NULL`. See `br.sample()` help for more.

Value

If `use.xts=FALSE`, a list where each element is the historical data of a security as a data frame.
If `use.xts=TRUE` and `merge.xts=FAlSE`, a list where each element is the historical data of a security as an xts object.
If `use.xts=TRUE` and `merge.xts=TRUE`, an xts oject where where each column is the historical data of a security.

Example

```
con=NULL # Open simulated connection and  load some data
br.hist(con, c("MSFT US", "AMZN US"), addtype=TRUE)
```

```
## Error in br.hist(con, c("MSFT US", "AMZN US"), addtype = TRUE): could not find function ".br.check.ty
```

```
br.close(con) # Use the token to release the connection
```

```
## Error in eval(expr, envir, enclos): could not find function "br.close"
```

See Also

br.hist.csv

br.idx

Description

Returns the historical data for the constituents of an index in xts or list format. It replaces `br.bulk.idx`.

Usage

```
br.idx(con, index, field="PX_LAST", start=Sys.Date()-7, end.date=Sys.Date(),

            include.idx=TRUE, showtype=FALSE,
            use.xts=TRUE, merge.xts=TRUE,

            option.names = NULL, option.values = NULL,
            only.trading.days = TRUE,

            nsec=10, sec.names = NULL,

            price=TRUE,
            mean=ifelse(price, 10, 0.1), sd=1, jitter=0,
            same.dates=FALSE, empty.sec=0,
            weekend=TRUE, holidays=NULL)
```

Arguments

con the connection token returned from br.open(). If `NULL` simulated values are generated.
index string denoting the index ticker with or without the final security type label ('Index')
include.idx if TRUE (default) returns also historical data for the index.
nsec number of simulated index constituents. Ignored if `con!=NULL`, it defaults to 10.
sec.names character vector with names of sampled index constituents. Ignored if `con!=NULL`. By default security names are like 'memb1', 'memb2', etc.

For other arguments see the function `br.hist`.

Details

If `con=NULL`, values are simulated by means of `br.sample()`. This function is used with default values, except for `nrow, nsec1, price, start, same.dates, no.na, empty.sec, sec.names`.

Value

If `use.xts=FALSE`, a list where each element is the historical data of a constituent as a data frame.
If `use.xts=TRUE` and `merge.xts=FAlSE`, a list where each element is the historical data of a constituent as an xts object.
If `use.xts=TRUE` and `merge.xts=TRUE`, an xts oject where where each column is the historical data of a constituent .
If `include.idx=TRUE`, the last column or element will be the historical data of the index.

br.sample

Description

Return simulated historical data for n securities in xts or df format.

Usage

```
br.sample(nrow=NULL,  price=TRUE,
               start=Sys.Date() - 7, end.date=Sys.Date(),
               field="FIELD",
               use.xts=TRUE,
               mean=ifelse(price, 10, 0.1), sd=1, jitter=0,
               rand.dates=TRUE, weekend=TRUE, holidays=NULL)
```

Arguments

nrow number of simulated data points for each security; if `same.dates=FALSE`, the number of rows for each sampled security will be a random number not exceeding nrow, else it will be nrow for all securities. Actual number of rows depends on the value of `rand.dates`, `weekend`, `holidays`.
price if TRUE (default), simulated values are non-negative.
start start date. Can be a Date object or an ISO string without separators (YYYYMMDD). Defaults to current date.
end.date end date. Same format as `start`. Defaults to current date.
field case insensitive string denoting the Bloomberg field queried. Defaults to "FIELD".
use.xts if TRUE (the default) time series are formatted as xts objects else as a data frame.
mean mean of security generated values. If `price=TRUE`, default to 10 else defaults to 0.1.
sd sd of security generated values. It defaults to 1.
jitter modifies each security mean by adding adding a random value in [-jitter, jitter]. Defaults to 0.
rand.dates if TRUE, all sampled securities will refer to the same dates and for each security the number will equal nrow. If FALSE (default), date values and number will randomly differ. For each security the random number will not exceed `nrow`.
weekend if TRUE (default), weekend dates are removed.
holidays list of dates to be removed,

Details

`br.sample()` assumes by default that data for some securities might not be available on certain days and time series might be misaligned (see "Missing observations and misalignment" in `br.hist()`), therefore the date values and count for each time series generated will randomly differ, with `nrow` as the maximum number of days. If you want all time series to share tha same dates, set `rand.dates=FALSE`. In this case, time series produced are aligned and you don't see any merge NA, the acutal dates generated depends on the value of `weekend` and `holidays`. If there are no holidays falling in time windows queried and `weekend=FALSE` the number of generated dates equals `nrow`.

Value

If `use.xts=FALSE`, a data frame object, where the first column is the vector with all generated dates merged and each subsequent column contains the sampled data of a security. If `use.xts=TRUE`, an xts object, where each element is the sampled data of a security, while the dates will be part of the xts time object. In both cases if `rand.dates=TRUE` generated data points might likely have different length

XXXX and the the date gaps will be filled with NAs, except if `no.na=TRUE`. If the generated values are only NAs the output will be converted to a 0-rows xts or data frame, containing only security labels accessible with `dimnames(*)[[2]]`.