Package ‘qrmdta’

December 6, 2019

Version 2019-12-03-1
Encoding UTF-8
Title Data Sets for Quantitative Risk Management Practice
Description Various data sets (stocks, stock indices, constituent data, FX, zero-coupon bond yield curves, volatility, commodities) for Quantitative Risk Management practice.
Author Marius Hofert [aut, cre], Kurt Hornik [aut], Alexander J. McNeil [aut]
Maintainer Marius Hofert <marius.hofert@uwaterloo.ca>
Depends R (>= 3.5.0)
Imports xts
Suggests knitr, qrmtools, lattice
Enhances
License GPL-2 | GPL-3
NeedsCompilation no
Repository CRAN
Date/Publication 2019-12-06 11:30:06 UTC

R topics documented:

commodities ................................................................. 2
crypto ................................................................. 3
default ................................................................. 3
fx ................................................................. 4
interest_rates ........................................................... 5
losses ................................................................. 6
commodities

Description

Data sets containing commodities.

Usage

```r
data("OIL_Brent")
data("GOLD")
```

Format

*xts* objects containing the Brent Crude price in USD per barrel (for *OIL_Brent*) and the World Gold Council gold price in USD per troy ounce (for *GOLD*).

Author(s)

Marius Hofert

Source

The data was obtained from Federal Reserve Economic Data (FRED) via Quandl on 2016-01-03 with the function `get_data()` from *qrmtools*.

Examples

```r
data("OIL_Brent")
data("GOLD")
```
### crypto

**Cryptocurrency Prices in USD**

**Description**

Bitcoin, Ethereum, Litecoin and Ripple prices in USD (from their first available date onwards).

**Usage**

```r
data("crypto")
```

**Format**

- `xts` object containing cryptocurrency prices in USD of Bitcoin (ticker symbol “BTC-USD”), Ethereum (ticker symbol “ETH-USD”), Litecoin (ticker symbol “LTC-USD”) and Ripple (ticker symbol “XRP-USD”) from their first available date onwards.

**Author(s)**

Marius Hofert

**Source**

The data was obtained from Yahoo Finance on 2018-05-29 via the function `get_data()` from `qrm-tools`.

**Examples**

```r
data("crypto")
str(crypto)
library(xts)
plot.zoo(crypto, main = "Cryptocurrencies in USD", xlab = "Time")
```

### default

**Standard & Poor’s Default Data**

**Description**

A three-dimensional array containing the default data for A-, BBB-, BB-, B- and C-rated companies for the years from 1981 to 2000.

**Usage**

```r
data("SP_defaults")
```
Format


Author(s)

Marius Hofert

Source

Standard & Poor's Credit Monitor

Examples

data("SP_defaults")

---

fx Foreign Exchange Rate Data

Description

Foreign exchange rate data with respect to USD and GBP.

Usage

data("CAD_USD")
data("GBP_USD")
data("EUR_USD")
data("CHF_USD")
data("JPY_USD")
data("CNY_USD")
data("CAD_GBP")
data("USD_GBP")
data("EUR_GBP")
data("CHF_GBP")
data("JPY_GBP")
data("CNY_GBP")

Format

Details

Interpretation: As an example, EUR_USD contains the EUR/USD exchange rate, so a value $x$ in EUR_USD indicates that 1 EUR is worth $x$ USD at that point in time.

Author(s)

Marius Hofert

Source

The data was obtained from OANDA (http://www.oanda.com/) on 2016-01-03 via the function get_data() from qrmtools.

Examples

data("CAD_USD")
data("GBP_USD")
data("EUR_USD")
data("CHF_USD")
data("JPY_USD")
data("CNY_USD")
data("CAD_GBP")
data("USD_GBP")
data("EUR_GBP")
data("CHF_GBP")
data("JPY_GBP")
data("CNY_GBP")

interest_rates
Interest-Rate Data

Description

Zero-coupon bond yield curves in CAD and USD.

Usage

data("ZCB_CAD")
data("ZCB_USD")

Format

ZCB_CAD: xts object containing, in each row, zero-coupon bond yield curves in percent for 120 times to maturity (ranging from 0.25 to 30 years); only trading days from 1991-01-02 to 2015-08-31 with available values for all maturities are included.

ZCB_USD: xts object containing, in each row, zero-coupon bond yield curves in percent for 30 times to maturity (ranging from 1 to 30 years); only trading days from 1985-11-25 to 2015-12-29 with available values for all maturities are included.
Author(s)

Marius Hofert

Source

ZCB_CAD was created from data obtained from http://www.bankofcanada.ca/rates/interest-rates/bond-yield-curves/ multiplied by 100. ZCB_USD was obtained from https://www.quandl.com/data/FED/SVENY-US-Treasury-Zero-Coupon-Yield-Curve via Quandl. Both data sets were drawn on 2016-01-03 (ZCB_USD via the function get_data() from qrmtools).

Examples

data("ZCB_CAD")
data("ZCB_USD")
mat <- as.matrix(ZCB_USD['2015-01-01/2015-12-31',])
df <- data.frame(Day = rep(1:nrow(mat), each = ncol(mat)),
                 Maturity = rep(1:ncol(mat), nrow(mat)),
                 Value = as.vector(t(mat)))
lattice::wireframe(Value ~ Day * Maturity, data = df,
                 alpha.regions = 0.5,
                 scales = list(arrows = FALSE, col = "black"),
                 par.settings = list(axis.line = list(col = "transparent")))

Description

Loss Datasets

Danish fire insurance claims in 1M DKK in Denmark from 1980-01-03 to 1990-12-31. Largest 1% of simulated losses of Norwegian bank DNB.

Usage

data("fire")
data("DNB")

Format

fire: univariate xts object with 2167 observations.

DNB: (25000, 3)-matrix containing the largest 1% of simulated (market risk, credit risk, asset risk) losses of DNB; see Aas and Puccetti (2014, Section 2).

Author(s)

Marius Hofert
Source
  fire: Originally Mette Rytgaard (Copenhagen Re).
  DNB: Originally Kjersti Aas and Giovanni Puccetti.

References

Examples
  ```r
  library(xts)
  ## Danish fire losses
  data("fire")
  str(fire)
  stopifnot(inherits(fire, "xts"), length(fire) == 2167)
  plot.zoo(fire, ylab = "Fire insurance claim")

  ## Largest 1% of simulated DNB losses
  data("DNB")
  stopifnot(dim(DNB) == c(25000, 3))
  ```

---

stock_data (Single) Stock Data

Description
  Single stock data; only Radioshack at the moment.

Usage
  ```r
data("RSHCQ")
  ```

Format
  An *xts* object containing adjusted close prices of Radioshack (RSHCQ; ticker symbol “RSHCQ”) from 1982-01-04 to 2015-01-20.

Author(s)
  Marius Hofert

Source
  Radioshack defaulted early 2015. Yahoo Finance did not provide adjusted close prices thereafter. We thus used the adjusted close prices from 1982-01-04 to 2015-01-20 which we drew from Yahoo Finance on 2015-01-21 via the function get_data() from qrmtools.
Examples

```r
data("RSHCQ")
```

---

**stock_indices**

*Stock Index Data*

### Description

Single stock indices.

### Usage

```r
data("SP500")
data("DJ")
data("NASDAQ")
data("FTSE")
data("SMI")
data("EURSTOXX")
data("CAC")
data("DAX")
data("CSI")
data("HSI")
data("SSEC")
data("NIKKEI")
```

### Format

**xts** objects containing adjusted close prices of the S&P 500 (SP500; ticker symbol “^GSPC”), Dow Jones (DJ; ticker symbol “^DJI”), NASDAQ 100 (NASDAQ; ticker symbol “^NDX”), FTSE 100 (FTSE; ticker symbol “^FTSE”), Swiss Market Index (SMI; ticker symbol “^SMI”), Euro Stoxx 50 (EURSTOXX; ticker symbol “^STOXX50E”), Cotation Assistée en Continu (CAC; ticker symbol “^FCHI”), Deutscher Aktienindex (DAX; ticker symbol “^GDAXI”), China Securities Index (CSI; ticker symbol “000300.SS”), Hang Seng Index (HSI; ticker symbol “^HSI”), Shanghai Stock Exchange Composite Index (SSEC; ticker symbol “000001.SS”) and the NIKKEI (NIKKEI; ticker symbol “^N225”) from their first date of availability to 2015-12-31.

### Author(s)

Marius Hofert

### Source

The data was obtained from Yahoo Finance on 2016-01-03 via the function `get_data()` from **qrm-tools**.
stock_indices_constituents

Examples

data("SP500")
data("DJI")
data("NASDAQ")
data("FTSE")
data("SMI")
data("EURSTOXX")
data("CAC")
data("DAX")
data("CSI")
data("HSI")
data("SSEC")
data("NIKKEI")

stock_indices_constituents

Stock Index Constituents Data

Description

Constituent data of various stock indices.

Usage

data("SP500_const")
data("DJ_const")
data("FTSE_const")
data("EURSTX_const")
data("HSI_const")

Format

_xts_ objects containing adjusted close prices of the constituents of the respective stock indices. These are the S&P 500 constituents (_SP500_const_ with corresponding Global Industry Classification Standard (GICS) information _SP500_const_info_; see [https://en.wikipedia.org/wiki/List_of_S%26P_500_companies](https://en.wikipedia.org/wiki/List_of_S%26P_500_companies); given these tickers, the data was obtained from [https://finance.yahoo.com/](https://finance.yahoo.com/) as of 2015-10-12, the Dow Jones constituents (_DJ_const_; see [https://finance.yahoo.com/q/cp?s=%5EDJI](https://finance.yahoo.com/q/cp?s=%5EDJI)) as of 2016-01-03, the FTSE 100 constituents (_FTSE_const_; see [https://uk.finance.yahoo.com/q/cp?s=%5EFTSE](https://uk.finance.yahoo.com/q/cp?s=%5EFTSE)) as of 2016-01-03 (the data was only available for 98 constituents), the Euro Stoxx 50 constituents (_EURSTX_const_; see [https://uk.finance.yahoo.com/q/cp?s=%5ESTOXX50E](https://uk.finance.yahoo.com/q/cp?s=%5ESTOXX50E)) as of 2016-01-03 (the data was only available for 98 constituents) and the Hang Seng Index constituents (_HSI_const_; see [https://uk.finance.yahoo.com/q/cp?s=%5EHSI](https://uk.finance.yahoo.com/q/cp?s=%5EHSI)) as of 2016-01-03.

The constituents data ranges from the first date at least one of the constituents is available (with missing data if not available) to 2015-12-31.
**Author(s)**

Marius Hofert

**Source**

The data was obtained from the respective URLs on 2016-01-03 via the function `get_data()` from `qrmtools`.

Note that for the S&P 500 constituents, the data was rounded to two decimal places to reduce the file size of the data set.

**Examples**

data("SP500_const")
data("DJ_const")
data("FTSE_const")
data("EURSTX_const")
data("HSI_const")

---

**volatility**

<table>
<thead>
<tr>
<th>Volatility Index</th>
</tr>
</thead>
</table>

**Description**

Volatility index data.

**Usage**

data("VIX")

**Format**

An `xts` object containing the volatility index (`VIX`; ticker symbol “^VIX”) from its first date of availability to 2015-12-31.

**Author(s)**

Marius Hofert

**Source**

The data was obtained from Yahoo Finance on 2016-01-03 via the function `get_data()` from `qrmtools`.

**Examples**

data("VIX")
Index

*Topic datasets
  commodities, 2
  crypto, 3
  default, 3
  fx, 4
  interest_rates, 5
  losses, 6
  stock_data, 7
  stock_indices, 8
  stock_indices_constituents, 9
  volatility, 10

CAC (stock_indices), 8
CAD_GBP (fx), 4
CAD_USD (fx), 4
CHF_GBP (fx), 4
CHF_USD (fx), 4
CNY_GBP (fx), 4
CNY_USD (fx), 4
commodities, 2
crypto, 3
CSI (stock_indices), 8

DAX (stock_indices), 8
default, 3
DJ (stock_indices), 8
DJ_const (stock_indices_constituents), 9
DNB (losses), 6

EUR_GBP (fx), 4
EUR_USD (fx), 4
EURSTOXX (stock_indices), 8
EURSTX_const
  (stock_indices_constituents), 9

fire (losses), 6
FTSE (stock_indices), 8
FTSE_const
  (stock_indices_constituents), 9
fx, 4

GBP_USD (fx), 4
GOLD (commodities), 2

HSI (stock_indices), 8
HSI_const (stock_indices_constituents), 9

interest_rates, 5

JPY_GBP (fx), 4
JPY_USD (fx), 4

losses, 6

matrix, 6

NASDAQ (stock_indices), 8
NIKKEI (stock_indices), 8

OIL_Brent (commodities), 2

RSHCQ (stock_data), 7

SMI (stock_indices), 8
SP500 (stock_indices), 8
SP500_const
  (stock_indices_constituents), 9
SP500_const_info
  (stock_indices_constituents), 9

SP_defaults (default), 3
SSEC (stock_indices), 8
stock_data, 7
stock_indices, 8
stock_indices_constituents, 9

USD_GBP (fx), 4

VIX (volatility), 10

volatility, 10

xts, 2–10

ZCB_CAD (interest_rates), 5
ZCB_USD (interest_rates), 5