

Package: autoslider.trade (via r-universe)

October 2, 2026

Title Slide Automation for Trading Tables, Listings and Figures

Version 0.0.1

Description A downstream package of 'autoslider.core' that produces tables, listings and figures for finance trading, in the same style as 'autoslider'. Where 'autoslider.core' automates clinical study outputs, this package automates trading outputs from price and trade data: performance tables, equity curves and trade listings.

License Apache License (>= 2.0)

Depends R (>= 4.1.0)

Imports assertthat, autoslider.core, cowplot, formatters, ggplot2, rlistings, rtables, stats

Suggests roxygen2 (>= 7.0.0), testthat (>= 3.0.0)

Config/roxygen2/version 8.1.0

Config/testthat/edition 3

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

Config/pak/sysreqs libcairo2-dev cmake libfontconfig1-dev libfreetype6-dev libfribidi-dev make libharfbuzz-dev libicu-dev libjpeg-dev libpng-dev libtiff-dev libuv1-dev libwebp-dev libxml2-dev libssl-dev libnode-dev

Repository <https://shajoezhu.r-universe.dev>

Date/Publication 2026-10-01 16:42:09 UTC

RemoteUrl <https://github.com/shajoezhu/autoslider.trade>

RemoteRef HEAD

RemoteSha 812f6b8c129d90a8e578968be22b3b00af22cef8

Contents

autoslider.trade-package	2
eg_ohlc	2
eg_prices	3
eg_trades	4
g_candle_slide	4
g_equity_slide	5
l_trades_slide	6
nz_tickers	7
pptx_to_pdf	7
t_performance_slide	8
Index	10

autoslider.trade-package	<i>autoslider.trade Package</i>
--------------------------	---------------------------------

Description

Trading tables, listings and figures, downstream of autoslider.core. Outputs follow the autoslider naming convention: t*_slide() for tables, l*_slide() for listings and g*_slide() for figures.

Author(s)

Maintainer: Joe Zhu <sha.joe.zhu@gmail.com>
Authors:
• Joe Zhu <sha.joe.zhu@gmail.com>

eg_ohlc	<i>Example daily OHLCV prices</i>
---------	-----------------------------------

Description

A synthetic daily OHLCV series for three NZX instruments, used by the candlestick figure and its tests. homepage-stock stores no price history (only the ticker list) and fetching market data is out of scope for this package, so the series is generated deterministically.

Usage

eg_ohlc

Format

A data frame with 360 rows and 7 columns:

SYMBOL Trading code, one of AIA.NZ, AIR.NZ, ANZ.NZ

DATE Trade date, Date

OPEN Opening price, numeric

HIGH Intraday high, numeric

LOW Intraday low, numeric

CLOSE Closing price, numeric

VOLUME Shares traded, numeric

Source

Synthetic data created for this package.

eg_prices

Example daily closing prices

Description

A small set of daily closing prices for three trading codes, used in examples and tests.

Usage

```
eg_prices
```

Format

A data frame with 15 rows and 3 columns:

SYMBOL Trading code, one of AAA, BBB, CCC

DATE Trade date, Date

CLOSE Closing price, numeric

Source

Synthetic data created for this package.

eg_trades

Example trades

Description

A small set of trades used in examples and tests.

Usage

```
eg_trades
```

Format

A data frame with 5 rows and 5 columns:

SYMBOL Trading code

DATE Trade date, Date

SIDE BUY or SELL

QTY Quantity traded, integer

PRICE Execution price, numeric

Source

Synthetic data created for this package.

g_candle_slide

Candlestick figure with technical indicators

Description

A richer take on the `quantmod::chartSeries` plot used in `homepage-stock`: daily candlesticks with Bollinger Bands, moving averages and high/low annotations in the top panel, then a volume panel with a volume moving average, an RSI panel and a MACD panel. The four panels are stacked with `cowplot`.

Usage

```
g_candle_slide(
  prices,
  symbol = code_col(),
  date = "DATE",
  open = "OPEN",
  high = "HIGH",
  low = "LOW",
  close = "CLOSE",
```

```

    volume = "VOLUME",
    ma = c(20L, 50L),
    title = "Candlestick"
  )

```

Arguments

prices	data.frame of OHLCV prices
symbol	character Name of the trading code column
date	character Name of the date column
open	character Name of the open column
high	character Name of the high column
low	character Name of the low column
close	character Name of the close column
volume	character Name of the volume column
ma	integer Window(s) for the moving-average lines overlaid on the price panel. Set to NULL to omit them.
title	character Plot title

Details

Up days are drawn in green and down days in red. When more than one trading code is supplied, only the first is plotted and the rest are ignored with a message.

Value

A ggplot object, ready to be rendered with `generate_slides()`

Examples

```

g_candle_slide(eg_ohlc)

# Only the moving averages, no Bollinger Bands shown as separate colour
g_candle_slide(eg_ohlc, ma = 20L)

```

g_equity_slide

Equity curve figure

Description

Plot the price history of one or more trading codes. By default every instrument is rescaled to start at 1 so that several instruments with very different price levels can be compared on the same axes.

Usage

```
g_equity_slide(  
  prices,  
  symbol = code_col(),  
  date = "DATE",  
  close = "CLOSE",  
  normalize = TRUE,  
  title = "Equity Curve"  
)
```

Arguments

prices	data.frame of prices
symbol	character Name of the trading code column
date	character Name of the date column
close	character Name of the price column
normalize	logical Should every instrument start at 1? Defaults to TRUE.
title	character Plot title

Value

A ggplot object, ready to be rendered with generate_slides()

Examples

```
g_equity_slide(eg_prices)  
  
# Absolute prices instead of a common starting point  
g_equity_slide(eg_prices, normalize = FALSE)
```

<i>l_trades_slide</i>	<i>Trade listing</i>
-----------------------	----------------------

Description

A listing of executed trades, keyed by trading code. Each trade is shown with its date, side, quantity and price, plus the computed notional value.

Usage

```
l_trades_slide(trades, symbol = code_col())
```

Arguments

trades	data.frame of trades
symbol	character Name of the trading code column

Value

A listing_df object, ready to be rendered with generate_slides()

Examples

```
l_trades_slide(eg_trades)
```

nz_tickers	<i>NZ trading universe</i>
------------	----------------------------

Description

The full list of New Zealand instruments the user tracks on their homepage-stock kanban. Copied verbatim from that repository (168 instruments); non-ASCII company names are preserved.

Usage

```
nz_tickers
```

Format

A data frame with 168 rows and 2 columns:

name Company or fund name, character

symbol Trading code on the NZX, e.g. AIA.NZ, character

Source

Copied from homepage-stock/data/nz_list.csv.

pptx_to_pdf	<i>Convert PowerPoint files to PDF</i>
-------------	--

Description

Rendered slide decks are easier to share as PDF. pptx_to_pdf() converts one or more .pptx files, choosing automatically between the two converters it can find: LibreOffice (soffice --headless) or Microsoft PowerPoint through Windows COM automation. The latter is what makes the function usable from WSL, where no Linux converter is installed but a Windows Office is reachable.

Usage

```
pptx_to_pdf(path, output_dir = NULL)
```

Arguments

path	character vector of paths to .pptx files
output_dir	character Directory to write the PDFs into. Defaults to the directory of each input file.

Value

A character vector of the created PDF paths, invisibly

Examples

```
if (interactive()) {
  generate_slides(t_performance_slide(eg_prices), "performance.pptx")
  pptx_to_pdf("performance.pptx")
}
```

t_performance_slide	<i>Performance summary table</i>
---------------------	----------------------------------

Description

A table of per-instrument summary statistics: the number of observations, the first and last price, the return over the period and the maximum drawdown, expressed as percentages.

Usage

```
t_performance_slide(
  prices,
  symbol = code_col(),
  date = "DATE",
  close = "CLOSE",
  digits = 2L
)
```

Arguments

prices	data.frame of prices
symbol	character Name of the trading code column
date	character Name of the date column
close	character Name of the price column
digits	integer Number of decimal places in the output

Value

An rtables object, ready to be rendered with generate_slides()

t_performance_slide

9

Examples

```
t_performance_slide(eg_prices)
```

Index

* datasets

- eg_ohlc, [2](#)
- eg_prices, [3](#)
- eg_trades, [4](#)
- nz_tickers, [7](#)

autoslider.trade
 (autoslider.trade-package), [2](#)
autoslider.trade-package, [2](#)

eg_ohlc, [2](#)
eg_prices, [3](#)
eg_trades, [4](#)

g_candle_slide, [4](#)
g_equity_slide, [5](#)

l_trades_slide, [6](#)

nz_tickers, [7](#)

pptx_to_pdf, [7](#)

t_performance_slide, [8](#)