Backtest Graphics in R

Yanrong Song
Columbia University

Zijie Zhu
Columbia University

David Kane
Harvard University

Ziqi Lu
Harvard University

Karan Tibrewal
Williams Collage

Fan Zhang
Williams Collage

Abstract

The **backtestGraphics** package provides an interactive graphical interface to visualize backtests results for a variety of financial instruments (equities, futures, credit default swaps, et cetera).The **Shiny** interface, returned by the package, contains a sidebar panel with summary and detailed performance statistics, such as number of instruments, cumulative and annualized profit and loss, Sharpe ratio, top three drawdowns, and the three best and worst performers in the backtest. The main panel of interface houses interactive plots of profit and loss (P&L), net market value (NMV) and gross market value (GMV). Lastly, the package also supports vizualization of multiple strategies, sub-strategies, and overlapping portfolios that may be present in a single backtest result.

Keywords: backtesting, vizualization, R.

1. Introduction

Backtesting is the process of testing trading strategies on prior time horizons to measure the effectiveness of the given strategy. It helps investors understand and optimize their trading strategies [Campbell, Enos, Gerlanc, and Kane (2008)]. The **backtestGraphics** package provides an interactive **Shiny** interface to visualize backtest results for a variety of financial instruments (equities, futures, and credit defualt swaps, among others) [Chang, Cheng, Allaire, Sievert, Schloerke, Xie, Allen, McPherson, Dipert, and Borges (2024)]. These visualizations enable users to employ their human perception to process a lot of backtest data quickly and approximately [Bostock (2012)].

It is important to note here that **backtestGraphics** doesn't run backtests, but instead provides graphical visualization of backtest results. To illustrate this subtle distinction, consider a

simple trading strategy that buys the top 10 shares of the S&P 500, and shorts the bottom 10. Now, to test how the strategy performs historically, the user may use backtesting software like *Quantopian*, or even R packages like **backtest** [Campbell *et al.* (2008)].The output from such backtesting is generally in the form of large data frames which are difficult to interpret. As of such, the user may feed these data frames, i.e. the holding data, to **backtestGraphics**. The package then constructs interactive **dygraph** plots, and calculates essential summary statistics which are easy to interpret and explore [Vanderkam, Allaire, Owen, Gromer, and Thieurmel (2018)].

The **Shiny** interface returned by the package consists of a sidebar panel that includes a "Summary" and a "Detail" tab. The former provides the user with summary statistics, such as average gross market value (GMV), net market value (NMV), number of instruments, cumulative and annualized profit and loss (P&L), Sharpe ratio and best and worst performing months. The "Detail" tab provides information about the best and worst performers, as well as the biggest drawdowns.

The main panel of the **Shiny** interface houses interactive plots for cumulative and point-in-time P&L, NMV, GMV and number of contracts. The interactive nature of these plots provide the user with additional flexibility in exploring her backtest results by enabling her to zoom into specific time periods, or learn the value of a given response variable on a specific date.

Additionally, to accomodate for more complex backtests, **backtestGraphics** allows the user to subset seamlessly between overlapping portfolios, and multiple strategies and sub-strategies. For example, suppose the user splits his portfolio into two halves: the first half trades using the aforementioned trading strategy on a weekly basis, while the second uses the same strategy on a bi-weekly basis. Although these strategies overlap every other week, the user may want to explore how a particular strategy does in isolation. In order to do so, the user simply has to select the appropriate options from the dropdown menus provided in the side bar panel.

2. Data

The package comes with three sample data frames: `commodity`, `equity`, and `credit`. These are backtest results for commodity futures, equities, and credit default swaps (CDS), respectively. We use these data frames to demonstrate the capabilities of the backtestGraphics package. (Note: The user may also use her own data frame, as long as it adheres to the format specified in the documentation).

2.1. Equity

Let us begin with some details of `equity`. Equities are financial instruments that represent an ownership claim entitling the holder to dividends and ownership rights of the company for which the equities are issued [Johnson (2014)].The `equity` data frame contains holding data for a trading strategy dealing with equities.

```
 name        date sector        nmv         pnl
  A.Z 2005-05-02    MAT -131926.1   3158.4525
 ABFS 2005-05-02    IND -131926.1  -4058.6214
 ABS. 2005-05-02    CNS -131926.1   -733.2929
```

```
 A.Z 2005-05-03    MAT -128767.7  3644.3684
ABFS 2005-05-03    IND -135984.7  -335.9721
ABS. 2005-05-03    CNS -132659.4   733.2935
 A.Z 2005-05-04    MAT -125123.3 -1457.7478
ABFS 2005-05-04    IND -136320.7 -1595.8684
ABS. 2005-05-04    CNS -131926.1 -2466.5317
```

- `name:` column contains the each stock's symbol.

- `date:` column contains the trading date. (Note: this column must be of `Date` type).

- `sector:` column contains the sector of each stock.

- `nmv:` column contains the NMV at the beginning of the trading day for each stock. (Note: all values must be converted to the same currency).

- `pnl:` column contains P&L of each stock. (Note: all values must be converted to the same currency).

2.2. Commodity

Now, let us look at `commodity`. Simply put, a commodity futures contract is a legally binding agreement that provides for the delivery of the specified value of the said commodity at a specific time period in the future [Kleinman (2013)]. The `commodity` data frame consists of holding data from a trading strategy dealing in such commodity futures.

```
          name id       date       sector portfolio   strategy
         brent CO 2003-02-03       energy         1 Strategy 1
         brent CO 2003-02-03       energy         6 Strategy 1
  cattle feeder FC 2003-02-03    livestock         1 Strategy 3
      cbt wheat  W 2003-02-03  agriculture         1 Strategy 3
      cbt wheat  W 2003-02-07  agriculture         1 Strategy 3
 substrategy        gmv     nmv       pnl contract
Strategy 1.1    921600  921600  -21600.0       30
Strategy 1.2    445440  445440  -10441.5       14
Strategy 3.1  10426000  -58500   -7280.0      260
Strategy 3.2   3616300   14500    5800.0      232
Strategy 3.2   3738100   23200   23200.0      232
```

The columns `name`, `date`, `sector`, `nmv`, and `pnl` have the same meaning as the ones in `equity`. The remaining columns are listed below: (Note: these columns are optional).

- `ID:` column contains ID information for different commodities.

- `strategy:` column contains the strategy number.

- `substrategy:` column contains the sub-strategy number.

- `gmv:` column contains the GMV of a given commodity at the start of each trading day (Note: all values must be converted to the same currency).

- `contract:` column contains the number of contracts of a given commodity at the start of each trading day.

2.3. Credit

Finally, we look at `credit`. Credit Default Swaps (CDS) are a type of credit derivative that transfers the credit risk from one group of investors to another, in exchange for payment [Song, Zhu, Kane, and et al. (2024)]. The `credit` data frame contains holding data from a trading strategy dealing in such CDS.

```
          name       date              sector  strategy  gmv        nmv
  AA.Senior.5 2008-10-01 Basic Materials          daily 9e+06 259259.3
  AA.Senior.5 2008-10-01 Basic Materials        monthly 9e+06 259259.3
  AA.Senior.5 2008-10-01 Basic Materials      quarterly 9e+06 259259.3
 ABY.Senior.5 2008-10-01 Basic Materials          daily 9e+06 259259.3
 ABY.Senior.5 2008-10-01 Basic Materials        monthly 9e+06 259259.3
 ABY.Senior.5 2008-10-01 Basic Materials      quarterly 9e+06 259259.3
  AA.Senior.5 2008-10-02 Basic Materials          daily 9e+06 170731.7
 ABY.Senior.5 2008-10-02 Basic Materials          daily 9e+06 170731.7
  AA.Senior.5 2008-10-03 Basic Materials          daily 9e+06 170731.7
 ABY.Senior.5 2008-10-03 Basic Materials          daily 9e+06 170731.7
         pnl
    570.9305
      0.0000
  -55605.8350
  -65521.3738
 1343776.6544
  356766.3732
   39295.1220
 -149936.6291
   75865.3799
  349670.0335
```

Most of the columns here are the same as those in `eequity` and `commodity`. Notice that the `strategy` column here is used to denote backtest results for different trading frequencies. Towards this, the user must provide all backtest observations for each instrument, even when the position on the instrument is held and unchanged. This is because **backtestGraphics** uses the time gaps between each observation to determine the trading frequencies, and to calculate the annualized statistics.

3. Using the backtestGraphics Interface

In this section, we discuss the functionality of various components of the **Shiny** interface returned by the **backtestGraphics** function. To use backtestGraphics, the user is required to pass in a data frame with `date`, `ID/name`, `NMV` and `P&L` columns to the **backtestGraphics** function. Additionally, the user can also pass in optional columns, such as `sector`, `GMV`, `contracts`, `strategy`, `substrategy`, and `portfolio`.

As an example, let us look at the interface for the `commodity` data frame. Type the following command, and click the "Visualize" button on the Shiny interface returned by the function call.

```
R> library(backtestGraphics)
R> backtestGraphics(x = commodity)
```
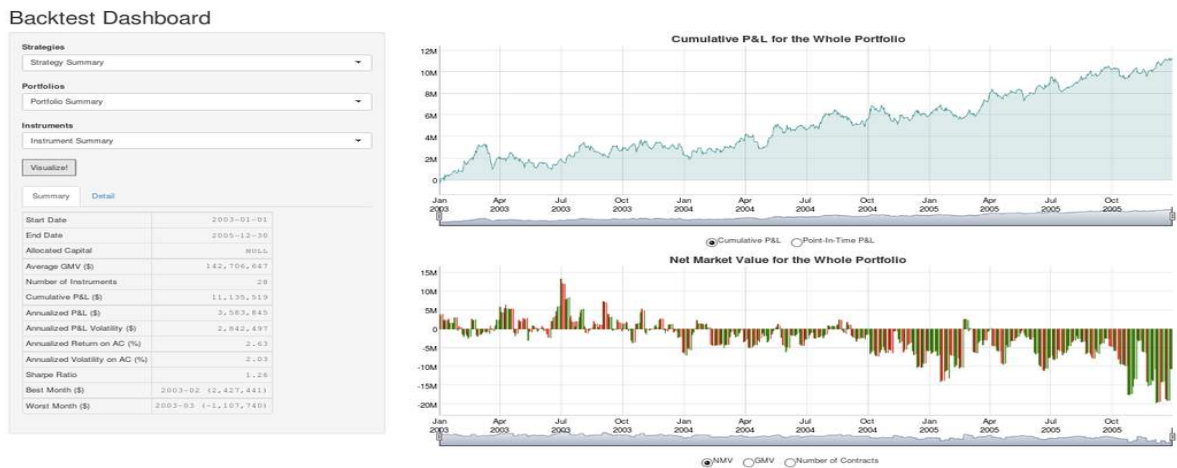

Figure 1: The `backtestGraphics` interface for the `commodity` data frame. The interface houses three dropdown menus on the top-left to toggle between Strategies, Portfolios, and Instruments. The user may click on the "Vizualize" button to look at the summary statistics and the interactive plots. Finally, the user can also switch between plots with the radio buttons at the bottom of the plots.

Now, let us individually examine the various components of the Shiny interface:

3.1. Dropdown Menus

The interface contains three dropdown menus on the top-left to toggle between different Strategies, Portfolios, and Instruments. By selecting different combinations of these menus, the user can visualize different subsets of strategies, portfolios and instruments.

If the given data frame does not have a `strategy` column, or a `portfolio` column, the respective dropdown menus will be fixed to "Strategy Summary" and "Portfolio Summary". If the user selects a combination that is incompatible with the given data frame, the interface will generate an error message and prompt the user to select a different combination.

Figure 2 provides a screenshot of the interface's dropdown menus.

Figure 2: Dropdown menus: The user can subset her backtest results along different strategies, portfolios and instruments by selecting appropriate choices from the dropdown menus.

3.2. The "Summary" Tab

The "Summary" tab is present in the left-panel of the `Shiny` interface. It provides summary statistics for the backtest result. Figure 3 contains a screenshot of the "Summary" tab for our commodity data frame, along with corresponding documentation.

Start Date	2003-01-01
End Date	2005-12-30
Allocated Capital	NULL
Average GMV ($)	142,706,647
Number of Instruments	28
Cumulative P&L ($)	11,135,519
Annualized P&L ($)	3,583,845
Annualized P&L Volatility ($)	2,842,497
Annualized Return on AC (%)	2.63
Annualized Volatility on AC (%)	2.03
Sharpe Ratio	1.26
Best Month ($)	2003-02 (2,427,441)
Worst Month ($)	2003-03 (-1,107,740)

Figure 3: The "Summary" tab for the `commodity` data frame. On clicking the "Visualize" button, the function slices the data frame according to inputs in the drop down menus to calculate summary statistics for the specified data subset.

Documentation of summary statistics:

- Start and End date : The backtest period.

- Allocated Capital : The amount of capital allocated to the portfolio. We use this number to calculate return on allocated capital. If the user does not specify this value, the corresponding entry will be `NULL` and 'backtestGraphics' will use the highest GMV to calculate return on allocated capital.

- Average GMV : Average GMV of the portfolio over the backtest period. GMV is calculated by taking the absolute value of NMV of each instrument.

- Number of Instruments : The number of different instruments in the given subset of the data frame.

- Cumulative P&L : Sum of all P&L over the backtest period.

- Annualized P&L : This is calculated as the Average P&L times an annualization factor. Our package observes the date interval between observations in the data frame to determine the trading frequency. The assumed annualization factor of daily trading is $\sqrt{252}$, while that for other trading frequency is $\sqrt{\frac{365}{date\ gap}}$.

- Annualized P&L volatility : The standard deviation of all annualized P&L's.

- Annualzied Return : Annualized average return on allocated capital. We calculate average return rates by dividing P&L by the allocated capital amount. The average return rates are then annualized by multiplying them with the annualization factor.

- Sharpe Ratio : A measure of risk-adjusted return. We calculate this by dividing the mean P&L by the standard deviation of P&L. We assume that risk-free rate $= 0$.

- Best Month and Worst Month : The month with the highest P&L, and the month with the lowest P&L

The "Detail" Tab

The "Detail" tab is also present in the left-panel of the `Shiny` interface. It provides information about the best and worst performers, as well as the biggest drawdowns. Figure 4 displays a screenshot of the "Detail" tab for our `commodity` data frame, along with corresponding documentation.

Top Three Drawdowns:

Start Date	End Date	P&L ($)
2003-03-11	2003-06-18	-2,306,913
2003-10-30	2004-01-09	-1,770,827
2004-10-11	2004-11-15	-1,732,102

Three Best Performers:

Tickers	P&L ($)
JX: tcom kerosene	2,780,786
FC: cattle feeder	1,730,847
CL: wti	1,285,581

Three Worst Performers:

Tickers	P&L ($)
SB: sugar	-1,085,512
QW: white sugar	-838,242
LH: lean hogs	-579,077

Figure 4: "Detail" tab with the `commodity` data frame.

Documentation of detailed statistics:

- Top three drawdowns : The three biggest declines in P&L from peak to trough. The table contains the start and end dates of the drawdowns, along with the actual value of the drawdowns.

- Best and worst three performers : The three best performers are the instruments with the highest three cumulative P&L's. Similarly, the three worst performers are the instruments with the lowest three cumulative P&L's. (Note: even if the user is looking at a subset of individual instruments, we will display information about the best and worst performers across all instruments. However, if the user is looking at data from a specific sector, we will display information about the best and worst performers in the specified sector).

3.3. Plots

The right-panel of our `Shiny` interface provides interactive plots for cumulative and point-in-time P&L, NMV, GMV and number of contracts.

Figure 5 provides a screenshot of the plots.

Figure 5: A screenshot of the plots with the `commodity` data frame. The P&L plots is displayed on the top, while the market value plots are on the bottom. Radio buttons at the bottom of the plots allow the user to seamlessly switch between plots.

We have made the plots interactive using the **dygraphs** package [Vanderkam *et al.* (2018)]. The user can zoom in to a specific time periods in a plot by dragging the corresponding region with the mouse, or changing the time slider bar, which is located below the plots. To go back to the original time scale, the user can merely double click on the respective plot. Additionally, by hovering the mouse over the plot, the user can get specific values of the response variable of the plot, on the specific date. The color of the plots is based on the sign of the point-in-time P&L. Green bars represent profits, while the red bars represent losses.

4. Conclusion

The **backtestGraphics** package provides a simple `Shiny` interface to visualize backtest results. Inside the package, we have provided three sample data frames. The user can use the code provided in this document to test the package with these data frames.

Our interface provides the user with summary, as well as detailed statistics about the backtest result. The user can subset her data according to strategies, sub-strategies, and instruments. Additionally, we also provide interactive plots that show movement of important variables over the backtest period.

These are powerful tools to uncover trends and anomalies in the data. By decomposing the data frame into different sectors and instruments, the user can gain important insights into his trading strategies over different sectors and time periods. The interactive plots provide additional flexibility to dive into the nitty-gritty details of the backtest.

Going forward, the **backtestGraphics** package can easily be customized to suit the needs of a specific user. Such customization may include, among others, adding new response variables to plots, redesigning plots, or introducing additional summary statistics. We invite users to take advantage of this, and redesign the **backtestGraphics** interface according to their preferences.

References

Bostock M (2012). *Time Series Visualization with Cubism.js*. URL https://bost.ocks.org/mike/cubism/intro/#0.

Campbell K, Enos J, Gerlanc D, Kane D (2008). "Backtests." *R News*, **7**(1).

Chang W, Cheng J, Allaire J, Sievert C, Schloerke B, Xie Y, Allen J, McPherson J, Dipert A, Borges B (2024). *shiny: Web Application Framework for R*. R package version 1.10.0, URL https://CRAN.R-project.org/package=shiny.

Johnson RS (2014). *Equity Markets and Portfolio Analysis*. John Wiley & Sons.

Kleinman G (2013). *Trading Commodities and Financial Futures*. FT Press, Upper Saddle River, NJ 07458.

Song Y, Zhu Z, Kane D, et al (2024). *creditr: Credit Default Swaps*. R package version 0.6.2, URL https://CRAN.R-project.org/package=creditr.

Vanderkam D, Allaire J, Owen J, Gromer D, Thieurmel B (2018). *dygraphs: Interface to 'Dygraphs' Interactive Time Series Charting Library*. R package version 1.1.1.6, URL https://CRAN.R-project.org/package=dygraphs.

Affiliation:

Yanrong Song
Columbia University in the City of New York
Financial Engineering
URL: https://github.com/yanrong-stacy-song