Backtest Graphics

by David Kane, Ziqi Lu, Karan Tibrewal, Fan Zhang, Miller Zijie Zhu

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). It provides handy facilities for visualizing backtest data, including:

- A panel to display summary performance statistics, like number of instruments, cumulative and annualized profit and loss, sharpe ratio and the best and worst performing months.
- A panel to display detailed performance statistics, like top three drawdowns, and the three best and worst performers in the backtest.
- Interactive plots of profit and loss (P&L), net market value (NMV) and gross market value (GMV).
- Seamless subsetting of backtest data to accommodate for visualization of multiple strategies, sub-strategies, and overlapping portfolios that may be present in a single backtest result.

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 et al. (2008)]. For more on backtesting, *Quantitative Value* [Gray and Carlisle (2013)] is a good introductory book.

backtestGraphics is not designed to run backtests, but instead, to enable users to employ their human perception to process lots of backtest data quickly and approximately [Bostock (2012)]. We provide fast and flexible tools that help in detecting trends and anomalies in the data. Our package returns a **Shiny** interface that allows the user to visually explore her backtest results [Chang et al. (2015)]. The interface contains a sidebar panel which includes "Summary" and "Detail" tabs. The former provides the user with summary statistics, such as average gross market value (GMV), number of instruments, cumulative and annualized profit and loss (P&L), sharpe ratio and best and worst performing months. The "Detail" tab provides the user with information about the top three drawdowns and the three best and worst performers in the backtest. The main panel of the interface houses **dygraph** plots of cumulative P&L, daily P&L, NMV and GMV [Vanderkam and Allaire (2015)]. 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.

Backtests vary in complexity. A simple backtest employs a single strategy in a single portfolio, while a more complex backtest might use multiple strategies and sub-strategies. In order to accommodate for this, our package provides two dropdown menus that allows the user to subset her backtest results in accordance to such overlapping portfolios for different (sub)strategies at different time periods.

Data

The package comes with three sample data frames: `commodity`, `equity`, and `credit`. These are backtest results for commodity futures, equities, and CDS, respectively. We use these data frames to demonstrate the capabilities of the backtestGraphics package.

Let us begin with some details about `equity`:

```
Source: local data frame [9 x 5]

    name       date sector       nmv        pnl
   (chr)     (date) (chr)      (dbl)      (dbl)
1   A.Z 2005-05-02    MAT -131926.1  3158.4525
2  ABFS 2005-05-02    IND -131926.1 -4058.6214
3  ABS. 2005-05-02    CNS -131926.1  -733.2929
4   A.Z 2005-05-03    MAT -128767.7  3644.3684
```

```
5  ABFS 2005-05-03    IND -135984.7  -335.9721
6  ABS. 2005-05-03    CNS -132659.4   733.2935
7   A.Z 2005-05-04    MAT -125123.3 -1457.7478
8  ABFS 2005-05-04    IND -136320.7 -1595.8684
9  ABS. 2005-05-04    CNS -131926.1 -2466.5317
```

- [,1] `name` column contains the name of each stock.
- [,2] `date` column contains the trading date. *(Note: this column must be of `Date` type).*
- [,3] `sector` column contains the sector of each stock.
- [,4] `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).*
- [,5] `pnl` column contains P&L of each stock. *(Note: all values must be converted to the same currency).*

Now, let us look at `commodity`:

```
Source: local data frame [5 x 11]

           name    id      date      sector portfolio   strategy  substrategy
          (chr) (chr)    (date)       (chr)     (int)      (chr)       (chr)
1         brent    CO 2003-02-03      energy         1 Strategy 1 Strategy 1.1
2         brent    CO 2003-02-03      energy         6 Strategy 1 Strategy 1.2
3 cattle feeder    FC 2003-02-03   livestock         1 Strategy 3 Strategy 3.1
4      cbt wheat    W 2003-02-03 agriculture         1 Strategy 3 Strategy 3.2
5      cbt wheat    W 2003-02-07 agriculture         1 Strategy 3 Strategy 3.2
Variables not shown: gmv (dbl), nmv (dbl), pnl (dbl), contract (dbl)
```

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).*

- [,2] ID column contains ID information for different commodities.
- [,6] `strategy` column contains the strategy number.
- [,7] `substrategy` column contains the sub-strategy number.
- [,8] `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).*
- [,11] `contract` column contains the number of contracts of a given commodity at the start of each trading day.

Finally, we look at `credit`:

```
Source: local data frame [10 x 7]

           name        date           sector  strategy    gmv        nmv
          (chr)      (date)            (chr)     (chr)   (dbl)      (dbl)
1   AA.Senior.5 2008-10-01 Basic Materials       daily 9e+06   259259.3
2   AA.Senior.5 2008-10-01 Basic Materials     monthly 9e+06   259259.3
3   AA.Senior.5 2008-10-01 Basic Materials   quarterly 9e+06   259259.3
4  ABY.Senior.5 2008-10-01 Basic Materials       daily 9e+06   259259.3
5  ABY.Senior.5 2008-10-01 Basic Materials     monthly 9e+06   259259.3
6  ABY.Senior.5 2008-10-01 Basic Materials   quarterly 9e+06   259259.3
7   AA.Senior.5 2008-10-02 Basic Materials       daily 9e+06   170731.7
8  ABY.Senior.5 2008-10-02 Basic Materials       daily 9e+06   170731.7
9   AA.Senior.5 2008-10-03 Basic Materials       daily 9e+06   170731.7
10 ABY.Senior.5 2008-10-03 Basic Materials       daily 9e+06   170731.7
Variables not shown: pnl (dbl)
```

Most of the columns here are the same as those in e`equity` 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.

Use `backtestGraphics`

To use `backtestGraphics`, the user is required to pass in a data frame with information on date, ID/name, NMV and P&L. Optional information can also be passed into `backtestGraphics`, such as both instrument name and ID, sector, GMV, number of contracts, strategy, substrategy and portfolio number. We will explain in "Interface" section how to calculate summary statistics and generate plots.

The three data frames in the package have the same column names as the default of `backtestGraphics` function. To look at backtestGraphics interface for `commodity` data, type the following command, and click on the "Visualize" button. Shiny will generate the interface in Figure 1.

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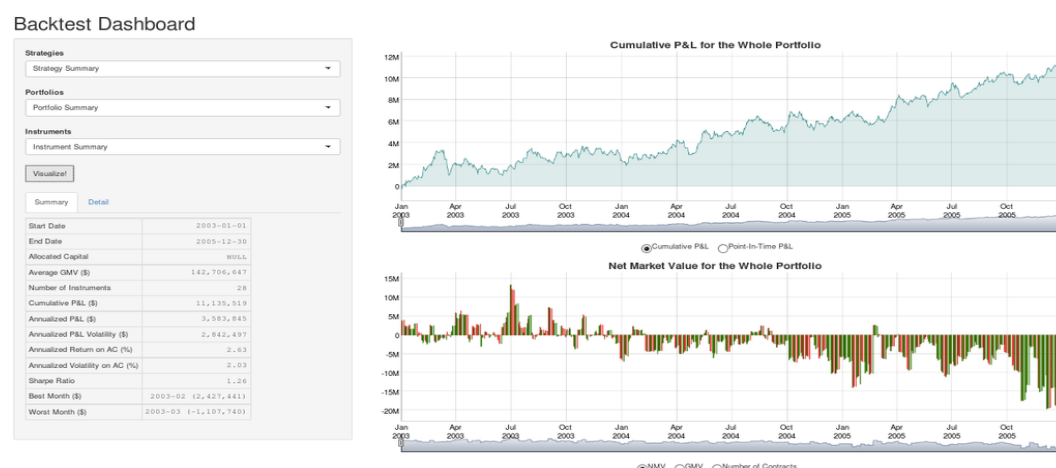

Figure 1: `backtestGraphics` interface. This interface contains three dropdown menus at the top-left. The user can click the "Visualize" button to look at summary statistics at the bottom left and interactive plots on the right. The user can also switch between graphs with radio buttons at the bottom of the plots.

Dropdown Menu

The interface has three dropdown menus for strategies, portfolios and instruments. The user can look at backtest results for a combination of strategy, portfolio, and instrument. If the data frame does not have the strategy column or the portfolio column, the two dropdown menus will be fixed to "Strategy Summary" or "Portfolio Summary". If the user selects a combination that is not available in the data frame, the interface will give an error message and ask the user to select a new combination.

Figure 2 is a screenshot of the interface's dropdown menu.
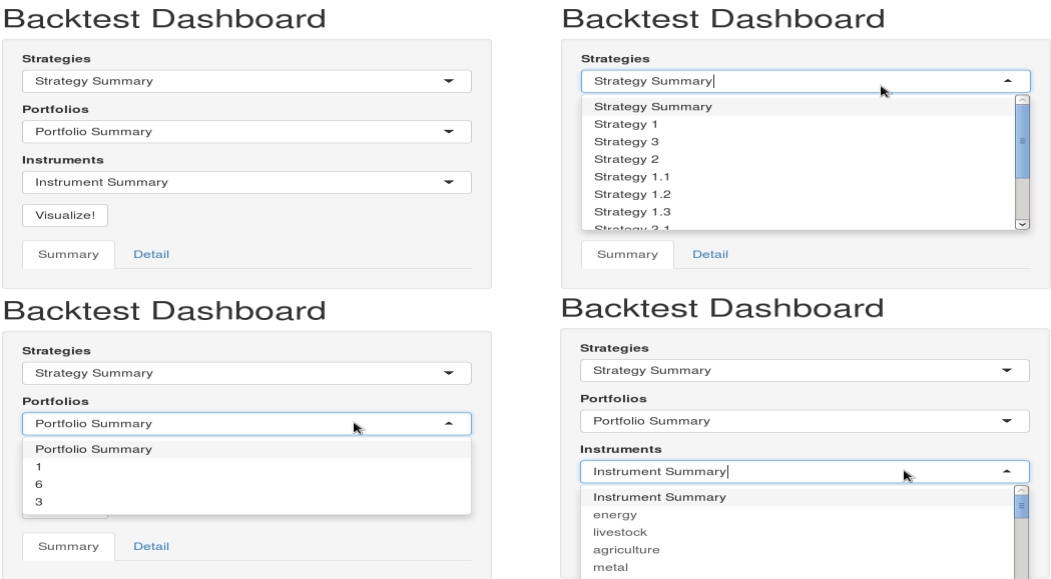
Figure 2: Dropdown menu. The user can look at backtest results for a combination of strategies, portfolios and instruments by selecting different choices in the dropdown menus. If the user selects a combination that is not available , the interface will give a clear error message and ask the user to select a new combination.

"Summary" Tab and "Detail" Tab

The interface has two tabs on the left: "Summary" tab and "Detail" tab. "Summary" tab shows summary statistics of the user-specified data, while "Detail" tab gives comparison across different instruments and different time periods. This information is calculated from the input data frame every time the user changes her combination of strategy, portfolio and instrument. Note that if the user is looking at data for individual instruments or instrument summary, the best/worst three performers are always selected from the entire portfolio and should remain the same.

"Summary" Tab

Figure 3 is a screenshot of the "Summary" tab.

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: "Summary" tab with commodity data. After the user clicks on the "Visualize" button, the function slices the data frame and calculates the summary statistics for the specific data subset. These are the main performance statistics.

- Start and End date : The backtest period.
- Allocated Capital : The amount of capital allocated to the portfolio. The function uses this number to calculate return on allocated capital. If the user does not specify allocated capital, this entry will be `NULL` and `backtestGraphics` will use biggest GMV to calculate return on allocated capital.
- Average GMV : Average of GMV over the backtest period. GMV is calculated by taking the absolute values of NMV of each instrument.
- Number of Instruments : The number of different instruments in the data frame or the user-specified subset of the data frame.
- Cumulative P&L : Sum of all P&L over the backtest period.
- Annualized P&L : Average P&L times an annualization factor. The function can observe the date gaps between observations and determine the trading frequency. The assumed annualization factor of daily trading is $\sqrt{252}$. For other trading frequency, the assumed annualization factor 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. If the user specifies the allocated capital amount, the functon will divide P&L by the allocated capital to calculate average return rates. If not, the function will divide P&L by the biggest GMV in the data. The function then multiplies average return by the annualization factor.
- Sharpe Ratio : A measure of risk-adjusted return. The function divides the mean P&L by the standard deviation of P&L to calculate sharpe ratio. We assume risk-free rate = 0.
- Best Month and Worst Month : The month with the highest P&L and the month with the lowest P&L

"Detail" Tab

Figure 4 is a screenshot of the "Detail" tab.

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 commodity data. This tab contains information about the best and worst performers, as well as the biggest drawdowns.

- 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 as well as the actual values of these drawdowns.
- Best and worst three performers : The best and worst three instruments with three highest and three lowest cumulative P&L's. If the user is looking at the subset of data about an individual instrument or instrument summary, the function will display the best and worst three performers across all instruments. If the user is looking at data about a specific sector, the function will instead display the best and worst three performers in the specified sector.

Plots

The interface displays interactive plots for cumulative and point-in-time P&L, NMV, GMV and number of contracts. Radio buttons at the bottom allow the user to quickly switch between graphs. The cumulative P&L plot is a filled line chart, and all others are bar charts.

Figure 5 is a screenshot of the plots.

Figure 5: Plots with `commodity` data. The plot of P&L is at the top and the plot of market values is at the bottom. Green bars represent profit and red bars represent loss. The user can zoom in on the plots by either selecting the corresponding region with mouse or changing the time slider bar.

The plots are interactive using the **dygraphs** package [Vanderkam and Allaire (2015)]. The user can zoom in to look at any specific time period on the plots. She can either select and drag the corresponding region with mouse, or change the time slider bar beneath the plots. The user can go back to the initial scale by double clicking the plot. Color of the plots is based on the sign of point-in-time P&L. Green bars represent profit and red bars represent loss. If the cursor is hovering above a specific bar, the legends of the plot will give specific values of the date and the response variable of that bar. Note that the data is automatically rounded. For example, a P&L of "2,856,000" will be presented as "2.86M".

Compared to summary statistics, interactive plots are more intuitive because the user can directly visualize the performance of different instruments across adjustable time period. By decomposing the data frame into different sectors and instruments, the user can closely look at profitability of different strategies as well as that of different sectors at different time periods.

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.

Authors

David Kane
Managing Director
Hutchin Hill Capital
101 Federal Street, Boston, USA
dave.kane@gmail.com

Ziqi Lu
Economics and Mathematics
Williams College
Williamstown, MA, USA
ziqi.lu@williams.edu

Karan Tibrewal
Mathematics and Computer Science
Williams College
Williamstown, MA, USA
karan.tibrewal@williams.edu

Fan Zhang
Economics and Statistics
Williams College
Williamstown, MA, USA
fan.zhang@williams.edu

Miller Zijie Zhu
Computer Science and Economics
Williams College
Williamstown, MA, USA
zijie.miller.zhu@gmail.com

Bibliography

M. Bostock. *Time Series Visualization with Cubism.js*, 2012. URL https://bost.ocks.org/mike/cubism/intro/#0. [p1]

K. Campbell, J. Enos, D. Gerlanc, and D. Kane. Backtests. *R News*, 7(1), MAY 2008. [p1]

W. Chang, J. Cheng, J. Allaire, Y. Xie, and J. McPherson. *shiny: Web Application Framework for R*, 2015. URL http://CRAN.R-project.org/package=shiny. R package version 0.12.1. [p1]

W. Gray and T. Carlisle. *Quantitative Value*. John Wiley & Sons, Inc., Hoboken, New Jersey, 2013. [p1]

D. Vanderkam and J. Allaire. *dygraphs: Interface to Dygraphs Interactive Time Series Charting Library*, 2015. URL http://CRAN.R-project.org/package=dygraphs. R package version 0.4.5. [p1, 7]