Backtests

Kyle Campbell, Jeff Enos, Daniel Gerlanc and David Kane

Introduction

The backtest package provides facilities for exploring portfolio-based conjectures about financial instruments (stocks, bonds, swaps, options, et cetera). For example, consider a claim that stocks for which analysts are raising their earnings estimates perform better than stocks for which analysts are lowering estimates. We want to examine if, on average, stocks with raised estimates have higher future returns than stocks with lowered estimates and whether this is true over various time horizons and across different categories of stocks. Colloquially, “backtest” is the term used in finance for such tests.

Background

To demonstrate the capabilities of the backtest package we will consider a series of examples based on a single real-world data set. StarMine\(^1\) is a San Francisco research company which creates quantitative equity models for stock selection. According to the company:

StarMine Indicator is a 1-100 percentile ranking of stocks that is predictive of future analyst revisions. StarMine Indicator improves upon basic earnings revisions models by:

- Explicitly considering management guidance.
- Incorporating SmartEstimates, StarMine’s superior estimates constructed by putting more weight on the most accurate analysts.
- Using a longer-term (forward 12-month) forecast horizon (in addition to the current quarter).

StarMine Indicator is positively correlated to future stock price movements. Top-decile stocks have annually outperformed bottom-decile stocks by 27 percentage points over the past ten years across all global regions.

These ranks and other attributes of stocks are in the starmine data frame, available as part of the backtest package.

```r
> data(starmine)
> names(starmine)
```

1. "date"
2. "id"
3. "symbol"
4. "name"
5. "country"
6. "sector"
7. "sec"
8. "ind"
9. "size"
10. "smi"
11. "liq"
12. "ret.0.1.m"
13. "ret.0.6.m"
14. "ret.1.0.m"
15. "ret.6.0.m"
16. "ret.12.0.m"
17. "m.dollar.volume.20.d"
18. "m.dollar.volume.120.d"
19. "cap.usd"
20. "cap"
21. "sales"
22. "net.income"
23. "common.equity"

starmine contains selected attributes such as sector, market capitalisation, country, and various measures of return for a universe of approximately 6,000 securities. The data is on a monthly frequency from January, 1995 through November, 1995. The number of observations varies over time from a low of 4,528 in February to a high of 5,194 in November.

```r
> date count
1995-01-31 4593
1996-02-28 4528
1995-03-31 4569
1995-04-30 4708
1995-05-31 4724
1995-06-30 4748
1995-07-31 4878
1995-08-31 5092
1995-09-30 5185
1995-10-31 5109
1995-11-30 5194
```

The smi column contains the StarMine Indicator score for each security and date if available. Here is a sample of rows and columns from the data frame:

```r
> data(starmine)
```

```r
> date name ret.0.1.m ret.0.6.m smi
1995-01-31 Lojack Corp 0.09 0.8 96
1995-02-28 Raymond Corp 0.05 0.1 85
1995-02-28 Lojack Corp 0.08 0.7 90
1995-03-31 Lojack Corp 0.15 1.0 49
1995-08-31 Supercuts Inc -0.11 -0.5 57
1995-10-31 Lojack Corp -0.40 -0.2 22
1995-11-30 Lojack Corp 0.20 0.4 51
```

Most securities (like LoJack above) have multiple entries in the data frame, each for a different date. The row for Supercuts indicates that, as of the close of business on August 31, 1995, its smi was 57. During the month of September, its return (i.e., ret.0.1.m) was -11%.

\(^1\)See www.starmine.com for details.
A simple backtest

Backtests are run by calling the function backtest to produce an object of class backtest.

```r
> bt <- backtest(starmine, in.var = "smi", +    ret.var = "ret.0.1.m", by.period = FALSE)
```

The `starmine` data frame contains all the information necessary to conduct the backtest. `in.var` and `ret.var` identify the columns containing the input and return variables, respectively. The `backtest` function splits observations into 5 (the default) quantiles, or “buckets,” based on the value of `in.var`. Lower (higher) buckets contain smaller (larger) values of `in.var`. Each quantile contains an approximately equal number of observations. This backtest creates quantiles according to values in the `smi` column of `starmine`.

```
[1,21] (21,40] (40,59] (59,82] (82,100] 6765 6885 6642 6600 6496
```

`backtest` calculates the average return within each bucket. From these averages we calculate the spread, or the difference between the average return of the highest and lowest buckets.

Calling `summary` on the resulting object of class `backtest` reports the `in.var`, `ret.var`, and `by.var` used. We will use a `by.var` in later backtests.

```r
> summary(bt)
Backtest conducted with:
 1 in.var: smi;
 1 ret.var: ret.0.1.m;
and no by.var;
do.spread: TRUE;
by.period: FALSE.

  spread
low  2  3  4 high
pooled 0.011 0.013 0.016 0.02  0.032  0.021
```

This backtest is an example of a pooled backtest. In such a backtest, we assume that all observations are exchangeable. This means that a quantile may contain observations for any stock and from any date. Quantiles may contain multiple observations for the same stock.

The backtest summary shows that the average return for the highest bucket was 3.2%. This value is the mean one month forward return of stocks with `smi` values in the highest quantile. As the observations are exchangeable, we use every observation in the `starmine` data frame with a non-missing `smi` value. This means that the returns for LoJack from both 1995-01-31 and 1995-02-28 would contribute to the 3.2% mean of the high bucket.

The backtest suggests that StarMine’s model predicted performance reasonably well. On average, stocks in the highest quantile returned 3.2% while stocks in the lowest quantile returned 1.1%. The spread of 2.1% suggests that stocks with high ratings perform better than stocks with low ratings.

Natural backtests

A natural backtest requires that the frequency of returns and observations be the same.

A natural backtest approximates the following implementation methodology: in the first period form an equal weighted portfolio with long positions in the stocks in the highest quantile and short positions in the stocks in the lowest quantile. Each stock has an equal weight in the portfolio; if there are 5 stocks on the long side, each stock has a weight of 20%. Subsequently rebalance the portfolio every time the `in.var` values change. If the observations have a monthly frequency, the `in.var` values change monthly and the portfolio must be rebalanced accordingly. When the `in.var` values change, rebalancing has the effect of exiting positions that have left the top and bottom quantiles and entering positions that have entered the top and bottom quantiles. If the data contains monthly observations, we will form 12 portfolios per year.

To create a simple natural backtest, we again call `backtest` using `ret.0.1.m`. This is the only return value in `starmine` for which we can construct a natural backtest of `smi`.

```r
> bt <- backtest(starmine, id.var = "id", +    date.var = "date", in.var = "smi", +    ret.var = "ret.0.1.m", natural = TRUE, +    by.period = FALSE)
```

Natural backtests require a `date.var` and `id.var`, the names of the columns in the data frame containing the dates of the observations and unique security identifiers, respectively. Calling `summary` displays the results of the backtest:

```r
> summary(bt)
Backtest conducted with:
 1 in.var: smi;
 1 ret.var: ret.0.1.m;
and no by.var;
do.spread: TRUE;
by.period: FALSE.

  spread
low  2  3  4 high
1995-01-31 0.003 0.011 0.003 -0.0001  0.019  0.016
1995-02-28 -0.008 -0.003 0.003  0.0072  0.013  0.021
1995-03-31  0.029  0.017 0.013  0.0225  0.037  0.008
1995-04-30 -0.002 -0.003 0.002 -0.0054  0.005  0.007
1995-05-31  0.010  0.013 0.019  0.0228  0.044  0.034
1995-06-30  0.072  0.059 0.057  0.0708  0.101  0.030
1995-07-31  0.033  0.030 0.034  0.0323  0.052  0.018
1995-08-31 -0.004  0.006 0.017  0.0119  0.024  0.028
1995-09-30 -0.055 -0.030 -0.031 -0.0219 -0.014  0.041
1995-10-31  0.030  0.032 0.040  0.0430  0.038  0.008
1995-11-30  0.013  0.016 0.021  0.0294  0.037  0.024
MEAN  0.011  0.014 0.016  0.0193  0.032  0.021
```
Focus on the mean return of the highest quantile for 1995-02-28 of 1.3%. backtest calculated this value by first computing the 5 quantiles of the input variable smi over all observations in starmine. Among the observations that fall into the highest quantile, those with date 1995-02-28 contribute to the mean return of 1.3%. It is important to note that the input variable quantiles are computed over the whole dataset, as opposed to within each category that may be defined by a date.var or by.var.

The bottom row of the table contains the mean quantile return over all dates. On account of the way we calculate quantile means, a single stock will have more effect on the quantile mean if during that month there are fewer stocks in the quantile. Suppose that during January there are only 2 stocks in the low quantile. The return of a single stock in January will account for \( \frac{1}{2} \) of the quantile mean. In a natural backtest, the weight of a single observation depends on the number of observations for that period.

Calling summary yields information beyond that offered by the summary method of a pooled backtest. The first piece of extra information is average turnover. Turnover is the percentage of the portfolio we would have to change each month if we implemented the backtest as a trading strategy. For example, covering all the shorts and shorting new stocks would yield a turnover of 50% because we changed half the portfolio. We trade stocks when they enter or exit the extreme quantiles due to in.var changes. On average, we would turn over 50% of this portfolio each month.

The second piece of extra information is mean spread. The spread was positive each month, so on average the stocks with the highest smi values outperformed the stocks with the lowest smi values. On average, stocks in the highest quantile outperformed stocks in the lowest quantile by 2%. The third piece of extra information, the standard deviation of spread, is 1%. The spread varied from month to month, ranging from a low of close to 0% to a high of over 4%.

We define the fourth piece of extra information, raw (non-annualized) Sharpe ratio, as \( \frac{\text{return}}{\text{risk}} \). We set return equal to mean spread return and use the standard deviation of spread return as a measure of risk.

More than one in.var

backtest allows for more than one in.var to be tested simultaneously. Besides using smi, we will test market capitalisation in dollars, cap.usd. This is largely a nonsense variable since we do not expect large cap stocks to outperform small cap stocks — if anything, the reverse is true historically.

```r
> bt <- backtest(starmine, id.var = "id",
+    date.var = "date", in.var = c("smi",
+    "cap.usd"), ret.var = "ret.0.1.m",
+    natural = TRUE, by.period = FALSE)
```

Because more than one in.var was specified, only the spread returns for each in.var are displayed, along with the summary statistics for each variable.

```r
> summary(bt)
```

<table>
<thead>
<tr>
<th>Backtest conducted with:</th>
<th>2 in-vars: smi, cap.usd; 1 ret.var: ret.0.1.m; and no by.var; do.spread: TRUE; by.period: FALSE.</th>
</tr>
</thead>
<tbody>
<tr>
<td>sà! cap.usd</td>
<td>1995-01-31 0.016 -0.0138 1995-02-28 0.021 0.0017 1995-03-31 0.008 -0.0023 1995-04-30 0.007 -0.0052</td>
</tr>
<tr>
<td></td>
<td>1995-05-31 0.034 -0.0568 1995-06-30 0.030 -0.0143 1995-07-31 0.018 -0.0008 1995-08-31 0.028 0.0051</td>
</tr>
<tr>
<td></td>
<td>1995-09-30 0.041 0.0321 1995-10-31 0.008 0.0127 1995-11-30 0.024 0.0029</td>
</tr>
</tbody>
</table>

**summary stats for in.var = smi:**

| average turnover: 0.5 | mean spread: 0.02 | sd spread: 0.01 | raw sharpe ratio: 2 |

**summary stats for in.var = cap.usd:**

| average turnover: 0.1 | mean spread: -0.004 | sd spread: 0.02 | raw sharpe ratio: -0.2 |

Viewing the results for the two input variables side-by-side allows us to compare their performance easily. As we expected, cap.usd as an input variable did not perform as well as smi over our backtest period. While smi had a positive return during each month, cap.usd had a negative return in 6 months and a negative mean spread. In addition, the spread
returns for cap.usd were twice as volatile as those of sml.

There are several plotting facilities available in backtest that can help illustrate the difference in performance between these two signals. These plots can be made from a natural backtest with any number of input variables. Below is a bar chart of the monthly returns of the two signals together:

```r
> plot(bt, type = "return")
```

![Spread Return](image1)

Figure 1: Monthly return spreads.

Returns for sml were consistently positive. Returns for cap.usd were of low quality, but improved later in the period. cap.usd had a particularly poor return in June. We can also plot cumulative returns for each input variable:

```r
> plot(bt.save, type = "cumreturn.split")
```

![Cumulative Spread Return](image2)

Figure 2: Cumulative spread and quantile returns.

The top region in this plot shows the cumulative return of each signal on the same return scale, and displays the total return and worst drawdown of the entire backtest period. The bottom region shows the cumulative return of the individual quantiles over time. We can see that sml’s top quantile performed best and lowest quantile performed worst. In contrast, cap.usd’s lowest quantile was its best performing.

Though it is clear from the summary above that sml generated about 5 times as much turnover as cap.usd, a plot is available to show the month-by-month turnover of each signal:

```r
> plot(bt, type = "turnover")
```

![Turnover](image3)

Figure 3: Monthly turnover.

This chart shows that the turnover of sml was consistently around 50% with lower turnover in September and October, while the turnover of cap.usd was consistently around 10%.

**Using by.var**

In another type of backtest we can look at quantile spread returns by another variable. Specifying by.var breaks up quantile returns into categories defined by the levels of the by.var column in the input data frame. Consider a backtest of sml by sector:

```r
> bt <- backtest(starmine, in.var = "smi", ret.var = "ret.0.1.m", by.var = "sector", by.period = FALSE)
> summary(bt)
```

Backtest conducted with:

1. in.var: sml;
2. ret.var: ret.0.1.m;
3. and by.var: sector;
This backtest categorises observations by the quantiles of smi and the levels of sector. The highest spread return of 2.6% occurs in Shops. Since smi quantiles were computed before the observations were split into groups by sector, however, we can not be sure how much confidence to place in this result. There could be very few observations in this sector or one of the top and bottom quantiles could have a disproportionate number of observations, thereby making the return calculation suspect. counts provides a simple check.

```r
> counts(bt)

$smi

    low 2 3 4 high
Durbl 348 349 261 231 223
Enrgy 246 250 158 130  64
HiTec 647 660 824 1004 1432
Hlth 380 377 410 464  64
Manuf 1246 1265 1279 1395 1576
Money 959 1265 1244 1095  875
NoDur 615 563 528 441  371
Other 1034 940 784 760  710
Shops 870 714 710 697  548
Telcm 186 177 140 129  95
Utils 152 245 252 198 130
```

While there seems to be an adequate number of observations in Shops, it is important to note that there are approximately 60% more observations contributing to the mean return of the lowest quantile than to the mean return of the highest quantile, 870 versus 548. Overall, we should be more confident in results for Manuf and Money due to their larger sample sizes. We might want to examine the result for HiTec more closely, however, since there are more than twice the number of observations in the highest quantile than the lowest.

by.var can also be numeric, as in this backtest using cap.usd:

```r
> bt <- backtest(starmine, by.var = cap.usd)
> summary(bt)

Backtest conducted with:
1 in.var: smi;
1 ret.var: ret.0.1.m;
and by.var: cap.usd;
do.spread: TRUE;
by.period: FALSE.

low 2 3 4 high
1 0.0105 0.0139 0.0236 0.028 0.038 0.028
2 0.0078 0.0093 0.0216 0.025 0.046 0.038
3 0.0186 0.0072 0.0167 0.031 0.034 0.016
4 0.0124 0.0142 0.0139 0.019 0.038 0.026
5 0.0080 0.0124 0.0087 0.010 0.025 0.017
6 0.0126 0.0121 0.0191 0.021 0.026 0.013
7 0.0080 0.0070 0.0160 0.019 0.034 0.026
8 0.0050 0.0181 0.0101 0.014 0.027 0.022
9 0.0104 0.0153 0.0167 0.014 0.028 0.018
10 0.0156 0.0207 0.0133 0.023 0.026 0.011
```

Since cap.usd is numeric, the observations are now split by two sets of quantiles. Those listed across the top are, as before, the input variable quantiles of smi. The row names are the quantiles of cap.usd. The buckets parameter of backtest controls the number of quantiles. The higher returns in the lower quantiles of cap.usd suggests that smi performs better in small cap stocks than in large cap stocks.

## Multiple return horizons

Using backtest we can also analyse the performance of a signal relative to multiple return horizons. Below is a backtest that considers one month and six month forward returns together:

```r
> bt <- backtest(starmine, in.var = "smi",
+    ret.var = c("ret.0.1.m", "+  "ret.0.6.m"), by.period = FALSE)
> summary(bt)

Backtest conducted with:
1 in.var: smi;
2 ret-vars: ret.0.1.m, ret.0.6.m;
and no by.var;
do.spread: TRUE;
by.period: FALSE.

low 2 3 high
ret.0.1.m 0.011 0.015 0.018 0.03 0.019
ret.0.6.m 0.112 0.121 0.142 0.17 0.059
```

The performance of smi over these two return horizons tells us that the power of the signal degrades after the first month. Using six month forward return, ret.0.6.m, the spread is 6%. This is only 3 times larger than the 2% spread return in the first month despite covering a period which is 6
times longer. In other words, the model produces 2% spread returns in the first month but only 4% in the 5 months which follow.

Conclusion

The backtest package provides a simple collection of tools for performing portfolio-based tests of financial conjectures. A much more complex package, portfolioSim, provides facilities for historical portfolio performance analysis using more realistic assumptions. Built on the framework of the portfolio package, portfolioSim tackles the issues of risk exposures and liquidity constraints, as well as arbitrary portfolio construction and trading rules. Above all, the flexibility of R itself allows users to extend and modify these packages to suit their own needs. Before reaching that level of complexity, however, backtest provides a good starting point for testing a new conjecture.

Kyle Campbell, Jeff Enos, Daniel Gerlanc and David Kane
Kane Capital Management
Cambridge, Massachusetts, USA
Kyle.W.Campbell@williams.edu, jeff@kanecap.com, dgerlanc@gmail.com and david@kanecap.com

Bibliography


2See Enos and Kane (2006) for an introduction to the portfolio package.