Analysing equity portfolios in R

Using the portfolio package

by David Kane and Jeff Enos

Introduction

R is used by major financial institutions around the world to manage billions of dollars in equity (stock) portfolios. Unfortunately, there is no open source R package for facilitating this task. The **portfolio** package is meant to fill that gap. Using **portfolio**, an analyst can create an object of class `portfolio`, examine its *exposures* to various factors, calculate its *performance* over time, and determine the *contributions* to performance from various categories of stocks. Exposures, performance and contributions are the basic building blocks of portfolio analysis.

One Period, Long-Only

Consider a simple long-only portfolio formed from the universe 30 stocks in the Dow Jones Industrial Average (DJIA) in January, 2005. Start by loading and examining the input data.

```
> library(portfolio)
> data(dow.jan.2005)
> summary(dow.jan.2005)

    symbol               name
 Length:30          Length:30
 Class :character   Class :character
 Mode  :character   Mode  :character




     price                 sector
 Min.   : 21.0   Industrials   :6
 1st Qu.: 32.1   Staples       :6
 Median : 42.2   Cyclicals     :4
 Mean   : 48.6   Financials    :4
 3rd Qu.: 56.0   Technology    :4
 Max.   :103.3   Communications:3
                 (Other)       :3
    cap.bil        month.ret
 Min.   : 22.6   Min.   :-0.12726
 1st Qu.: 53.9   1st Qu.:-0.05868
 Median : 97.3   Median :-0.02758
 Mean   :126.0   Mean   :-0.02914
 3rd Qu.:169.3   3rd Qu.: 0.00874
 Max.   :385.9   Max.   : 0.04468


> head(dow.jan.2005)

     symbol                           name price
140      AA                      ALCOA INC    31
214      MO              ALTRIA GROUP INC    61
270     AXP           AMERICAN EXPRESS CO    56
294     AIG AMERICAN INTERNATIONAL GROUP    66
946      BA                     BOEING CO    52
1119    CAT               CATERPILLAR INC    98
          sector cap.bil month.ret
140    Materials      27   -0.0608
214      Staples     125    0.0447
270   Financials      71   -0.0515
294   Financials     171    0.0094
946  Industrials      43   -0.0226
1119 Industrials      33   -0.0822
```

The DJIA consists of exactly 30 large US stocks. We provide a minimal set of information for constructing a long-only portfolio. Note that `cap.bil` is market capitalization in billions of dollars, `price` is the per share closing price on December 31, 2004, and `month.ret` is the one month return from December 31, 2004 through January 31, 2005.

In order to create an object of class `portfolio`, we can use this data and form a portfolio on the basis of a "nonsense" variable like `price`.

```
> p <- new("portfolioBasic", date = as.Date("2004-12-31"),
+     id.var = "symbol", in.var = "price",
+     sides = "long", ret.var = "month.ret",
+     data = dow.jan.2005)

An object of class "portfolioBasic" with 6 positions

Selected slots:
name: Unnamed portfolio
date: 2004-12-31
in.var: price
ret.var: month.ret
type: equal
size: quintile


> summary(p)

Portfolio: Unnamed portfolio

       count       weight
Long:      6            1

Top/bottom positions by weight:
   id pct
1 AIG  17
2 CAT  17
3 IBM  17
4 JNJ  17
5 MMM  17
6 UTX  17
```

In other words, we have formed a portfolio of the highest priced 6 stocks out of the 30 stocks in the DJIA. The `id.var` argument causes the portfolio to use `symbol` as the key for identifying securities throughout the analysis. `in.var` selects the variable by which stocks are ranked in terms of desirability. `sides` set to `long` specifies that we want a long-only portfolio. `ret.var` selects the return variable for measuring performance. In this case, we are interested in how the portfolio does for the month of January 2005.

The default arguments to `portfolioBasic` form equal-weighted positions; in this case, each of the 6 stocks has 16.67% of the resulting portfolio. The defaults also lead to a portfolio made up of the best 20% (or `quintile`) of the universe of stocks provided to the `data` argument. Since 20% of 30 is 6, there are 6 securities in the resulting portfolio.

Exposures

Once we have a portfolio, the next step is to analyse its exposures. These can be calculated with respect to both numeric or factor variables. The method `exposure` will accept a vector of variable names.

```
> exposure(p, exp.var = c("price", "sector"))

An object of class "exposure"
numeric
  variable exposure
1    price       85

sector
     variable exposure
2 Industrials     0.50
1  Financials     0.17
3     Staples     0.17
4  Technology     0.17
```

The weighted average price of the portfolio is 85. In other words, since the portfolio is equal-weighted, the average share price of AIG, CAT, IBM, JNJ, MMM, and UTX is 85. This is a relatively high price, but makes sense since we explicitly formed the portfolio by taking the 6 highest priced stocks out of the DJIA.

Each of the 6 stocks is assigned a sector and 3 of them are Industrials. This compares to the 20% of the entire universe of 30 stocks that are in this sector. In other words, the portfolio has a much higher exposure to Industrials than the universe as a whole. Similarly, the portfolio has no stocks from the Communications, Cyclicals or Energy sectors despite the fact that these make up almost 27% of the DJIA universe.

Performance

Time plays two roles in the `portfolio` class. First, there is the moment of portfolio formation. This is the instant when all of the data, except for future returns, is correct. After this moment, of course, things change. The price of AIG on December 31, 2004 was $65.67, but by January 5, 2005 it was $67.35.

The second role played by time on the `portfolio` class concerns future returns. `ret.var` specifies a return variable which measures the performance of individual stocks going forward. These returns can be of any duration — an hour, a day, a month, a year — but should generally start from the moment of portfolio formation. In this example, we are using one month forward returns. Now that we know the portfolio's exposures at the start of the time period, we can examine its performance for the month of January.

```
> summary(performance(p))

Performance summary:

Total return:  -1.71 %

Best/Worst performers:
   id weight     ret contrib
2 CAT   0.17 -0.0822 -0.0137
3 IBM   0.17 -0.0523 -0.0087
6 UTX   0.17 -0.0258 -0.0043
1 AIG   0.17  0.0094  0.0016
4 JNJ   0.17  0.0202  0.0034
5 MMM   0.17  0.0279  0.0047
```

The portfolio lost 1.7% of its value in January. The worst performing stock was CAT (Caterpillar), down more than 8%. The best performing stock was MMM (3M), up almost 3%. The `contrib` (contribution) of each stock to the overall performance of the portfolio is simply its weight in the portfolio multiplied by its return. The sum of the 6 individual contributions yields -1.7%.

Contributions

The contributions of individual stocks are not that interesting in and of themselves. More important is to examine summary statistics of the contributions across different categories of stocks. Consider the use of the `contribution` method:

```
> contribution(p, contrib.var = c("sector"))

An object of class "contribution"
sector
         variable weight contrib    roic
11 Communications   0.00  0.0000  0.0000
21  Conglomerates   0.00  0.0000  0.0000
31      Cyclicals   0.00  0.0000  0.0000
41         Energy   0.00  0.0000  0.0000
1      Financials   0.17  0.0016  0.0094
2     Industrials   0.50 -0.0134 -0.0267
5       Materials   0.00  0.0000  0.0000
3         Staples   0.17  0.0034  0.0202
4      Technology   0.17 -0.0087 -0.0523
```

```
6       Utilities   0.00  0.0000  0.0000
```

`contribution`, like `exposure`, accepts a vector of variable names. In the case of `sector`, the contribution object displays the 10 possible values, the total weight of the stocks for each level and the sum of the contributions for those stocks. Only 4 of the 10 levels are represented by the stocks in the portfolio. The other 6 levels have zero weight and, therefore, zero contributions.

The sector with the biggest weight is Industrials, with half of the portfolio. Those 3 stocks did poorly, on average, in January and are therefore responsible for -1.3% in total losses. There is only a single Technology stock in the portfolio, IBM. Because it was down 5% for the month, its contribution was -0.87%. Since IBM is the only stock in its sector in the portfolio, the contribution for the sector is the same as the contribution for IBM.

The last column in the display shows the `roic` — the return on invested capital — for each level. This captures the fact that raw contributions are a function of both the total size of positions and their return. For example, the reason that the total contribution for Industrials is so large is mostly because they account for such a large proportion of the portfolio. The individual stocks performed poorly, on average, but not that poorly. Much worse was the performance of the Technology stock. Although the total contribution from Technology was only 60% of that of Industrials, this was on a total position weight only 1/3 as large. In other words, the return on total capital was *much worse* in Technology than in Industrials even though Industrials accounted for a larger share of the total losses.

Think about `roic` as useful in determining contributions on the margin. Imagine that you have the chance move $1 out of one sector and into another. Where should than initial dollar come from? Not from the sector with the worst total contribution. Instead, the marginal dollar should come from the sector with the worst `roic` and should be placed into the sector with the best `roic`. In this example, we should move money out of Technology and into Staples.

`contribution` can also work with numeric variables.

```
> contribution(p, contrib.var = c("cap.bil"))

An object of class "contribution"
cap.bil
       rank    variable weight  contrib    roic
1   1 - low (22.6,50.9]   0.17 -0.01370 -0.0822
2         2 (50.9,71.1]   0.33  0.00034  0.0010
11        3  (71.1,131]   0.00  0.00000  0.0000
3         4   (131,191]   0.50 -0.00379 -0.0076
21 5 - high   (191,386]   0.00  0.00000  0.0000
```

Analysing contributions only makes sense in the context of categories into which each position can be placed. So, we need to break up a numeric variable like `cap` into discrete, exhaustive categories. The `contribution` function provides various options for doing so, but most users will be satisfied with the default behavior of forming 5 equal sized quintiles based on the distribution of the variable in the entire universe.

In this example, we see that there are no portfolio holdings among the biggest 20% of stocks in the DJIA. Half the portfolio comes from stocks in the second largest quintile. Within each category, the analysis is the same as that above for sectors. The worst performing category into terms of total contribution is the smallest quintile. This category also has the lowest `roic`.

One Period Long-Short

Having examined a very simple long-only portfolio in order to explain the concepts behind *exposures*, *performance* and *contributions*, it is time to consider a more complex case, a long-short portfolio which uses the same underlying data.

```
> p <- new("portfolioBasic", date = as.Date("2004-12-31"),
+     id.var = "symbol", in.var = "price",
+     type = "linear", sides = c("long",
+         "short"), ret.var = "month.ret",
+     data = dow.jan.2005)

An object of class "portfolioBasic" with 12 positions

Selected slots:
name: Unnamed portfolio
date: 2004-12-31
in.var: price
ret.var: month.ret
type: linear
size: quintile


> summary(p)

Portfolio: Unnamed portfolio

       count      weight
Long:      6           1
Short:     6          -1

Top/bottom positions by weight:
     id   pct
12  UTX  28.6
5   IBM  23.8
2   CAT  19.0
8   MMM  14.3
1   AIG   9.5
10  PFE  -9.5
9  MSFT -14.3
11  SBC -19.0
6  INTC -23.8
4   HPQ -28.6
```

Besides changing to a long-short portfolio, we have also provided a value of "linear" to the `type` argument. As the summary above shows, this yields a portfolio in which the weights on the individual positions are (linearly) proportional to their share prices. At the end of 2004, the lowest priced stock in the DJIA was HPQ (Hewlett-Packard) at $20.97. Since we are using price as our measure of desirability, HPQ is the biggest short position. Since we have not changed the default value for `size` from "quintile," there are still 6 positions per side.

The display for the exposure object is now somewhat different to accommodate the structure of a long-short portfolio.

```
> exposure(p, exp.var = c("price", "sector"))

An object of class "exposure"
numeric
  variable long short exposure
1    price   93   -24       68

sector
         variable  long  short exposure
2     Industrials 0.619  0.000    0.619
1      Financials 0.095  0.000    0.095
3         Staples 0.048 -0.095   -0.048
5  Communications 0.000 -0.238   -0.238
4      Technology 0.238 -0.667   -0.429
```

The `long` exposure to price is simply the weighted average of price on the long side of the portfolio, where the weighting is done in proportion to the size of the position in each stock. The same is true on the short side. Since a linear weighting used here emphasises the tail of the distribution, the long exposure is greater than the long exposure of the equal weighted portfolio considered above, $93 versus $85.

Since the weights on the short side are actually negative — in the sense that we are negatively exposed to positive price changes in these stocks — the weighted average on the short side for a positive variable like price is also negative. Another way to read this is to note that the weighted average price on the short side is about $24 but that the portfolio has a negative exposure to this number because these positions are all on the short side.

One reason for the convention of using a negative sign for short side exposures is that doing so makes the overall exposure of the portfolio into a simple summation of the long and short exposures. (Note the assumption that the weights on both sides are equal. In future versions of the **portfolio** package, we hope to weaken these and other requirements.) For this portfolio, the overall exposure is 68. Because the portfolio is long the high priced stocks and short the low priced ones, the portfolio has a positive exposure to the price factor. Colloquially, we are "long price."

A similar analysis applies to sector exposures. We have 62% of our long holdings in Industrials but zero of our short holdings. We are, therefore, 62% long Industrials. We have 24% of the longs holdings in Technology, but more 67% of the short holdings; so we are 43% short Technology.

The performance object is similar for a long-short portfolio.

```
> summary(performance(p))

Performance summary:

Total return:  2.19 %

Best/Worst performers:
     id weight     ret contrib
2   CAT  0.190 -0.0822 -0.0157
5   IBM  0.238 -0.0523 -0.0125
12  UTX  0.286 -0.0258 -0.0074
3   DIS -0.048  0.0299 -0.0014
1   AIG  0.095  0.0094  0.0009
8   MMM  0.143  0.0279  0.0040
6  INTC -0.238 -0.0402  0.0096
10  PFE -0.095 -0.1015  0.0097
11  SBC -0.190 -0.0662  0.0126
4   HPQ -0.286 -0.0658  0.0188
```

The portfolio was up 2.2% in January. By default, the summary of the `performance` object only provides the 5 worst and best contributions to return. HPQ was the biggest winner because, though it was only down 7% for the month, its large weighting caused it to contribute almost 2% to overall performance. The 19% weight of CAT in the portfolio placed it as only the third largest position on the long side, but its -8% return for the month made it the biggest drag on performance.

The `contribution` function provides similar output for a long-short portfolio.

```
> contribution(p, contrib.var = c("cap.bil",
+     "sector"))

An object of class "contribution"
cap.bil
       rank    variable weight contrib    roic
1  1 - low (22.6,50.9]  0.095 -0.0157 -0.1644
2        2 (50.9,71.1]  0.381  0.0140  0.0367
3        3  (71.1,131]  0.095  0.0126  0.1323
4        4   (131,191]  0.310 -0.0010 -0.0033
5 5 - high   (191,386]  0.119  0.0120  0.1010

sector
          variable weight contrib   roic
1   Communications  0.119  0.0112  0.094
11   Conglomerates  0.000  0.0000  0.000
21       Cyclicals  0.000  0.0000  0.000
31          Energy  0.000  0.0000  0.000
2       Financials  0.048  0.0009  0.019
3      Industrials  0.310 -0.0191 -0.062
41       Materials  0.000  0.0000  0.000
```

```
4          Staples  0.071  0.0106  0.149
5       Technology  0.452  0.0183  0.040
51       Utilities  0.000  0.0000  0.000
```

As in the last example, the `weight` column contains the sum of the absolute value of weights in each category, scaled to $[0,1]$. In this case, we have 45% of the total capital (or weight) of the long and short sides *considered together* invested in the Technology sector. We know from the exposure results above that most of this is invested on the short side, but in the context of contributions, it does not matter on which side the capital is deployed.

Multi-Period, Long-Short

Analysing a portfolio for a single time period is a useful starting point, but any serious research will require considering a collection of portfolios over time. The `help` pages of the **portfolio** package provide detailed instructions on how to construct a `portfolioHistory` object. Here, we will just load up the example object provided with the package and consider the associated methods for working with it.

```
> data(global.2004.history)
> global.2004.history

An object of class "portfolioHistory"
Contains:

Object of class "exposureHistory"
Number of periods: 12
Period range: 2003-12-31 -- 2004-11-30
Data class: exposure
Variables: numeric currency sector

Object of class "performanceHistory"
Number of periods: 12
Period range: 2003-12-31 -- 2004-11-30
Data class: performance

Object of class "contributionHistory"
Number of periods: 12
Period range: 2003-12-31 -- 2004-11-30
Data class: contribution
Variables: currency sector liq.w
```

Note that the portfolios analysed in this object were created on a monthly frequency for 2004 using a universe of the 500 largest global stocks each month. Market capitalization in billions of dollars (`cap.bil`) was used as a measure of desirability, so the portfolio is long the mega-large caps and short the merely typical large caps.

We have designed the architecture of the **portfolio** package so that working with multi-period portfolios is similar to working with single period portfolios. For starters, we can call the `exposure` method on objects of class `portfolioHistory` just as we called them on objects of class `portfolio`.

```
> exposure(global.2004.history)

Mean exposure:
numeric
   variable  long  short exposure
11  cap.bil 109.2 -10.48     98.7
1     liq.w   1.2   0.82      2.0

currency
   variable    long    short exposure
6       JPY 0.06298 -0.15893 -0.09595
8       SEK 0.00095 -0.04021 -0.03926
5       HKD 0.00000 -0.01728 -0.01728
1       AUD 0.00124 -0.00573 -0.00449
7       NOK 0.00000 -0.00395 -0.00395
9       SGD 0.00000 -0.00044 -0.00044
3       EUR 0.17990 -0.17690  0.00300
4       GBP 0.11359 -0.10912  0.00447
2       CHF 0.04234 -0.01923  0.02311
10      USD 0.59900 -0.46820  0.13080

sector
         variable   long   short exposure
10      Utilities 0.0080 -0.0883 -0.08029
3       Cyclicals 0.0498 -0.1247 -0.07495
7       Materials 0.0011 -0.0588 -0.05764
6     Industrials 0.0415 -0.0976 -0.05615
2   Conglomerates 0.0000 -0.0036 -0.00356
9      Technology 0.0878 -0.0873  0.00055
8         Staples 0.2452 -0.2317  0.01353
5      Financials 0.2454 -0.1925  0.05298
4          Energy 0.1033 -0.0346  0.06872
1  Communications 0.1914 -0.0539  0.13759
```

The difficulty in considering exposures — basically a static concept about how the portfolio looks at this moment — in the context of a time series of `portfolio` objects is that it is not obvious how to summarize the exposures over time, especially in the context of a portfolio whose overall size is changing. In this release of the **portfolio** package, we will simply report the average exposure for all time periods.

For this example portfolio, it is easy to see that the portfolio is long the biggest stocks. The average market capitalization is over \$100 billion on the long side versus only \$10 billion on the short side. This cap exposure causes the portfolio to take a similar "bet" on liquidity, since larger stocks are, on average, more liquid than smaller stocks. The `liq.w` variable is our measure of liquidity — basically how much stock is traded on a "typical" day — scaled to be $N(0,1)$ for this universe.

The `currency` and `sector` exposures are, likewise, simple averages of the of the 12 monthly exposures. Even though the portfolio is constructed without reference to `currency` or `sector`, it still ends up with non-trivial exposures to these variables. In a typical month, the portfolio is 13% long Communications and 10% short Japan. In future versions of the package, we plan on adding other analytics as well as plotting methods.

The `performance` method aggregates performance for each individual time period into an overall figure and provides a listing of the largest contributions.

```
> performance(global.2004.history)

Performance summary (frequency = 12):

                ret
mean        -0.0081
mean (ann) -0.0971
sd          0.0152
sd (ann)    0.0527
sharpe     -1.8439

Best period:
                 date   ret
2004-03-31 2004-03-31 0.015


Worst period:
                 date    ret
2004-10-31 2004-10-31 -0.029

Top/Bottom performers (total contribution):
          id contrib
140     EBAY  0.0042
231   MRW.LN  0.0039
234       MU  0.0036
338      XOM  0.0034
196      JNJ  0.0033
211      LTR -0.0042
323 VOLVB.SS -0.0045
329      WLP -0.0047
227      MON -0.0049
11      5201 -0.0056
```

The best month for the portfolio was April 2004 when it earned 1.5%. The worst month was a loss of almost 3% in November. The biggest contributor on the positive side was EBAY. The mean annualized return of the portfolio is almost -10% was the annualized volatility is 5.3%. Sharpe ratios — defined[1] as the mean return divided by the standard deviation of return — are generally not provided for unprofitable portfolios, but the ratio here is -1.8.

Note that the `portfolioHistory` object does not keep a record of every position for each time period. By design, it only retains summary information, albeit on a stock-by-stock basis. We want to be able to use this package for high frequency data — say, a decade worth of daily portfolios — so it is not easy (or even possible) to manipulate an object which maintain information for every stock for each time period.

As with the `exposure` method, the `contribution` method works with `portfolioHistory` objects in the same way that it works with `portfolio` objects.

```
> contribution(global.2004.history)

Mean contribution:
currency
   variable  weight    contrib    roic
1       AUD 0.00349 -0.0001418 -0.0262
2       CHF 0.03079  0.0000087 -0.0029
11      DKK 0.00000  0.0000000  0.0000
3       EUR 0.17840 -0.0010476 -0.0062
4       GBP 0.11135 -0.0004812 -0.0041
5       HKD 0.00864  0.0001175 -0.0072
6       JPY 0.11095  0.0000511 -0.0027
21      NOK 0.00197 -0.0001276 -0.0375
7       SEK 0.02058 -0.0011139 -0.0467
31      SGD 0.00022 -0.0000096 -0.0126
8       USD 0.53360 -0.0053475 -0.0100

sector
         variable weight   contrib    roic
1  Communications 0.1261  0.000848  0.0081
11  Conglomerates 0.0018 -0.000075 -0.0104
2       Cyclicals 0.0896 -0.002543 -0.0286
3          Energy 0.0708  0.000815  0.0138
4      Financials 0.2248 -0.001019 -0.0042
5     Industrials 0.0715 -0.001492 -0.0199
6       Materials 0.0307 -0.000247 -0.0125
7         Staples 0.2451 -0.003618 -0.0148
8      Technology 0.0901  0.000259  0.0023
9       Utilities 0.0495 -0.001193 -0.0261

liq.w
  variable weight contrib   roic
1  1 - low  0.263 -0.0039 -0.014
2        2  0.161 -0.0055 -0.035
3        3  0.067 -0.0014 -0.020
4        4  0.095 -0.0017 -0.016
5 5 - high  0.414  0.0038  0.009
```

Again, the basic approach consists of calculating the contributions for each time period separately, saving those results, and the presenting an average for the entire period.

Conclusion

The current release of the portfolio package is meant to serve as a proof-of-concept. Relatively sophisticated portfolio analytics are possible using an open source package. Although money management is largely a zero-sum game — every dollar that I make is a dollar that someone else has lost — there is no reason why we might not cooperate on the tools that we all use.

[1] The definition of the Sharpe ratio subtracts the risk free rate from the mean return in the numerator. But, in the context of a long-short portfolio, this is typically ignored.

David Kane and Jeff Enos
Kane Capital Management, Cambridge, Massachusetts, USA
`david@kanecap.com` *and* `jeff@kanecap.com`