Abstract

This vignette provides an overview of calculating portfolio returns through time with an emphasis on the math used to develop the `Return.portfolio` function in **PerformanceAnalytics**. We first introduce some basic definitions, then give simple examples of computing portfolio returns in a prices and shares framework as well as a returns and weights framework. We then introduce `Return.portfolio` and demonstrate the function with a few examples.

Contents

1 Basic definitions

Suppose we have a portfolio of N assets. The value of asset i, V_i, in the portfolio is defined as

$$V_i = \lambda_i * P_i$$

where:

$$\lambda_i \text{ is the number of shares of asset } i$$
$$P_i \text{ is the price of asset } i$$

The total portfolio value, V_p, is defined as

$$V_P = \sum_{i=1}^{N} V_i$$

The weight of asset i, w_i, in the portfolio is defined as

$$w_i = V_i / V_P$$

where:

$$V_i \text{ is the value of asset } i$$
$$V_P \text{ is the total value of the portfolio}$$

The portfolio return at time t, R_t, is defined as

$$R_t = \frac{V_{p_t} - V_{p_{t-1}}}{V_{p_{t-1}}}$$

$$V_{p_t} \text{ is the portfolio value at time } t$$

2 Simple Example: Prices and Shares Framework

Suppose we have a portfolio of $N = 2$ assets, asset A and asset B. The prices for assets A and B are given as

```
prices = cbind(c(5, 7, 6, 7),
               c(10, 11, 12, 8))
dimnames(prices) = list(paste0("t",0:3), c("A", "B"))
prices

##    A  B
## t0 5 10
## t1 7 11
## t2 6 12
## t3 7  8
```

We wish to form an equal weight portfolio, that is, form a portfolio where

$$w_i = \frac{1}{N} \text{ for } i \in 1, \ldots, N.$$

Let $V_{P0} = 1000$ be the portfolio value at t_0.
Step 1: Compute the number of shares of each asset to purchase.

$$\begin{aligned} w_i &= \frac{V_i}{V_P} \\ &= \frac{\lambda_i * P_i}{V_P} \end{aligned}$$

Solve for λ_i.

$$\lambda_i = \frac{w_i * V_P}{P_i}$$

$$\begin{aligned} \lambda_A &= \frac{w_A * V_P 0}{P_A} = \frac{0.5 * \$1000}{\$5} = 100 \\ \lambda_B &= \frac{w_B * V_P 0}{P_B} = \frac{0.5 * \$1000}{\$10} = 50 \end{aligned}$$

```
V_P0 = 1000
N = ncol(prices)
w = rep(1 / N, N)
lambda = w * V_P0 / prices["t0",]
lambda

##   A   B
## 100  50
```

Step 2: Compute the asset value and portfolio value for $t \in 0, \ldots, 3$.

```
# Compute the value of the assets
V_assets <- matrix(0, nrow(prices), ncol(prices), dimnames=dimnames(prices))
for(i in 1:nrow(prices)){
  V_assets[i,] = prices[i,] * lambda
}
V_assets

##      A   B
## t0 500 500
## t1 700 550
## t2 600 600
## t3 700 400
```

```
# Compute the value of the portfolio
V_P = rowSums(V_assets)
V_P

##   t0   t1   t2   t3
## 1000 1250 1200 1100
```

Step 3: Compute the portfolio returns for $t \in 1, \ldots, 3$.

```
# Compute the portfolio returns
R_t = diff(V_P) / V_P[1:3]
R_t

##       t1       t2       t3
##  0.25000 -0.04000 -0.08333
```

Step 4: Compute the weights of each asset in the portfolio for $t \in 0, \ldots, 3$

```
weights = V_assets / V_P
weights

##         A      B
## t0 0.5000 0.5000
## t1 0.5600 0.4400
## t2 0.5000 0.5000
## t3 0.6364 0.3636
```

We have shown that calculating portfolio weights, values, and returns is simple in a prices and shares framework. However, calculating these metrics becomes more challenging in a weights and returns framework.

3 Example: Weights and Returns Framework

We will use the monthly returns during 1997 of the first 5 assets in the edhec dataset for the following example.

```r
library(PerformanceAnalytics)

## Loading required package:  zoo
##
## Attaching package:  'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
##
## Loading required package:  xts
##
## Attaching package:  'PerformanceAnalytics'
##
## The following object is masked from 'package:graphics':
##
##     legend

data(edhec)
R = edhec["1997", 1:5]
colnames(R) = c("CA", "CTAG", "DS", "EM", "EMN")
R

##                CA    CTAG      DS      EM    EMN
## 1997-01-31 0.0119  0.0393  0.0178  0.0791 0.0189
## 1997-02-28 0.0123  0.0298  0.0122  0.0525 0.0101
## 1997-03-31 0.0078 -0.0021 -0.0012 -0.0120 0.0016
## 1997-04-30 0.0086 -0.0170  0.0030  0.0119 0.0119
## 1997-05-31 0.0156 -0.0015  0.0233  0.0315 0.0189
## 1997-06-30 0.0212  0.0085  0.0217  0.0581 0.0165
## 1997-07-31 0.0193  0.0591  0.0234  0.0560 0.0247
## 1997-08-31 0.0134 -0.0473  0.0147 -0.0066 0.0017
## 1997-09-30 0.0122  0.0198  0.0350  0.0229 0.0202
## 1997-10-31 0.0100 -0.0098 -0.0064 -0.0572 0.0095
## 1997-11-30 0.0000  0.0133  0.0054 -0.0378 0.0041
## 1997-12-31 0.0068  0.0286  0.0073  0.0160 0.0066
```

Suppose that on 1996-12-31 we wish to form an equal weight portfolio such that the weight for asset i is given as:

$$w_i = \frac{1}{N} \quad \text{for } i \in 1, \ldots, N$$

where N is equal to the number of assets.

```r
N = ncol(R)
weights = xts(matrix(rep(1 / N, N), 1), as.Date("1996-12-31"))
colnames(weights) = colnames(R)
weights
```

```
##            CA CTAG  DS  EM EMN
## 1996-12-31 0.2  0.2 0.2 0.2 0.2
```

There are two cases we need to consider when calculating the beginning of period (bop) value.

Case 1: The beginning of period t is a rebalancing event. For example, the rebalance weights at the end of `1996-12-31` take effect at the beginning of `1997-01-31`. This means that the beginning of `1997-01-31` is considered a rebalance event.

The beginning of period value for asset i at time t is given as

$$V_{bop_{t,i}} = w_i * V_{t-1}$$

where w_i is the weight of asset i and V_{t-1} is the end of period (eop) portfolio value of the prior period.

Case 2: The beginning of period t is not a rebalancing event.

$$V_{bop_{t,i}} = V_{eop_{t-1,i}}$$

where $V_{eop_{t-1,i}}$ is the end of period value for asset i from the prior period.

The end of period value for asset i at time t is given as

$$V_{eop_{t,i}} = (1 + R_{t,i}) * V_{bop_{t,i}}$$

Here we demonstrate this and compute values for the periods 1 and 2. For the first period, $t = 1$, we need an initial value for the portfolio value. Let $V_0 = 1$ denote the initial portfolio value. Note that the initial portfolio value can be any arbitrary number. Here we use $V_0 = 1$ for simplicity.

```
V_0 = 1
bop_value = eop_value = matrix(0, 2, ncol(R))
```

Compute the values for $t = 1$.

```
t = 1
bop_value[t,] = coredata(weights) * V_0
eop_value[t,] = coredata(1 + R[t,]) * bop_value[t,]
```

Now compute the values for $t = 2$.

```
t = 2
bop_value[t,] = eop_value[t-1,]
eop_value[t,] = coredata(1 + R[t,]) * bop_value[t,]
```

It is seen that the values for the rest of the time periods can be computed by iterating over $t \in 1, \ldots, T$ where $T = 12$ in this example.

The weight of asset i at time t is calculated as

$$w_{t,i} = \frac{V_{t,i}}{\sum_{i=0}^{N} V_{t,i}}$$

Here we compute both the beginning and end of period weights.

```r
bop_weights = eop_weights = matrix(0, 2, ncol(R))
for(t in 1:2){
  bop_weights[t,] = bop_value[t,] / sum(bop_value[t,])
  eop_weights[t,] = eop_value[t,] / sum(eop_value[t,])
}
bop_weights

##        [,1]   [,2]  [,3]   [,4]   [,5]
## [1,] 0.2000 0.2000 0.200 0.2000 0.2000
## [2,] 0.1958 0.2011 0.197 0.2088 0.1972

eop_weights

##        [,1]   [,2]   [,3]   [,4]   [,5]
## [1,] 0.1958 0.2011 0.1970 0.2088 0.1972
## [2,] 0.1936 0.2023 0.1948 0.2147 0.1946
```

The portfolio returns at time t are calculated as

$$R_{P_t} = \frac{V_t - V_{t-1}}{V_{t-1}}$$

```r
V = c(V_0, rowSums(eop_value))
R_P = diff(V) / V[1:2]
R_P

## [1] 0.03340 0.02376
```

The contribution of asset i at time t is calculated as

$$contribution_{t,i} = \frac{V_{eop_{t,i}} - V_{bop_{t,i}}}{\sum_{i=1}^{N} V_{bop_{t,i}}}$$

```r
contribution = matrix(0, 2, ncol(R))
for(t in 1:2){
  contribution[t,] = (eop_value[t,] - bop_value[t,]) / sum(bop_value[t,])
}
contribution

##          [,1]     [,2]     [,3]    [,4]     [,5]
## [1,] 0.002380 0.007860 0.003560 0.01582 0.003780
## [2,] 0.002409 0.005994 0.002403 0.01096 0.001992
```

Note that contribution can also be calculated as

$$contribution_{t,i} = R_{t,i} * w_{t,i}$$

4 Return.portfolio Examples

```
args(Return.portfolio)

## function (R, weights = NULL, wealth.index = FALSE, contribution = FALSE,
##     geometric = TRUE, rebalance_on = c(NA, "years", "quarters",
##         "months", "weeks", "days"), value = 1, verbose = FALSE,
##     ...)
## NULL
```

If no `weights` are specified, then an equal weight portfolio is computed. If `rebalance_on=NA` then a buy and hold portfolio is assumed. See `?Return.portfolio` for a detailed explanation of the function and arguments.

```
# Equally weighted, buy and hold portfolio returns
Return.portfolio(R)

##            portfolio.returns
## 1997-01-31          0.033400
## 1997-02-28          0.023762
## 1997-03-31         -0.001413
## 1997-04-30          0.003678
## 1997-05-31          0.017767
## 1997-06-30          0.025914
## 1997-07-31          0.036970
## 1997-08-31         -0.005006
## 1997-09-30          0.022081
## 1997-10-31         -0.012352
## 1997-11-30         -0.003844
## 1997-12-31          0.012936

# Equally weighted, rebalanced quarterly portfolio returns
Return.portfolio(R, rebalance_on="quarters")

##            portfolio.returns
## 1997-01-31          0.033400
## 1997-02-28          0.023762
## 1997-03-31         -0.001413
## 1997-04-30          0.003680
## 1997-05-31          0.017661
## 1997-06-30          0.025452
## 1997-07-31          0.036500
## 1997-08-31         -0.005137
## 1997-09-30          0.022049
## 1997-10-31         -0.010780
## 1997-11-30         -0.002621
## 1997-12-31          0.012986

# Equally weighted, rebalanced quarterly portfolio returns.
```

```r
# Use verbose=TRUE to return additional information
# including asset values and weights
Return.portfolio(R, rebalance_on="quarters", verbose=TRUE)
```

```
## $returns
##            portfolio.returns
## 1997-01-31          0.033400
## 1997-02-28          0.023762
## 1997-03-31         -0.001413
## 1997-04-30          0.003680
## 1997-05-31          0.017661
## 1997-06-30          0.025452
## 1997-07-31          0.036500
## 1997-08-31         -0.005137
## 1997-09-30          0.022049
## 1997-10-31         -0.010780
## 1997-11-30         -0.002621
## 1997-12-31          0.012986
##
## $contribution
##                  CA       CTAG         DS        EM       EMN
## 1997-01-31 0.002380  0.0078600  0.0035600  0.015820 0.0037800
## 1997-02-28 0.002409  0.0059940  0.0024032  0.010964 0.0019917
## 1997-03-31 0.001510 -0.0004249 -0.0002337 -0.002576 0.0003113
## 1997-04-30 0.001720 -0.0034000  0.0006000  0.002380 0.0023800
## 1997-05-31 0.003135 -0.0002938  0.0046568  0.006352 0.0038110
## 1997-06-30 0.004252  0.0016336  0.0043611  0.011874 0.0033311
## 1997-07-31 0.003860  0.0118200  0.0046800  0.011200 0.0049400
## 1997-08-31 0.002636 -0.0096663  0.0029028 -0.001345 0.0003361
## 1997-09-30 0.002444  0.0038749  0.0070493  0.004659 0.0040215
## 1997-10-31 0.002000 -0.0019600 -0.0012800 -0.011440 0.0019000
## 1997-11-30 0.000000  0.0026626  0.0010848 -0.007205 0.0008368
## 1997-12-31 0.001392  0.0058171  0.0014783  0.002942 0.0013561
##
## $BOP.Weight
##                CA   CTAG     DS     EM    EMN
## 1997-01-31 0.2000 0.2000 0.2000 0.2000 0.2000
## 1997-02-28 0.1958 0.2011 0.1970 0.2088 0.1972
## 1997-03-31 0.1936 0.2023 0.1948 0.2147 0.1946
## 1997-04-30 0.2000 0.2000 0.2000 0.2000 0.2000
## 1997-05-31 0.2010 0.1959 0.1999 0.2016 0.2016
## 1997-06-30 0.2006 0.1922 0.2010 0.2044 0.2019
## 1997-07-31 0.2000 0.2000 0.2000 0.2000 0.2000
## 1997-08-31 0.1967 0.2044 0.1975 0.2038 0.1977
## 1997-09-30 0.2003 0.1957 0.2014 0.2035 0.1991
## 1997-10-31 0.2000 0.2000 0.2000 0.2000 0.2000
## 1997-11-30 0.2042 0.2002 0.2009 0.1906 0.2041
## 1997-12-31 0.2047 0.2034 0.2025 0.1839 0.2055
```

```
##
## $EOP.Weight
##                CA   CTAG     DS     EM    EMN
## 1997-01-31 0.1958 0.2011 0.1970 0.2088 0.1972
## 1997-02-28 0.1936 0.2023 0.1948 0.2147 0.1946
## 1997-03-31 0.1954 0.2022 0.1948 0.2124 0.1951
## 1997-04-30 0.2010 0.1959 0.1999 0.2016 0.2016
## 1997-05-31 0.2006 0.1922 0.2010 0.2044 0.2019
## 1997-06-30 0.1997 0.1890 0.2002 0.2109 0.2001
## 1997-07-31 0.1967 0.2044 0.1975 0.2038 0.1977
## 1997-08-31 0.2003 0.1957 0.2014 0.2035 0.1991
## 1997-09-30 0.1984 0.1953 0.2040 0.2036 0.1987
## 1997-10-31 0.2042 0.2002 0.2009 0.1906 0.2041
## 1997-11-30 0.2047 0.2034 0.2025 0.1839 0.2055
## 1997-12-31 0.2035 0.2065 0.2014 0.1844 0.2042
##
## $BOP.Value
##                CA   CTAG     DS     EM    EMN
## 1997-01-31 0.2000 0.2000 0.2000 0.2000 0.2000
## 1997-02-28 0.2024 0.2079 0.2036 0.2158 0.2038
## 1997-03-31 0.2049 0.2141 0.2060 0.2272 0.2058
## 1997-04-30 0.2113 0.2113 0.2113 0.2113 0.2113
## 1997-05-31 0.2131 0.2077 0.2119 0.2138 0.2138
## 1997-06-30 0.2164 0.2074 0.2169 0.2205 0.2178
## 1997-07-31 0.2213 0.2213 0.2213 0.2213 0.2213
## 1997-08-31 0.2256 0.2344 0.2265 0.2337 0.2268
## 1997-09-30 0.2286 0.2233 0.2298 0.2322 0.2272
## 1997-10-31 0.2332 0.2332 0.2332 0.2332 0.2332
## 1997-11-30 0.2356 0.2310 0.2317 0.2199 0.2355
## 1997-12-31 0.2356 0.2340 0.2330 0.2116 0.2364
##
## $EOP.Value
##                CA   CTAG     DS     EM    EMN
## 1997-01-31 0.2024 0.2079 0.2036 0.2158 0.2038
## 1997-02-28 0.2049 0.2141 0.2060 0.2272 0.2058
## 1997-03-31 0.2065 0.2136 0.2058 0.2244 0.2062
## 1997-04-30 0.2131 0.2077 0.2119 0.2138 0.2138
## 1997-05-31 0.2164 0.2074 0.2169 0.2205 0.2178
## 1997-06-30 0.2210 0.2092 0.2216 0.2334 0.2214
## 1997-07-31 0.2256 0.2344 0.2265 0.2337 0.2268
## 1997-08-31 0.2286 0.2233 0.2298 0.2322 0.2272
## 1997-09-30 0.2314 0.2277 0.2379 0.2375 0.2317
## 1997-10-31 0.2356 0.2310 0.2317 0.2199 0.2355
## 1997-11-30 0.2356 0.2340 0.2330 0.2116 0.2364
## 1997-12-31 0.2372 0.2407 0.2347 0.2150 0.2380
```