Portfolio specification, constraints, and objectives

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Ross Bennett
Instructor

datacamp

Workflow overview

General portfolio optimization problem workflow in `PortfolioAnalytics`:

- Portfolio specification
- Add constraints and objectives
- Run optimization
- Analyze optimization results

Workflow: portfolio specification

`portfolio.spec(assets = NULL, ...)`

```
# Character vector of assets
portfolio.spec(assets = c("SP00", "DJIA", "Nasdaq", "FTSE100", "DAX", "CAC40"))

# Named vector of assets with initial weights
initial_weights <- c("SP500" = 0.5, "FTSE100" = 0.3, "NIKKEI" = 0.2)
portfolio.spec(assets = initial_weights)

# Scalar of number of assets
portfolio.spec(assets = 4)
```

```
add.constraint(portfolio,
               type = c("weight_sum", "box", "full_investment",...),
               ...)
```

```
# Initialize portfolio specification
p <- portfolio.spec(assets = 4)

# Add full investment constraint
p <- add.constraint(portfolio = p, type = "weight_sum",
                    min_sum = 1, max_sum = 1)

# Add box constraint
p <- add.constraint(portfolio = p, type = "box",
                    min = 0.2, max = 0.6)
```

```
add.objective(portfolio,
              type = c("return", "risk", ...),
              name,
              arguments = NULL,
              ... )
```

```
# Initialize portfolio specification
p <- portfolio.spec(assets = 4)
# Add mean return objective
p <- add.objective(portfolio = p, type = "return",name = "mean")

# Add expected shortfall risk objective
p <- add.objective(portfolio = p, type = "risk", name = "ES",
                   arguments = list(p= 0.9, method = "gaussian")
```

Let's practice!

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Running optimizations

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Ross Bennett
Instructor

datacamp

Single period optimization

- Single period optimization with `optimize.portfolio()`
- Optimization with periodic rebalancing (backtesting) with `optimize.portfolio.rebalancing()`

Single period optimization

```
optimize.portfolio(R, portfolio = NULL,
                   optimize_method = c("DEoptim", "random", "ROI", ...),
                   search_size = 20000, trace = TRUE,
                   momentFUN = "set.portfolio.moments",
                   ...)
```

```
optimize.portfolio.rebalancing(R, portfolio = NULL,
                               optimize_method = c("DEoptim", "random", "ROI", ...),
                               search_size = 20000, trace = TRUE,
                               rebalance_on = "quarters",
                               training_period,
                               rolling_window,
                               momentFUN = "set.portfolio.moments",
                               ...)
```

Optimization methods

The following optimization methods are supported:

Global Solvers:

- *DEoptim*: Differential Evolution Optimization
- *random*: Random Portfolios Optimization
- *GenSA*: Generalized Simulated AnnealingAnalyze optimization results
- *pso*: Particle Swarm Optimization

LP and QP Solvers:

- *ROI*: R Optimization Infrastructure for linear and quadratic programming solvers

```
data(edhec)
ret <- edhec[,1:6]
# Portfolio
p <- portfolio.spec(assets = colnames(ret))
p <- add.constraint(portfolio = p, type = "full_investment")
p <- add.constraint(portfolio = p, type = "long_only")
p <- add.objective(portfolio = p, type = "risk", name = "StdDev")

# Optimizations
opt_single <- optimize.portfolio(R = ret, portfolio = p, optimize_method = "ROI")

opt_rebal <- optimize.portfolio.rebalancing(R = ret, portfolio = p,
                                            optimize_method = "ROI",
                                            rebalance_on = "years",
                                            training_period = 60,
                                            rolling_window = 60)
```

Let's practice!

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Analyzing optimization results

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Ross Bennett
Instructor

datacamp

Workflow: analyze results

Visualization	Data Extraction
`plot()`	`extractObjectiveMeasures()`
`chart.Concentration()`	`extractStats()`
`chart.EfficientFrontier()`	`extractWeights()`
`chart.RiskReward()`	`print()`
`chart.RiskBudget()`	`summary()`
`chart.Weights()`	

Example: extract weights

```
# Extract the optimal weights
extractWeights(opt)
```

```
Convertible Arbitrage   CTA Global      Distressed Securities
         0.000000e+00   6.515184e-02          5.840055e-18

Emerging Markets     Equity Market Neutral       Event Driven
  -8.501425e-18            9.348482e-01          4.105887e-18
```

Example: extract weights

```
head(extractWeights(opt_rebal), n = 3)
```

```
           Convertible Arbitrage   CTA Global   Distressed Securities
2001-12-31           0.12986589   0.06849445              0.00000000
2002-12-31           0.08738164   0.08645814              0.00000000
2003-12-31           0.09177469   0.03192720              0.02419038?
           Emerging Markets  Equity Market Neutral  Event Driven
2001-12-31     7.113112e-18              0.8016397  -1.608927e-16
2002-12-31    -2.553006e-19              0.8261602  -3.837233e-17
2003-12-31     0.000000e+00              0.8521077   2.991493e-19
```

Example: extract weights

```
# Extract the optimal weights
extractWeights(opt)
```

```
Convertible Arbitrage   CTA Global      Distressed Securities
         0.000000e+00   6.515184e-02          5.840055e-18

Emerging Markets     Equity Market Neutral       Event Driven
  -8.501425e-18            9.348482e-01          4.105887e-18
```

Example: extract weights

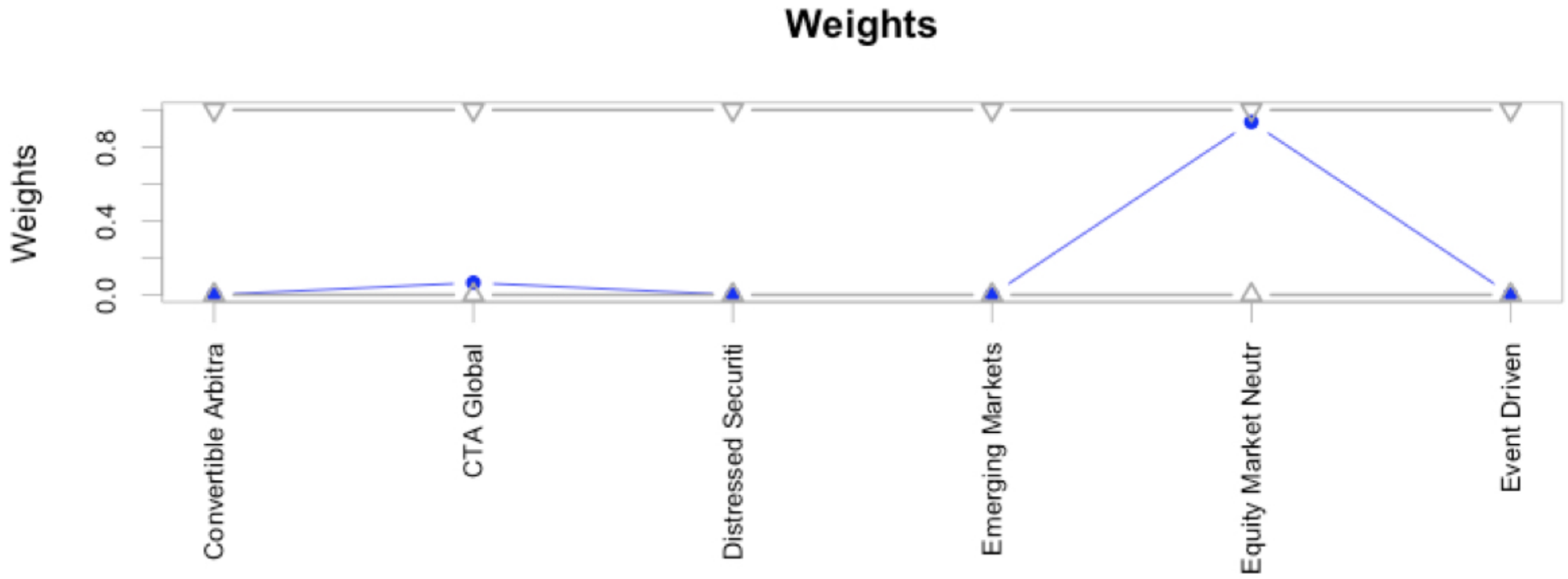
```
head(extractWeights(opt_rebal), n = 3)
```

```
           Convertible Arbitrage CTA Global
```

```
            Convertible Arbitrage   CTA Global   Distressed Securities
2001-12-31             0.12986589   0.06849445             0.00000000
2002-12-31             0.08738164   0.08645814             0.00000000
2003-12-31             0.09177469   0.03192720             0.02419038?
            Emerging Markets  Equity Market Neutral  Event Driven
2001-12-31     7.113112e-18              0.8016397  -1.608927e-16
2002-12-31    -2.553006e-19              0.8261602  -3.837233e-17
2003-12-31     0.000000e+00              0.8521077   2.991493e-19
```

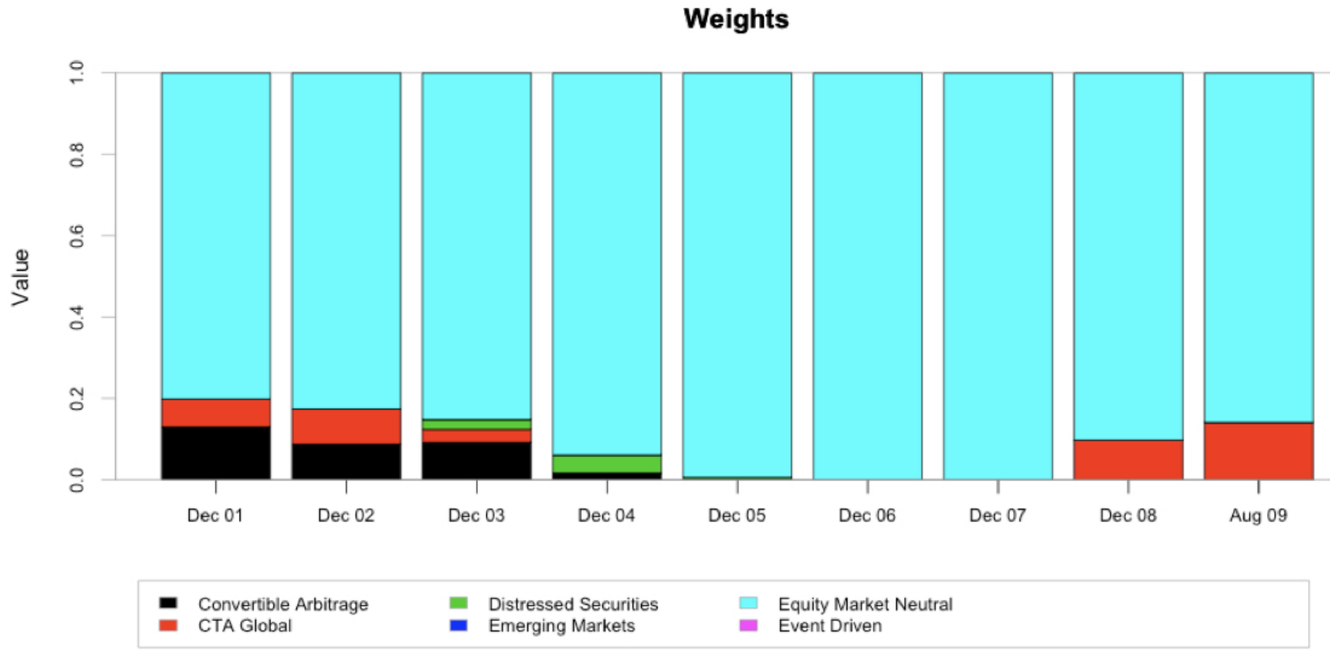
Example: chart weights

```
# Chart the weights
chart.Weights(opt)
chart.Weights(opt_rebal)
```



Example: chart weights

```
# Chart the weights
chart.Weights(opt)
chart.Weights(opt_rebal)
```



```
# Extract the objective measures
extractObjectiveMeasures(opt)
```

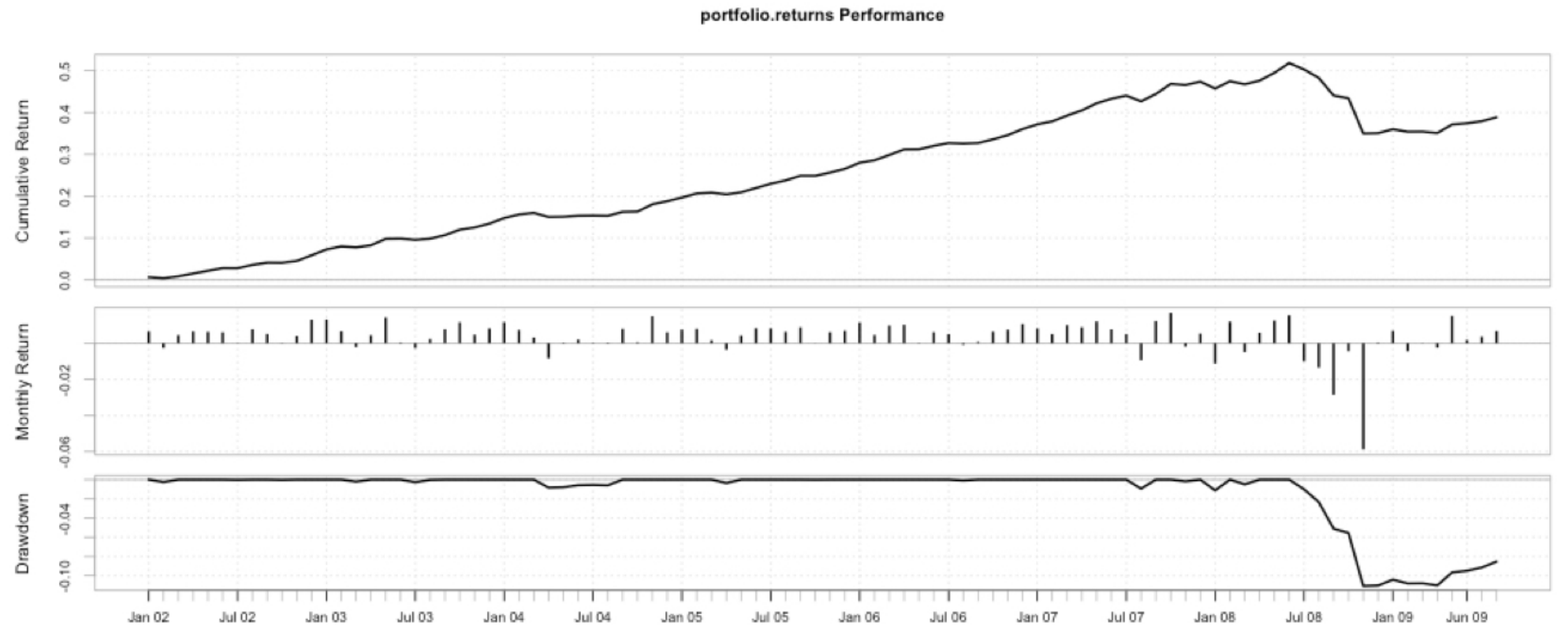
```
$StdDev
     StdDev
0.008855401
```

```
head(extractObjectiveMeasures(opt_rebal))
```

```
                StdDev
2001-12-31 0.006521328
2002-12-31 0.005886103
2003-12-31 0.005656744
2004-12-31 0.005855993
2005-12-31 0.004308911
2006-12-31 0.004198900
```

Example: optimization analysis

```
# Compute the rebalancing returns
rr <- Return.portfolio(ret, weights = extractWeights(opt_rebal))
charts.PerformanceSummary(rr)
```



Let's practice!

INTERMEDIATE PORTFOLIO ANALYSIS IN R