Advanced Moment Estimation & Optimization with PortfolioAnalytics

R/Finance 2015

Ross Bennett

Overview

- Discuss Portfolio Optimization
- Introduce PortfolioAnalytics
- Demonstrate PortfolioAnalytics with Examples

Modern Portfolio Theory

"Modern" Portfolio Theory (MPT) was introduced by Harry Markowitz in 1952.

In general, MPT states that an investor's objective is to maximize portfolio expected return for a given amount of risk.

General Objectives

- Maximize a measure of gain per unit measure of risk
- Minimize a measure of risk

How do we define risk? What about more complex objectives and constraints?

Portfolio Optimization Objectives

- Minimize Risk
  - Volatility
  - Tail Loss (VaR, ES)
  - Other Downside Risk Measure
- Maximize Risk Adjusted Return
  - Sharpe Ratio, Modified Sharpe Ratio
  - Several Others
- Risk Budgets
  - Equal Component Contribution to Risk (i.e. Risk Parity)
  - Limits on Component Contribution
- Maximize a Utility Function
  - Quadratic, CRRA, etc.

PortfolioAnalytics Overview

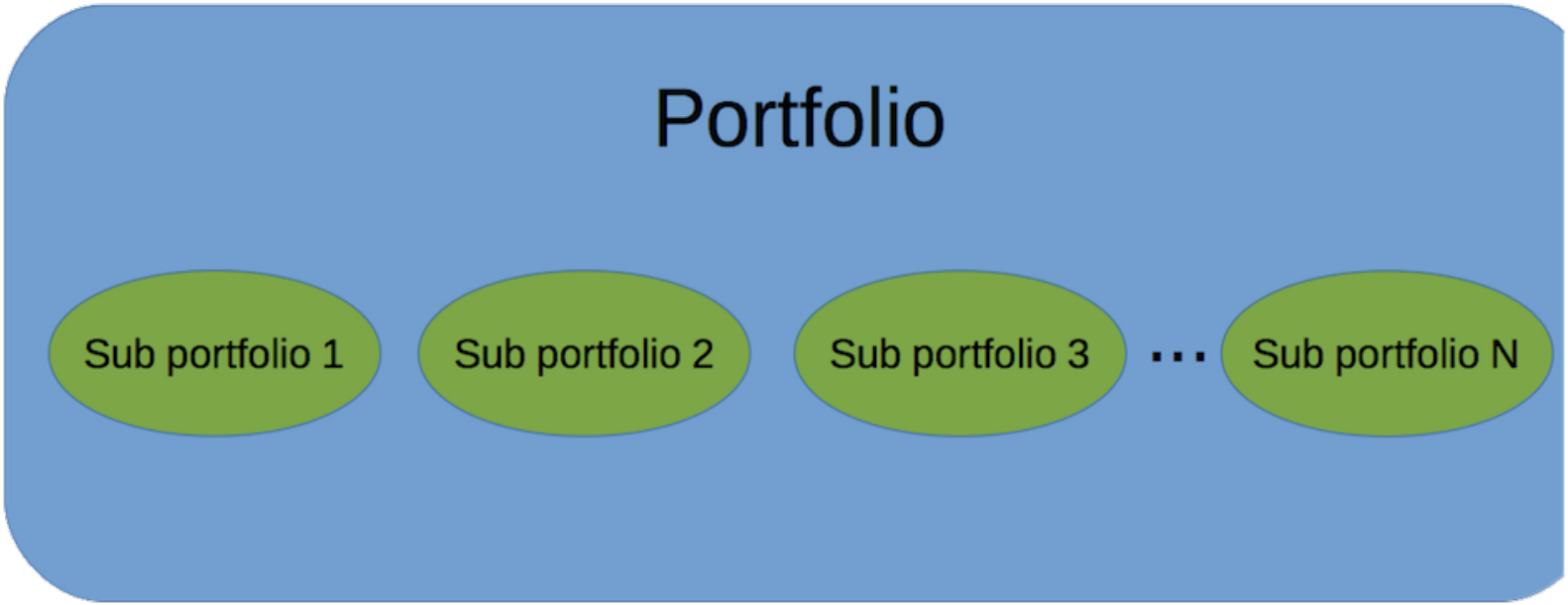
PortfolioAnalytics is an R package designed to provide numerical solutions and visualizations for portfolio optimization problems with complex constraints and objectives.

- Support for multiple constraint and objective types
- Modular constraints and objectives
- An objective function can be any valid R function
- Support for user defined moment functions
- Visualizations
- Solver agnostic
- Support for parallel computing

New in PortfolioAnalytics

- Pushed to CRAN
- Regime Switching Framework
- Multilayer Optimization
- Rank Based Optimization
- Factor Model Moment Estimates
- Improved Random Portfolios Algorithm
- More demos, vignettes, and documentation

Multilayer Optimization



Support Multiple Solvers

Linear and Quadratic Programming Solvers

- R Optimization Infrastructure (ROI)
  - GLPK (Rglpk)
  - Symphony (Rsymphony)
  - Quadprog (quadprog)

Global (stochastic or continuous solvers)

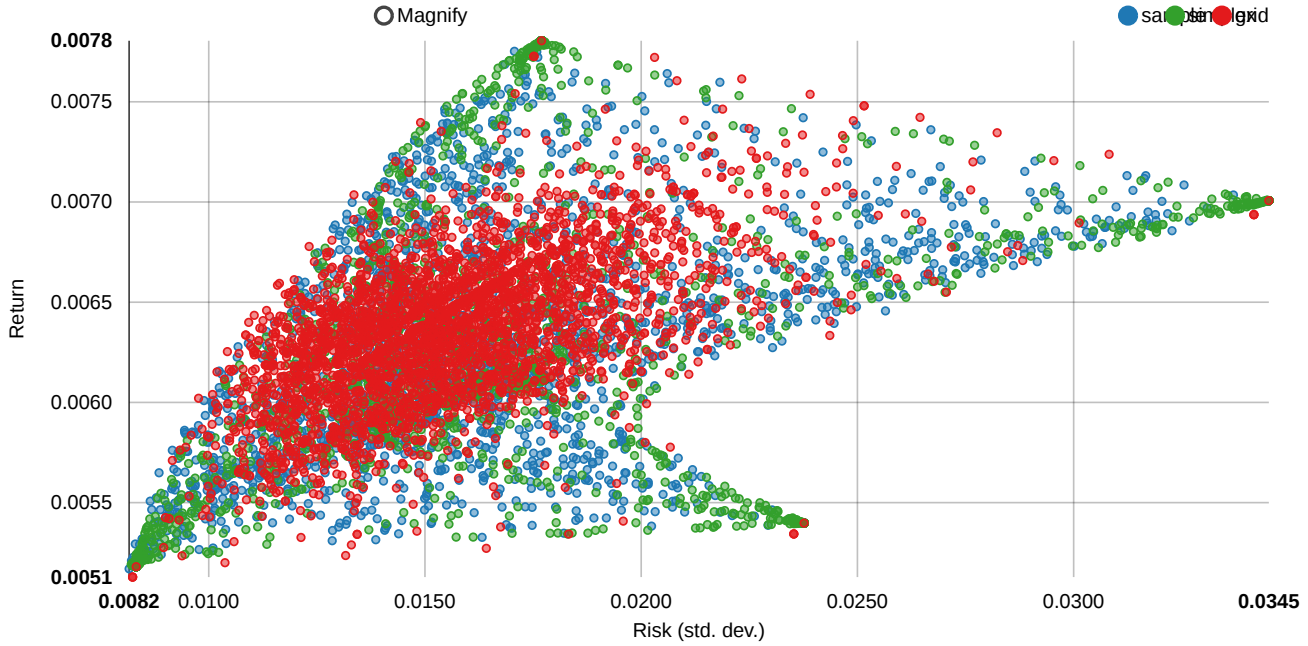
- Random Portfolios
- Differential Evolution (DEoptim)
- Particle Swarm Optimization (pso)
- Generalized Simulated Annealing (GenSA)

Random Portfolios

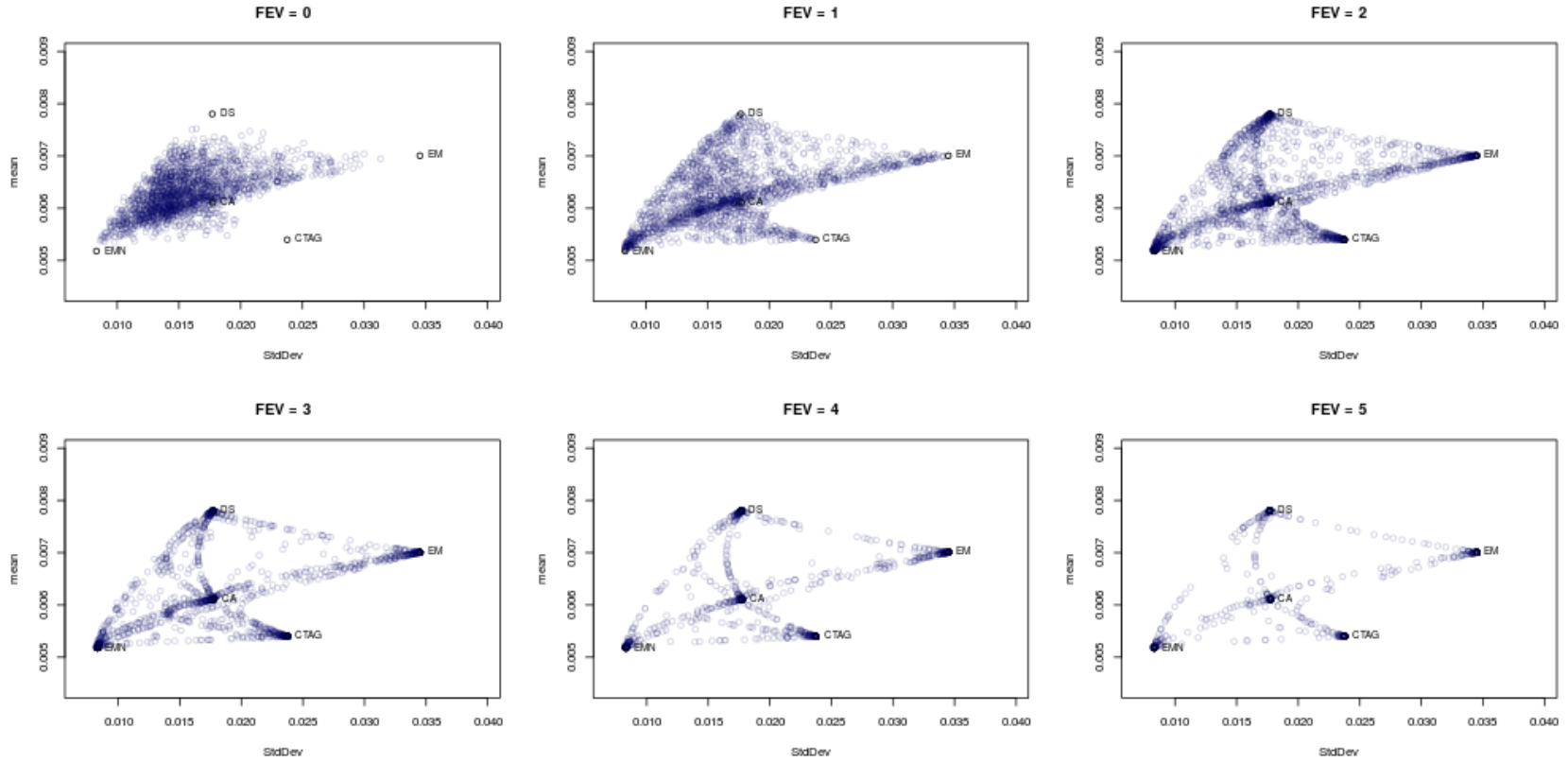
PortfolioAnalytics has three methods to generate random portfolios.

1. The sample method to generate random portfolios is based on an idea by Pat Burns.
2. The simplex method to generate random portfolios is based on a paper by W. T. Shaw.
3. The grid method to generate random portfolios is based on the `gridSearch` function in the NMOF package.

Comparison of Random Portfolio Methods (Interactive!)



Random Portfolios: Simplex Method



Workflow: Specify Portfolio

```
args(portfolio.spec)
```

```
## function (assets = NULL, category_labels = NULL, weight_seq = NULL,
##     message = FALSE)
## NULL
```

Initializes the portfolio object that holds portfolio level data, constraints, and objectives

Workflow: Add Constraints

```
args(add.constraint)
```

```
## function (portfolio, type, enabled = TRUE, message = FALSE, ...,
##     indexnum = NULL)
## NULL
```

Supported Constraint Types

- Sum of Weights
- Box
- Group
- Factor Exposure
- Position Limit
- and many more

Workflow: Add Objectives

```
args(add.objective)
```

```
## function (portfolio, constraints = NULL, type, name, arguments = NULL,
##     enabled = TRUE, ..., indexnum = NULL)
## NULL
```

Supported Objective types

- Return
- Risk
- Risk Budget
- Weight Concentration

Workflow: Run Optimization

```
args(optimize.portfolio)
```

```
## function (R, portfolio = NULL, constraints = NULL, objectives = NULL,
##     optimize_method = c("DEoptim", "random", "ROI", "pso", "GenSA"),
##     search_size = 20000, trace = FALSE, ..., rp = NULL, momentFUN = "set.portfolio.moments
##     message = FALSE)
## NULL
```

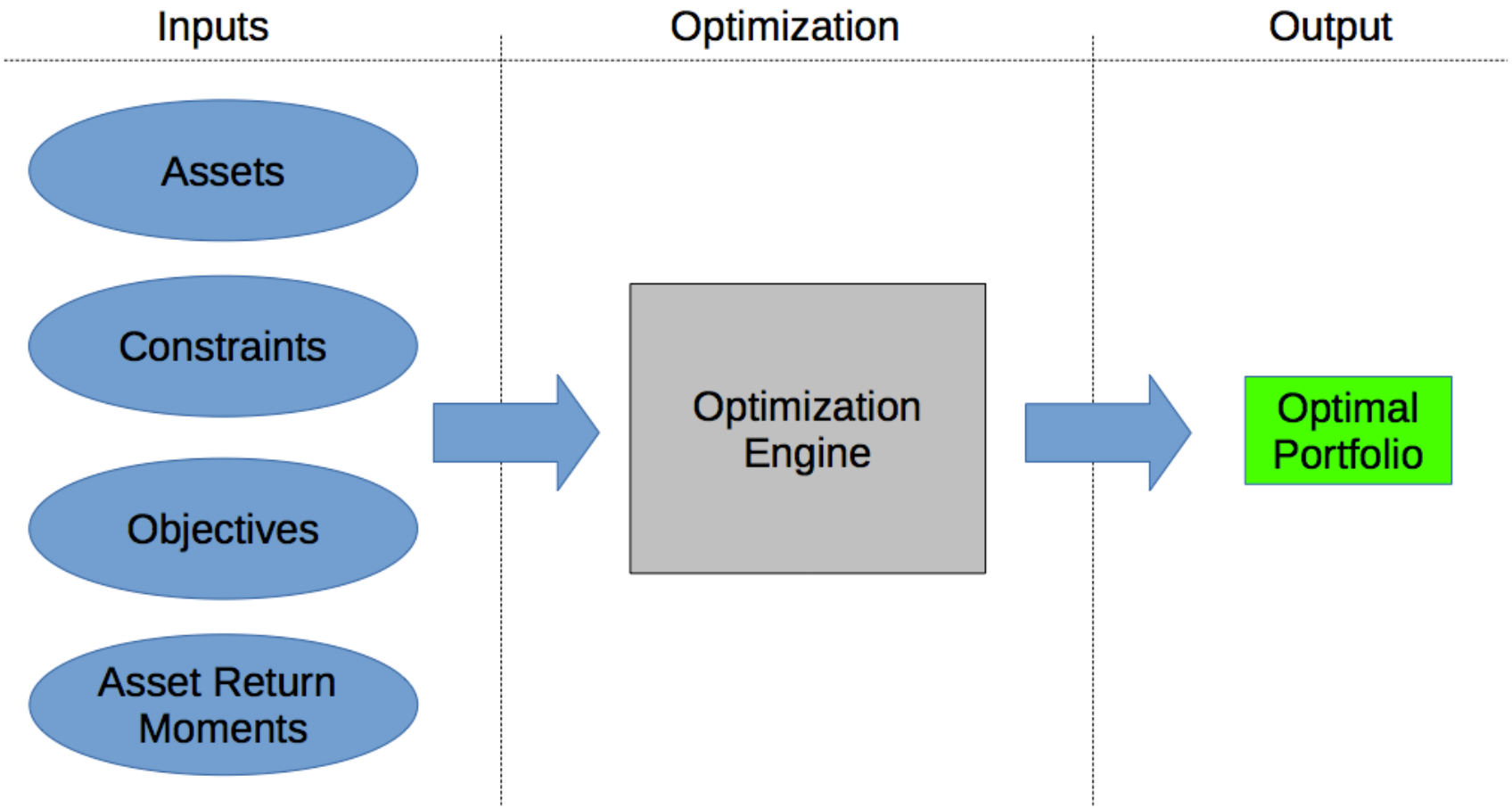
```
args(optimize.portfolio.rebalancing)
```

```
## function (R, portfolio = NULL, constraints = NULL, objectives = NULL,
##     optimize_method = c("DEoptim", "random", "ROI"), search_size = 20000,
##     trace = FALSE, ..., rp = NULL, rebalance_on = NULL, training_period = NULL,
##     rolling_window = NULL)
## NULL
```

Workflow: Analyze Results

VISUALIZATION	DATA EXTRACTION
plot	extractObjectiveMeasures
chart.Concentration	extractStats
chart.EfficientFrontier	extractWeights
chart.RiskReward	print
chart.RiskBudget	summary
chart.Weights	

Portfolio Optimization



Estimating Moments

Ledoit and Wolf (2003):

"The central message of this paper is that nobody should be using the sample covariance matrix for the purpose of portfolio optimization."

- Sample
- Shrinkage Estimators
- Factor Model
- Expressing Views

Meucci Fully Flexible Views

- Reference Model

$$X \sim f_X$$

- Views

$$V \equiv g(X) \sim f_V$$

- Express View on Ranking

$$m\{V_1\} \geq m\{V_2\} \geq \ldots \geq m\{V_K\}$$

- Posterior

$$\tilde{f}_x \equiv \underset{f \in V}{\operatorname{argmin}}\, entropy(f, f_x)$$

Almgren-Chriss Portfolios from Sorts

- Define $S_1, S_2, \ldots, S_n$ as the investment universe of n assets
- Defining Sorts
  - Single complete sort

$$r_1 \geq r_2 \geq \ldots \geq r_n$$

  - Sector based sort
  - Deciles and other divisions
  - Single complete sort with longs and shorts
  - others
- Centroid vector, c, is defined as the center of mass of the set Q.
  - where Q is the space of consistent expected returns

Example 1 and 2

Consider an allocation to hedge funds using the EDHEC-Risk Alternative Index as a proxy.

- Simple example of expressing a view on the order of the expected returns of assets using both Meucci's Fully Flexible Views and the Almgren-Chriss framework
- Establish a quantitative rule for our view on the order of the expected returns of assets and test on historical data.

Example 1: Data and Portfolio

```
# Load package and data
library(PortfolioAnalytics)
source("data_prep.R")
R <- edhec[,1:4]
funds <- colnames(R)

# Construct portfolio
meanSD.portf <- portfolio.spec(assets=funds)
meanSD.portf <- add.constraint(portfolio=meanSD.portf, type="weight_sum",
                               min_sum=0.99, max_sum=1.01)
meanSD.portf <- add.constraint(portfolio=meanSD.portf, type="box",
                               min=0.05, max=0.5)
meanSD.portf <- add.objective(portfolio=meanSD.portf, type="risk", name="StdDev")
meanSD.portf <- add.objective(portfolio=meanSD.portf, type="return", name="mean")

# Generate random portfolios for use in the optimization.
rp <- random_portfolios(meanSD.portf, 5000)
```

Example 1: Express Views

```
# Here we express views on the relative rank of the asset returns
# E{ R[,2] < R[,3] < R[,1] < R[,4] }
asset.rank <- c(2, 3, 1, 4)
```

Meucci: Fully Flexible Views Framework

```
p <- rep(1 / nrow(R), nrow(R))
m.moments <- meucci.ranking(R, p, asset.rank)
```

Almgren and Chriss: Portfolios from Sorts

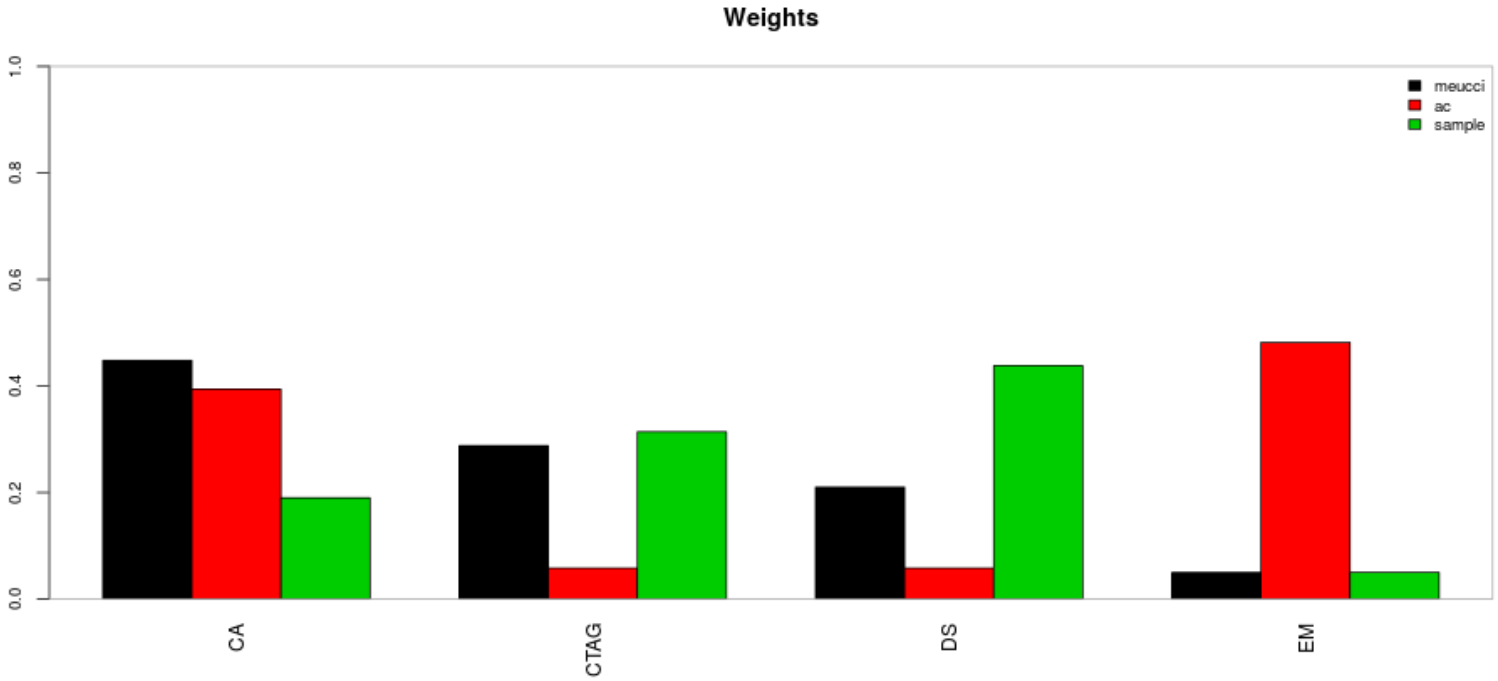
```
ac.moments <- list()
ac.moments$mu <- ac.ranking(R, asset.rank)
# Sample estimate for second moment
ac.moments$sigma <- cov(R)
```

Example 1: Optimization

```
# Use moments output from meucci.ranking
opt.meucci <- optimize.portfolio(R, portfolio=meanSD.portf,
                                 optimize_method="random",
                                 rp=rp,
                                 trace=TRUE,
                                 momentargs=m.moments)

# Use first moment from ac.ranking. Note second moment is sample covariance
opt.ac <- optimize.portfolio(R, portfolio=meanSD.portf,
                             optimize_method="random",
                             rp=rp,
                             trace=TRUE,
                             momentargs=ac.moments)
```

Example 1: Optimization Results Optimal Weights



Example 2: Custom Moment Function

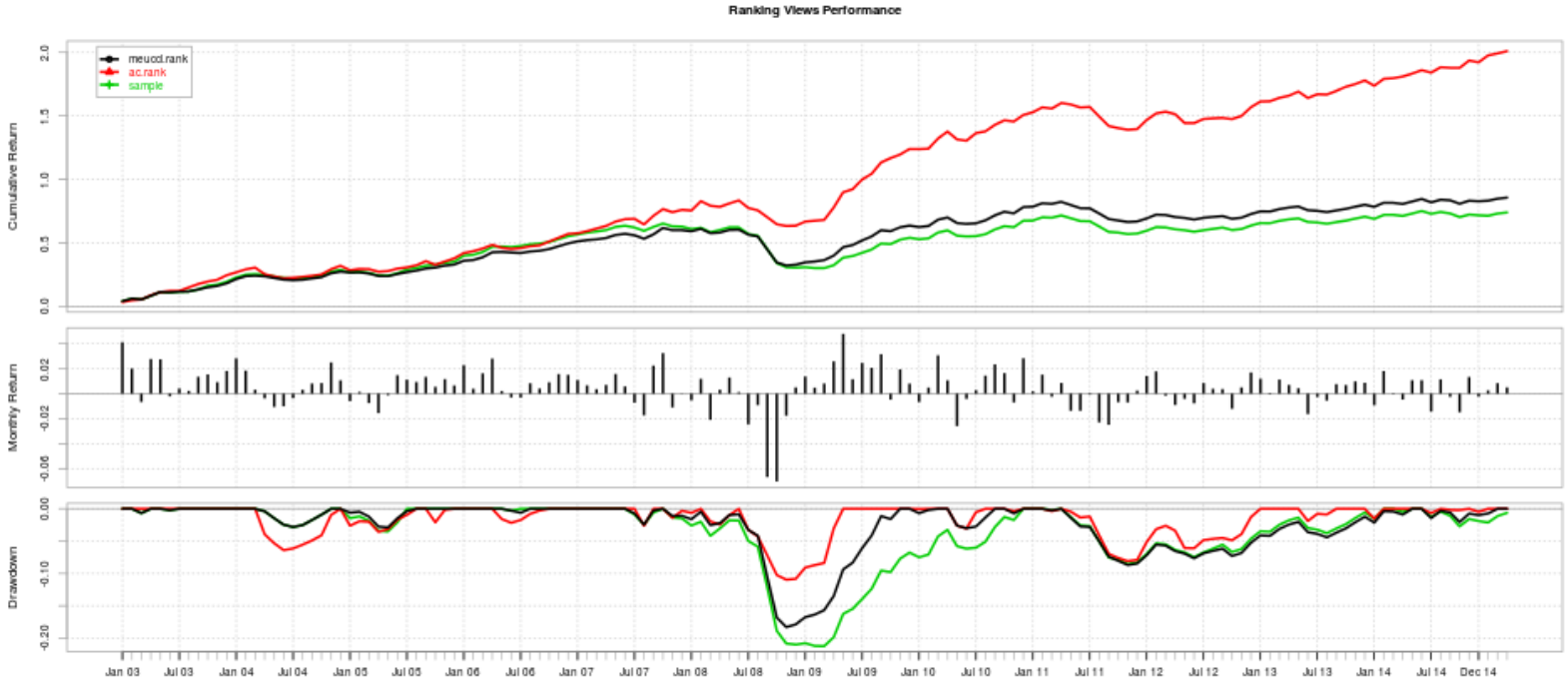
```
moment.ranking <- function(R, n=1, method=c("meucci", "ac")){
  method <- match.arg(method)
  tmpR <- apply(tail(R, n), 2, function(x) prod(1 + x) - 1)
  # Assume that the assets with the highest return will continue to outperform
  asset.rank <- order(tmpR)
  switch(method,
         meucci = {
           p <- rep(1 / nrow(R), nrow(R))
           moments <- meucci.ranking(R, p, asset.rank)
         },
         ac = {
           moments <- list()
           moments$mu <- ac.ranking(R, asset.rank)
           moments$sigma <- cov(R)
         })
  moments
}
```

Example 2: Optimization with Periodic Rebalancing

```
opt.bt.meucci <- optimize.portfolio.rebalancing(R, portfolio=meanSD.portf,
                                                optimize_method="random", rp=rp,
                                                rebalance_on="quarters",
                                                training_period=72,
                                                momentFUN="moment.ranking",
                                                n=3, method="meucci")

opt.bt.ac <- optimize.portfolio.rebalancing(R, portfolio=meanSD.portf,
                                            optimize_method="random", rp=rp,
                                            rebalance_on="quarters",
                                            training_period=72,
                                            momentFUN="moment.ranking",
                                            n=3, method="ac")
```

Example 2: Optimization Results



Example 3: Minimum Expected Shortfall with Risk Budget Limit

Consider an allocation to equity sectors using 9 sector ETFs.

- Raw returns data and sample moment estimates
- Cleaned returns data and factor model moment estimates

Example 3: Data and Portfolio

```
# Data
R.raw <- ret.sector
R <- Return.clean(R.raw, "boudt")
funds <- colnames(R)

# Construct initial portfolio with basic constraints.
ES.portf <- portfolio.spec(assets=funds)
ES.portf <- add.constraint(portfolio=ES.portf, type="weight_sum",
                           min_sum=0.99, max_sum=1.01)
ES.portf <- add.constraint(portfolio=ES.portf, type="long_only")
ES.portf <- add.objective(portfolio=ES.portf, type="risk", name="ES",
                          arguments=list(p=0.95))
ES.portf <- add.objective(portfolio=ES.portf, type="risk_budget",
                          name="ES", max_prisk=0.25,
                          arguments=list(p=0.95))
# Generate random portfolios
rp <- random_portfolios(ES.portf, 5000)
```

Example 3: Fit Statistical Factor Model

```
# This is not necessary for the optimization, but demonstrates how to extract
# the moments for use in custom objective function
fit <- statistical.factor.model(R, k=3)

# Extract the moments
sigma <- extractCovariance(fit)
m3 <- extractCoskewness(fit)
m4 <- extractCokurtosis(fit)
```

Example 3: Custom Moment Function

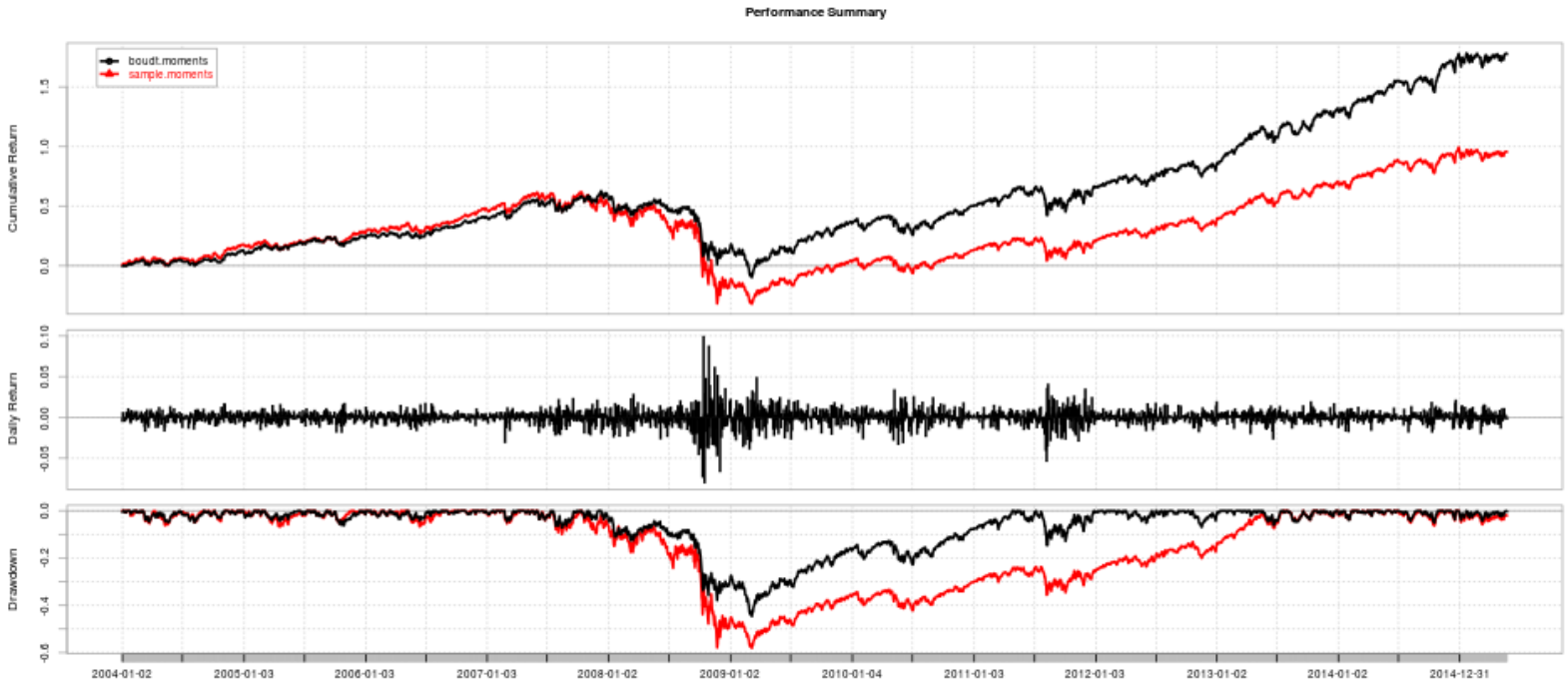
```
fm.moments <- function(R, k=1){
  fit <- statistical.factor.model(R=R, k=k)
  momentargs <- list()
  momentargs$mu <- matrix(rep(0, ncol(R)),ncol=1)
  momentargs$sigma <- extractCovariance(fit)
  momentargs$m3 <- extractCoskewness(fit)
  momentargs$m4 <- extractCokurtosis(fit)
  return(momentargs)
}
```

Example 3: Optimization with Periodic Rebalancing

```
# Higher moment estimates from statistical factor model
minES.boudt <- optimize.portfolio.rebalancing(R=R, portfolio=ES.portf,
                                              momentFUN=fm.moments, k=3,
                                              optimize_method="random", rp=rp,
                                              rebalance_on="quarters",
                                              training_period=1250,
                                              trace=TRUE)

# Sample estimates of the moments
minES.sample <- optimize.portfolio.rebalancing(R=R.raw, portfolio=ES.portf,
                                               optimize_method="random", rp=rp,
                                               rebalance_on="quarters",
                                               training_period=1250,
                                               trace=TRUE)
```

Example 3: Optimization Results



Conclusion

- Introduced the goals and summary of PortfolioAnalytics
- Demonstrated the flexibility through examples
- Plans for continued development
  - Interface to *parma*
  - Additional solvers
  - "Gallery" of examples

Acknowledgements

Many thanks to...

- Google: funding Google Summer of Code (GSoC) for 2013 and 2014
- UW CF&RM Program: continued work on PortfolioAnalytics
- GSoC Mentors: Brian Peterson, Peter Carl, Doug Martin, and Guy Yollin
- R/Finance Committee

PortfolioAnalytics Links

- PortfolioAnalytics on CRAN
- PortfolioAnalytics on R-Forge

Source code for the slides

- https://github.com/rossb34/PortfolioAnalyticsPresentation2015

and view it here

- http://rossb34.github.io/PortfolioAnalyticsPresentation2015/

Any Questions?

References and Useful Links

- ROI
- DEoptim
- pso
- GenSA
- PerformanceAnalytics
- Patrick Burns Random Portfolios
- W.T. Shaw Random Portfolios
- Improved Forecasts of Higher-Order Co-moments and Implications for Portfolio Selection
- Higher Order Comoments of Multifactor Models and Asset Allocation
- Shiny App