Welcome to the course!

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Ross Bennett
Instructor

What you will learn

- Build on fundamental concepts from "Introduction to Portfolio Analysis in R"
- Explore advanced concepts in the portfolio optimization process
- Use the R package `PortfolioAnalytics` to solve portfolio optimization problems that mirror real world problems

Modern Portfolio Theory

- Modern Portfolio Theory (MPT) was introduced by Harry Markowitz in 1952.
- MPT states that an investor's objective is to maximize portfolio expected return for a given amount of risk.
- Common Objectives:
  - Maximize a measure of gain per unit measure of risk
  - Minimize a measure of risk

Mean - Standard Deviation Example: Setup

```
library(PortfolioAnalytics)
data(edhec)
data <- edhec[,1:8]
```

```
# Create the portfolio specification
port_spec <- portfolio.spec(colnames(data))
port_spec <- add.constraint(portfolio = port_spec, type = "full_investment")
port_spec <- add.constraint(portfolio = port_spec, type = "long_only")
port_spec <- add.objective(portfolio = port_spec, type = "return", name = "mean")
port_spec <- add.objective(portfolio = port_spec, type = "risk", name = "StdDev")
```

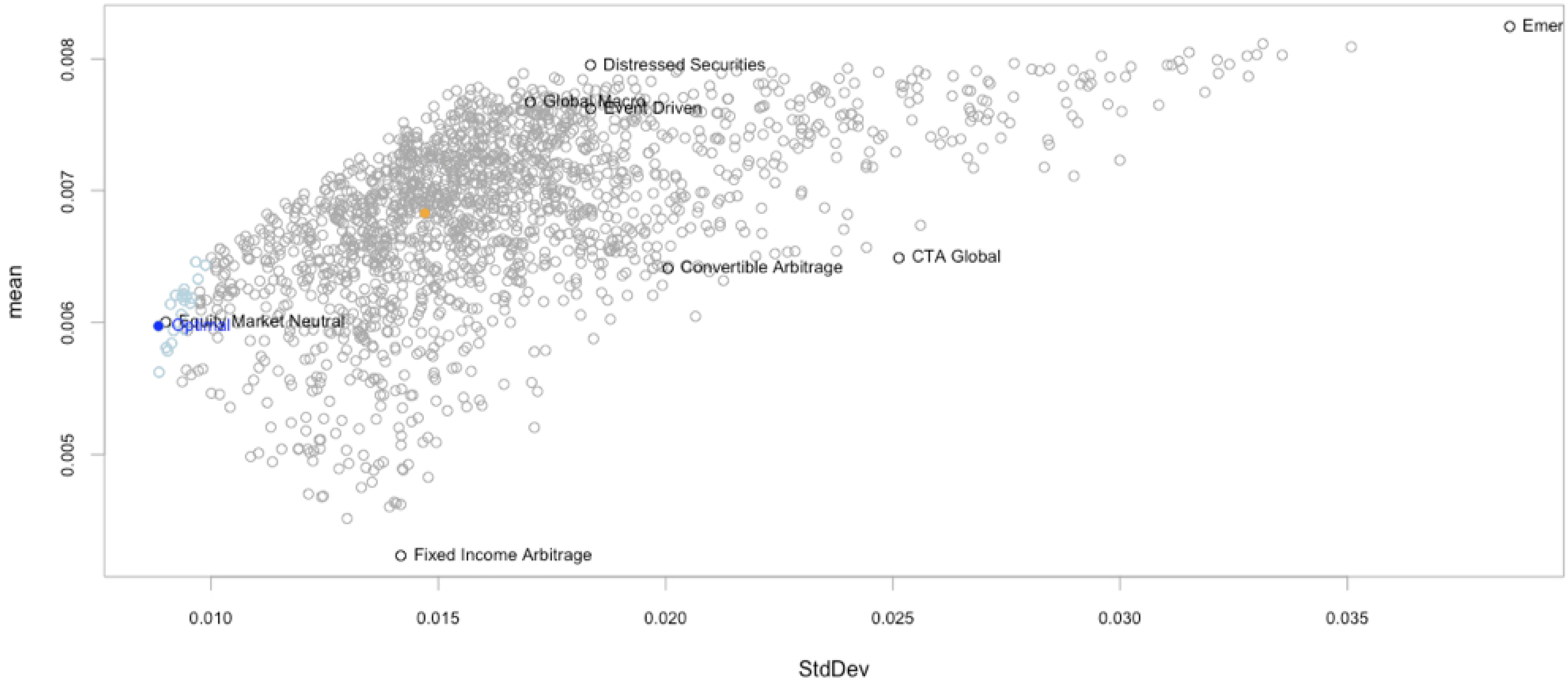
```
**************************************************
PortfolioAnalytics Portfolio Specification
**************************************************
Call:
portfolio.spec(assets = colnames(data))

Number of assets: 8
Asset Names
[1] "Convertible Arbitrage"  "CTA Global"             "Distressed Securities"
[4] "Emerging Markets"       "Equity Market Neutral"  "Event Driven"
[7] "Fixed Income Arbitrage" "Global Macro"

Constraints
Enabled constraint types
        - full_investment
        - long_only

Objectives:
Enabled objective names
        - mean
        - StdDev
```

Mean - Standard Deviation Example: Optimize

```r
# Run optimization and chart results in risk-reward space
opt <- optimize.portfolio(data,
                          portfolio = port_spec,
                          optimize_method = "random",
                          trace = TRUE)
```

```r
chart.RiskReward(opt,
                 risk.col = "StdDev",
                 return.col = "mean",
                 chart.assets = TRUE)
```

Mean - Standard Deviation Example: Optimize



Let's practice!

Challenges of portfolio optimization

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Ross Bennett
Instructor

Challenges

- Many solvers are not specific to portfolio optimization
- Understanding the capabilities and limits of solvers to select the appropriate solver for the problem or formulate the problem to fit the solver
- Difficult to switch between solvers
- Closed-Form solver (e.g., quadratic programming)
- Global solver (e.g., differential evolution optimization)

Quadratic utility

- Maximize: $\omega^T * \mu - \lambda * \omega^T * \Sigma * \omega$
- Subject to:

$$\omega_i >= 0$$

$$\sum_{i=1}^{n} \omega_i = 1$$

- ω is the weight vector
- μ is the expected return vector
- λ is the risk aversion parameter
- Σ is the variance - covariance matrix

Quadratic programming solver

- Use the R package `quadprog` to solve the quadratic utility optimization problem
- `solve.QP()` solves quadratic programming problems of the form:

$$min(-d^T b + \frac{1}{2} b^T D b)$$

- Subject to the constraint:

$$A^T b >= b_0$$

```r
library(quadprog)
data(edhec)
dat <- edhec[,1:4]

# Create the constraint matrix
Amat <- cbind(1, diag(ncol(dat)), -diag(ncol(dat)))
# Create the constraint vector
bvec <- c(1, rep(0, ncol(dat)), -rep(1, ncol(dat)))
 # Create the objective matrix
Dmat <- 10 * cov(dat)
# Create the objective vector
dvec <- colMeans(dat)

# Specify number of equality constraints
meq <- 1

# Solve the optimization problem
opt <- solve.QP(Dmat, dvec, Amat, bvec, meq)
```

Let's practice!

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Introduction to PortfolioAnalytics

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Ross Bennett
Instructor

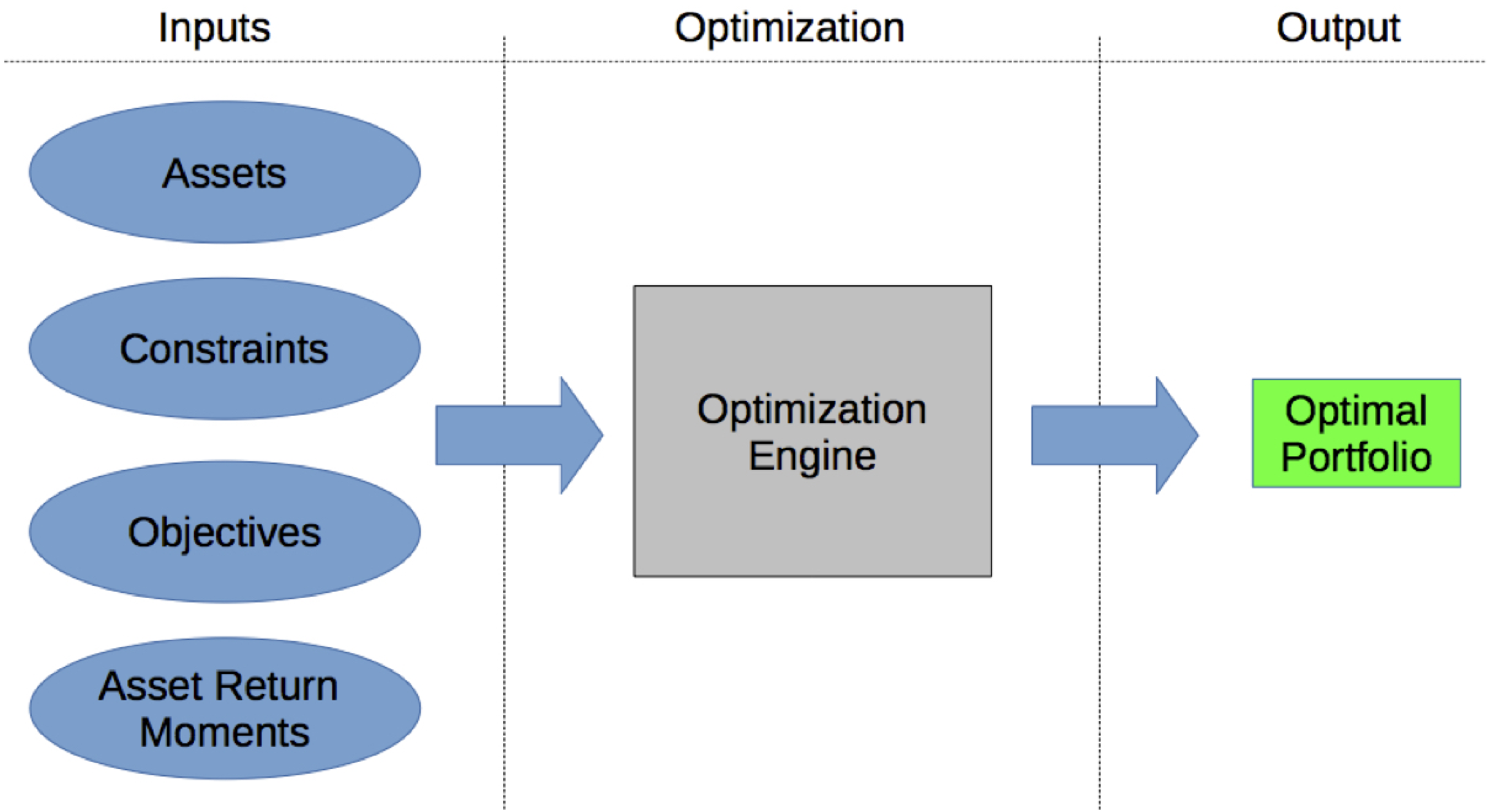
PortfolioAnalytics

`PortfolioAnalytics` is designed to provide numerical solutions and visualizations for portfolio optimization problems with complex constraints and objectives

Supports:

- Multiple and modular constraint and objective types
- An objective function can be any valid R function
- User defined moment functions (covariance matrix, return projections)
- Visualizations
- Solver agnostic
- Parallel computing

PortfolioAnalytics framework



Let's practice!

Portfolio specification, constraints, and objectives

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Ross Bennett
Instructor

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)
```

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

```r
# 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!

Running optimizations

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Ross Bennett
Instructor

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

```r
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

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

```r
# 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
```

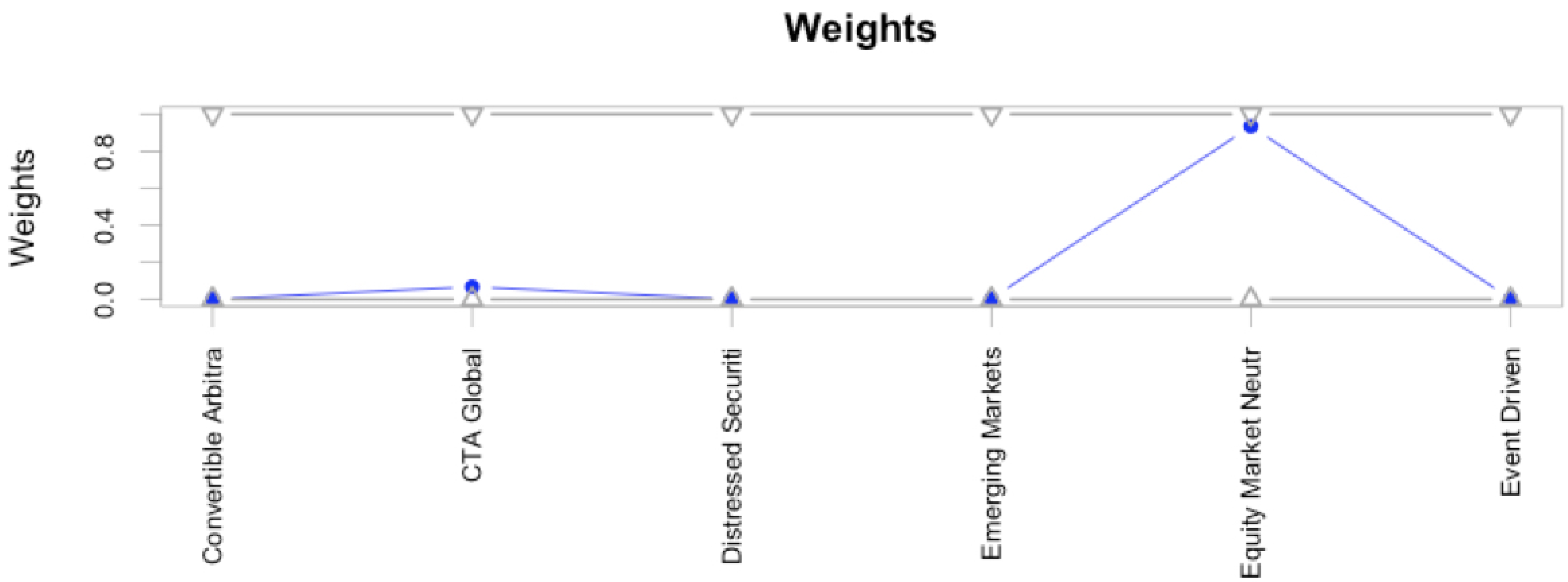
Example: extract weights

```
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
```

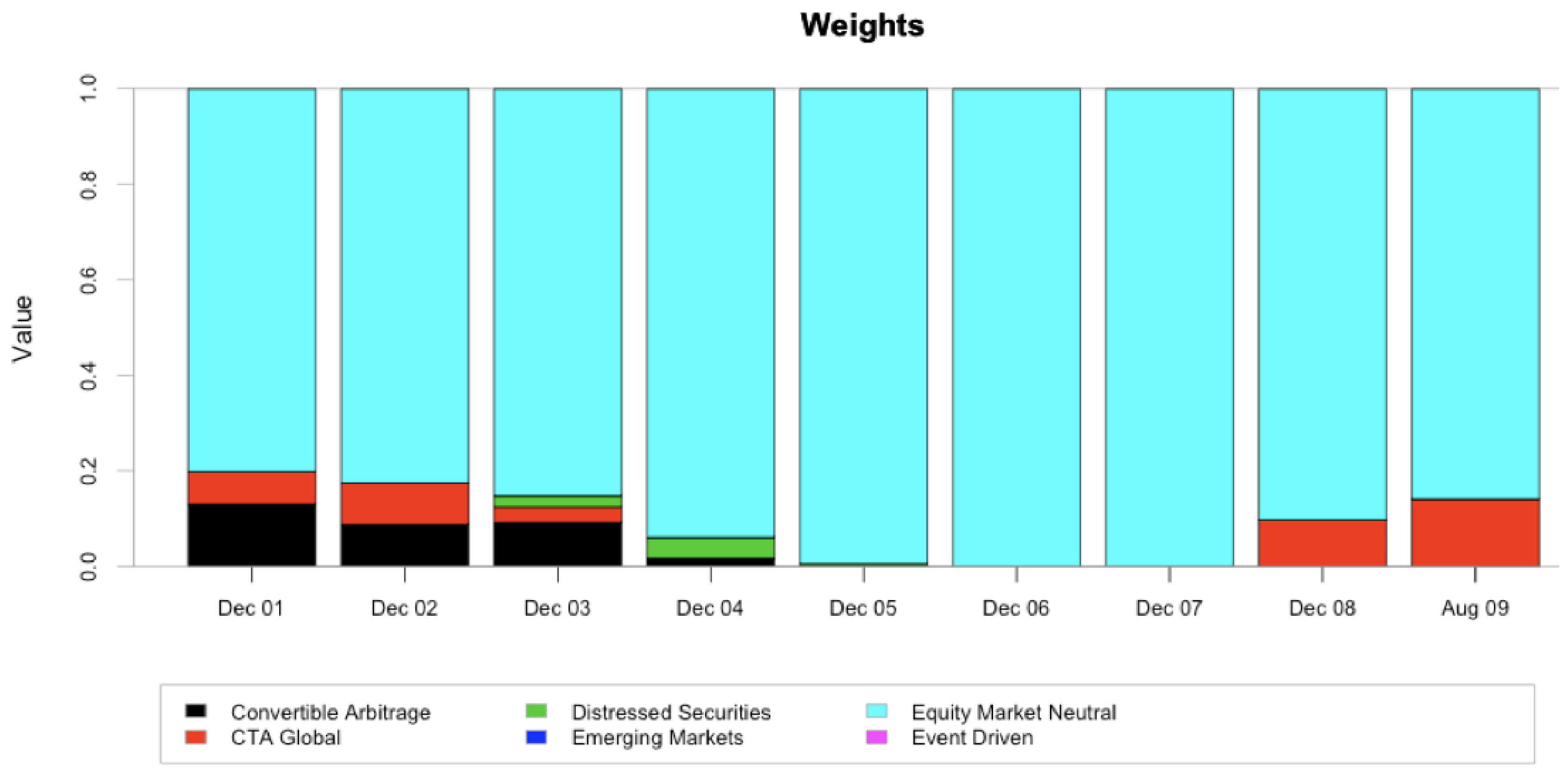
Example: chart weights

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



Example: chart weights

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



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

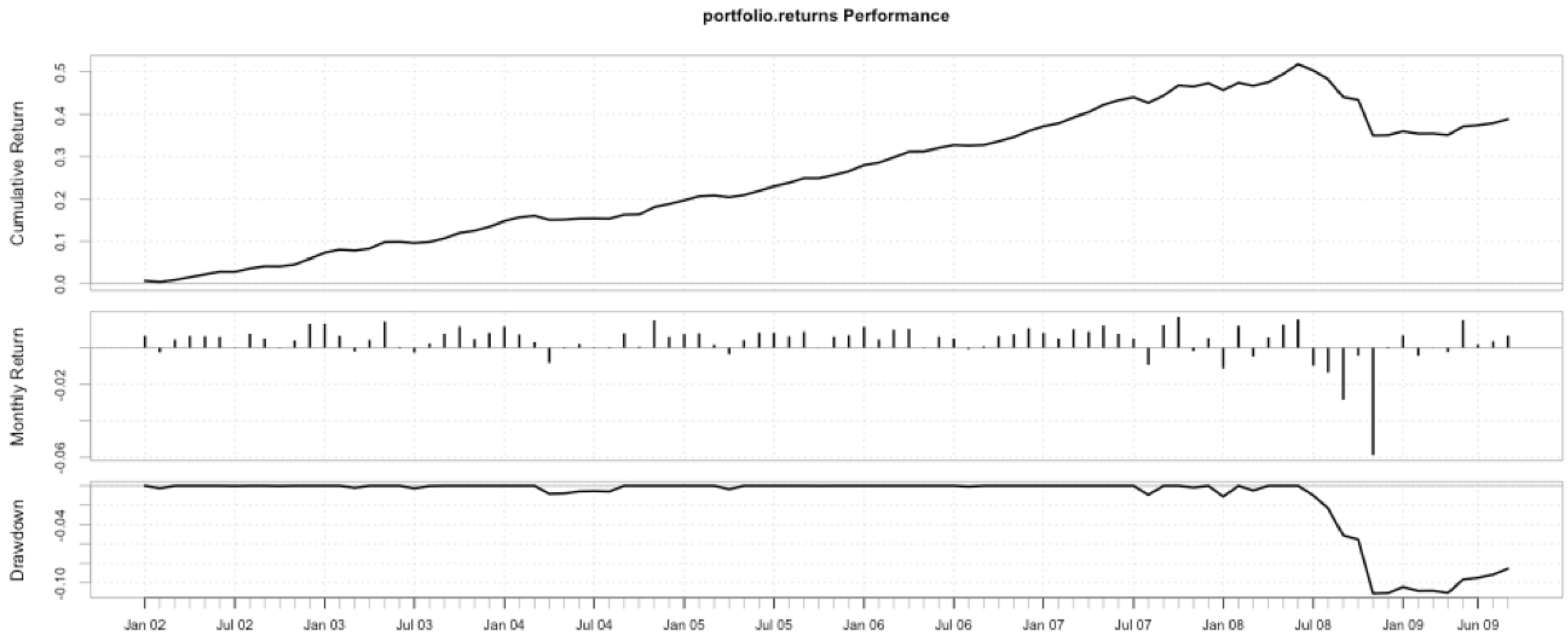
```
$StdDev
     StdDev
0.008855401
```

```r
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

Asset return oments

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Ross Bennett
Instructor

Optimization Inputs

Portfolio optimization problem inputs:

- Assets
- Constraints
- Objectives
- Moments of asset returns

Asset return moments

- First Moment: expected returns vector
- Second Moment: variance-covariance matrix
- Third Moment: coskewness matrix
- Fourth Moment: cokurtosis matrix

Asset return moments

Moments to estimate are determined by objectives and constraints:

- Mean - Variance
  - Expected returns vector
  - Covariance matrix
- Minimum Variance
  - Covariance matrix

Asset return moment estimates

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."*

20 Asset Portfolio:

Method	Sample	k = 3 factors
# of parameters	210	86

- Methods:
  - Sample
  - Shrinkage estimators
  - Factor model
  - Expressing views
  - Robust statistics

Calculating moments in PortfolioAnalytics

```
set.portfolio.moments(R,
                      portfolio,
                      method = c("sample", "boudt", "black_litterman", "meucci"),
                      ...)
```

`set.portfolio.moments()` supports several methods:

- Sample
- Boudt
- Black-Litterman
- Meucci

Example: moments in PortfolioAnalytics

```
# Sample vs Boudt
sample_moments <- set.portfolio.moments(R = asset_returns,
                                        portfolio = port_spec)

boudt_moments <- set.portfolio.moments(R = asset_returns,
                                       portfolio = port_spec,
                                       method = "boudt",
                                       k = 1)
```

Example: moments in PortfolioAnalytics

```
round(sample_moments$sigma, 6)
```

```
          [,1]      [,2]      [,3]  ...
[1,]  0.000402 -0.000034  0.000262  ...
[2,] -0.000034  0.000632 -0.000037  ...
[3,]  0.000262 -0.000037  0.000337  ...
[4,]  0.000429 -0.000010  0.000568  ...
```

```
round(boudt_moments$sigma, 6)
```

```
          [,1]      [,2]      [,3]  ...
[1,]  0.000403 -0.000016  0.000224  ...
[2,] -0.000016  0.000636 -0.000019  ...
[3,]  0.000224 -0.000019  0.000337  ...
[4,]  0.000523 -0.000044  0.000614  ...
```

Let's practice!

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Custom moment functions

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Ross Bennett
Instructor

Custom moment functions

A custom moment function is a user defined function

- Arguments:
  - `R` for asset returns
  - `portfolio` for the portfolio specification object
- Return a named list where the elements represent the moments
  - `mu` : Expected returns vector
  - `sigma` : Variance-covariance matrix
  - `m3` : Coskewness matrix
  - `m4` : Cokurtosis matrix

Example: custom moment function

```r
library(MASS)

custom_fun <- function(R, portfolio, rob_method = "mcd"){
    out <- list()
    out$sigma <- cov.rob(R, method = rob_method)
    return(out)
```

```r
# Passing the rob_method argument to custom_fun
optimize.portfolio(R, portfolio, momentFUN = custom_fun,
                   rob_method = "mcd")
optimize.portfolio(R, portfolio, momentFUN = custom_fun,
                   rob_method = "mve")
```

Let's practice!

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Objective functions

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Ross Bennett
Instructor

Objective functions

Objective functions compute the objective value. In `PortfolioAnalytics`, objective functions can be any valid R function.

- Common portfolio risk measures
  - standard deviation, expected shortfall, value at risk, component contribution to risk, maximum drawdown, Sharpe Ratio
- Common benchmark relative performance measures
  - information ratio, tracking error, excess return, maximum relative drawdown

Custom objective functions

User defined functions as objective functions

- Argument naming:
  - `R` for asset returns
  - `weights` for the portfolio weights
  - `mu`, `sigma`, `m3`, `m4` for the moments
- Returns a single value

```r
# Annualized sharpe ratio
sr_annualized <- function(R, weights, sigma, scale, rfr){

    # Geometric annualized return
    r <- Return.annualized(Return.portfolio(R, weights), scale = scale)
    # Annual excess return
    re <- r - rfr

    # Annualized portfolio standard deviation
    pasd <- sqrt(as.numeric(t(weights) %*%
                 sigma %*% weights)) * sqrt(scale)

    return(re / pasd)
}
```

```r
data(edhec)
asset_returns <- edhec[,1:4]
# Setup spec and add constraints
port_spec <- portfolio.spec(assets = colnames(asset_returns))
port_spec <- add.constraint(portfolio = port_spec,
                            type = "full_investment")
port_spec <- add.constraint(portfolio = port_spec,
                            type = "long_only")
# Add custom objective function
port_spec <- add.objective(portfolio = port_spec,
                           type = "return", name = "sr_annualized",
                           arguments = list(scale = 12, rfr = 0.02))
```

Let's practice!

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Real world example

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Ross Bennett
Instructor

Real world example

- Solve a portfolio optimization problem similar to the types of problems in the industry
- Apply techniques learned throughout the course
  - Specify a portfolio with constraints and objectives
  - Run the optimization with period rebalancing on historical data
  - Analyze the results
  - Refine constraints, objectives, and moment estimates
- Data
  - EDHEC-Risk Alternative Indexes monthly returns {6}
  - Jan 1997 - March 2016

```r
data(indexes)
returns <- indexes[,1:4]
# Equal weight benchmark
n <- ncol(returns)
equal_weights <- rep(1 / n, n)

benchmark_returns <- Return.portfolio(R = returns,
                                      weights = equal_weights,
                                      rebalance_on = "years")
colnames(benchmark_returns) <- "benchmark"
# Benchmark performance
table.AnnualizedReturns(benchmark_returns)
```

```
                          benchmark
Annualized Return            0.0775
Annualized Std Dev           0.1032
Annualized Sharpe (Rf=0%)    0.7509
```

Base portfolio definition

- Define a portfolio specification to be used as the base case
- The base portfolio specification is meant to be a simple approach with relaxed constraints and basic objectives

```
# Base portfolio specification
base_port_spec <- portfolio.spec(assets = colnames(returns))
base_port_spec <- add.constraint(portfolio = base_port_spec,
                                 type = "full_investment")
base_port_spec <- add.constraint(portfolio = base_port_spec,
                                 type = "long_only")
base_port_spec <- add.objective(portfolio = base_port_spec,
                                type = "risk", name = "StdDev")
```

Let's practice!

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Optimization backtest

INTERMEDIATE PORTFOLIO ANALYSIS IN R

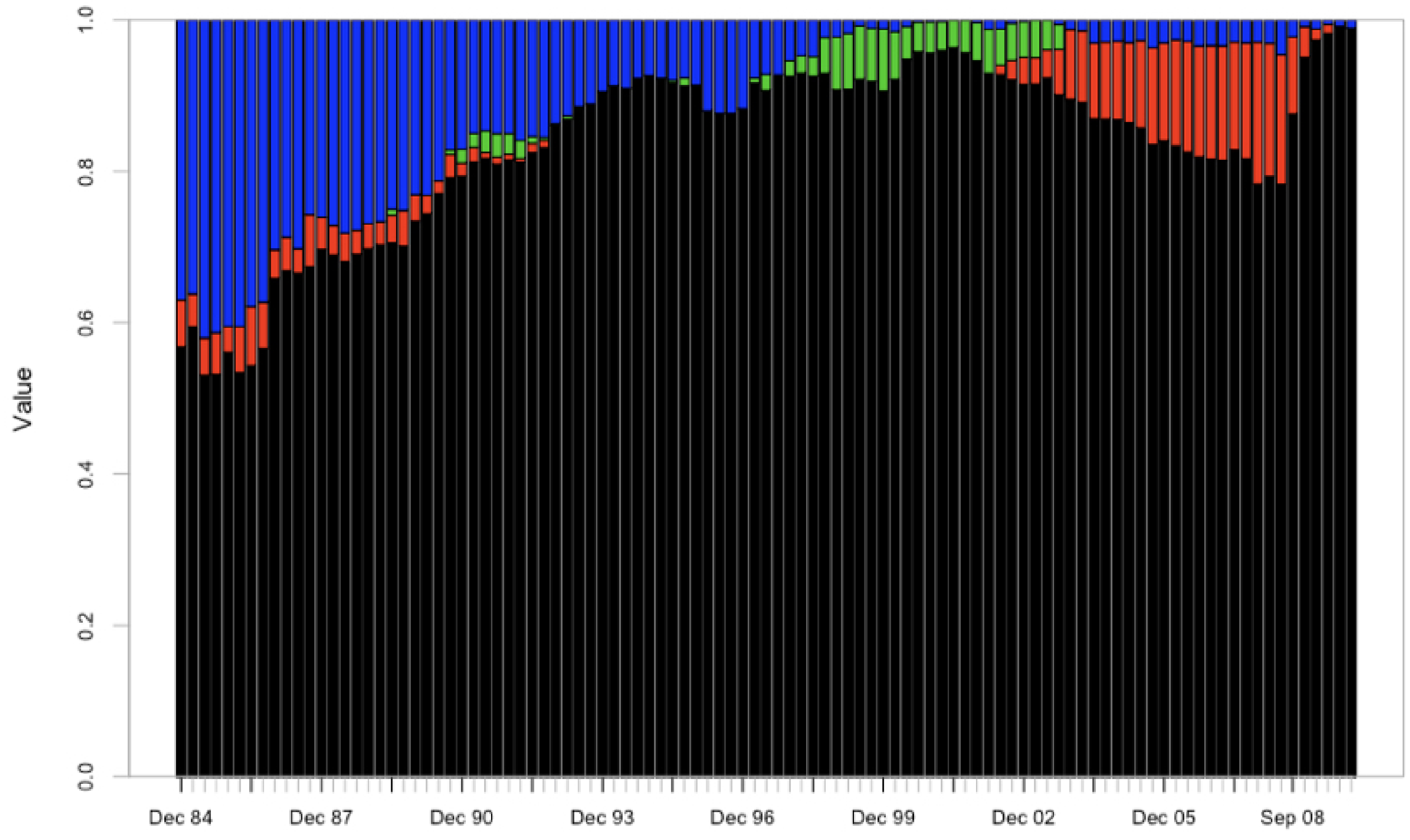
Ross Bennett
Instructor

Optimization backtest: execution

```
# Run the optimization with periodic rebalancing
opt_base <- optimize.portfolio.rebalancing(R = returns,
                    optimize_method = "ROI",
                    portfolio = base_port_spec,
                    rebalance_on = "quarters",
                    training_period = 60,
                    rolling_window = 60)
# Calculate portfolio returns
base_returns <- Return.portfolio(returns,
                                 extractWeights(opt_base))
colnames(base_returns) <- "base"
```

Optimization backtest: analysis

```
# Chart the optimal weights
chart.Weights(opt_base)
```

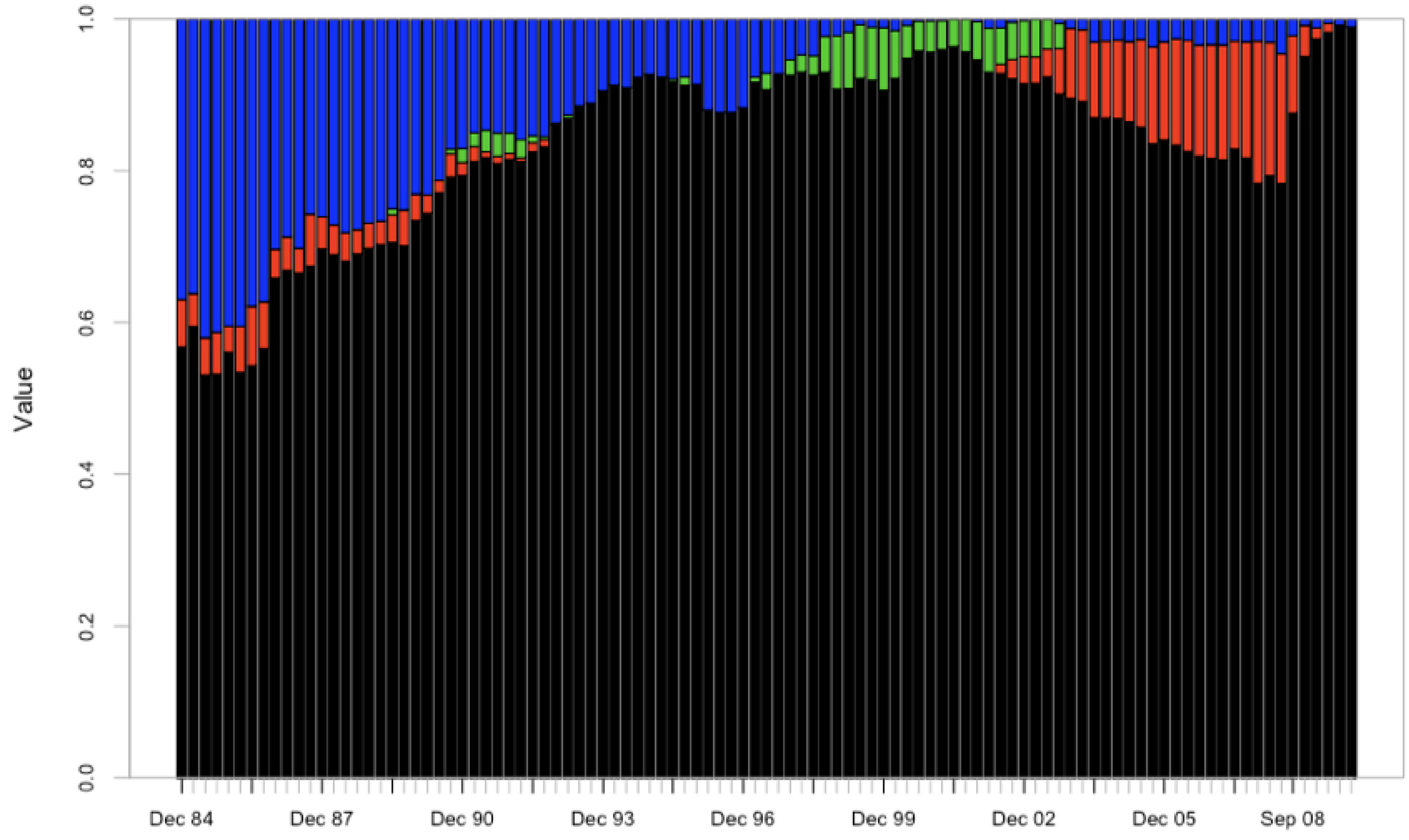


Optimization backtest: analysis

```
# Merge benchmark and portfolio returns
ret <- cbind(benchmark_returns, base_returns)

# Annualized performance
table.AnnualizedReturns(ret)
```

```
                          benchmark   base
Annualized Return            0.0775 0.0772
Annualized Std Dev           0.1032 0.0436
Annualized Sharpe (Rf=0%)    0.7509 1.7714
```

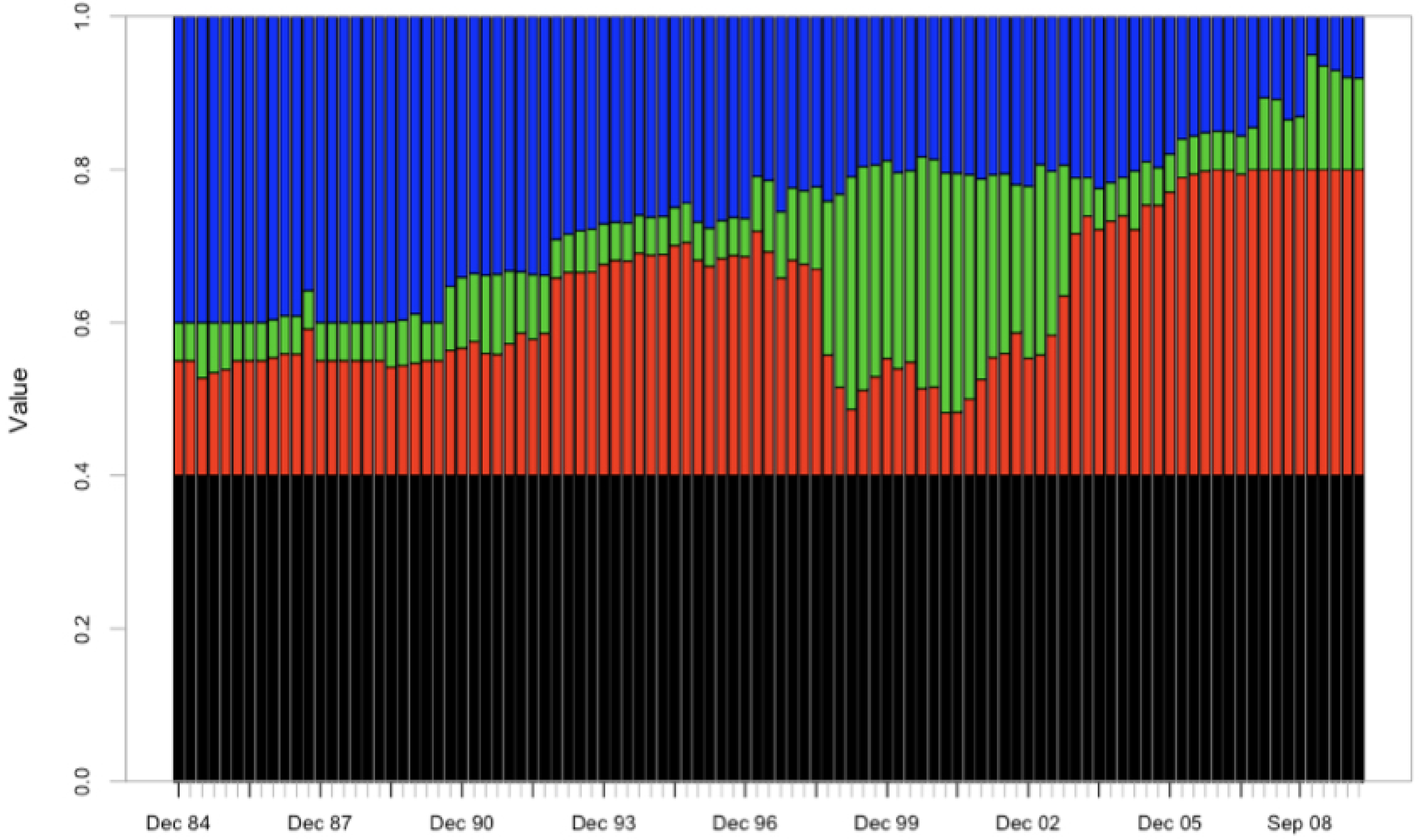
Optimization backtest: refine constraints



```r
# Make a copy of the portfolio specification
box_port_spec <- base_port_spec
# Update the constraint
box_port_spec <- add.constraint(portfolio = box_port_spec,
                                type = "box",
                                min = 0.05, max = 0.4,
                                indexnum = 2)
# Backtest
opt_box <- optimize.portfolio.rebalancing(R = returns,
                                          optimize_method = "ROI",
                                          portfolio = box_port_spec,
                                          rebalance_on = "quarters",
                                          training_period = 60,
                                          rolling_window = 60)
# Calculate portfolio returns
box_returns <- Return.portfolio(returns, extractWeights(opt_box))
colnames(box_returns) <- "box"
```

Optimization backtest: analysis refined constraints

```
# Chart the optimal weights
chart.Weights(opt_box)
```



Optimization backtest: analysis refined constraints

```
# Merge box portfolio returns
ret <- cbind(ret, box_returns)
# Annualized performance
table.AnnualizedReturns(ret)
```

```
                          benchmark   base    box
Annualized Return            0.0775 0.0772 0.0760
Annualized Std Dev           0.1032 0.0436 0.0819
Annualized Sharpe (Rf=0%)    0.7509 1.7714 0.9282
```

Let's practice!

Congratulations!

INTERMEDIATE PORTFOLIO ANALYSIS IN R

Ross Bennett
Instructor

Congratulations!

INTERMEDIATE PORTFOLIO ANALYSIS IN R