

Análise de Carteiras usando o R - Parte 4

Bibliografia – BKM, cap. 7

Claudio Lucinda

FEA-RP/USP

Risco Não Diversificável

- A seguir, temos uma figura calculada com os retornos das 37 ações que tem bastante negócios (negociadas em 2/3 dos dias nos 14 anos entre 2004 e 2018).
- Basicamente temos os desvios-padrão dos retornos para portfólios com 1, 2, ..., 37 ações, igualmente ponderados e rebalanceados mensalmente.
- Só temos nas duas pontas porque existem limites computacionais para isso.

Risco Não Diversificável

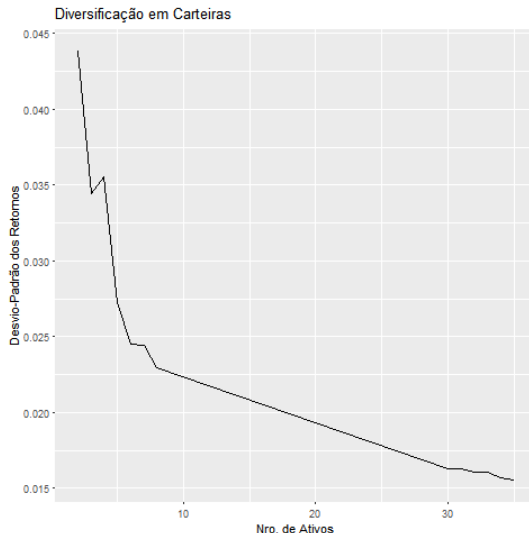


Figura 1: Carteira e Risco

Fronteira Eficiente e o Pacote PortfolioAnalytics

PortfolioAnalytics

- Nas aulas anteriores, nós utilizamos algumas funções do PortfolioAnalytics para alguns cálculos de carteira.
- Hoje iremos usar a funcionalidade mais legal do pacote, que é a capacidade de especificar e otimizar as carteiras de uma forma muito flexível.
- Inicialmente vamos fazer isso de uma forma mais geral para depois aplicarmos pro nosso caso.
- O código nesse caso é o `Opt_Portfolio_Analytics.R`

PortfolioAnalytics

- O processo de otimização de carteiras no PortfolioAnalytics tem quatro etapas:
 - Especificar a carteira - Quais são os ativos componentes
 - Adicionar restrições e objetivos
 - Otimizar
 - Analisar os resultados da otimização
- O código que vai ser usado aqui é o `eff_frontier.R`

Otimizadores

- Existem quatro métodos de otimização suportados pelo PortfolioAnalytics
 - Solvers LP e QP:
 - ROI: Um pacote que centraliza solvers de LP e QP
 - Outros Métodos:
 - DEoptim: Differential Evolution Optimization
 - random: Otimização por Portfólios Aleatórios
 - GenSA: Generalized Simulated Annealing
 - pso: Particle Swarm Optimization

Inicializando o PortfolioAnalytics

- O primeiro passo é especificar a lista de ativos que fazem parte da sua carteira:
- O passo seguinte é implementar o portfólio:

```
port_spec<-portfolio.spec(colnames(data_r))
port_spec <- add.constraint(portfolio = port_spec,
                           type = "full_investment")
port_spec <- add.objective(portfolio = port_spec,
                          type = "return",
                          name = "mean")
port_spec <- add.objective(portfolio = port_spec,
                          type = "risk",
                          name = "StdDev")
```


Objetivos de Risco e Retorno

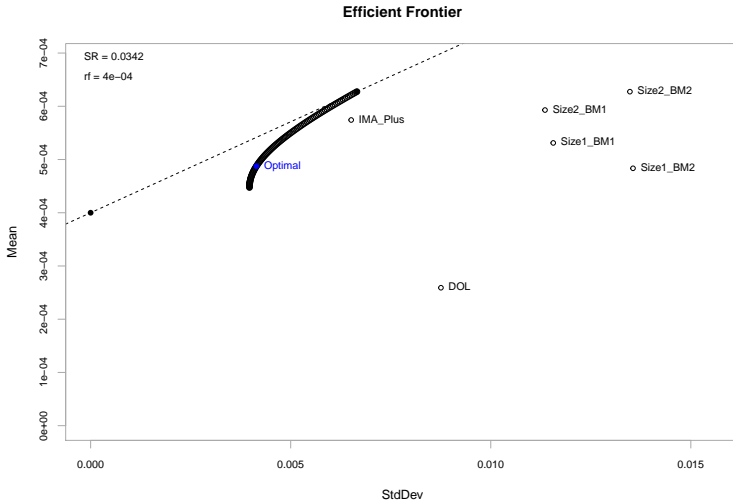
- O interessante deste objetivo de risco é que podemos usar todas aquelas medidas de risco para distribuição não normal que vimos anteriormente:
 - StdDev - O básico, o desvio-padrão dos retornos
 - ES - Expected Shortfall
 - VaR - *Value at Risk*
- O objetivo de retorno é uma função que vc especifica. Usualmente é o comando `mean`.

Otimizando a carteira

- Você otimiza a carteira com o comando `optimize.portfolio`.
- Os exemplos a seguir mostram dois tipos de otimização:
 - Usando Portfólios Aleatórios
 - Usando um otimizador formal

```
opt3 <- optimize.portfolio(data_r, portfolio = port_spec,  
                           optimize_method = "ROI",  
                           trace = TRUE, maxSR=TRUE,  
                           message=TRUE)
```

Fronteira Eficiente



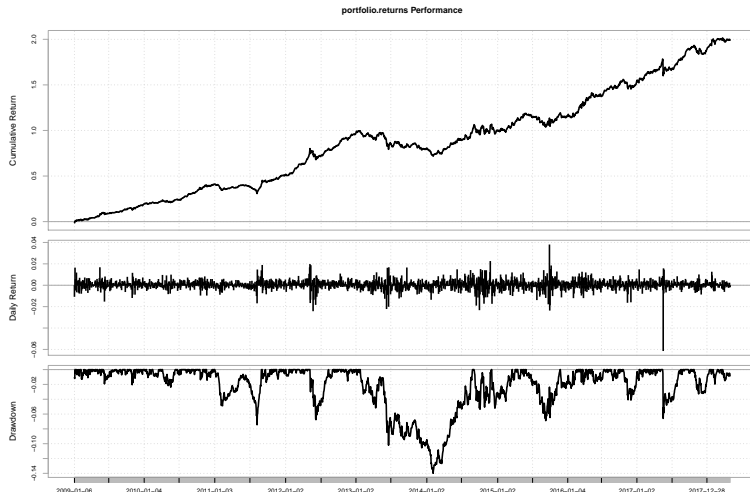
Visualização dos Pesos

```
chart.Weights(opt3)
```



Desempenho

```
base_port <- Return.portfolio(data_r, weights=extractWeights(opt
charts.PerformanceSummary(base_port)
```

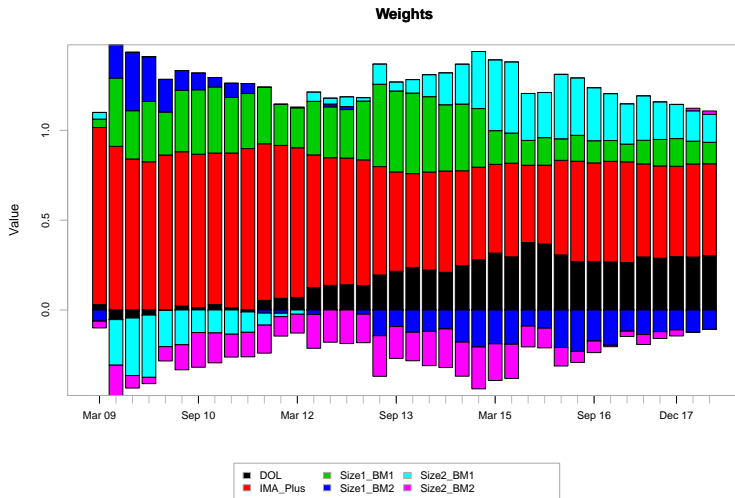


Backtesting e Avaliando o desempenho

- A função `optimize.portfolio.rebalancing` permite que você refaça o problema periodicamente
- Isso é muito bom para fazer *backtesting*.

[illegible]

Visualizando o Portifólio - Backtesting



#

Problemas mais Complexos

Portifólio que Minimiza o CTL

- Vamos agora construir uma carteira que minimiza o *Conditional Tail Loss*

```
port_spec.ES<-portfolio.spec(colnames(data_r))
port_spec.ES <- add.constraint(port_spec.ES,
                              type="weight_sum",
                              min_sum=0.99,
                              max_sum=1.01)
port_spec.ES <- add.objective(port_spec.ES,
                              type="return",
                              name="mean",
                              multiplier=0)
port_spec.ES <- add.objective(port_spec.ES,
                              type="risk",
                              name="ES")
```


Otimizando

- Para isso, utilizamos o ROI:

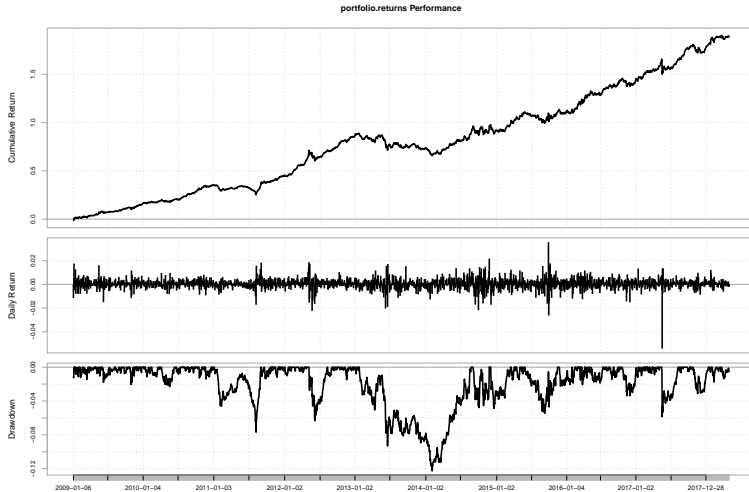
```
opt.minES <- optimize.portfolio(data_r, port_spec.ES,  
                                optimize_method="ROI")
```

Mostrando os pesos

```
chart.Weights(opt.minES)
```



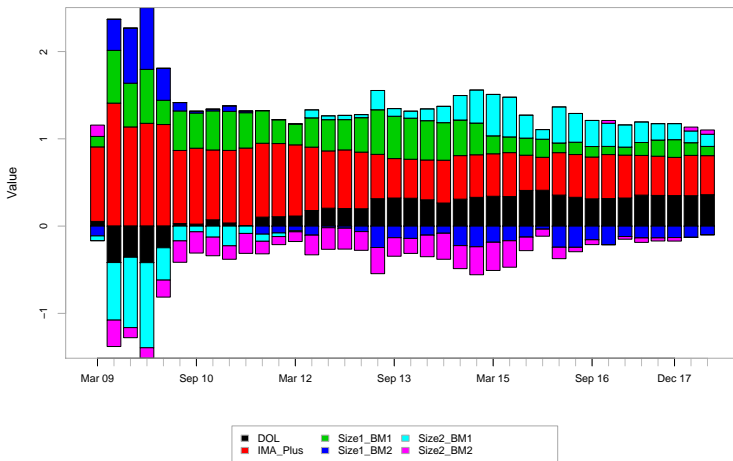
Mostrando o desempenho



Backtesting

Pesos

Weights



Comparando as carteiras

```
# Juntando os retornos
ret <- cbind(base_port, port.ES)
# Desempenho anualizado
table.AnnualizedReturns(ret, Rf=riskfree)
```

##	portfolio.returns	portfolio.returns.1
## Annualized Return	0.1278	0.1235
## Annualized Std Dev	0.0758	0.0729
## Annualized Sharpe (Rf=10.08%)	0.2598	0.2166