

Análise de Carteiras usando o R - Parte 5

Bibliografia – BKM, cap. 7

Claudio Lucinda

FEA-RP/USP

Otimizando Carteiras

Otimizando Carteiras

- Vamos agora entender melhor o que o PortfolioAnalytics faz.
- Existem Funções para achar uma carteira ótima, o que é consistente com a ideia de implementação de uma estratégia de investimento baseada em carteira ótima.
- Existem Funções para desenhar a fronteira eficiente, o que é útil para fins didáticos e para comparar o que você está fazendo em termos de investimento com o ótimo.

Um exemplo básico

```
port_spec<-portfolio.spec(colnames(IBOV>Returns_Final))
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")
```

Encontrando a Carteira que Maximiza o Índice de Sharpe – Sem Risk-Free

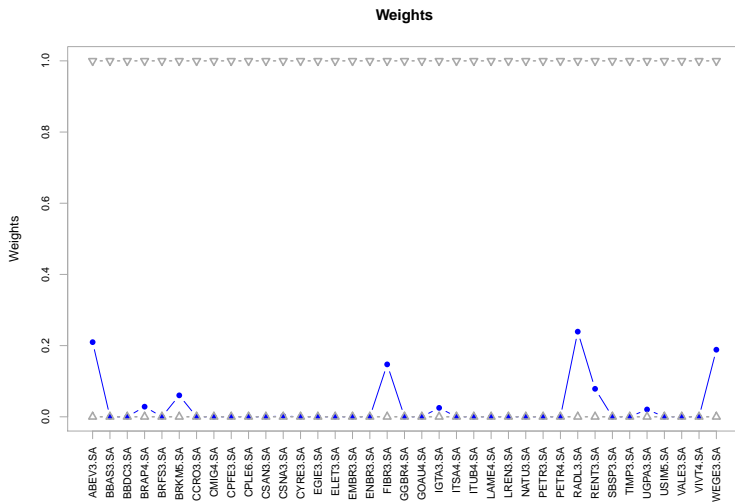
```
opt_1 <- optimize.portfolio(IBOV>Returns_Final,  
                             portfolio = port_spec,  
                             optimize_method = "ROI",  
                             trace = TRUE, maxSR=TRUE,  
                             message=TRUE)
```

Carteira Ótima

```
## *****
## PortfolioAnalytics Optimization
## *****
##
## Call:
## optimize.portfolio(R = IBOV>Returns_Final, portfolio = port_spec,
##   optimize_method = "ROI", trace = TRUE, maxSR = TRUE, message = TRUE)
##
## Optimal Weights:
## ABEV3.SA BBAS3.SA BBDC3.SA BRAP4.SA BRFS3.SA BRKM5.SA CCRO3.SA CMIG4.SA
## 0.2097 0.0000 0.0000 0.0284 0.0000 0.0602 0.0000 0.0000
## CPFE3.SA CPLE6.SA CSAN3.SA CSNA3.SA CYRE3.SA EGIE3.SA ELET3.SA EMBR3.SA
## 0.0000 0.0000 0.0000 0.0023 0.0000 0.0000 0.0000 0.0000
## ENBR3.SA FIBR3.SA GGBR4.SA GOAU4.SA IGTA3.SA ITSA4.SA ITUB4.SA LAME4.SA
## 0.0000 0.1472 0.0000 0.0000 0.0251 0.0000 0.0000 0.0000
## LREN3.SA NATU3.SA PETR3.SA PETR4.SA RADL3.SA RENT3.SA SBSP3.SA TIMP3.SA
## 0.0000 0.0000 0.0000 0.0000 0.2392 0.0786 0.0000 0.0000
## UGPA3.SA USIM5.SA VALE3.SA VIVT4.SA WEGE3.SA
## 0.0206 0.0000 0.0000 0.0000 0.1885
##
## Objective Measure:
## StdDev
## 0.04188
##
## mean
## 0.02494
```

Graficamente – Pesos

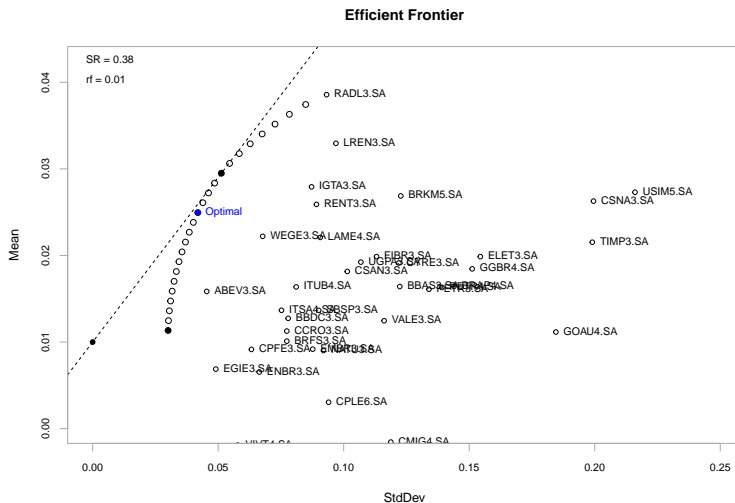
```
chart.Weights(opt_1)
```



Carteira ótima e Fronteira Eficiente

```
chart.EfficientFrontier(opt_1, match.col = "StdDev",  
                        n.portfolios = 25,  
                        xlim = NULL, ylim = NULL,  
                        cex.axis = 0.8,  
                        element.color = "darkgray",  
                        main = "Efficient Frontier",  
                        RAR.text = "SR", rf = 0.01,  
                        tangent.line = TRUE,  
                        cex.legend = 0.8,  
                        chart.assets = TRUE,  
                        labels.assets = TRUE,  
                        pch.assets = 21,  
                        cex.assets = 0.8)
```


Carteira ótima e Fronteira Eficiente – Figura



Exemplo 02 – Função Média-Variância

- Lembrando que a função Média-Variância é dada por $w'r - \frac{\lambda}{2} w'Vw$, em que λ é um parâmetro de aversão ao risco.
- Especificando

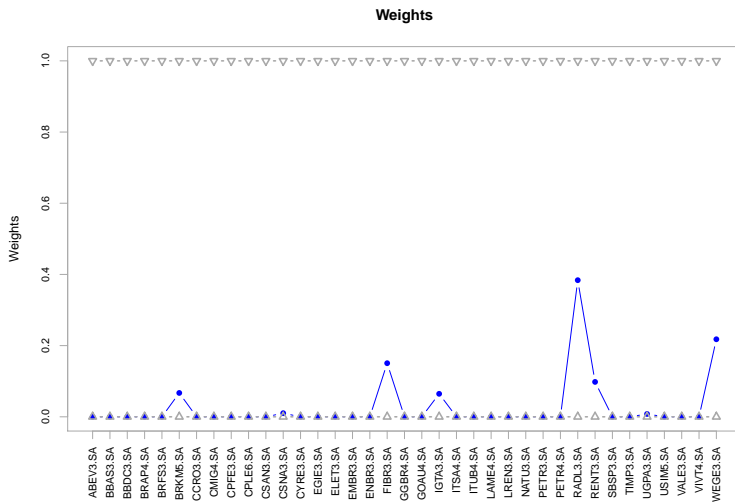
```
port_spec2 <- portfolio.spec(colnames(IBOV>Returns_Final))
port_spec2 <- add.constraint(portfolio = port_spec2,
                             type = "full_investment")
port_spec2 <- add.constraint(portfolio = port_spec2,
                             type = "long_only")
port_spec2 <- add.objective(portfolio = port_spec2,
                            type = "return",
                            name = "mean")
port_spec2 <- add.objective(portfolio = port_spec2,
                            type = "risk",
                            name = "StdDev",
                            risk_aversion=4)
```


Carteira Ótima nesse caso

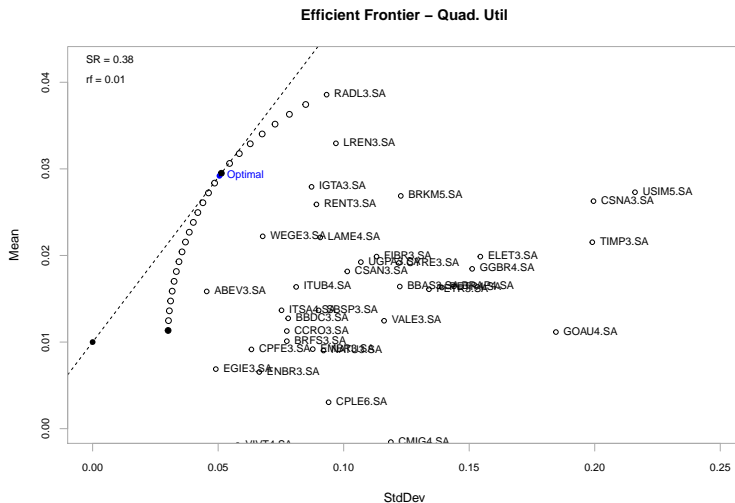
```
## *****
## PortfolioAnalytics Optimization
## *****
##
## Call:
## optimize.portfolio(R = IBOV>Returns_Final, portfolio = port_spec2,
##   optimize_method = "ROI", trace = TRUE, message = TRUE)
##
## Optimal Weights:
## ABEV3.SA BBAS3.SA BBDC3.SA BRAP4.SA BRFS3.SA BRKM5.SA CCRO3.SA CMIG4.SA
## 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0670 0.0000 0.0000
## CPFE3.SA CPLE6.SA CSAN3.SA CSNA3.SA CYRE3.SA EGIE3.SA ELET3.SA EMBR3.SA
## 0.0000 0.0000 0.0000 0.0102 0.0000 0.0000 0.0000 0.0000
## ENBR3.SA FIBR3.SA GGBR4.SA GOAU4.SA IGTA3.SA ITSA4.SA ITUB4.SA LAME4.SA
## 0.0000 0.1506 0.0000 0.0000 0.0646 0.0000 0.0000 0.0000
## LREN3.SA NATU3.SA PETR3.SA PETR4.SA RADL3.SA RENT3.SA SBSP3.SA TIMP3.SA
## 0.0000 0.0000 0.0000 0.0000 0.3838 0.0980 0.0000 0.0000
## UGPA3.SA USIM5.SA VALE3.SA VIVT4.SA WEGE3.SA
## 0.0080 0.0000 0.0000 0.0000 0.2179
##
## Objective Measure:
## mean
## 0.0292
##
## StdDev
## 0.05052
```

Graficamente – Pesos

```
chart.Weights(opt_2)
```



Graficamente

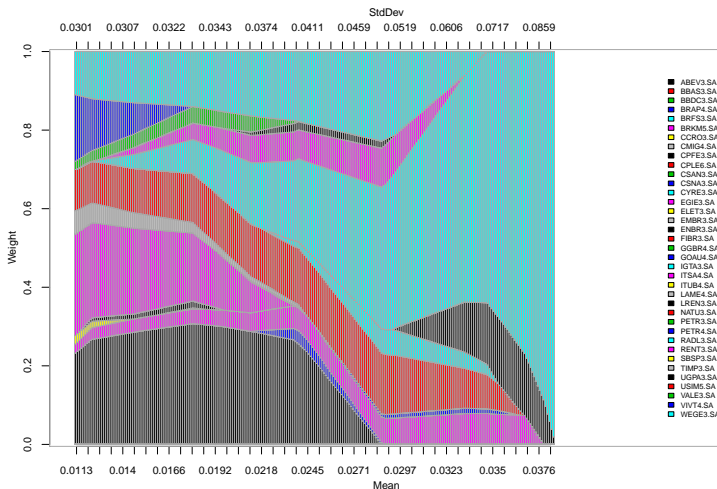


Pesos ao longo da Fronteira Eficiente

- A função `chart.EF.Weights` mostra a evolução dos pesos dos diferentes ativos à medida em que a gente se move ao longo da fronteira eficiente.
- No meu caso, fica ruim de ver porque tem muitos ativos e o R não consegue encontrar cores para pintar tudo direitinho.

```
chart.EF.Weights(meanvar.ef,  
                  colorset = NULL,  
                  n.portfolios = 25,  
                  by.groups = FALSE,  
                  match.col = "StdDev",  
                  main = "",  
                  cex.lab = 0.8,  
                  cex.axis = 0.8,  
                  cex.legend = 0.25,  
                  legend.labels = NULL,  
                  element.color = "darkgray",  
                  legend.loc = "right")
```


Pesos ao longo da Fronteira Eficiente – Graficamente



Backtesting

Backtesting

- Existe a função `optimize.portfolio.rebalancing` para fazer o problema de otimização de forma recursiva no passado.
- Ou seja, fazer o mesmo procedimento repetidamente com os dados passados para avaliar o desempenho da alocação
- Isto também é chamado de *Backtesting*.
- Vamos fazer isso com uma versão do que fizemos, só que para dados diários.
- O código tá em `eff_frontier_daily.R`

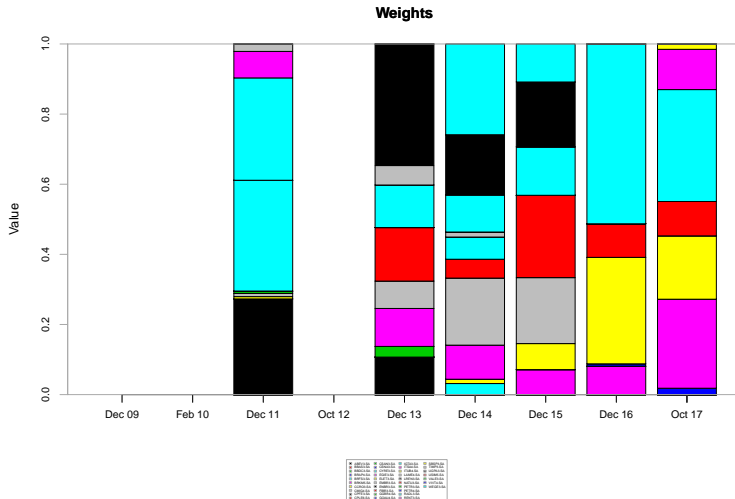
A carteira

```
IBOV>Returns<-na.omit(IBOV>Returns)

port_spec<-portfolio.spec(colnames(IBOV>Returns))
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")
```


Extraindo os pesos - por Ano

```
chart.Weights(opt2_rebal, ylim=range(0:1), cex.legend = 0.25)
```



Extraindo as medidas de desempenho por ano

```
extractObjectiveMeasures(opt2_rebal)
```

##		StdDev	mean
##	2009-12-23	NA	NA
##	2010-02-23	NA	NA
##	2011-12-29	0.014676516	0.0039343459
##	2012-10-29	NA	NA
##	2013-12-30	0.010322322	0.0010540743
##	2014-12-30	0.009063321	0.0010123504
##	2015-12-30	0.010310740	0.0009235861
##	2016-12-29	0.015792578	0.0022310451
##	2017-10-02	0.015042474	0.0020078040

Extraindo os valores dos pesos

- A função para isso é a `extractWeights`

```
extractWeights(opt2_rebal)
```

- Mas aqui eu vou usar o endereçamento direto pra um período só

```
opt2_rebal$opt_rebalancing$`2017-10-02`
```


Extraindo os valores dos pesos

```
## *****
## PortfolioAnalytics Optimization
## *****
##
## Call:
## optimize.portfolio(R = R[(ifelse(ep - rolling_window >= 1, ep -
##     rolling_window, 1)):ep, ], portfolio = portfolio, optimize_method = optimize_method,
##     search_size = search_size, trace = trace, parallel = FALSE,
##     maxSR = TRUE, rp = rp)
##
## Optimal Weights:
## ABEV3.SA BBAS3.SA BBDC3.SA BRAP4.SA BRFS3.SA BRKM5.SA CCR03.SA CMIG4.SA
## 0.0000 0.0000 0.0000 0.0182 0.0000 0.2539 0.0000 0.0000
## CPFE3.SA CPLE6.SA CSAN3.SA CSNA3.SA CYRE3.SA EGIE3.SA ELET3.SA EMBR3.SA
## 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.1806 0.0000
## ENBR3.SA FIBR3.SA GGBR4.SA GOAU4.SA IGTA3.SA ITSA4.SA ITUB4.SA LAME4.SA
## 0.0000 0.0981 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
## LREN3.SA NATU3.SA PETR3.SA PETR4.SA RADL3.SA RENT3.SA SBSP3.SA TIMP3.SA
## 0.0000 0.0000 0.0000 0.0000 0.0000 0.3191 0.1148 0.0153 0.0000
## UGPA3.SA USIM5.SA VALE3.SA VIVT4.SA WEGE3.SA
## 0.0000 0.0001 0.0000 0.0000 0.0000
##
## Objective Measure:
## StdDev
## 0.01504
##
##
## mean
## 0.002008
```