Design of Risk Parity Portfolios

Convex Group - HKUST

2018-08-26

Contents

This vignette illustrates the design of risk-parity portfolios, widely used by practitioners in the financial industry, with the package `riskParityPortfolio` (with a comparison with other packages) and gives a description of the algorithms used.

1 Comparison with other packages

Existing packages that I quickly found (a more thorough search is due):

- Package cccp contains function `rp` (and the package itself may be useful for solving conic optimization problems)
- Package FinCovRegularization contains function `RiskParity` (the package may be useful for covariance matrix regularization, which is one of my upcoming projects…)
- Here you can find some simple code for risk parity portfolio: http://nakisa.org/bankr-useful-financial-r-snippets/risk-parity-portfolios-r/

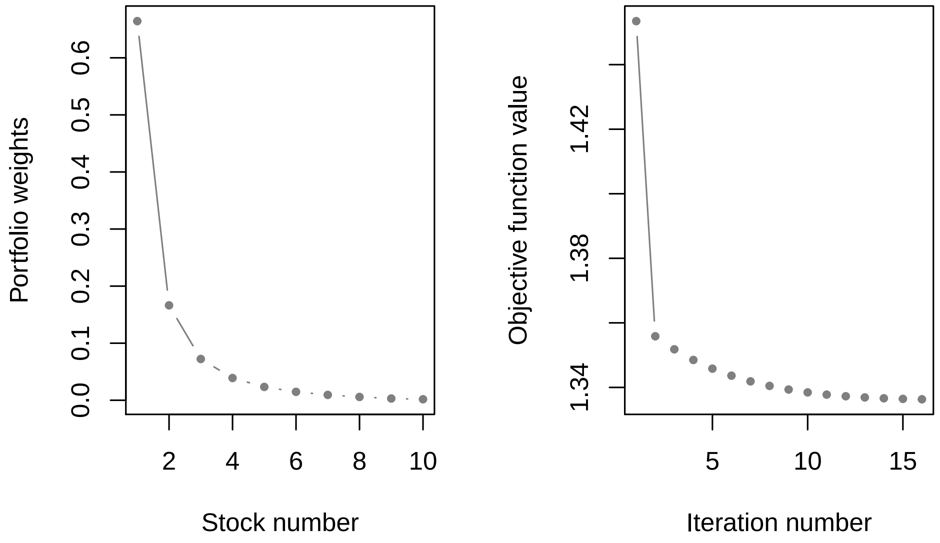
Websites to check:

- Many links to docs related to risk parity here: https://www.r-bloggers.com/risk-parity/
- Nice intro articles that I can use in my slides for the course:
  - https://www.ipe.com/risk-parity-nice-idea-awkward-reality/40026.article
  - http://news.morningstar.com/pdfs/gmohiddenrisks.pdf
  - https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1707478
- More links:
  - https://pdfs.semanticscholar.org/094d/24b924caa659442065401999d7a77e06953e.pdf
  - https://cdn2.hubspot.net/hubfs/2529352/Blog/2010_03_nepc_risk_parity.pdf?t=1501681499185
  - https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1271972

2 Usage of the package

```r
library(riskParityPortfolio)
stock_index <- c(1:10)
Sigma <- diag(c(1:10) ^ 2)
mu <- runif(10)
res <- riskParityPortfolio(mu, Sigma, nu = 0., type = "1", ftol = 1e-4)
```

```r
niter <- length(res$obj_fun)
par(mfrow = c(1, 2))
plot(stock_index, res$portfolio_weights, type = "b", pch=19, cex=.6,
     col = scales::alpha("black", .5), xlab = "Stock number",
     ylab = "Portfolio weights")
plot(c(1:niter), res$obj_fun, type = "b", pch=19, cex=.6,
     col = scales::alpha("black", .5), xlab = "Iteration number",
     ylab = "Objective function value")
```



3 Explanation of the algorithms

References