Exercises

Portfolio Optimization: Theory and Application
Chapter 11 – Risk Parity Portfolios

Daniel P. Palomar (2025). *Portfolio Optimization: Theory and Application.* Cambridge University Press.

portfoliooptimizationbook.com

Exercise 11.1: Change of variable

Show why $\boldsymbol{\Sigma x} = \boldsymbol{b}/\boldsymbol{x}$ can be equivalently solved as $\boldsymbol{Cx} = \boldsymbol{b}/\boldsymbol{x}$, where $\boldsymbol{C}$ is the correlation matrix defined as $\boldsymbol{C} = \boldsymbol{D}^{-1/2}\boldsymbol{\Sigma}\boldsymbol{D}^{-1/2}$ with $\boldsymbol{D}$ a diagonal matrix containing $\mathrm{diag}(\boldsymbol{\Sigma})$ along the main diagonal. Would it be possible to use instead $\boldsymbol{C} = \boldsymbol{M}^{-1/2}\boldsymbol{\Sigma}\boldsymbol{M}^{-1/2}$, where $\boldsymbol{M}$ is not necessarily a diagonal matrix?

Exercise 11.2: Naive diagonal risk parity portfolio

If the covariance matrix is diagonal, $\boldsymbol{\Sigma} = \boldsymbol{D}$, then the system of nonlinear equations $\boldsymbol{\Sigma x} = \boldsymbol{b}/\boldsymbol{x}$ has the closed-form solution $\boldsymbol{x} = \sqrt{\boldsymbol{b}/\mathrm{diag}(\boldsymbol{D})}$. Explore whether a closed-form solution can be obtained for the rank-one plus diagonal case $\boldsymbol{\Sigma} = \boldsymbol{u}\boldsymbol{u}^\mathsf{T} + \boldsymbol{D}$.

Exercise 11.3: Vanilla convex risk parity portfolio

The solution to the formulation

$$\begin{array}{ll} \underset{\boldsymbol{x}\geq\boldsymbol{0}}{\text{maximize}} & \boldsymbol{b}^\mathsf{T}\log(\boldsymbol{x}) \\ \text{subject to} & \sqrt{\boldsymbol{x}^\mathsf{T}\boldsymbol{\Sigma}\boldsymbol{x}} \leq \sigma_0 \end{array}$$

is

$$\lambda\boldsymbol{\Sigma x} = \boldsymbol{b}/\boldsymbol{x} \times \sqrt{\boldsymbol{x}^\mathsf{T}\boldsymbol{\Sigma}\boldsymbol{x}}.$$

Can you solve for λ and rewrite the solution in a more compact way without λ?

Exercise 11.4: Newton's method

Newton's method requires computing the direction $\boldsymbol{d} = \mathsf{H}^{-1}\nabla f$ or, equivalently, solving the system of linear equations $\mathsf{H}\,\boldsymbol{d} = \nabla f$ for $\boldsymbol{d}$. Explore whether a more efficient solution is possible by exploiting the structure of the gradient and Hessian:

$$\begin{aligned} \nabla f &= \boldsymbol{\Sigma}\boldsymbol{x} - \boldsymbol{b}/\boldsymbol{x},\\ \mathsf{H} &= \boldsymbol{\Sigma} + \mathrm{Diag}(\boldsymbol{b}/\boldsymbol{x}^2). \end{aligned}$$

Exercise 11.5: MM algorithm

The MM algorithm requires the computation of the largest eigenvalue $\lambda_{\max}$ of matrix $\boldsymbol{\Sigma}$, which can be obtained from the eigenvalue decomposition of the matrix. A more efficient alternative is the *power iteration method*. Program both methods and compare their computational complexity.

Exercise 11.6: Coordinate descent vs. SCA methods

Consider the vanilla convex formulation

$$\underset{\boldsymbol{x}\geq\boldsymbol{0}}{\text{minimize}} \quad \tfrac{1}{2}\boldsymbol{x}^\mathsf{T}\boldsymbol{\Sigma}\boldsymbol{x} - \boldsymbol{b}^\mathsf{T}\log(\boldsymbol{x}).$$

Implement the cyclical coordinate descent method and the parallel SCA method in a high-level programming language (e.g., R, Python, Julia, or MATLAB) and compare the convergence against the CPU time for these two methods. Then, re-implement these two methods in a low-level programming language (e.g., C, C++, C#, or Rust) and compare the convergence again. Comment on the difference observed.