Efficient Computation of Risk and Jacobians for Risk-Parity Portfolio

Convex Group - HKUST

2018-09-03

Contents

This note contains compact expressions for the risk and corresponding Jacobians for efficient implementation

1 General risk-parity portfolio formulation

The risk-parity portfolio formulation is of the form [1]:

$$\begin{array}{ll} \underset{\mathbf{w}}{\textsf{minimize}} & R(\mathbf{w}) \\ \textsf{subject to} & \mathbf{1}^T\mathbf{w} = 1, \quad \mathbf{w} \in \mathcal{W}, \end{array}$$

where the risk term is of the form

$$R(\mathbf{w}) = \sum_{i,j=1}^{N} (g_{ij}(\mathbf{w}))^2$$

or simply

$$R(\mathbf{w}) = \sum_{i=1}^{N} (g_i(\mathbf{w}))^2.$$

This problem can be solved directly with some nonlinear solver (for which we need to be able to compute the risk term $R(\mathbf{w})$ (even better if the gradient is computed) as well as with the Successive Convex Approximation (SCA) method developed in [2]. The algorithm iteratively solves a sequence of QP problems of the form:

$$\begin{array}{ll} \underset{\mathbf{w}}{\textsf{minimize}} & \tilde{U}\left(\mathbf{w}, \mathbf{w}^k\right) = \frac{1}{2}\mathbf{w}^T\mathbf{Q}^k\mathbf{w} + \mathbf{w}^T\mathbf{q}^k + \lambda F\left(\mathbf{w}\right) \\ \textsf{subject to} & \mathbf{1}^T\mathbf{w} = 1, \quad \mathbf{w} \in \mathcal{W}, \end{array}$$

where

$$\begin{aligned} \mathbf{g}\left(\mathbf{w}^k\right) &\triangleq \left[g_1\left(\mathbf{w}^k\right), \ldots, g_N\left(\mathbf{w}^k\right)\right]^T \\ \mathbf{A}^k\left(\mathbf{w}^k\right) &\triangleq \left[\nabla g_1\left(\mathbf{w}^k\right), \ldots, \nabla g_N\left(\mathbf{w}^k\right)\right]^T, \\ \mathbf{Q}^k &\triangleq 2\left(\mathbf{A}^k\right)^T\mathbf{A}^k + \tau\mathbf{I}, \\ \mathbf{q}^k &\triangleq 2\left(\mathbf{A}^k\right)^T\mathbf{g}\left(\mathbf{w}^k\right) - \mathbf{Q}^k\mathbf{w}^k. \end{aligned}$$

To effectively implement the SCA method we need efficient computation of the risk contribution terms contained in $\mathbf{g}(\mathbf{w})$ and their gradients contained in the Jacobian matrix $\mathbf{A}(\mathbf{w}) = [\nabla g_1(\mathbf{w}), \ldots, \nabla g_N(\mathbf{w})]^T$.

Notation:

- Define the i-th risk contribution: $x_i = w_i(\boldsymbol{\Sigma}\mathbf{w})_i$ (in R: `x <- w*(Sigma %*% w)`)

- Gradient of $R(\mathbf{w})$: $\nabla_{\mathbf{w}} R = \left[\frac{\partial R}{\partial w_1}, \ldots, \frac{\partial R}{\partial w_N}\right]^T$
- Jacobian of $\mathbf{w}$: $\mathsf{J}_{\mathbf{w}}\mathbf{x} = \frac{\partial \mathbf{x}}{\partial \mathbf{w}^T}$ (note that the Jacobian of a scalar function is the traspose of the gradient)
- For the single index case: $\mathbf{g}(\mathbf{w}) = [g_1(\mathbf{w}), \ldots, g_N(\mathbf{w})]^T$
- For the double index case: $\mathbf{G}(\mathbf{w}) = (g_{ij}(\mathbf{w}))$ and $\mathbf{g}(\mathbf{w}) = \mathsf{vec}(\mathbf{G}(\mathbf{w}))$
- M-notation [2]: for unification purposes, one can express the risk contributions and risk term in terms of the matrix $\mathbf{M}_i$, whose i-th row is equal to that of $\boldsymbol{\Sigma}$ and is zero elsewhere. Then it follows that $x_i = w_i(\boldsymbol{\Sigma}\mathbf{w})_i = \mathbf{w}^T\mathbf{M}_i\mathbf{w}$ (although for computational purposes it is far more efficient to use former expression of x_i than the latter).

2 Original formulation

Let's focus on one specific risk expression:

$$R(\mathbf{w}) = \sum_{i,j=1}^{N} \left(w_i \left(\boldsymbol{\Sigma}\mathbf{w}\right)_i - w_j \left(\boldsymbol{\Sigma}\mathbf{w}\right)_j\right)^2 = \sum_{i,j=1}^{N} (x_i - x_j)^2 = 2N\sum_i x_i^2 - 2\left(\sum_i x_i\right)^2$$

which can be efficiently coded as `risk <- 2*N*sum(x^2) - 2*sum(x)^2`.

Let's compute now the gradient of $R(\mathbf{w})$:

- $\frac{\partial R}{\partial x_i} = 4(Nx_i - \sum_i x_i) \implies \nabla_{\mathbf{x}} R = 4(N\mathbf{x} - (\mathbf{1}^T\mathbf{x})\mathbf{1})$
- $\frac{\partial x_i}{\partial w_j} = w_i\boldsymbol{\Sigma}_{ij} + \delta_{ij}(\boldsymbol{\Sigma}\mathbf{w})_i \implies \mathsf{J}_{\mathbf{w}}\mathbf{x} = \mathsf{Diag}(\mathbf{w})\boldsymbol{\Sigma} + \mathsf{Diag}(\boldsymbol{\Sigma}\mathbf{w})$
- chain rule: using Jacobians is $\mathsf{J}_{\mathbf{w}} R = \mathsf{J}_{\mathbf{x}} R \cdot \mathsf{J}_{\mathbf{w}}\mathbf{x}$, using gradients is $(\nabla_{\mathbf{w}} R)^T = (\nabla_{\mathbf{x}} R)^T \cdot \mathsf{J}_{\mathbf{w}}\mathbf{x}$ or, more conveniently, $\nabla_{\mathbf{w}} R = (\mathsf{J}_{\mathbf{w}}\mathbf{x})^T \cdot \nabla_{\mathbf{x}} R$:

  $$\nabla_{\mathbf{w}} R = 4(\boldsymbol{\Sigma}\mathsf{Diag}(\mathbf{w}) + \mathsf{Diag}(\boldsymbol{\Sigma}\mathbf{w}))(N\mathbf{x} - (\mathbf{1}^T\mathbf{x})\mathbf{1}),$$

  which can be coded as `risk_grad <- 4*(Sigma*w + diag(Sigma %*% w)) %*% *(N*x-sum(x)*rep(1, N))`.

If we are interested in implenting the SCA method, then we need an expression for the risk contributions contained in $\mathbf{g}$ as well as its Jacobian matrix $\mathbf{A}$.

The risk contributions are

$$g_{ij}(\mathbf{w}) = w_i(\boldsymbol{\Sigma}\mathbf{w})_i - w_j(\boldsymbol{\Sigma}\mathbf{w})_j = x_i - x_j,$$

which can be efficiently coded as `g <- rep(x, times = N) - rep(x, each = N)`. So another way to compute $R(\mathbf{w})$ is with `sum(g^2)`, but it's not as efficient as the previous computation since $\mathbf{g}$ has N^2 elements.

Matrix $\mathbf{A}$ is more involved to compute. Using the M-notation: $\nabla g_{ij} = (\mathbf{M}_i + \mathbf{M}_i^T + \mathbf{M}_j + \mathbf{M}_j^T)\mathbf{w}$ (recall that $\mathbf{A} = [\nabla g_1, \ldots, \nabla g_N]^T$). But we need an efficient way to compute this...

This derivation is in the making:

Observations:

1. each terms $\nabla g_{ij} = (\mathbf{M}_i + \mathbf{M}_i^T + \mathbf{M}_j + \mathbf{M}_j^T)\mathbf{w}$ is symmetric, i.e., $\nabla g_{ij} = \nabla g_{ji}$

2. we need $\mathbf{A}$ only through the terms $\mathbf{A}^T\mathbf{A}$ (which is symmetric) and $\mathbf{A}^T\mathbf{g}$, which can simplify things a lot

3. can we compute efficiently $(\mathbf{M}_i + \mathbf{M}_i^T)\mathbf{w}$ for all $i = 1, \ldots, N$ at once?. Yes!!:

   $$[\mathbf{M}_1\mathbf{w}, \ldots, \mathbf{M}_N\mathbf{w}] = \mathsf{Diag}(\boldsymbol{\Sigma}\mathbf{w})$$

   and

   $$\left[\mathbf{M}_1^T\mathbf{w}, \ldots, \mathbf{M}_N^T\mathbf{w}\right] = \boldsymbol{\Sigma}\mathsf{Diag}(\mathbf{w})$$

4. TBD:

   $$\begin{aligned}\nabla g_{ij} &= (\mathbf{M}_i + \mathbf{M}_i^T + \mathbf{M}_j + \mathbf{M}_j^T)\mathbf{w} \\ &= \mathbf{M}_i\mathbf{w} + \mathbf{M}_i^T\mathbf{w} + \mathbf{M}_j\mathbf{w} + \mathbf{M}_j^T\mathbf{w}\end{aligned}$$

References

[1] Y. Feng and D. P. Palomar, *A Signal Processing Perspective on Financial Engineering.* Foundations; Trends in Signal Processing, Now Publishers, 2016.

[2] Y. Feng and D. P. Palomar, "SCRIP: Successive convex optimization methods for risk parity portfolios design," *IEEE Trans. Signal Process.*, vol. 63, no. 19, pp. 5285–5300, 2015.