Simple example

Daniel P. Palomar

Hong Kong University of Science and Technology (HKUST)

2019-07-14

Contents

This note contains an example under consideration by Alexis Nortier.

1 Risk-parity portfolio formulation

The risk-parity portfolio formulation is of the form [1][2]:

$$\begin{array}{ll} \underset{\mathbf{w}}{\textsf{minimize}} & R(\mathbf{w}) \\ \textsf{subject to} & \mathbf{1}^T\mathbf{w} = 1 \\ & \mathbf{w} \geq \mathbf{0}, \end{array}$$

where the risk term is of the form (double summation) $R(\mathbf{w}) = \sum_{i,j=1}^{N}(g_{ij}(\mathbf{w}))^2$ or simply (single summation) $R(\mathbf{w}) = \sum_{i=1}^{N}(g_i(\mathbf{w}))^2$.

2 Parameter definition for the test

Consider the following parameters:

$$\mathbf{\Sigma} = \left[\begin{array}{ccc} 1.0000 & 0.0015 & -0.0119 \\ 0.0015 & 1.0000 & -0.0308 \\ -0.0119 & -0.0308 & 1.0000 \end{array}\right]$$

$$\mathbf{b} = \left[\begin{array}{c} 0.1594 \\ 0.0126 \\ 0.8280 \end{array}\right]$$

and the corresponding code:

```r
Sigma <- rbind(c(1.0000, 0.0015, -0.0119),
               c(0.0015, 1.0000, -0.0308),
               c(-0.0119, -0.0308, 1.0000))
b <- c(0.1594, 0.0126, 0.8280)
```

The optimal solution is known to be

$$\mathbf{w}^{\star} = \begin{bmatrix} 0.2799 \\ 0.0877 \\ 0.6324 \end{bmatrix},$$

but the Matlab implementation seems to converge to

$$\mathbf{w} = \begin{bmatrix} 0.3106 \\ 0.0000 \\ 0.6895 \end{bmatrix}.$$

Let's explore this problem with the R package riskParityPortfolio.

3 Vanilla formulation

This problem is a vanilla formulation because it just contains the risk-parity term subject to the budget constraint and the no-shortselling constraint. Therefore, it can be reformulated as a convex problem and the global optimal solution can be obtained:

```r
library(riskParityPortfolio)

res <- riskParityPortfolio(Sigma, b = b)
res$w
#> [1] 0.27986280 0.08774909 0.63238811
res$risk_contribution/b
#> [1] 0.478381 0.478381 0.478381
```

4 Formulation "rc-over-var vs b"

Even though we really have a vanilla formulation, we can still consider a direct nonconvex formulation. Consider the risk expression:

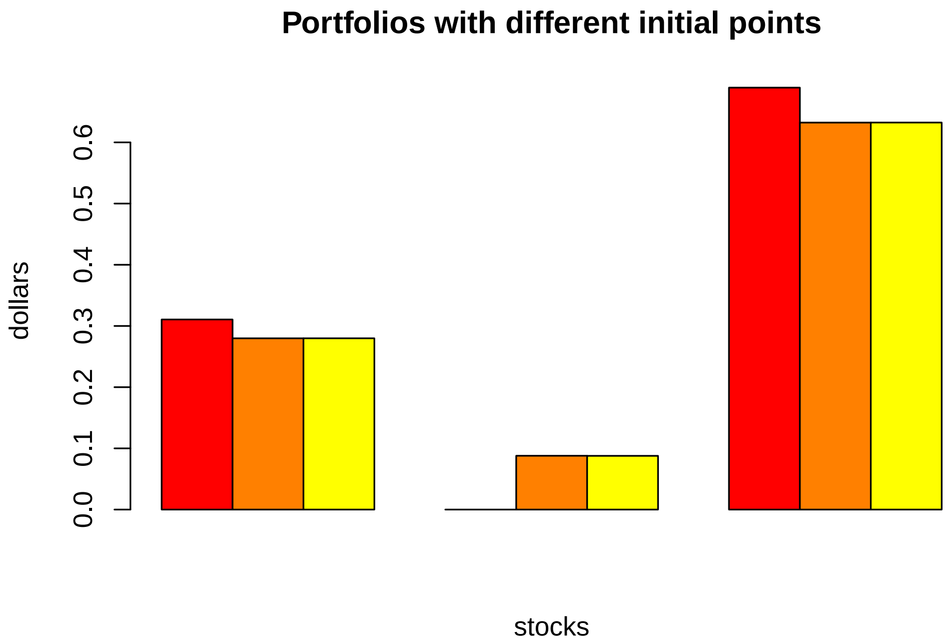
$$R(\mathbf{w}) = \sum_{i=1}^{N} \left(\frac{w_i \left(\boldsymbol{\Sigma}\mathbf{w}\right)_i}{\mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w}} - b_i \right)^2.$$

The general solver `alabama` is sensitive to the initial point in this formulation (the first case gets stuck in a local minimum):

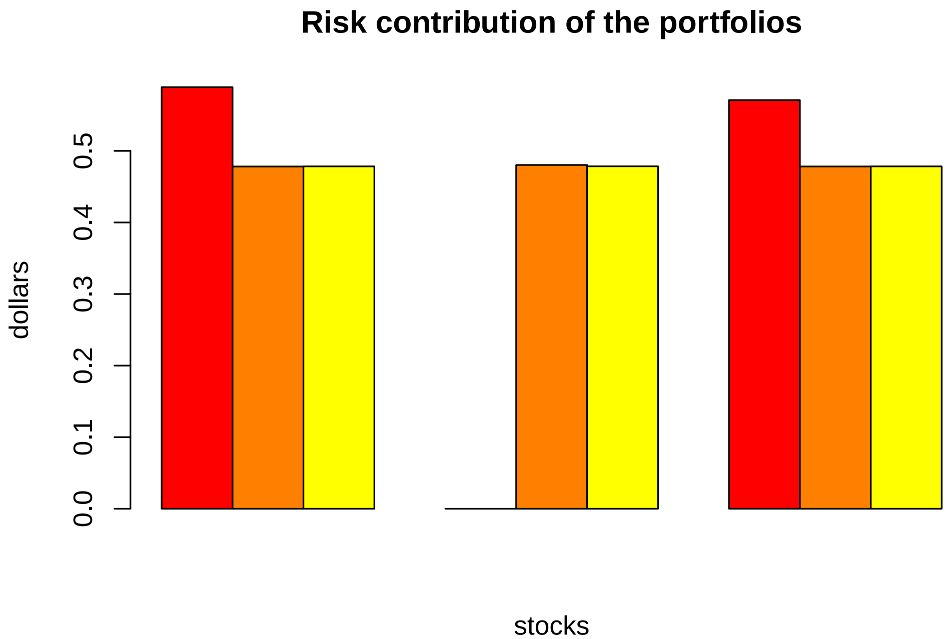
```r
set.seed(234)
res_ala1 <- riskParityPortfolio(Sigma, b = b, w0 = b,
                                formulation = "rc-over-var vs b",
                                method = "alabama")
#> Warning in riskParityPortfolio(Sigma, b = b, w0 = b, formulation = "rc-
#> over-var vs b", : The problem is a vanilla risk parity portofolio, but
#> a nonconvex formulation has been chosen. Consider not specifying the
#> formulation argument in order to get the guaranteed global solution.
res_ala1$w
#> [1] 3.105582e-01 5.543940e-08 6.894418e-01
```

```r
tail(res_ala1$obj_fun, 1)
#> [1] 0.0002381401

res_ala2 <- riskParityPortfolio(Sigma, b = b, w0 = c(1, 1, 1)/3,
                                formulation = "rc-over-var vs b",
                                method = "alabama")
#> Warning in riskParityPortfolio(Sigma, b = b, w0 = c(1, 1, 1)/3, formulation
#> = "rc-over-var vs b", : The problem is a vanilla risk parity portofolio,
#> but a nonconvex formulation has been chosen. Consider not specifying the
#> formulation argument in order to get the guaranteed global solution.
res_ala2$w
#> [1] 0.27980257 0.08789919 0.63229830
tail(res_ala2$obj_fun, 1)
#> [1] 4.202766e-09

res_ala3 <- riskParityPortfolio(Sigma, b = b, w0 = (w0 <- runif(3))/sum(w0),
                                formulation = "rc-over-var vs b",
                                method = "alabama")
#> Warning in riskParityPortfolio(Sigma, b = b, w0 = (w0 <- runif(3))/
#> sum(w0), : The problem is a vanilla risk parity portofolio, but a nonconvex
#> formulation has been chosen. Consider not specifying the formulation
#> argument in order to get the guaranteed global solution.
res_ala3$w
#> [1] 0.27985991 0.08775626 0.63238382
tail(res_ala3$obj_fun, 1)
#> [1] 9.590756e-12
```

```r
# plot the portfolios
barplot(rbind(res_ala1$w, res_ala2$w, res_ala3$w),
        main = "Portfolios with different initial points",
        xlab = "stocks", ylab = "dollars", beside = TRUE, col = heat.colors(3))
```



```r
# plot the risk contributions
barplot(rbind(res_ala1$risk_contribution/b, res_ala2$risk_contribution/b, res_ala3$risk_contribution/b),
        main = "Risk contribution of the portfolios",
        xlab = "stocks", ylab = "dollars", beside = TRUE,  col = heat.colors(3))
```

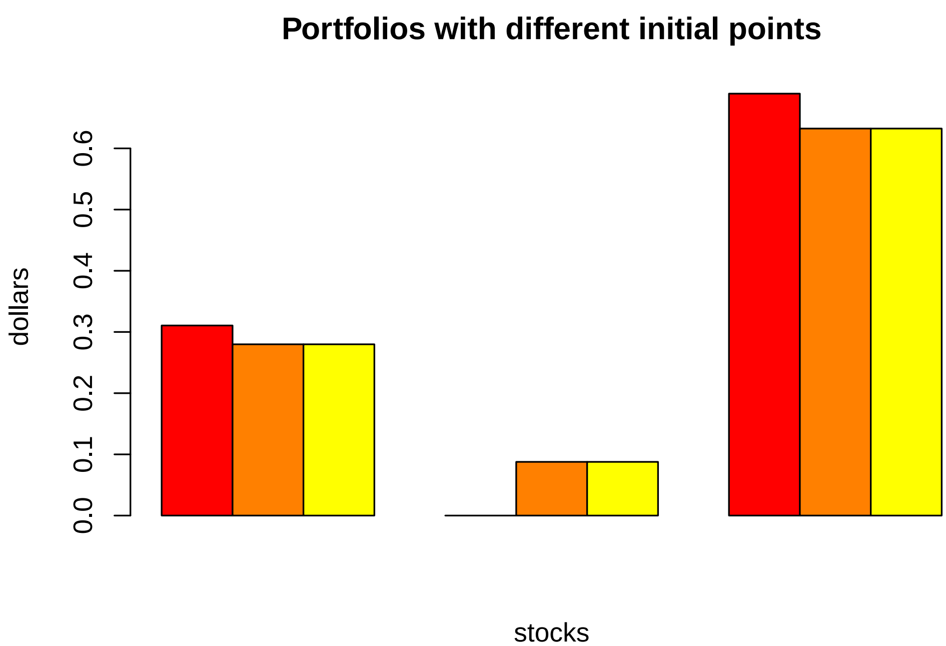


The SCA method gives the same results:

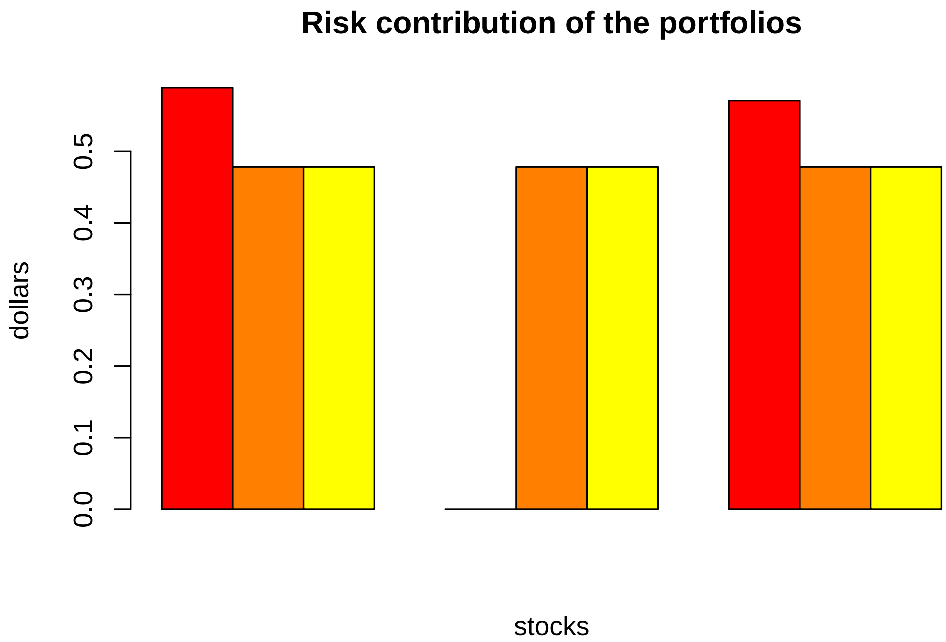
```r
set.seed(234)
res_sca1 <- riskParityPortfolio(Sigma, b = b, w0 = b,
                                formulation = "rc-over-var vs b",
                                method = "sca")
#> Warning in riskParityPortfolio(Sigma, b = b, w0 = b, formulation = "rc-
#> over-var vs b", : The problem is a vanilla risk parity portofolio, but
#> a nonconvex formulation has been chosen. Consider not specifying the
#> formulation argument in order to get the guaranteed global solution.
res_sca1$w
#> [1] 3.105589e-01 7.304669e-34 6.894411e-01
tail(res_sca1$obj_fun, 1)
#> [1] 0.00023814

res_sca2 <- riskParityPortfolio(Sigma, b = b, w0 = c(1, 1, 1)/3,
                                formulation = "rc-over-var vs b",
                                method = "sca")
#> Warning in riskParityPortfolio(Sigma, b = b, w0 = c(1, 1, 1)/3, formulation
#> = "rc-over-var vs b", : The problem is a vanilla risk parity portofolio,
#> but a nonconvex formulation has been chosen. Consider not specifying the
#> formulation argument in order to get the guaranteed global solution.
res_sca2$w
#> [1] 0.27986280 0.08774909 0.63238811
tail(res_sca2$obj_fun, 1)
#> [1] 1.733337e-33

res_sca3 <- riskParityPortfolio(Sigma, b = b, w0 = (w0 <- runif(3))/sum(w0),
```

```
                                 formulation = "rc-over-var vs b",
                                 method = "sca")
#> Warning in riskParityPortfolio(Sigma, b = b, w0 = (w0 <- runif(3))/
#> sum(w0), : The problem is a vanilla risk parity portofolio, but a nonconvex
#> formulation has been chosen. Consider not specifying the formulation
#> argument in order to get the guaranteed global solution.
res_sca3$w
#> [1] 0.27986280 0.08774909 0.63238811
tail(res_sca3$obj_fun, 1)
#> [1] 1.733337e-33
```

```
# plot the portfolios
barplot(rbind(res_sca1$w, res_sca2$w, res_sca3$w),
        main = "Portfolios with different initial points",
        xlab = "stocks", ylab = "dollars", beside = TRUE, col = heat.colors(3))
```




```
# plot the risk contributions
barplot(rbind(res_sca1$risk_contribution/b, res_sca2$risk_contribution/b, res_sca3$risk_contribution/b)
        main = "Risk contribution of the portfolios",
        xlab = "stocks", ylab = "dollars", beside = TRUE,  col = heat.colors(3))
```



5 Formulation "rc-over-sd vs b-times-sd"

Consider now the risk expression:

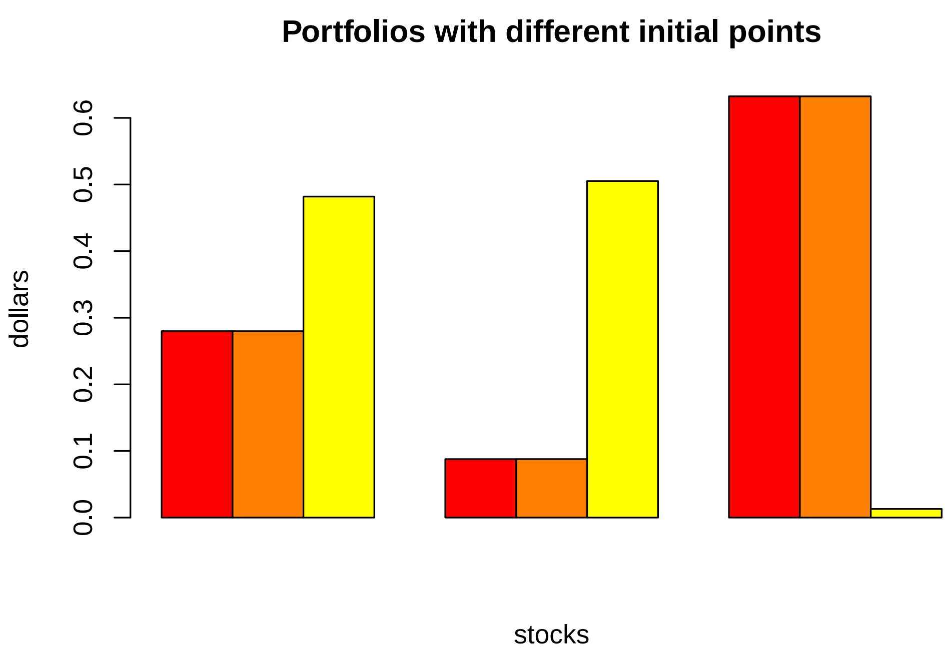
$$R(\mathbf{w}) = \sum_{i=1}^{N} \left(\frac{w_i \left(\mathbf{\Sigma w}\right)_i}{\sqrt{\mathbf{w}^T \mathbf{\Sigma w}}} - b_i \sqrt{\mathbf{w}^T \mathbf{\Sigma w}} \right)^2 = \sum_{i=1}^{N} \left(\frac{r_i}{\sqrt{\mathbf{1}^T \mathbf{r}}} - b_i \sqrt{\mathbf{1}^T \mathbf{r}} \right)^2.$$

The general solver `alabama` is again sensitive to the initial point in this formulation (the third case gets stuck in a local minimum):

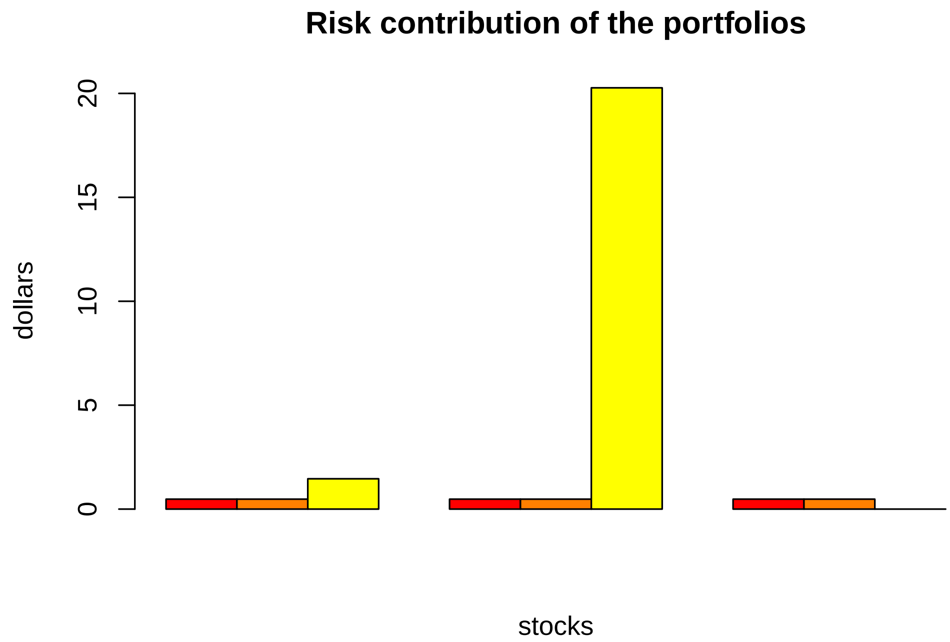
```
set.seed(234)
res_ala1 <- riskParityPortfolio(Sigma, b = b, w0 = b,
                                formulation = "rc-over-sd vs b-times-sd",
                                method = "alabama")
#> Warning in riskParityPortfolio(Sigma, b = b, w0 = b, formulation = "rc-
#> over-sd vs b-times-sd", : The problem is a vanilla risk parity portofolio,
#> but a nonconvex formulation has been chosen. Consider not specifying the
#> formulation argument in order to get the guaranteed global solution.
res_ala1$w
#> [1] 0.27986397 0.08774612 0.63238991
tail(res_ala1$obj_fun, 1)
#> [1] 7.827213e-13

res_ala2 <- riskParityPortfolio(Sigma, b = b, w0 = c(1, 1, 1)/3,
                                formulation = "rc-over-sd vs b-times-sd",
                                method = "alabama")
#> Warning in riskParityPortfolio(Sigma, b = b, w0 = c(1, 1, 1)/3, formulation
#> = "rc-over-sd vs b-times-sd", : The problem is a vanilla risk parity
#> portofolio, but a nonconvex formulation has been chosen. Consider not
#> specifying the formulation argument in order to get the guaranteed global
#> solution.
res_ala2$w
```

```r
#> [1] 0.27984754 0.08779533 0.63235712
tail(res_ala2$obj_fun, 1)
#> [1] 2.110713e-10

res_ala3 <- riskParityPortfolio(Sigma, b = b, w0 = (w0 <- runif(3))/sum(w0),
                                formulation = "rc-over-sd vs b-times-sd",
                                method = "alabama")
#> Warning in riskParityPortfolio(Sigma, b = b, w0 = (w0 <- runif(3))/
#> sum(w0), : The problem is a vanilla risk parity portofolio, but a nonconvex
#> formulation has been chosen. Consider not specifying the formulation
#> argument in order to get the guaranteed global solution.
res_ala3$w
#> [1] 0.48186292 0.50518794 0.01294915
tail(res_ala3$obj_fun, 1)
#> [1] 0.5110162
```

```r
# plot the portfolios
barplot(rbind(res_ala1$w, res_ala2$w, res_ala3$w),
        main = "Portfolios with different initial points",
        xlab = "stocks", ylab = "dollars", beside = TRUE, col = heat.colors(3))
```




```r
# plot the risk contributions
barplot(rbind(res_ala1$risk_contribution/b, res_ala2$risk_contribution/b, res_ala3$risk_contribution/b)
        main = "Risk contribution of the portfolios",
        xlab = "stocks", ylab = "dollars", beside = TRUE,  col = heat.colors(3))
```

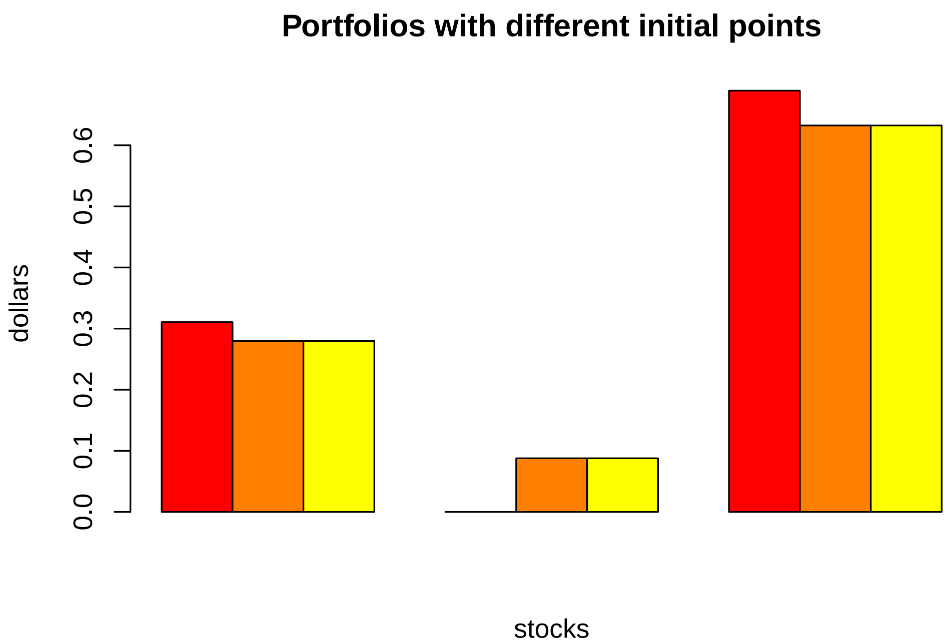


The SCA method is also sensitive to the initial point (the first case gets stuck in a local minimum):

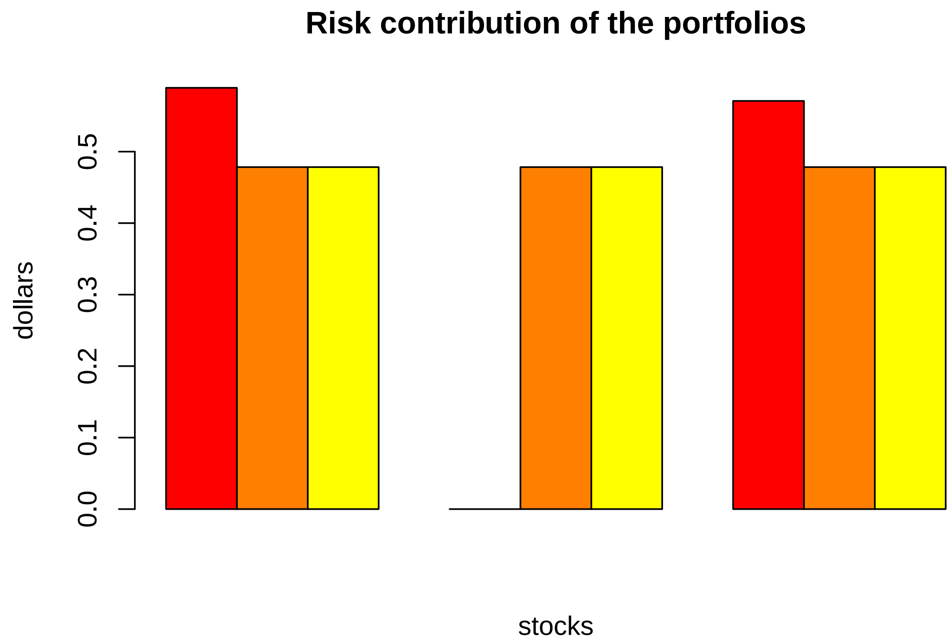
```
set.seed(234)
res_sca1 <- riskParityPortfolio(Sigma, b = b, w0 = b,
                                 formulation = "rc-over-sd vs b-times-sd",
                                 method = "sca")
#> Warning in riskParityPortfolio(Sigma, b = b, w0 = b, formulation = "rc-
#> over-sd vs b-times-sd", : The problem is a vanilla risk parity portofolio,
#> but a nonconvex formulation has been chosen. Consider not specifying the
#> formulation argument in order to get the guaranteed global solution.
res_sca1$w
#> [1] 3.106056e-01 1.181161e-19 6.893944e-01
tail(res_sca1$obj_fun, 1)
#> [1] 0.0001349449

res_sca2 <- riskParityPortfolio(Sigma, b = b, w0 = c(1, 1, 1)/3,
                                 formulation = "rc-over-sd vs b-times-sd",
                                 method = "sca")
#> Warning in riskParityPortfolio(Sigma, b = b, w0 = c(1, 1, 1)/3, formulation
#> = "rc-over-sd vs b-times-sd", : The problem is a vanilla risk parity
#> portofolio, but a nonconvex formulation has been chosen. Consider not
#> specifying the formulation argument in order to get the guaranteed global
#> solution.
res_sca2$w
#> [1] 0.27986280 0.08774909 0.63238811
tail(res_sca2$obj_fun, 1)
#> [1] 1.372225e-32

res_sca3 <- riskParityPortfolio(Sigma, b = b, w0 = (w0 <- runif(3))/sum(w0),
                                 formulation = "rc-over-sd vs b-times-sd",
                                 method = "sca")
#> Warning in riskParityPortfolio(Sigma, b = b, w0 = (w0 <- runif(3))/
#> sum(w0), : The problem is a vanilla risk parity portofolio, but a nonconvex
#> formulation has been chosen. Consider not specifying the formulation
```

```r
#> argument in order to get the guaranteed global solution.
res_sca3$w
#> [1] 0.27986280 0.08774909 0.63238811
tail(res_sca3$obj_fun, 1)
#> [1] 7.82409e-34
```

```r
# plot the portfolios
barplot(rbind(res_sca1$w, res_sca2$w, res_sca3$w),
        main = "Portfolios with different initial points",
        xlab = "stocks", ylab = "dollars", beside = TRUE, col = heat.colors(3))
```




```r
# plot the risk contributions
barplot(rbind(res_sca1$risk_contribution/b, res_sca2$risk_contribution/b, res_sca3$risk_contribution/b)
        main = "Risk contribution of the portfolios",
        xlab = "stocks", ylab = "dollars", beside = TRUE,  col = heat.colors(3))
```



6 Conclusion

The vanilla risk-parity portfolio is a convex problem and it can be solved optimally. However, if instead one uses a direct nonconvex formulation (which is required when having different constrains or additional objectives apart from the risk-parity one), there are local minima.

References

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

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