Gestion de Portefeuille

Ex 7: Risk Parity and Risk Budgeting

Version: 22 févr. 2024

1 Données

On utilisera les données de l'article de Litterman et He.

```
spl <- function (
  s,      # input string
  delim = ',' # delimiter
) {
  unlist(strsplit(s,delim))
}


 data =
'1,0.4880,0.4780,0.5150,0.4390,0.5120,0.4910
 0.4880,1,0.6640,0.6550,0.3100,0.6080,0.7790
 0.4780,0.6640,1,0.8610,0.3550,0.7830,0.6680
 0.5150,0.6550,0.8610,1,0.3540,0.7770,0.6530
 0.4390,0.3100,0.3550,0.3540,1,0.4050,0.3060
 0.5120,0.6080,0.7830,0.7770,0.4050,1,0.6520
 0.4910,0.7790,0.6680,0.6530,0.3060,0.6520,1'

  Corrmat = matrix( as.double(spl( gsub('\n', ',', data), ',')),
                    nrow = length(spl(data, '\n')), byrow=TRUE)

  stdevs = c(16.0, 20.3, 24.8, 27.1, 21.0,  20.0, 18.7)/100
  w.eq = c(1.6, 2.2, 5.2, 5.5, 11.6, 12.4, 61.5)/100
  # Prior covariance of returns
  Sigma = Corrmat * (stdevs %*% t(stdevs))
```

Rendements d'équilibre

```
# risk aversion parameter
delta = 2.5
Pi = delta * Sigma %*% w.eq
```

Assets	Std Dev	Weq	PI
Australia	16	1.6	3.9
Canada	20.3	2.2	6.9
France	24.8	5.2	8.4
Germany	27.1	5.5	9
Japan	21	11.6	4.3
UK	20	12.4	6.8
USA	18.7	61.5	7.6

2 Questions

2.1 Calculer une allocation telle que les contributions au risque du portefeuille sont identiques pour tous les titres (optimisation non-linéaire).

On doit avoir:

$$\min_w \sum_i (CR_i - CR_{i-1})^2$$

Définissez la fonction objectif et la matrice de contraintes, puis utilisez solnl pour obtenir la solution.

```
f.obj <- function(w) {
  CR <- w * Sigma %*% w
  sum(diff(CR) ** 2)*1000
}

n <- length(asset.names)
Aeq <- matrix(rep(1,n), nrow=1)
Beq <- 1
lb <- rep(0, n)
ub <- rep(1,n)
w.0 <- rep(1/n, n)

res.optim <- solnl(X=w.0, objfun=f.obj, Aeq=Aeq, Beq=Beq, lb=lb,
                   ub=ub)
```

Solution:

Table 1: Portefeuille Risk Parity

	weight
Australia	19.61
Canada	13.90
France	10.72
Germany	9.79
Japan	17.61
UK	13.42
USA	14.95

2.2 Solution analytique (méthode de Newton).

Condition nécessaire pour avoir des contributions homogènes au risque:

$$w_i \frac{\partial \sigma_P}{\partial w_i} = w_j \frac{\partial \sigma_P}{\partial w_j} = \lambda$$

Soit $1/w = [1/w_1, \ldots, 1/w_n]$, la condition s'exprime, en omettant le dénominateur:

$$\Sigma w = \lambda \times 1/w$$

On définit la fonction $F(w, \lambda)$:

$$F(w, \lambda) = \begin{bmatrix} \Sigma w - \lambda \times 1/w \\ 1^T w - 1 \end{bmatrix}$$

et on recherche w^*, λ^* tels que $F(w^*, \lambda^*) = 0$.

```r
f.obj <- function(x) {
  n <- length(x)
  w <- matrix(x[1:(n-1)], ncol=1)
  lambda <- x[n]
  res.1 <- Sigma %*% w - lambda * 1/w
  res.2 <- sum(w) - 1
  as.vector(rbind(res.1, res.2))
}

x.0 <- c(rep(1/n, n), .1)
res.newton = nleqslv(x.0, f.obj)
```

La solution est:

Table 2: Portefeuille Risk Parity (Newton)

	weight
Australia	19.61
Canada	13.90
France	10.72
Germany	9.79
Japan	17.61
UK	13.42
USA	14.96

2.3 Vérification avec la librairie RiskParityPortfolio

```r
library(riskParityPortfolio)
rpp_vanilla <- riskParityPortfolio(Sigma)
```

Table 3: Portefeuille Risk Parity

	weight
Australia	19.61
Canada	13.90
France	10.72
Germany	9.79
Japan	17.61
UK	13.42
USA	14.96