A Simple ALM Cash Flow Matching using Excel and R

Posted on September 26, 2021 by sang-heon lee in R bloggers | 0 Comments

This post implements a simple asset-liability cash flow matching using Excel and R. This example is a linear programming (LP) model but will serve as a foundation for the advanced ALM modeling techniques such as two-stage or multi-stage stochastic linear programming (2SSLP, MSSLP).

We borrow a basic asset-liability cash flow matching model as an example from the next book(pdf).

- Optimization Methods in Finance (Gerard Cornuejols and Reha Tütüncü)

LP model : Asset/Liability Cash Flow Matching

The page 47 of their book states a problem of financing short term cash commitments as follows.

source : Optimization Methods in Finance (Gerard Cornuejols and Reha Tütüncü)

During the 6 months, we should be funding short term capital to fulfill the cash flow requirements. Given that sources of funds have different interests and terms, we can set up a mathematical optimization problem for this work. We will use the company's wealth \(v\) in June as a objective function and the following decision variables:

- \(x_i\) : the line of credit in month i(=1,2,3,4,5)
- \(y_i\) : CP issued in month i(=1,2,3)
- \(z_i\) : the excess funds in month i(=1,2,3,4,5,6)
- \(z_6 = \nu\)

\[\begin{align} &\max_{x_i, y_i, z_i} \nu \\ s.t. &\\ &x_1 + y_1 \quad\quad\quad\quad\quad\quad\quad\quad\quad\quad – z_1 = 150 \\ &x_2 + y_2 \quad\quad – 1.01x_1 + 1.003z_1 – z_2 = 100 \\ &x_3 + y_3 \quad\quad – 1.01x_2 + 1.003z_2 – z_3 = -200 \\ &x_4 – 1.02y_1 – 1.01x_3 + 1.003z_3 – z_4 = 200 \\ &x_5 – 1.02y_2 – 1.01x_4 + 1.003z_4 – z_5 = -50 \\ & \quad – 1.02y_3 – 1.01x_5 + 1.003z_5 – \nu = -300 \\ \\ &x_i, y_i, z_i \ge 0, \quad x_i \le 100 \end{align}\]
We can solve this problem by using either Excel or R.

Excel

Making a table for given problem is followed by setting Excel Solver with initial guesses (yellow range of **F4:J4, F5:H5**).

The above table consists of **cash flow requirements**, **funding actions or instruments** (the line of credit and CP), **the associated cost and proceeds**, and finally **wealth or excess funds**. We add cost and proceeds from funding and excess funds. Excel formula will be explained later.

Given a table for optimization problem, Excel Solver is set as follows.

```
[Set Objective]
   $K$9

[To]
   Max

[By Changing Variable Cells]
   $F$4:$J$4,$F$5:$H$5

[Subject to the Constraints]
   $F$4:$J$4 <= $D$4
   $F$4:$J$4 >= $C$4
   $F$9:$K$9 >= $C$9

[Make Unconstrained Variables Non-Negative]
   Checked
```
Colored by Color Scripter

The answer for this problem is also presented in their book as follows.

Output from Excel Solver delivers the same result as their book's answer in the following way.

R code

Using **ROI, ROI.plugin.neos** R packages, we can solve the above linear optimization problem using NEOS server.

Refer the following post for more details of ROI, ROI.plugin.neos R package.

- Using NEOS Optimization Solver in R code

The following R code implements the same asset-liability cash flow matching problem with the help of ROI, ROI.plugin.neos.

```
#========================================================#
# Quantitative ALM, Financial Econometrics & Derivatives
# ML/DL using R, Python, Tensorflow by Sang-Heon Lee
#
# https://kiandlee.blogspot.com
#————————————————————————-#
# A simple Asset/Liability CF Matching problem
#========================================================#

graphics.off()  # clear all graphs
rm(list = ls()) # remove all files from your workspace

library(ROI)
library(ROI.plugin.neos)

# Gross Interest Rates
Rx = 1.01    # Xi : the line of credit
Ry = 1.02    # Yi : CP 90d
Rz = 1.003   # Zi : excess funds

# decision variables
# -> x1,2,3,4,5, y1,2,3, z1,2,3,4,5, v

# Left hand side matrix
v.LHS <- c(
    1,0,0,0,0,    1,0,0,   -1,0,0,0,0,  0,
   -Rx,1,0,0,0,    0,1,0,  Rz,-1,0,0,0,  0,
   0,-Rx,1,0,0,    0,0,1,  0,Rz,-1,0,0,  0,
   0,0,-Rx,1,0,  -Ry,0,0,  0,0,Rz,-1,0,  0,
   0,0,0,-Rx,1,  0,-Ry,0,  0,0,0,Rz,-1,  0,
   0,0,0,0,-Rx,  0,0,-Ry,  0,0,0,0,Rz,  -1
)
m.LHS <- matrix(v.LHS, nrow = 6, byrow = TRUE)

# v (14th decision variable) is the objective function
lp_obj <- L_objective(c(rep(0,13),1))

# LHS * X = RHS
lp_con <- L_constraint(
   L   = m.LHS,
   dir = rep("==", 6),
   rhs = c(150,100,-200,200,-50,-300))

# Lower & Upper bounds for decision variables
lp_bound <- V_bound(
   li = 1:14, ui = 1:14, lb = rep(0,14),
   ub = c(rep(100,5), rep(Inf,9)))
```


Recent Posts

```
# Set Problem
lp <- OP(objective = lp_obj,
         constraints = lp_con,
         bounds = lp_bound, maximum = TRUE)

# Solve Problem
opt <- ROI_solve(
    lp, solver = "neos",
    method = "mosek",
    email = "your email address")

# Print
cat("\nResult for simple A/L CF Matching \n\n",
    "Objective Function Value = ",
    round(opt$objval,4), "\n",
    "Decision  Variable Xi = ",
    round(opt$solution[1:5],4), "\n",
    "Decision  Variable Yi = ",
    round(opt$solution[6:9],4), "\n",
    "Decision  Variable Zi = ",
    round(opt$solution[10:14],4))
```

Colored by Color Scripter

We can easily find that results from R code is the same as the results of Excel and Cornuejols & Tütüncü's book.

```
 Result for simple A/L CF Matching

 Objective Function Value =  92.4969
 Decision  Variable Xi =  0 50.9804 0 0 0
 Decision  Variable Yi =  150 49.0196 203.4344 0
 Decision  Variable Zi =  0 351.9442 0 0 92.4969
```

Colored by Color Scripter cs

Concluding Remarks

From this post, we solve the simple ALM model but we think that it is not easy because this problem has many important aspects. Among them are flow and stock relationship, timing convention of cash flows, set-up for linear optimization problem and so on. Although this problem seems so simple, it has 14 decision variables which it is not small number.

Of course, using NEOS server for this problem may not be efficient because locally installed other R packages might deliver the fast and better performance. But when we deal with stochastic version of this kind of problem, the number of decision variables grows tremendously and the size of equations increases exponentially. In this case, the NEOS server is a good choice when we do not have an expensive commercial solver. \(\blacksquare\)

← **Previous post** **Next post** →

String pad to the column in R
5 New books added to Big Book of R
Finding Happiness in 'The Smoke'
Time for a new workshop series!
Bootstrap Confidence Interval R
Using R to Win Worldle
Call for talks deadline extended!
nanonext – how it provides a concurrency framework for R
rOpenSci News Digest, March 2022
How to remove Scientific Notation in R
Network Visualizations of Code Collections (funspotr part 3)
{tvthemes 1.3.0} is on CRAN: 'Steven Universe'-themed color palettes for ggplot2!
UPDATE: Successful R-based Test Package Submitted to FDA
How to Add Text to a Plot in R
How to calculate proportion in a table

Jobs for R-users

Junior Data Scientist / Quantitative economist
Senior Quantitative Analyst
R programmer
Data Scientist – CGIAR Excellence in Agronomy (Ref No: DDG-R4D/DS/1/CG/EA/06/20)
Data Analytics Auditor, Future of Audit Lead @ London or Newcastle

python-bloggers.com (python/data-science news)

Explaining a Keras _neural_ network predictions with the-teller
Object Oriented Programming in Python – What and Why?
Dunn Index for K-Means Clustering Evaluation
Installing Python and Tensorflow with Jupyter Notebook Configurations
How to Get Twitter Data using Python
Visualizations with Altair
Spelling Corrector Program in Python

Full list of contributing R-bloggers

Archives

Select Month

Other sites

Jobs for R-users
SAS blogs