**Cryptocurrency Portfolio Optimization**

Rohit Garg ([f2005636@gmail.com](mailto:f2005636@gmail.com))

**Cryptocurrencies**

A cryptocurrency a digital asset designed to work as a medium of exchange that uses cryptography to secure its transactions. Cryptocurrencies are classified as a subset of digital currencies and are also classified as a subset of alternative currencies and virtual currencies.

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Mean Variance Model**

For the mathematical formulation of Mean-Variance model by Markowitz, we need some definitions. They are explained as follows:

* By asset Xi, we mean a random variable with finite variance.
* By portfolio, we mean the combination of assets: P =­ ΣwiXi , where Σ­wi = 1
* By optimization, we mean a process of choosing the best wi coefficients (weights) so that our portfolio meets our needs (that is, it has a minimal risk on the given expected return or has the highest expected return on a given level of risk).

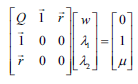
Minimum variance on a desired level of return: It is clear that wTQw is the variance of the portfolio and wr is the expected return. For the sum of the weights we have w1 =1, which means that we would like to invest 1 unit of cash.

**Equation 1: Minimum Variance**



It can be shown that this problem is equivalent to the following system of linear equations. Two rows and two columns are added to the covariance matrix, so we have conditions to determine the two Lagrange multipliers as well. We can expect a unique solution for this system.

**Equation 2: Linear Equation**



assets = IT[, -1]

return = log(tail(assets, -1) / head(assets, -1))

head(return)

tail(return)

<check returns manually>

Q = rbind(cov(return), rep(1, ncol(assets)), colMeans(return))

Q = cbind(Q, rbind(t(tail(Q, 2)), matrix(0, 2, 2)))

round(Q, 5)

mu = 0.005

b = c(rep(0, ncol(assets)), 1, mu)

round(b, 5)

w = solve(Q, b)

<I get weights here>

cr = cov(return)

nr = ncol(assets)

r = colMeans(return)

rbase = seq(min(r), max(r), length = 100)

s = sapply(rbase, function(x) {

y = head(solve(Q, c(rep(0, n), 1, x)), n)

y %\*% cr %\*% y

})

plot(s, rbase, xlab = 'Return', ylab = 'Variance')

plot(rbase, s, xlab = 'Variance', ylab = 'Return')

<I get efficient frontier here>

On the variance-return plane, the desired return-minimum variance curve is called Portfolio Frontier. Ignoring its downward sloping part, we get Efficient Frontier.

**Tangency Portfolio and Capital Market Line**

If R (riskless asset) is added to X (any risky portfolio); then those portfolios form a straight line on the mean-standard deviation plane. Any portfolio on this line is available by investing into R and X. It is clear that the best choice for X is the point where this line is tangent to Efficient Frontier. This tangency point is called the market portfolio or tangency portfolio, and the tangent of Efficient Frontier of risky assets at this point is called Capital Market Line (CML), which consists of the efficient portfolios of all the assets in this case.

n = 6

mu = 0.005

Q = cbind(cov(return), rep(0, n - 1))

Q = rbind(Q, rep(0, n))

rf = 0.0001

r = c(colMeans(return), rf)

Q = rbind(Q, rep(1, n), r)

Q = cbind(Q, rbind(t(tail(Q, 2)), matrix(0, 2, 2)))

b = c(rep(0, n), 1, mu)

round(Q, 6)

b

w = solve(Q, b)

w = head(w, -3)

w / sum(w)

w

<i get the weights>