Calculating the Efficient Frontier: Part 1

A Matrix Based Example of Mean-Variance Optimization using Octave

The concept of an "efficient frontier" was developed by Harry Markowitz in the 1950s. The efficient frontier shows us the minimum risk (i.e. standard deviation) that can be achieved at each level of expected return for a given set of risky securities.

Of course, to calculate the efficient frontier, we need to have an estimate of the expected returns and the covariance matrix for the set of risky securities which will used to build the optimal portfolio. These parameters are difficult (impossible) to forecast, and the optimal portfolio calculation is extremely sensitive to these parameters. For this reason, an efficient frontier based portfolio is difficult to successfully implement in practice. However, a familiarity with the concept is still very useful and will help to develop intuition about diversification and the relationship between risk and return.

Calculating the Efficient Frontier

In this post, I'll demonstrate how to calculate and plot the efficient frontier using the expected returns and covariance matrix for a set of securities.

In a future post, I'll demonstrate how to calculate the security weights for various points on this efficient frontier using the two-fund separation theorem.

Calculations

In order to calculate the efficient frontier using n assets, we need two inputs. First, we need the expected returns of each asset. The vector of expected returns will be designated $\bar{\mathbf{z}}$. The second input is the variance-covariance matrix for the n assets. This covariance matrix will be designated as $\mathbf{S}$. We also need a unity vector ($\mathbf{1}$) with the same length as the vector $\bar{\mathbf{z}}$.

Once we have this information, we can run the following calculations using a matrix based mathematical program such as Octave or Matlab.

$$A = \mathbf{1}'\mathbf{S}^{-1}\mathbf{1} > 0$$

$$B = \mathbf{1}'\mathbf{S}^{-1}\bar{\mathbf{z}}$$

$$C = \bar{\mathbf{z}}'\mathbf{S}^{-1}\bar{\mathbf{z}}$$

$$\Delta = AC - B^2 > 0$$

Using these values, the variance (σ^2) at each level of expected return (μ) is given by this equation:

$$\sigma^2 = \frac{A\mu^2 - 2B\mu + C}{\Delta}$$

You can see from the equation, that the efficient frontier is a parabola in mean-variance space.

Using the standard deviation (σ) rather than the variance, we have:

$$\sigma = \sqrt{\frac{A\mu^2 - 2B\mu + C}{\Delta}}$$

Example using Octave Script

As an example, lets consider four securities, A,B,C and D, with expected returns of 14%, 12%, 15%, and 7%. The expected return vector is:

$$\bar{\mathbf{z}} = \begin{bmatrix} 14 \\ 12 \\ 15 \\ 7 \end{bmatrix}$$

The covariance matrix for our example is shown below. In practice, the historical covariance matrix can be calculated by reading the historical returns into Octave or Matlab and using the *cov(X)* command. Note that the diagonal of the matrix is the variance of our four securities. So, if we take the square root of the diagonal, we can calculate the standard deviation of each asset (13.6%, 14%, 20.27%, and 5%).

$$\mathbf{S} = \begin{bmatrix} 185 & 86.5 & 80 & 20 \\ 86.5 & 196 & 76 & 13.5 \\ 80 & 76 & 411 & -19 \\ 20 & 13.5 & -19 & 25 \end{bmatrix}$$

The example script for computing the efficient frontier from these inputs is shown at the end of this post. It can be modified for any number of assets by updating the expected return vector and the covariance matrix.

The plot of the efficient frontier for our four assets is shown here:

Efficient Frontier with Individual Securities
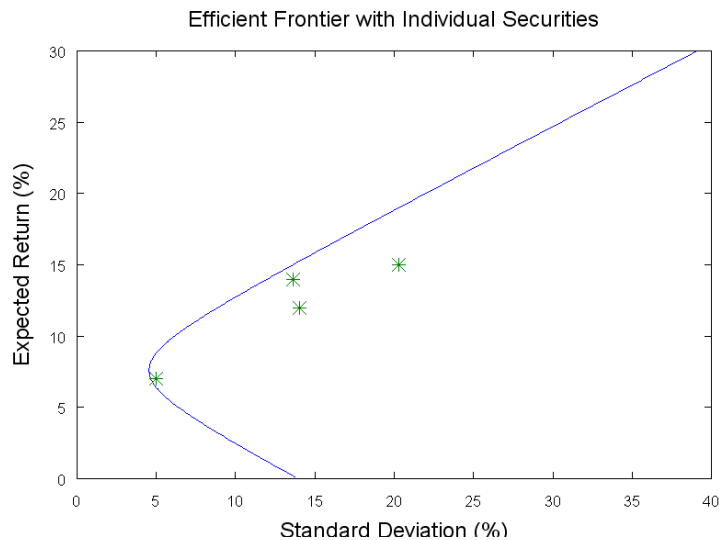
Derivation and References

Deriving the approach I have shown is beyond the scope of this post. However, for those who want to dive into the linear algebra, there are several excellent examples available online.

Derivation of Mean-Variance Frontier Equation using the Lagrangian (The Appendix B result is identical to what I show above, but the notation is a little different)

Old school derivation by a young professor who later went on to win a Nobel prize

Octave Code:

This script will also work in Matlab, but I've chosen to use Octave since it is opensource and available for free.

```octave
1   % Mean Variance Optimizer
2
3   % S is matrix of security covariances
4   S = [185 86.5 80 20; 86.5 196 76 13.5; 80 76 411 -19; 20 13.5 -19
5
6   % Vector of security expected returns
7   zbar = [14; 12; 15; 7]
8
9   % Unity vector..must have same length as zbar
10  unity = ones(length(zbar),1)
11
12  % Vector of security standard deviations
13  stdevs = sqrt(diag(S))
14
15  % Calculate Efficient Frontier
16  A = unity'*S^-1*unity
17  B = unity'*S^-1*zbar
18  C = zbar'*S^-1*zbar
19  D = A*C-B^2
20
21  % Efficient Frontier
22  mu = (1:300)/10;
23
24  % Plot Efficient Frontier
25  minvar = ((A*mu.^2)-2*B*mu+C)/D;
26  minstd = sqrt(minvar);
27
28  plot(minstd,mu,stdevs,zbar,'*')
29  title('Efficient Frontier with Individual Securities','fontsize',
30  ylabel('Expected Return (%)','fontsize',18)
31  xlabel('Standard Deviation (%)','fontsize',18)
```