

A Sequential Linear Programming Algorithm to Portfolio Optimization

MSCI 331

Operations Research 1: Introduction to Optimization

Fall 2006

Richard Hui, Rajesh Swaminathan, Sarah Vandaiyar

December 5, 2006

Contents

1	Executive Summary	1
2	Discussion	2
2.1	Problem Description	2
2.2	Mathematical Model	2
2.2.1	Analysis	3
2.3	Solution Approach	4
2.4	Implementation, Savings and Benefits	5
2.5	Computational Experiment	5
3	Conclusions	7
	Appendices	9
A	The MATLAB Code	9
B	Mini-Case Data	12

1 Executive Summary

In asset-liability management applications (ALM), the key objective is to optimize portfolio investments by either maximizing return or minimizing risk. Although risk is widely represented as a non-linear function (usually quadratic), portfolio managers traditionally need to work with linear risk functions to ensure “computational tractability”. The paper *GE Asset Management, Genworth Financial and GE Insurance Use a Sequential-Linear-Programming Algorithm to Optimize Portfolios* [1] introduces a *sequential linear programming* (SLP) algorithm that accounts for the non-linearity associated with risk measurement.

The algorithm approximates the efficient frontier—a relationship between risk and return—by first generating the highest possible return without considering risk, and solving the resulting LP relaxation. The next step is to generate another portfolio with a slightly smaller risk and return value. This is done by approximating the non-linear objective with a linear function tangent at the solution obtained in the previous step. A linear constraint is added to the LP to restrict the objective function to below a new target risk value. The solution to the resulting new LP generates the next portfolio. This process is repeated until the efficient frontier is adequately represented.

For our computational experiment, we formulated a hypothetical portfolio consisting of three stocks: Gap Inc., Ballard Power Systems, and Research in Motion. The algorithm was coded in MATLAB and then used to approximate the efficient frontier.

The algorithm has proven to be suitable for large-scale asset-liability optimizations involving thousands of securities. With this new tool, GE has been able to run the optimization multiple times as market variables change, rather than only when it is clear that analysis is needed.

2 Discussion

2.1 Problem Description

GE financialists used a multifactor-based portfolio optimization technique, wherein the risk is quadratic. Because most practical applications of this model require selecting thousands of securities, solving ALM optimization problems was beyond the computational limits of commercial quadratic solvers.

ALM portfolio managers have generally used LP approximations to simplify calculations. Although the objective and constraints in this simplified model are linear, the model still has some shortcomings. First, one cannot determine a meaningful efficient frontier because there is not one measure of risk. Second, the risk measures capture only the severity aspect of the portfolio risk and not the frequency, so that the risk is not fully characterized. Finally, computing the mismatch targets involves a trial-and-error process such that different managers will obtain different final portfolios.

2.2 Mathematical Model

The model developed by GE is based on modern portfolio theory proposed by Harry Markowitz [2], where the goal of portfolio optimization is to manage risk through diversification and obtain an optimal risk-return trade-off. This means that an investment opportunity can be meaningfully measured in terms of mean return and variance of risk.

The objective in using a mean-variance approach is to choose an efficient portfolio by offering the highest expected return for a given level of risk as measured by variance or standard deviation of returns.

To determine the minimum-variance frontier for a set of n assets, we must first determine the minimum and maximum expected returns possible. Then, we must determine the portfolio weights that will create the minimum variance for values of expected return z between the maximum (r_{max}) and minimum returns (r_{min}) of all possible portfolios. The **linear programming** (LP) representation of the portfolio optimization problem is:

$$\begin{aligned} \text{MIN} \quad & \sigma_p^2 = \sum_{i=1}^n \sum_{j=1}^n w_i w_j \text{Cov}(R_i, R_j) \\ \text{s.t.} \quad & E(R_p) = \sum_{j=1}^n w_j E(R_j) = z \\ & \sum_{j=1}^n w_j = 1 \\ & w_j \geq 0, \text{ for } j \in \{1, 2, 3, \dots, n\} \end{aligned}$$

where $E(R_p)$ represents the expected return of the portfolio, $E(R_j)$ is the expected return of the j^{th} asset, w_j is the weighting of each asset, and $\text{Cov}(R_i, R_j)$ is the covariance between two returns.

2.2.1 Analysis

The model developed by GE is based on Markowitz's mean-variance analysis is based on the following assumptions:

1. All investors are risk averse; they prefer less risk for the same level of expected return.
2. The expected returns, variances, and covariances of all assets are known.
3. Investors can ignore skewness and kurtosis of the distribution.
4. There are no transaction costs or taxes.

Strengths of the model include the ability of the investor to optimize a given portfolio. Because this model assumes that risk is a non-linear function, optimization poses limits on solvers because of the complexity of the problem. Furthermore, the mathematical representation of risk is still in its early stages of development. Most importantly, the model does not consider other risk factors besides market risk.

Today, portfolio managers are studying LP approximations to simplify the optimization problem associated with this approach.

2.3 Solution Approach

The algorithm computes the first portfolio on the efficient frontier that yields the highest possible return, ignoring risk. The next approximated portfolio has a smaller risk and expected return value. The non-linear risk function is approximated by a linear tangent centered around the solution obtained in the previous iteration. A new constraint is added to restrict allowed risk values to those less than the value obtained in the previous iteration. The algorithm iterates until the portfolio with the smallest possible risk/return has been reached. Each iteration computes the optimum portfolio for a given return which decreases by a step size ϵ each iteration. The variable ϵ dictates the run-time of the algorithm and the accuracy of the linear approximation. In the limit that ϵ goes to zero, the frontier will converge to the true efficient frontier (for a convex risk function). However, the code would then run for an infinitely long time.

Another approach would be to start with the lowest possible return and then iterate the algorithm with incremental gains in return. Our mini-case produces the efficient frontier in this manner.

2.4 Implementation, Savings and Benefits

The algorithm was implemented in MATLAB and distributed as a web-based application over GE's intranet. Once registered, portfolio managers had access to a variety of tools and functions with a standard web browser. Hence, there was no need to install any software on local machines. Built on the Java platform via a connector plugin called JMatLink, the web application prompts for user input and displays optimization results.

Since the implementation of the SLP algorithm in 2003, more than 30 portfolios totaling over \$30 billion have been optimized using this approach. For \$100 billion of assets, the present value (as of 2005) of benefits is approximated to be \$75 million or 0.075% over 5 years.

GE's SLP algorithm has proven to be suitable for large-scale asset-liability optimizations involving thousands of securities. The SLP approximates the efficient frontier in minutes where traditional commercial QP solvers would take hours. With this new tool, GE has been able to pro-actively run the optimization multiple times as market variables change.

2.5 Computational Experiment

For our mini-case, we chose three contemporary stocks from three different industry sectors. Our three companies were Gap Inc. (retail), RIM (technology), and Ballard (energy). The problem was to obtain the minimum-variance frontier based on historical data. Then, given a desired risk tolerance characterized by variance (or equivalently, standard deviation), we want to determine the optimal fraction of investments in each stock to obtain the highest expected return.

We obtained closing prices of these three stock prices over the last three months. This data has been plotted in Figure 2 (Appendix B) and summarized in Table 1. A more refined

version of the problem can be stated as follows: Given x dollars and an acceptable risk tolerance $y\%$ in terms of standard deviation, what are the best weights w_1 , w_2 and w_3 that determine what fraction of x should be invested in each stock respectively?

	GAP	RIM	Ballard
Mean Daily Returns	18.70	113.10	6.21
Standard Sample Deviation	1.341	21.151	0.645
<i>Correlations</i>			
GAP	1.000		
RIM	0.883	1.000	
Ballard	0.455	0.683	1.000

Table 1: Summary of daily stock closing prices from 17 Aug. – 17 Nov. 2006.

Although it might seem on first insight that investing all money in RIM is the ideal solution (since RIM has the highest mean return), this is only true if one is willing to take the maximum risk. The advantages of diversification have been studied extensively and in general, diversification leads to the same expected return but with lower risk. This is especially true when including into the portfolio stocks that have a correlation factor less than one.

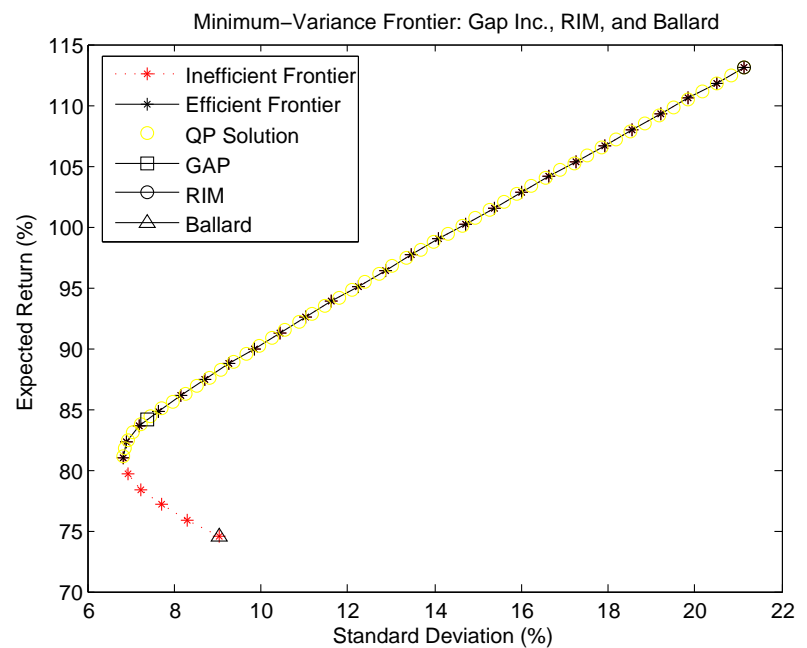


Figure 1: Minimum Variance Frontier for Mini-Case

We used MATLAB to iterate between levels of expected return between the minimum and maximum levels. For each given expected return, we used a first order Taylor approximation of the risk function centered around the weights obtained in the previous iteration to obtain the portfolio weights that resulted in the lowest risk for that level of expected return. The

coded algorithm repeats this process for each level of average return as we move from the minimum level ($r_{min} = 6.21$) to the maximum level ($r_{max} = 113.1$) to obtain the allocation weights that correspond to the least risk for that return. We performed 30 iterations in increments of 3.5% of average return.

The minimum-variance frontier is plotted in Figure 1. The MATLAB code is viewable in Appendix A of this document.

3 Conclusions

Our study of the SLP algorithm has proven time savings that translate to economic savings as a result of a simplification of Markowitz's mean-variance analysis. The simplification has resulted in decreasing run-time from exponential-time to polynomial-time with respect to the number of assets.

The mini-case problem was useful in our theoretical understanding of LP problems, as presented in MSCI 331. Furthermore, we were able to integrate fundamental OR principles to real-world applications.

References

- [1] Chalermkraivuth et al. “Sequential-Linear-Programming Algorithm to Optimize Portfolios.” *Interfaces* 35(5), pp. 370-380. Informs, 2005.
- [2] Markowitz, H. “Portfolio Finance.” *J. Finance* 7(1) pp. 77-91. 1952.
- [3] DeFusco et al. “Quantitative Methods For Investment Analysis.” 2nd Ed. CFA Institute, 2005.

Appendices

A The MATLAB Code

```
1 % SLP Algorithm for solving the PORTFOLIO OPTIMIZATION problem
2 % Rajesh Swaminathan
3 % MSCI 331 - #20194189
4 %
5 % Created : 11/18/2006
6
7 % clear all variables
8 clear
9
10 % specify the max. num of slp iterations
11 num_iterations = 30;
12
13 % define data
14
15 returns = [84.15 113.10 74.56];
16 risks = [7.377 21.151 9.032];
17 corrs = [1.000 0.883 0.455
18          0.883 1.000 0.683
19          0.455 0.683 1.000];
20
21 [rmin,i] = min(returns);
22 rmax = max(returns);
23
24 eps = (rmax - rmin) / num_iterations;
25 r = rmin;
26
27 for j = 1:3
28     if i == j
29         w(j) = 1.00;
30     else
31         w(j) = 0.00;
32     end
33 end
34
35 prisks(1) = risk(returns, risks, corrs, w);
36 preturns(1) = ereturn(returns, w);
```

```

37
38 % initialize the counter
39 k = 1;
40
41 while (r <= rmax)
42
43     % increment the counter
44     k = k + 1;
45
46     % increase the return by a fixed amount eps
47     r = r + eps;
48
49     % define equality and inequality constraints
50
51     % w1 >= 0 (non-negativity)
52     % w2 >= 0
53     % w3 >= 0
54     A = [-1  0  0;
55          0 -1  0;
56          0  0 -1];
57
58     % w1 + w2 + w3 = 1
59     % r1*w1 + r2*w2 + r3*w3 = r
60     Aeq = [[1 1 1]; returns];
61
62     b = [0; 0; 0];
63     beq = [1; r];
64
65     % compute the coefficients of w1,w2,w3 in the 1st order
66     % taylor about w(1), w(2), and w(3)
67
68     f(1) = 2*risks(1)^2*w(1) + 2*w(2)*risks(1)*risks(2);
69     f(1) = f(1) + 2*w(3)*risks(1)*risks(3);
70     f(2) = 2*risks(2)^2*w(2) + 2*w(1)*risks(2)*risks(1);
71     f(2) = f(2) + 2*w(3)*risks(2)*risks(3);
72     f(3) = 2*risks(3)^2*w(3) + 2*w(1)*risks(3)*risks(1);
73     f(3) = f(3) + 2*w(2)*risks(3)*risks(2);
74
75     w = linprog(f, A, b, Aeq, beq);
76
77     prisks(k) = risk(returns, risks, corrs, w);
78     preturns(k) = ereturn(returns, w');

```

```

79
80 end
81 %end while loop
82
83 plot(prisks, pretreturns, 'r*');
84 hold on
85
86 % replot only efficient frontier
87 x = [];
88 y = [];
89 [i,j] = min(prisks);
90
91 for i = 1:length(prisks)
92     if pretreturns(i) >= pretreturns(j);
93         x = [x prisks(i)];
94         y = [y pretreturns(i)];
95     end
96 end
97
98 plot(x,y,'k*-');
99
100 % plot from qp solver
101 [prisks, pretreturns, pweights] = ...
102 frontcon(returns, corr2cov(risks, corrs), num_iterations+20);
103 plot(prisks, pretreturns, 'yo');
104
105 % investment in one stock only
106 ws = [1,0,0
107       0,1,0
108       0,0,1];
109 shapes = ['s', 'o', '^'];
110 for i = 1:3
111     x = risk(returns, risks, corrs, ws(i,:));
112     y = ereturn(returns, ws(i,:));
113     plot(x, y, ['k-' shapes(i)]);
114 end
115
116 ylabel('Expected Return (%)');
117 xlabel('Standard Deviation (%)');
118 title('Minimum-Variance Frontier: Gap Inc., RIM, and Ballard');
119 legend({'Inefficient Frontier','Efficient Frontier','QP Solution', ...
120        'GAP','RIM','Ballard'},'Location','NorthWest');

```

B Mini-Case Data

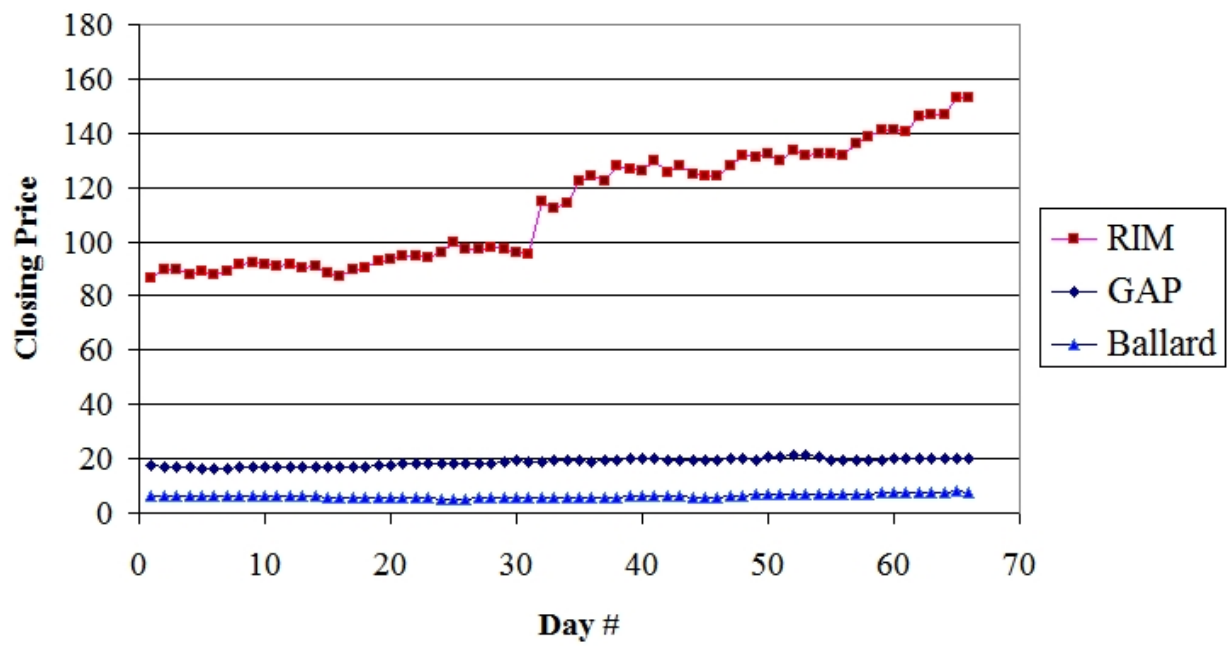


Figure 2: Closing price summary of GAP, RIM and Ballard stocks.

Day	Closing Price			Day	Closing Price		
	GAP	RIM	Ballard		GAP	RIM	Ballard
1	17.30	86.69	6.19	34	19.44	114.40	5.48
2	16.65	89.72	6.21	35	19.26	122.18	5.64
3	16.87	90.00	6.09	36	19.07	123.89	5.62
4	16.80	87.92	6.22	37	19.25	122.50	5.54
5	16.52	89.00	6.14	38	19.62	127.65	5.61
6	16.57	88.00	6.18	39	19.81	126.73	6.10
7	16.61	89.08	6.11	40	19.81	126.19	6.14
8	16.85	91.50	6.08	41	19.85	129.58	6.04
9	16.96	92.00	6.12	42	19.49	125.65	6.03
10	17.22	91.51	6.24	43	19.46	128.00	5.98
11	16.81	91.13	6.17	44	19.56	125.05	5.91
12	16.90	91.27	6.09	45	19.47	124.42	5.92
13	16.75	90.50	6.09	46	19.30	123.94	5.91
14	16.92	91.18	6.01	47	20.06	127.79	6.49
15	16.73	88.38	5.94	48	19.94	131.81	6.29
16	16.90	87.21	5.87	49	19.73	131.00	7.09
17	16.90	89.58	5.91	50	20.51	132.24	7.06
18	17.08	90.05	5.92	51	20.72	129.93	6.86
19	17.48	92.68	5.83	52	21.09	133.31	6.95
20	17.68	93.46	5.80	53	21.02	131.90	7.07
21	17.92	94.61	5.64	54	20.64	132.41	6.81
22	18.21	94.76	5.85	55	19.50	132.05	6.67
23	18.16	93.97	5.52	56	19.57	131.48	6.72
24	18.34	95.70	5.32	57	19.50	135.98	6.70
25	18.16	99.97	5.21	58	19.49	138.55	6.67
26	18.39	97.50	5.27	59	19.48	141.25	7.25
27	18.47	96.96	5.34	60	20.02	140.81	7.25
28	18.50	98.05	5.44	61	20.19	140.61	7.26
29	19.07	97.51	5.76	62	19.85	146.11	7.39
30	19.17	95.99	5.94	63	19.93	146.56	7.40
31	18.95	95.30	5.69	64	20.02	146.48	7.60
32	18.93	114.59	5.57	65	19.80	153.12	7.88
33	19.21	112.00	5.40	66	19.81	153.07	7.59

Table 2: Summary of daily stock closing prices from 17 Aug. – 17 Nov. 2006.