Risk & Asset Allocation (Spring)
Case for Week 4

John A. Dodson

February 11, 2015

1 Efficient Frontier

With the objective $\Psi_\alpha = \alpha' M$ for market vector $M = P - p$ (profit objective) with initial price vector p (dollars per share) and portfolio allocation α (shares), the efficient frontier is defined by

$$\alpha(m) = \arg \min_{\substack{\alpha \in \mathcal{C} \\ \alpha' \operatorname{E} M \geq m}} \alpha' \left(\operatorname{cov} M\right) \alpha$$

which is parameterized by m.

2 Quadratic Programming

A class of convex optimization problems that can be readily solved with polynomial-scale algorithms is termed quadratic programming,

$$\min_{Ax \leq b} x' H x$$

for $H \geq 0$ (positive semi-definite) and (not necessarily square) matrix A and corresponding vector b.

3 Constraints

Since covariance is positive semi-definite, as long as the constraints are affine, the efficient frontier problem is an example of quadratic programming.

Common constraints that can be expressed in this form, in addition to the minimum return constraint above, include the wealth constraint $p'\alpha \leq w$ with initial wealth w, and a shorting prohibition $\alpha_i \geq 0$.

It is also possible to represent a gross leverage constraint, although this is a little more challenging because we need to introduce dummy variables. Let

$$x = \alpha \mathbin{+\!\!+} \bar{\alpha}$$

where $\bar{\alpha}$ is the same length (say N) as α. $\bar{\alpha}$ will not enter the objective function, so we will define

$$H = \begin{pmatrix} \operatorname{cov} M & Z \\ Z & Z \end{pmatrix}$$

where Z is a square matrix of zeros. The wealth constrain is simple enough: $(p \# z)' x \le w$ where z is a vector of zeros. The reason we need $\bar{\alpha}$, is so that we can set up constraints of the form $\bar{\alpha}_i \ge \alpha_i$ and $\bar{\alpha}_i \ge -\alpha_i$, such that if one is binding then $\bar{\alpha}_i = |\alpha_i|$. We can then express a gross leverage constraint by $(z \# p)' x \le g\,w$, where $g \ge 1$ is the leverage limit. So all together, the $2N + 3$ constraints can be represented as

$$A = \begin{pmatrix} -\frac{w}{m}\left(\mathrm{E}\, M \# z\right)' \\ (p \# z)' \\ (z \# p)' \\ \operatorname{diag} p \# -\operatorname{diag} p \\ -\operatorname{diag} p \# -\operatorname{diag} p \end{pmatrix} \qquad b = w \begin{pmatrix} -1 \\ 1 \\ g \\ z \\ z \end{pmatrix}$$

Note that m, which is the lower bound on $\mathrm{E}\,\Psi_\alpha$, is itself bounded above.

$$0 \le m \le m_{\text{max}}$$

where $m_{\text{max}} = \max_{\alpha \in \mathcal{C}} \mathrm{E}\,\Psi_\alpha$ subject to the wealth and gross leverage constraints. The solution to this is

$$m_{\text{max}} = w \max\left(\frac{g+1}{2} \max_i \frac{\mathrm{E}\, M_i}{p_i} - \frac{g-1}{2} \min_i \frac{\mathrm{E}\, M_i}{p_i}, \max_i \frac{\mathrm{E}\, M_i}{p_i}, -g \min_i \frac{\mathrm{E}\, M_i}{p_i}\right)$$

4 Optimization

Note that solutions along the efficient frontier will probably not be "fully invested". The uninvested portion,

$$w - p' \alpha(m)$$

represent an implicit net cash allocation.

In this case, our investor's objective is profit, so (assuming zero interest rates) cash does not contribute to this objective. In other circumstances, cash may contribute, so we need to be careful to include the contribution from the implicit cash allocation.

Assuming that the investor satisfaction $\mathcal{S}(\alpha)$ is consistent with stochastic dominance, the optimal portfolio $\alpha^\star$ is on the efficient frontier, and we can use the two-step procedure to identify it.

$$\beta^\star = \arg \max_{0 \le \beta \le 1} \mathcal{S}\left(\alpha\left(\beta\, m_{\text{max}}\right)\right)$$

This is a (constrained) one-dimensional optimization. It can be a little intense, since you need to solve a quadratic programming problem at each step, and a little delicate, since $\alpha(m)$ is not continuous; but it is generally surmountable using modern algorithms.

5 MATLAB Implementation

The following is an example using MATLAB's `quadprog()` interior point algorithm.

```
function alpha=alpha_eff(w,g,p,mu,Sigma,beta)
% 'w' is net asset value constraint
% 'g' is the gross leverage constraint ('g>=1')
% 'p' is the initial price array, 'P=p*e^X' is the final price,
% 'X' is normal with mean 'mu' and covariance 'Sigma',
% the objective is profit, so the market vector is 'M=P-p'
% the efficient frontier is indexed by 'beta' in the range [0,1]
N=length(p);
EM=p.*(exp(mu+diag(Sigma)/2)-1);
covM=(p+EM)*(p+EM)'.*(exp(Sigma)-1);
m1=max(EM./p);
m2=min(EM./p);
m=max(eps,min(beta,1-eps))*...
    max(max(m1,-g*m2),(g+1)/2*m1-(g-1)/2*m2);
H=[covM zeros(N);zeros(N,2*N)];
A=[ [-EM'/m zeros(1,N)]
    [p' zeros(1,N)]
    [zeros(1,N) p']
    [diag(p) -diag(p)]
    [-diag(p) -diag(p)]];
b=w*[-1;1;g;zeros(2*N,1)];
x=quadprog(H,[],A,b,[],[],[],[],[],optimset(...
    'Algorithm','interior-point-convex',...
    'Display','off',...
    'TypicalX',w./p/length(p)));
alpha=x(1:N);
end
```

This could be called, for example, in conjunction with an index of satisfaction (defined elsewhere).

```
...
betaStar=fminbnd(@(beta)-satisfaction(p,mu,Sigma,...
    alpha_eff(w,p,mu,Sigma,beta)),0,1);
alphaStar=alpha_eff(w,p,mu,Sigma,betaStar);
...
```