Zarek McGee

MTH410- Optim 2

HW3; 02/14/2019

#4 code

function [xStar] = OptimPortfolio(m)

%INPUT: m is the number of 'l' points to test. Higher m -> smoother curve

l = linspace(1.2,2.5,m); %l is the level of return

Q = [1 1/3 -1/3; 1/3 2 0; -1/3 0 3]; %Covariance Matrix

invQ = inv(Q);

r = [1.1 2 3]'; %Expected value of return

%Shorthand for x formula used below

a = 0.5 \* ones(1,3) \* invQ \* ones(3,1);

b = 0.5 \* ones(1,3) \* invQ \* r;

c = 0.5 \* r' \* invQ \* r;

d = b^2 - a\*c;

%For each l, calculate a lam1 & lam2 then a 3x1 x vector

for i=1:m

lam1 = (b\*l(i)-c)/d;

lam2 = (b - a\*l(i))/d;

x = (0.5\*invQ\*ones(3,1)\*lam1 + 0.5\*invQ\*r\*lam2);

%Sigma Squared (variance)

sigSq(i) = x'\*Q\*x;

end

%Create plot of variance vs. l (top of curve is Efficient Frontier)

plot(sigSq,l); xlabel("Variance x'Qx"); ylabel("Level of return l");

%Find l value corresponding to min x’Qx

min = find(sigSq==min(sigSq));

lStar = l(min); lStar

%Calculate Lagrange Multipliers corr. to the Optimal Portfolio Distr.

lam1 = (b\*lStar-c)/d;

lam2 = (b - a\*lStar)/d;

%Optimal Portfolio Distr.

xStar = (0.5\*invQ\*ones(3,1)\*lam1 + 0.5\*invQ\*r\*lam2);

end

The output (below) gives us an estimate for the value *l\*,* the optimal 3x1 vector *x\** and the graph of the optimal portfolio distribution σ2(*x\*)*, where the Efficient Frontier is positively sloped portion of the curve.

>> OptimPortfolio(50)

**lStar = 1.7306**

**ans =**

**0.57360**

**0.17955**

**0.24685**

