```
function result = mvp(mean_r, CovMatrix, Rf, PlotDummy, TitleString,
 PlotName)

S = CovMatrix;
z = mean_r'; % note the prime to make z a column vector of means
N = length(S);

A = ones(N,1)'*inv(S)*ones(N,1);
B = ones(N,1)'*inv(S)*z;
C = z'*inv(S)*z;
D = A*C - B^2;

mu = [0:1.5*max(z)/50:1.5*max(z)]; %% Use this line for more general
 program
%mu = [0:.02/50:.02]; %% Use this line to make graphs all comparable

for i = 1:length(mu)
%     lam(i) = (C - mu(i)*B)/D;
%     gam(i) = (mu(i)*A - B)/D;
%     w_star(:,i) = lam(i)*inv(S)*ones(N,1) + gam(i)*inv(S)*z;
    sig_2(i) = (A*mu(i)^2 - 2*B*mu(i) + C)/D;
end

MVP_sig = sqrt(1/A);
MVP_r = B/A;
result.MVP_w = (inv(S)*ones(N,1))/(ones(N,1)'*inv(S)*ones(N,1));

TanP_sig = sqrt( (C - 2*Rf*B + Rf^2*A) / (B-A*Rf)^2 );
TanP_r = (C-B*Rf)/(B-A*Rf);
result.TanP_w = inv(S)*(z - Rf*ones(N,1))/(B - A*Rf);

result.frontier = [sqrt(sig_2)', mu'];
result.MVP = [MVP_sig, MVP_r];
result.TanP = [TanP_sig, TanP_r];

max_std = (max(mu)-Rf) / ((TanP_r-Rf)/TanP_sig);
X_axis = [0:max_std/100:max_std];
%Tangent_Line = Rf + X_axis*(TanP_r-Rf)/TanP_sig;

if PlotDummy ==1
    std_r = sqrt(diag(S));
    figure
        plot(result.frontier(:,1),result.frontier(:,2), std_r,
 z', '.', result.MVP(1), result.MVP(2), 'X')
        xlabel('$StdDev_{monthly}$', 'interpreter', 'latex');
        ylabel('$E[R_{monthly}]$', 'interpreter' , 'latex');
        legend('Eff. Frontier', 'Industries', 'Min. Var.
 Port.', 'Location','northwest')
        title([TitleString ': Mean-StdDev Efficient Frontier ']);
    saveas(gcf,[PlotName '.png'])
end
```

Not enough input arguments.

```
Error in mvp (line 3)
S = CovMatrix;
```

Published with MATLAB® R2018b