```
commissionPerShare = 0.005;
ispd = 1000;
% 1 minute data on EWA-EWC
% load('inputData_ETF', 'tday', 'syms', 'cl');
idxA=find(strcmp('EWA', syms));
idxC=find(strcmp('EWC', syms));
%
x=cl(:, idxA);
y=cl(:, idxC);
%
% plot(x);
% hold on;
% plot(y, 'g');
%
% legend('EWA', 'EWC');
% figure;
%
% scatter(x, y);
%
% figure;
%
% regression_result=ols(y, [x ones(size(x))]);
% hedgeRatio=regression_result.beta(1);
% plot(y-hedgeRatio*x);
% Assume a non-zero offset but no drift, with lag=1.
% results=cadf(y, x, 0, 1); % cadf is a function in the jplv7 (spatial-econometrics.com) package. We pick y to be the dependent variable again.
% Print out results
% prt(results);
% Output:
%  Augmented DF test for co-integration variables:      variable   1,variable   2
% CADF t-statistic        # of lags   AR(1) estimate
%      -3.64346635               1       -0.020411
%
%    1% Crit Value    5% Crit Value   10% Crit Value
%         -3.880          -3.359          -3.038

% Combine the two time series into a matrix y2 for input into Johansen test
y2=[y, x];
% results=johansen(y2, 0, 1); % johansen test with non-zero offset but zero drift, and with the lag k=1.
%
% % Print out results
% prt(results);

% Output:
%  Johansen MLE estimates
```

```
% NULL:              Trace Statistic      Crit 90%      Crit 95%      Crit 99%
% r <= 0   variable  1          19.983         13.429        15.494        19.935
% r <= 1   variable  2           3.983          2.705         3.841         6.635
%
% NULL:              Eigen Statistic      Crit 90%      Crit 95%      Crit 99%
% r <= 0   variable  1          16.000         12.297        14.264        18.520
% r <= 1   variable  2           3.983          2.705         3.841         6.635


% Adding IGE to the portfolio

idxI=find(strcmp('IGE', syms));
z=cl(:, idxI);
y3=[y2, z];
y3is = y3(1:ispd,:);
y3os = y3(1+ispd:end,:);

%y3=Prices5Min(1:3000,1:2);

results=johansen(y3is, 0, 1); % johansen test with non-zero offset but zero drift, and
with the lag k=1.

% Print out results
prt(results);

% Output:
%  Johansen MLE estimates
% NULL:              Trace Statistic      Crit 90%      Crit 95%      Crit 99%
% r <= 0   variable  1          34.429         27.067        29.796        35.463
% r <= 1   variable  2          17.532         13.429        15.494        19.935
% r <= 2   variable  3           4.471          2.705         3.841         6.635
%
% NULL:              Eigen Statistic      Crit 90%      Crit 95%      Crit 99%
% r <= 0   variable  1          16.897         18.893        21.131        25.865
% r <= 1   variable  2          13.061         12.297        14.264        18.520
% r <= 2   variable  3           4.471          2.705         3.841         6.635

results.eig % Display the eigenvalues

% ans =
%
%    0.0112
%    0.0087
%    0.0030
results.evec % Display the eigenvectors

% ans =
```

```
%
%   -1.0460   -0.5797   -0.2647
%    0.7600   -0.1120   -0.0790
%    0.2233    0.5316    0.0952
yportis=sum(repmat(results.evec(:, 1)', [size(y3is, 1) 1]).*y3is, 2); % (net) market value of portfolio
yportos=sum(repmat(results.evec(:, 1)', [size(y3os, 1) 1]).*y3os, 2);
% Find value of lambda and thus the halflife of mean reversion by linear regression fit
ylag=lag(yportis, 1);  % lag is a function in the jplv7 (spatial-econometrics.com) package.
deltaY=yportis-ylag;
deltaY(1)=[]; % Regression functions cannot handle the NaN in the first bar of the time series.
ylag(1)=[];
regress_results=ols(deltaY, [ylag ones(size(ylag))]); % ols is a function in the jplv7 (spatial-econometrics.com) package.
halflife=-log(2)/regress_results.beta(1);

fprintf(1, 'halflife=%f periods\n', halflife);

% halflife=22.662578  days
%
%  Apply a simple linear mean reversion strategy to EWA-EWC-IGE
lookback=round(halflife); % setting lookback to the halflife found above

numUnits =-(yportos-movingAvg(yportos, lookback))./movingStd(yportos, lookback);
numUnits(abs(numUnits) < 1) = 0; % This sets a threshold level of 1x SD
% capital invested in portfolio in dollars.  movingAvg and movingStd are functions from epchan.com/book2
positions=repmat(numUnits, [1 size(y3os, 2)]).*repmat(results.evec(:, 1)', [size(y3os, 1) 1]).*y3os;
pnl=sum(lag(positions, 1).*(y3os-lag(y3os, 1))./lag(y3os, 1), 2);

% results.evec(:, 1)' can be viewed as the capital allocation, while positions is the dollar capital in each ETF.

ret=pnl./sum(abs(lag(positions, 1)), 2); % return is P&L divided by gross market value of portfolio
ret(isnan(ret))=[];
pos=sum(abs(positions),2);
figure;
plot(cumprod(1+ret)-1); % Cumulative compounded return
fprintf(1, 'APR=%f Sharpe=%f\n', prod(1+ret).^(252/length(ret))-1, sqrt(252)*mean(ret)/std(ret));
% APR=0.125739 Sharpe=1.391310
```