Table of Contents

Q1 Stress testing

I copied and pasted the monthly data into Matlab into the variable names 'FFmat' and 'indmat' FFmat = [insert copied data here]; indmat=[insert copied data here]; I then saved them as lab4Q1.mat:

```
% save lab4Q1.mat
load lab4Q1
```

Q1(a) Fit multi-factor CAPM to each industry portfolio excess return series.

```
% Create Excess Returns
rf=FFmat(:,5);   % risk-free rate
ex_mark_ret=FFmat(:,2);  % excess market return
cnsmr_ret=indmat(:,2); cnsmr_ex_ret=cnsmr_ret-rf;  % excess returns
manuf_ret=indmat(:,3);manuf_ex_ret=manuf_ret-rf;
hitech_ret=indmat(:,4);hitech_ex_ret=hitech_ret-rf;
health_ret=indmat(:,5);health_ex_ret=health_ret-rf;
other_ret=indmat(:,6);other_ex_ret=other_ret-rf;
ymat=[cnsmr_ex_ret manuf_ex_ret hitech_ex_ret health_ex_ret
 other_ex_ret];  % rows are observations over time, columns are
 variables
size(ymat)   % size() command returns the the number of rows and
 columns of the matrix should be T by n

% FF factors
hml=FFmat(:,4);smb=FFmat(:,3);
xmat=[ones(length(ex_mark_ret),1) ex_mark_ret hml smb];   % creates X
 matrix for regression with three factors

% OLS regresssions
[Bc,BINTc,Rc,RINTc,STATSc] = regress(cnsmr_ex_ret,xmat);   % runs OLS
 regression
cnsmr_est=xmat*Bc;   % estimates of average return from SLR
[Bm,BINTm,Rm,RINTm,STATSm] = regress(manuf_ex_ret,xmat);   % runs OLS
 regression
```

```
manuf_est=xmat*Bm;
[Bhi,BINThi,Rhi,RINThi,STATShi] = regress(hitech_ex_ret,xmat);   %
 runs OLS regression
hitech_est=xmat*Bhi;
[Bhe,BINThe,Rhe,RINThe,STATShe] = regress(health_ex_ret,xmat);   %
 runs OLS regression
health_est=xmat*Bhe;
[Bo,BINTo,Ro,RINTo,STATSo] = regress(other_ex_ret,xmat);   % runs OLS
 regression
other_est=xmat*Bo;

% Coefficients and 95% Confidence intervals
[Bc(1) BINTc(1,:) Bc(2) BINTc(2,:) Bc(3) BINTc(3,:) Bc(4) BINTc(4,:);
 Bm(1) BINTm(1,:) Bm(2) BINTm(2,:) Bm(3) BINTm(3,:) Bm(4) BINTm(4,:);
 Bhi(1) BINThi(1,:) Bhi(2) BINThi(2,:) Bhi(3) BINThi(3,:) Bhi(4)
 BINThi(4,:);
 Bhe(1) BINThe(1,:) Bhe(2) BINThe(2,:) Bhe(3) BINThe(3,:) Bhe(4)
 BINThe(4,:);
 Bo(1) BINTo(1,:) Bo(2) BINTo(2,:) Bo(3) BINTo(3,:) Bo(4) BINTo(4,:)]


ans =

        1026           5


ans =

  Columns 1 through 7

    0.1244    0.0061    0.2426    0.9213    0.8980    0.9445   -0.0058
    0.0383   -0.0492    0.1259    0.9815    0.9643    0.9987    0.1591
    0.1321    0.0013    0.2629    0.9868    0.9611    1.0125   -0.3260
    0.3181    0.1113    0.5248    0.8917    0.8511    0.9323   -0.1888
   -0.2231   -0.3428   -0.1034    1.0549    1.0314    1.0785    0.3706

  Columns 8 through 12

   -0.0395    0.0279    0.0124   -0.0251    0.0498
    0.1342    0.1841   -0.0961   -0.1238   -0.0684
   -0.3633   -0.2887    0.0367   -0.0047    0.0781
   -0.2478   -0.1299   -0.0954   -0.1608   -0.0299
    0.3364    0.4047    0.0663    0.0284    0.1042
```

Q1(b) and (c) Estimate mean returns under different scenarios for each industry along with VaR and ES

Note: The Function getstuff() calculates the quantities required for parts (b) and (c). See the file "getstuff.m" for details. 'est' is the regression estimate (predicted values) for each scenario. VaRG is the VaR assuming a Gaussian distribution, ESG is the Expected Shortfall assuming a Gaussian distribution, VaRN is the VaR using Non-parametric methods, ESN is the Expected Shortfall using Non-parametric methods.

```
% Consumer
[c_est500, c_VaRG500, c_ESG500, c_VaRN500, c_ESN500] = getstuff([1 -5
 0 0],Bc,sqrt(STATSc(4)),Rc);
[c_est1000, c_VaRG1000, c_ESG1000, c_VaRN1000, c_ESN1000] =
 getstuff([1 -10 0 0],Bc,sqrt(STATSc(4)),Rc);
[c_est502, c_VaRG502, c_ESG502, c_VaRN502, c_ESN502] = getstuff([1 -5
 0 -2],Bc,sqrt(STATSc(4)),Rc);
[c_est1002, c_VaRG1002, c_ESG1002, c_VaRN1002, c_ESN1002] =
 getstuff([1 -10 0 -2],Bc,sqrt(STATSc(4)),Rc);
[c_est520, c_VaRG520, c_ESG520, c_VaRN520, c_ESN520] = getstuff([1 -5
 -2 0],Bc,sqrt(STATSc(4)),Rc);
[c_est1020, c_VaRG1020, c_ESG1020, c_VaRN1020, c_ESN1020] =
 getstuff([1 -10 -2 0],Bc,sqrt(STATSc(4)),Rc);
[c_est522, c_VaRG522, c_ESG522, c_VaRN522, c_ESN522] = getstuff([1 -5
 -2 -2],Bc,sqrt(STATSc(4)),Rc);
[c_est1022, c_VaRG1022, c_ESG1022, c_VaRN1022, c_ESN1022] =
 getstuff([1 -10 -2 -2],Bc,sqrt(STATSc(4)),Rc);

% display results in matrix
[c_est500 c_VaRG500 c_ESG500 c_VaRN500 c_ESN500;
 c_est502, c_VaRG502, c_ESG502, c_VaRN502, c_ESN502;
 c_est520, c_VaRG520, c_ESG520, c_VaRN520, c_ESN520;
 c_est522, c_VaRG522, c_ESG522, c_VaRN522, c_ESN522;
 c_est1000, c_VaRG1000, c_ESG1000, c_VaRN1000, c_ESN1000;
 c_est1002, c_VaRG1002, c_ESG1002, c_VaRN1002, c_ESN1002;
 c_est1020, c_VaRG1020, c_ESG1020, c_VaRN1020, c_ESN1020;
 c_est1022, c_VaRG1022, c_ESG1022, c_VaRN1022, c_ESN1022]

% Histogram of residuals
figure;hist(Rc,25);title('Consumer residuals');
% Skewness and Kurtosis of residuals
[skewness(Rc) kurtosis(Rc)]
% JB test for normality.
[h,p]=jbtest(Rc)

%Manufacturing
[ma_est500, ma_VaRG500, ma_ESG500, ma_VaRN500, ma_ESN500] =
 getstuff([1 -5 0 0],Bm,sqrt(STATSm(4)),Rm);
[ma_est1000, ma_VaRG1000, ma_ESG1000, ma_VaRN1000, ma_ESN1000] =
 getstuff([1 -10 0 0],Bm,sqrt(STATSm(4)),Rm);
[ma_est502, ma_VaRG502, ma_ESG502, ma_VaRN502, ma_ESN502] =
 getstuff([1 -5 0 -2],Bm,sqrt(STATSm(4)),Rm);
[ma_est1002, ma_VaRG1002, ma_ESG1002, ma_VaRN1002, ma_ESN1002] =
 getstuff([1 -10 0 -2],Bm,sqrt(STATSm(4)),Rm);
[ma_est520, ma_VaRG520, ma_ESG520, ma_VaRN520, ma_ESN520] =
 getstuff([1 -5 -2 0],Bm,sqrt(STATSm(4)),Rm);
[ma_est1020, ma_VaRG1020, ma_ESG1020, ma_VaRN1020, ma_ESN1020] =
 getstuff([1 -10 -2 0],Bm,sqrt(STATSm(4)),Rm);
[ma_est522, ma_VaRG522, ma_ESG522, ma_VaRN522, ma_ESN522] =
 getstuff([1 -5 -2 -2],Bm,sqrt(STATSm(4)),Rm);
```

```
[ma_est1022, ma_VaRG1022, ma_ESG1022, ma_VaRN1022, ma_ESN1022] =
 getstuff([1 -10 -2 -2],Bm,sqrt(STATSm(4)),Rm);
[ma_est500 ma_VaRG500 ma_ESG500 ma_VaRN500 ma_ESN500;
 ma_est502, ma_VaRG502, ma_ESG502, ma_VaRN502, ma_ESN502;
 ma_est520, ma_VaRG520, ma_ESG520, ma_VaRN520, ma_ESN520;
 ma_est522, ma_VaRG522, ma_ESG522, ma_VaRN522, ma_ESN522;
 ma_est1000, ma_VaRG1000, ma_ESG1000, ma_VaRN1000, ma_ESN1000;
 ma_est1002, ma_VaRG1002, ma_ESG1002, ma_VaRN1002, ma_ESN1002;
 ma_est1020, ma_VaRG1020, ma_ESG1020, ma_VaRN1020, ma_ESN1020;
 ma_est1022, ma_VaRG1022, ma_ESG1022, ma_VaRN1022, ma_ESN1022]

figure;hist(Rm,25);title('Manufacturing residuals');
[skewness(Rm) kurtosis(Rm)]
[h,p]=jbtest(Rm)   % JB test for normality.

%Hi_tech
[hi_est500, hi_VaRG500, hi_ESG500, hi_VaRN500, hi_ESN500] =
 getstuff([1 -5 0 0],Bhi,sqrt(STATShi(4)),Rhi);
[hi_est1000, hi_VaRG1000, hi_ESG1000, hi_VaRN1000, hi_ESN1000] =
 getstuff([1 -10 0 0],Bhi,sqrt(STATShi(4)),Rhi);
[hi_est502, hi_VaRG502, hi_ESG502, hi_VaRN502, hi_ESN502] =
 getstuff([1 -5 0 -2],Bhi,sqrt(STATShi(4)),Rhi);
[hi_est1002, hi_VaRG1002, hi_ESG1002, hi_VaRN1002, hi_ESN1002] =
 getstuff([1 -10 0 -2],Bhi,sqrt(STATShi(4)),Rhi);
[hi_est520, hi_VaRG520, hi_ESG520, hi_VaRN520, hi_ESN520] =
 getstuff([1 -5 -2 0],Bhi,sqrt(STATShi(4)),Rhi);
[hi_est1020, hi_VaRG1020, hi_ESG1020, hi_VaRN1020, hi_ESN1020] =
 getstuff([1 -10 -2 0],Bhi,sqrt(STATShi(4)),Rhi);
[hi_est522, hi_VaRG522, hi_ESG522, hi_VaRN522, hi_ESN522] =
 getstuff([1 -5 -2 -2],Bhi,sqrt(STATShi(4)),Rhi);
[hi_est1022, hi_VaRG1022, hi_ESG1022, hi_VaRN1022, hi_ESN1022] =
 getstuff([1 -10 -2 -2],Bhi,sqrt(STATShi(4)),Rhi);
[hi_est500 hi_VaRG500 hi_ESG500 hi_VaRN500 hi_ESN500;
 hi_est502, hi_VaRG502, hi_ESG502, hi_VaRN502, hi_ESN502;
 hi_est520, hi_VaRG520, hi_ESG520, hi_VaRN520, hi_ESN520;
 hi_est522, hi_VaRG522, hi_ESG522, hi_VaRN522, hi_ESN522;
 hi_est1000, hi_VaRG1000, hi_ESG1000, hi_VaRN1000, hi_ESN1000;
 hi_est1002, hi_VaRG1002, hi_ESG1002, hi_VaRN1002, hi_ESN1002;
 hi_est1020, hi_VaRG1020, hi_ESG1020, hi_VaRN1020, hi_ESN1020;
 hi_est1022, hi_VaRG1022, hi_ESG1022, hi_VaRN1022, hi_ESN1022]

figure;hist(Rhi,25);title('Hitech residuals');
[skewness(Rhi) kurtosis(Rhi)]
[h,p]=jbtest(Rhi)   % JB test for normality.

%Health
[he_est500, he_VaRG500, he_ESG500, he_VaRN500, he_ESN500] =
 getstuff([1 -5 0 0],Bhe,sqrt(STATShe(4)),Rhe);
[he_est1000, he_VaRG1000, he_ESG1000, he_VaRN1000, he_ESN1000] =
 getstuff([1 -10 0 0],Bhe,sqrt(STATShe(4)),Rhe);
[he_est502, he_VaRG502, he_ESG502, he_VaRN502, he_ESN502] =
 getstuff([1 -5 0 -2],Bhe,sqrt(STATShe(4)),Rhe);
[he_est1002, he_VaRG1002, he_ESG1002, he_VaRN1002, he_ESN1002] =
 getstuff([1 -10 0 -2],Bhe,sqrt(STATShe(4)),Rhe);
```

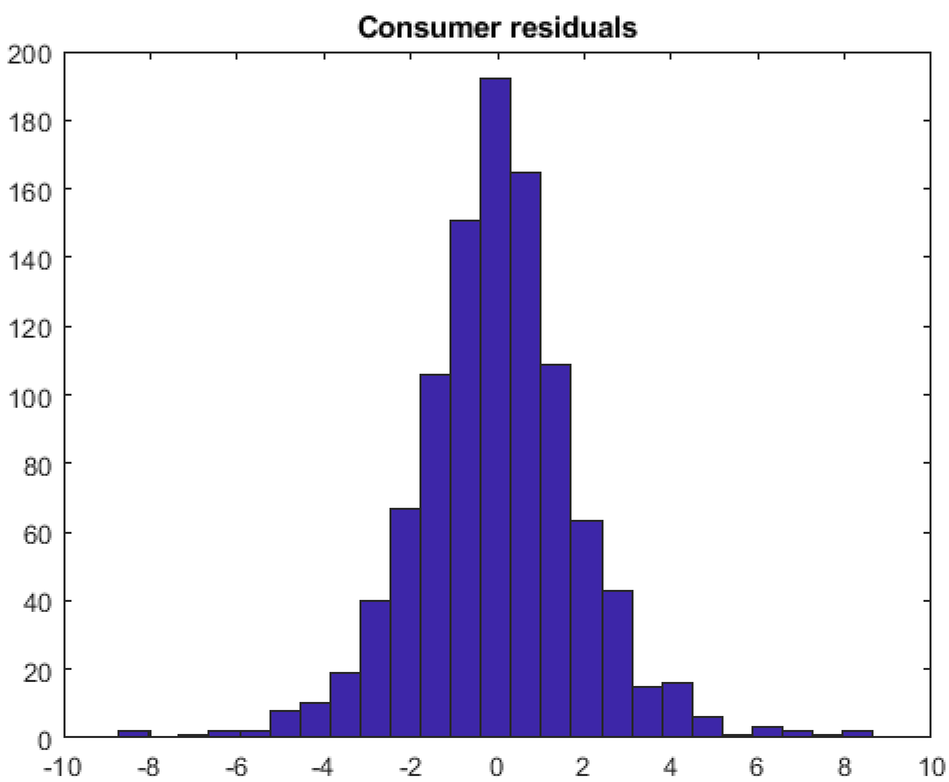
```
[he_est520, he_VaRG520, he_ESG520, he_VaRN520, he_ESN520] =
 getstuff([1 -5 -2 0],Bhe,sqrt(STATShe(4)),Rhe);
[he_est1020, he_VaRG1020, he_ESG1020, he_VaRN1020, he_ESN1020] =
 getstuff([1 -10 -2 0],Bhe,sqrt(STATShe(4)),Rhe);
[he_est522, he_VaRG522, he_ESG522, he_VaRN522, he_ESN522] =
 getstuff([1 -5 -2 -2],Bhe,sqrt(STATShe(4)),Rhe);
[he_est1022, he_VaRG1022, he_ESG1022, he_VaRN1022, he_ESN1022] =
 getstuff([1 -10 -2 -2],Bhe,sqrt(STATShe(4)),Rhe);
[he_est500 he_VaRG500 he_ESG500 he_VaRN500 he_ESN500;
 he_est502, he_VaRG502, he_ESG502, he_VaRN502, he_ESN502;
 he_est520, he_VaRG520, he_ESG520, he_VaRN520, he_ESN520;
 he_est522, he_VaRG522, he_ESG522, he_VaRN522, he_ESN522;
 he_est1000, he_VaRG1000, he_ESG1000, he_VaRN1000, he_ESN1000;
 he_est1002, he_VaRG1002, he_ESG1002, he_VaRN1002, he_ESN1002;
 he_est1020, he_VaRG1020, he_ESG1020, he_VaRN1020, he_ESN1020;
 he_est1022, he_VaRG1022, he_ESG1022, he_VaRN1022, he_ESN1022]

figure;hist(Rhe,25);title('Health residuals');
[skewness(Rhe) kurtosis(Rhe)]
[h,p]=jbtest(Rhe)   % JB test for normality.

%Other
[o_est500, o_VaRG500, o_ESG500, o_VaRN500, o_ESN500] = getstuff([1 -5
 0 0],Bo,sqrt(STATSo(4)),Ro);
[o_est1000, o_VaRG1000, o_ESG1000, o_VaRN1000, o_ESN1000] =
 getstuff([1 -10 0 0],Bo,sqrt(STATSo(4)),Ro);
[o_est502, o_VaRG502, o_ESG502, o_VaRN502, o_ESN502] = getstuff([1 -5
 0 -2],Bo,sqrt(STATSo(4)),Ro);
[o_est1002, o_VaRG1002, o_ESG1002, o_VaRN1002, o_ESN1002] =
 getstuff([1 -10 0 -2],Bo,sqrt(STATSo(4)),Ro);
[o_est520, o_VaRG520, o_ESG520, o_VaRN520, o_ESN520] = getstuff([1 -5
 -2 0],Bo,sqrt(STATSo(4)),Ro);
[o_est1020, o_VaRG1020, o_ESG1020, o_VaRN1020, o_ESN1020] =
 getstuff([1 -10 -2 0],Bo,sqrt(STATSo(4)),Ro);
[o_est522, o_VaRG522, o_ESG522, o_VaRN522, o_ESN522] = getstuff([1 -5
 -2 -2],Bo,sqrt(STATSo(4)),Ro);
[o_est1022, o_VaRG1022, o_ESG1022, o_VaRN1022, o_ESN1022] =
 getstuff([1 -10 -2 -2],Bo,sqrt(STATSo(4)),Ro);
[o_est500 o_VaRG500 o_ESG500 o_VaRN500 o_ESN500;
 o_est502, o_VaRG502, o_ESG502, o_VaRN502, o_ESN502;
 o_est520, o_VaRG520, o_ESG520, o_VaRN520, o_ESN520;
 o_est522, o_VaRG522, o_ESG522, o_VaRN522, o_ESN522;
 o_est1000, o_VaRG1000, o_ESG1000, o_VaRN1000, o_ESN1000;
 o_est1002, o_VaRG1002, o_ESG1002, o_VaRN1002, o_ESN1002;
 o_est1020, o_VaRG1020, o_ESG1020, o_VaRN1020, o_ESN1020;
 o_est1022, o_VaRG1022, o_ESG1022, o_VaRN1022, o_ESN1022]

figure;hist(Ro,25);title('Other residuals');
[skewness(Ro) kurtosis(Ro)]
[h,p]=jbtest(Ro)   % JB test for normality.

% SERs for industry regressions for part (b)
[sqrt(STATSc(4)) sqrt(STATSm(4)) sqrt(STATShi(4)) sqrt(STATShe(4))
 sqrt(STATSo(4))]
```

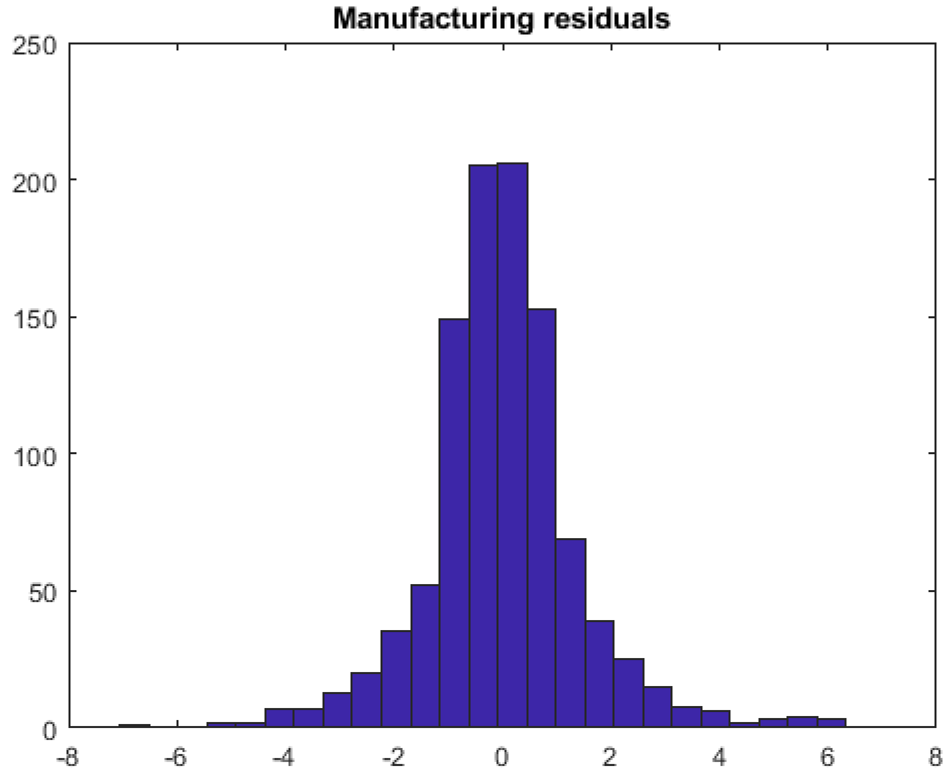
```
% mean estimates for part (b)
[c_est500 ma_est500 hi_est500 he_est500 o_est500;
 c_est502 ma_est502 hi_est502 he_est502 o_est502;
 c_est520 ma_est520 hi_est520 he_est520 o_est520;
 c_est522 ma_est522 hi_est522 he_est522 o_est522;
 c_est1000 ma_est1000 hi_est1000 he_est1000 o_est1000;
 c_est1002 ma_est1002 hi_est1002 he_est1002 o_est1002;
 c_est1020 ma_est1020 hi_est1020 he_est1020 o_est1020;
 c_est1022 ma_est1022 hi_est1022 he_est1022 o_est1022]

m_N500=length(ex_mark_ret(ex_mark_ret<-5));
m_pN500=length(ex_mark_ret(ex_mark_ret<-5))/length(ex_mark_ret);
m_N1000=length(ex_mark_ret(ex_mark_ret<-10));
m_pN1000=length(ex_mark_ret(ex_mark_ret<-10))/length(ex_mark_ret);
[m_N500 m_pN500;
 m_N1000 m_pN1000]


ans =

   -4.4820   -8.9255   -9.5807   -9.5492  -10.8464
   -4.5067   -8.9502   -9.6054   -9.5739  -10.8711
   -4.4704   -8.9140   -9.5691   -9.5376  -10.8348
   -4.4951   -8.9387   -9.5938   -9.5624  -10.8596
   -9.0883  -13.5319  -14.1870  -14.1555  -15.4527
   -9.1130  -13.5566  -14.2117  -14.1802  -15.4774
   -9.0767  -13.5203  -14.1754  -14.1440  -15.4412
   -9.1014  -13.5450  -14.2002  -14.1687  -15.4659


ans =

    0.0754    5.3068

Warning: P is less than the smallest tabulated value, returning
 0.001.

h =

     1


p =

   1.0000e-03


ans =

   -4.8691   -8.1598   -8.6450   -8.7813   -9.5452
   -4.6769   -7.9676   -8.4528   -8.5891   -9.3530
   -5.1873   -8.4780   -8.9632   -9.0996   -9.8634
   -4.9951   -8.2858   -8.7710   -8.9074   -9.6712
```

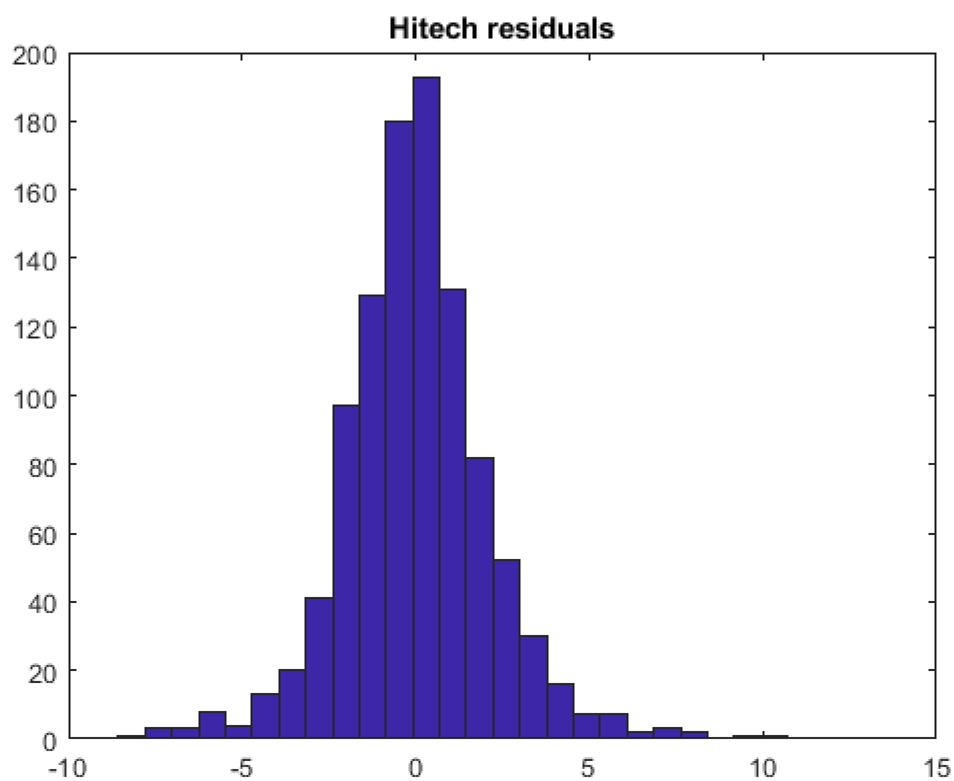
```
   -9.7765  -13.0672  -13.5524  -13.6887  -14.4526
   -9.5843  -12.8750  -13.3602  -13.4965  -14.2604
  -10.0947  -13.3854  -13.8706  -14.0070  -14.7708
   -9.9025  -13.1932  -13.6784  -13.8148  -14.5786


ans =

    0.3089    6.2502

Warning: P is less than the smallest tabulated value, returning
 0.001.

h =

     1


p =

   1.0000e-03


ans =

   -4.8020   -9.7186  -10.4435  -10.6395  -11.6570
   -4.8755   -9.7920  -10.5169  -10.7129  -11.7304
   -4.1500   -9.0665   -9.7914   -9.9874  -11.0049
   -4.2234   -9.1399   -9.8648  -10.0609  -11.0783
   -9.7362  -14.6527  -15.3776  -15.5737  -16.5911
   -9.8096  -14.7262  -15.4511  -15.6471  -16.6645
   -9.0841  -14.0007  -14.7256  -14.9216  -15.9390
   -9.1575  -14.0741  -14.7990  -14.9950  -16.0125


ans =

    0.2121    5.4096

Warning: P is less than the smallest tabulated value, returning
 0.001.

h =

     1


p =

   1.0000e-03


ans =
```

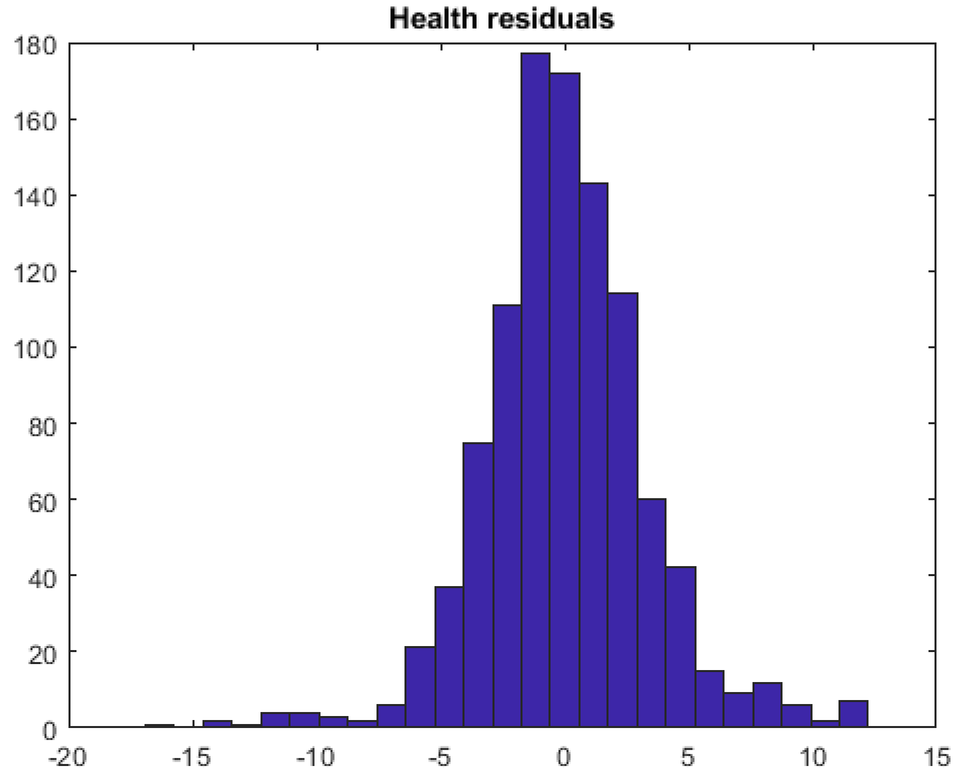
```
   -4.1405  -11.9095  -13.0549  -14.3829  -16.5973
   -3.9497  -11.7187  -12.8642  -14.1922  -16.4065
   -3.7628  -11.5318  -12.6773  -14.0052  -16.2196
   -3.5720  -11.3410  -12.4865  -13.8145  -16.0289
   -8.5990  -16.3680  -17.5135  -18.8415  -21.0558
   -8.4083  -16.1773  -17.3227  -18.6507  -20.8651
   -8.2213  -15.9903  -17.1358  -18.4638  -20.6782
   -8.0306  -15.7996  -16.9450  -18.2730  -20.4874


ans =

   -0.0564    5.6389

Warning: P is less than the smallest tabulated value, returning
 0.001.

h =

     1


p =

   1.0000e-03


ans =

   -5.4977   -9.9960  -10.6592  -10.7942  -12.9568
   -5.6304  -10.1287  -10.7919  -10.9269  -13.0895
   -6.2388  -10.7371  -11.4003  -11.5353  -13.6979
   -6.3715  -10.8698  -11.5330  -11.6680  -13.8306
  -10.7724  -15.2706  -15.9339  -16.0688  -18.2315
  -10.9050  -15.4033  -16.0666  -16.2015  -18.3642
  -11.5135  -16.0118  -16.6750  -16.8100  -18.9726
  -11.6462  -16.1444  -16.8077  -16.9426  -19.1053


ans =

   -0.4944   11.0257

Warning: P is less than the smallest tabulated value, returning
 0.001.

h =

     1


p =

   1.0000e-03
```

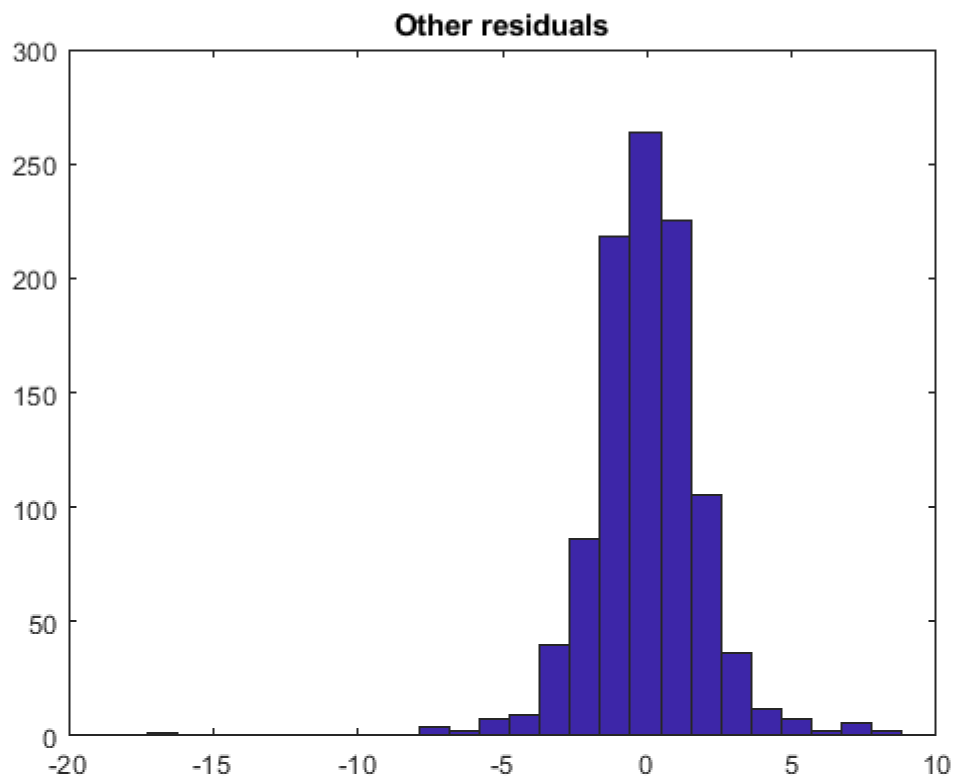
```
ans =

    1.9101    1.4145    2.1134    3.3396    1.9336


ans =

   -4.4820   -4.8691   -4.8020   -4.1405   -5.4977
   -4.5067   -4.6769   -4.8755   -3.9497   -5.6304
   -4.4704   -5.1873   -4.1500   -3.7628   -6.2388
   -4.4951   -4.9951   -4.2234   -3.5720   -6.3715
   -9.0883   -9.7765   -9.7362   -8.5990  -10.7724
   -9.1130   -9.5843   -9.8096   -8.4083  -10.9050
   -9.0767  -10.0947   -9.0841   -8.2213  -11.5135
   -9.1014   -9.9025   -9.1575   -8.0306  -11.6462


ans =

  117.0000    0.1140
   31.0000    0.0302
```




Manufacturing residuals
250
200
150
100
50
0
-8
-6
-4
-2
0
2
4
6
8


Hitech residuals
200
180
160
140
120
100
80
60
40
20
0
-10
-5
0
5
10
15


Health residuals
180
160
140
120
100
80
60
40
20
0
-20
-15
-10
-5
0
5
10
15

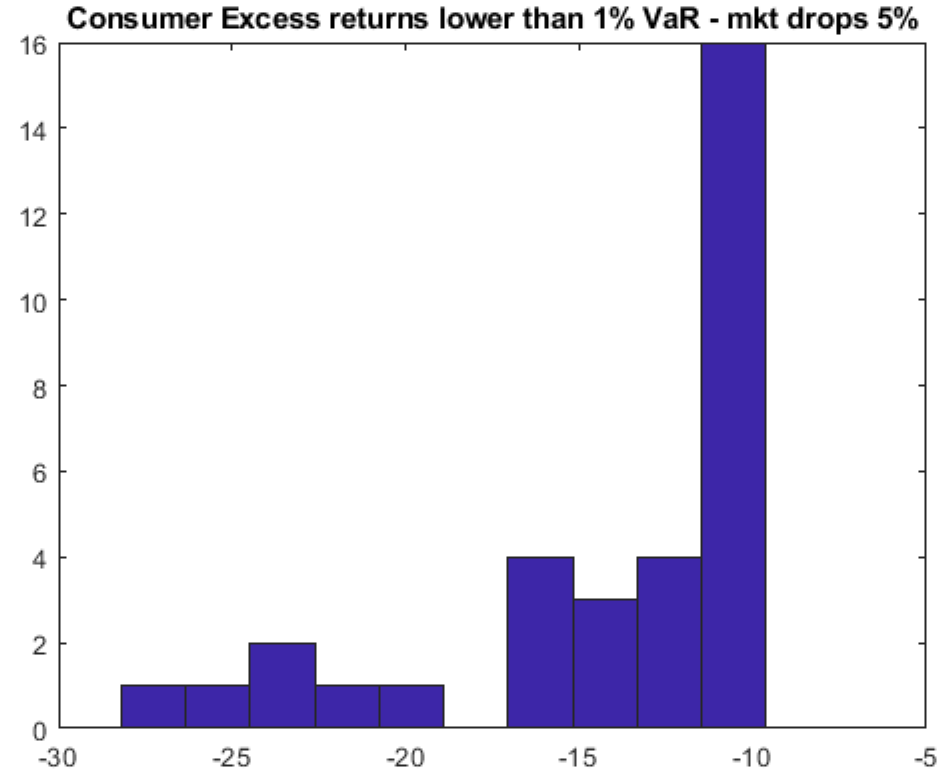

Other residuals
300
250
200
150
100
50
0
-20
-15
-10
-5
0
5
10

Q1(e) Count the number of times loses were greater than VaRs and market returns <-5,-10 in actual data series

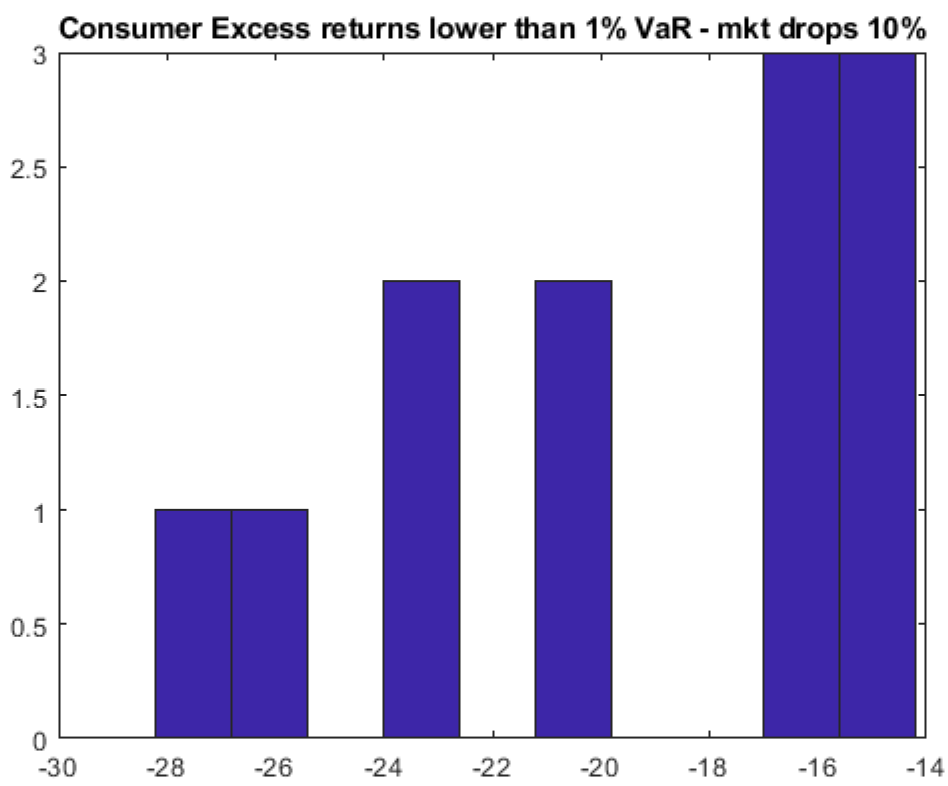
```
c_N500=sum(cnsmr_ex_ret<c_VaRN500 & ex_mark_ret<-5);
c_pN500=sum(cnsmr_ex_ret<c_VaRN500 & ex_mark_ret<-5)/
sum(ex_mark_ret<-5);
c_mN500=mean(cnsmr_ex_ret(cnsmr_ex_ret<c_VaRN500 & ex_mark_ret<-5));

% histograms of Excess returns lower than estimated VaR
figure;hist(cnsmr_ex_ret(cnsmr_ex_ret<c_VaRN500 & ex_mark_ret<-5));
title('Consumer Excess returns lower than 1% VaR - mkt drops 5%');

c_N1000=sum(cnsmr_ex_ret<c_VaRN1000 & ex_mark_ret<-10);
c_pN1000=sum(cnsmr_ex_ret<c_VaRN1000 & ex_mark_ret<-10)/
sum(ex_mark_ret<-10);
c_mN1000=mean(cnsmr_ex_ret(cnsmr_ex_ret<c_VaRN1000 &
 ex_mark_ret<-10));

figure;hist(cnsmr_ex_ret(cnsmr_ex_ret<c_VaRN1000 & ex_mark_ret<-10));
title('Consumer Excess returns lower than 1% VaR - mkt drops 10%');

ma_N500=sum(manuf_ex_ret<ma_VaRN500 & ex_mark_ret<-5);
ma_pN500=sum(manuf_ex_ret<ma_VaRN500 & ex_mark_ret<-5)/
sum(ex_mark_ret<-5);
ma_mN500=mean(manuf_ex_ret(manuf_ex_ret<ma_VaRN500 & ex_mark_ret<-5));
ma_N1000=sum(manuf_ex_ret<ma_VaRN1000 & ex_mark_ret<-10);
ma_pN1000=sum(manuf_ex_ret<ma_VaRN1000 & ex_mark_ret<-10)/
sum(ex_mark_ret<-10);
ma_mN1000=mean(manuf_ex_ret(manuf_ex_ret<ma_VaRN1000 &
 ex_mark_ret<-10));
hi_N500=sum(hitech_ex_ret<hi_VaRN500 & ex_mark_ret<-5);
hi_pN500=sum(hitech_ex_ret<hi_VaRN500 & ex_mark_ret<-5)/
sum(ex_mark_ret<-5);
hi_mN500=mean(hitech_ex_ret(hitech_ex_ret<hi_VaRN500 &
 ex_mark_ret<-5));
hi_N1000=sum(hitech_ex_ret<hi_VaRN1000 & ex_mark_ret<-10);
hi_pN1000=sum(hitech_ex_ret<hi_VaRN1000 & ex_mark_ret<-10)/
sum(ex_mark_ret<-10);
hi_mN1000=mean(hitech_ex_ret(hitech_ex_ret<hi_VaRN1000 &
 ex_mark_ret<-10));
he_N500=sum(health_ex_ret<he_VaRN500 & ex_mark_ret<-5);
he_pN500=sum(health_ex_ret<he_VaRN500 & ex_mark_ret<-5)/
sum(ex_mark_ret<-5);
he_mN500=mean(health_ex_ret(health_ex_ret<he_VaRN500 &
 ex_mark_ret<-5));
he_N1000=sum(health_ex_ret<he_VaRN1000 & ex_mark_ret<-10);
he_pN1000=sum(health_ex_ret<he_VaRN1000 & ex_mark_ret<-10)/
sum(ex_mark_ret<-10);
he_mN1000=mean(health_ex_ret(health_ex_ret<he_VaRN1000 &
 ex_mark_ret<-10));
```

```
o_N500=sum(other_ex_ret<o_VaRN500 & ex_mark_ret<-5);
o_pN500=sum(other_ex_ret<o_VaRN500 & ex_mark_ret<-5)/
sum(ex_mark_ret<-5);
o_mN500=mean(other_ex_ret(other_ex_ret<o_VaRN500 & ex_mark_ret<-5));
o_N1000=sum(other_ex_ret<o_VaRN1000 & ex_mark_ret<-10);
o_pN1000=sum(other_ex_ret<o_VaRN1000 & ex_mark_ret<-10)/
sum(ex_mark_ret<-10);
o_mN1000=mean(other_ex_ret(other_ex_ret<o_VaRN1000 &
 ex_mark_ret<-10));

[c_N500 c_pN500 c_pN500*m_pN500 c_mN500 c_N1000 c_pN1000
 c_pN1000*m_pN1000 c_mN1000;
 ma_N500 ma_pN500 ma_pN500*m_pN500 ma_mN500 ma_N1000 ma_pN1000
 ma_pN1000*m_pN1000 ma_mN1000;
 hi_N500 hi_pN500 hi_pN500*m_pN500 hi_mN500 hi_N1000 hi_pN1000
 hi_pN1000*m_pN1000 hi_mN1000;
 he_N500 he_pN500 he_pN500*m_pN500 he_mN500 he_N1000 he_pN1000
 he_pN1000*m_pN1000 he_mN1000;
 o_N500 o_pN500 o_pN500*m_pN500 o_mN500 o_N1000 o_pN1000
 o_pN1000*m_pN1000 o_mN1000]

%clear; % clear workspace
```

```
ans =

  Columns 1 through 7

   33.0000    0.2821    0.0322  -14.0842   12.0000    0.3871    0.0117
   42.0000    0.3590    0.0409  -13.5983   15.0000    0.4839    0.0146
   33.0000    0.2821    0.0322  -15.1315   11.0000    0.3548    0.0107
   10.0000    0.0855    0.0097  -21.5690    6.0000    0.1935    0.0058
   36.0000    0.3077    0.0351  -16.6686   14.0000    0.4516    0.0136

  Column 8

  -19.4983
  -18.7033
  -20.2564
  -24.4233
  -21.4400
```



Consumer Excess returns lower than 1% VaR - mkt drops 5%
16
14
12
10
8
6
4
2
0
-30
-25
-20
-15
-10
-5



Consumer Excess returns lower than 1% VaR - mkt drops 10%
3
2.5
2
1.5
1
0.5
0
-30
-28
-26
-24
-22
-20
-18
-16
-14

Q2 PCA and Factor Modelling

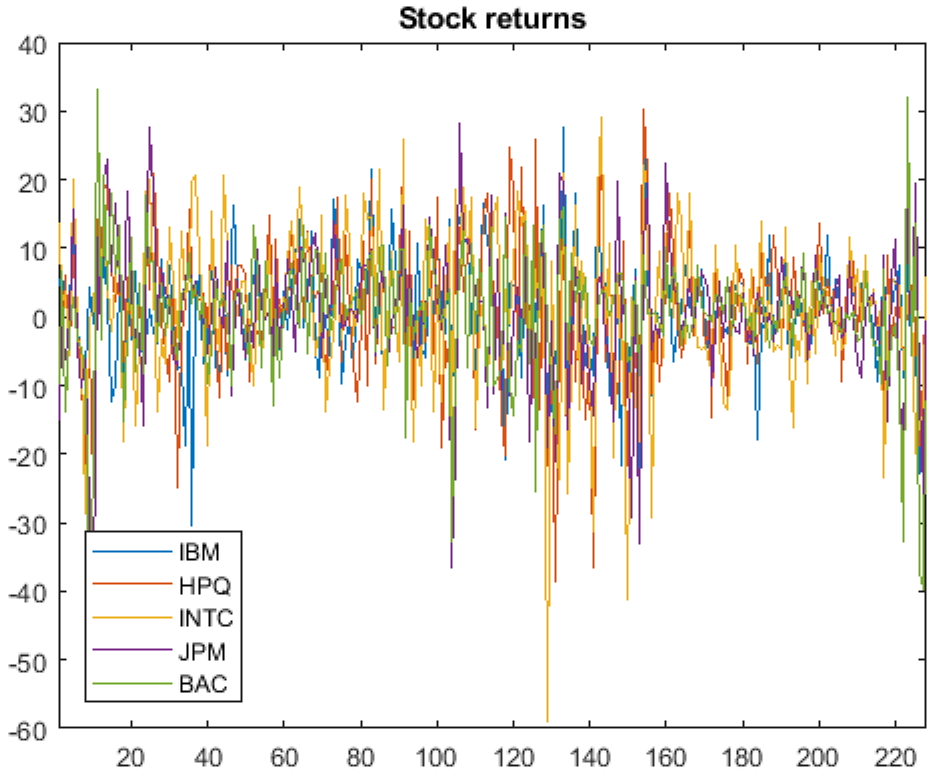
I copied the data from Tsay_FM_data.txt Tsay_data = [copied data here]; save Tsay_data.mat

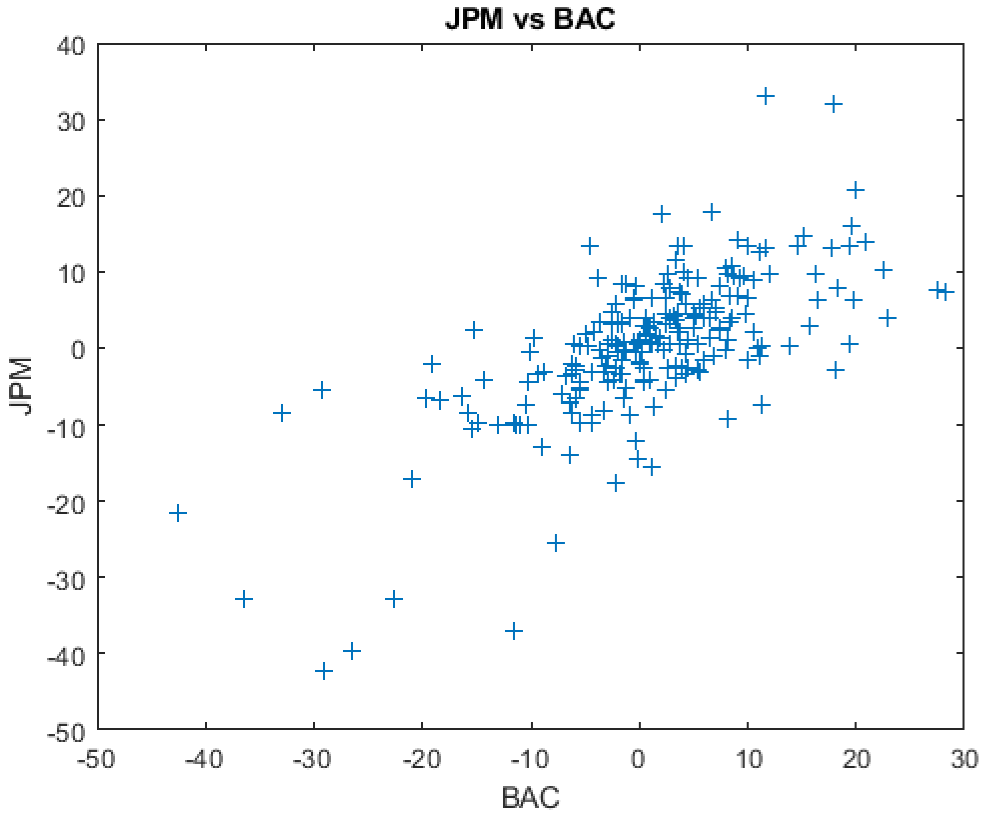
```
load('Tsay_data.mat'); % Equivalent command to "load Tsay_data.mat" or
 simply "load Tsay_data"
```

Q2(a) Correlation matrix

```
% plot each series on the one graph
figure;plot(Tsay_data);title('Stock returns');
legend('IBM','HPQ','INTC','JPM','BAC','Location','SouthWest');
xlim([ 1 length(Tsay_data)]);

% Correlation matrix
corr(Tsay_data)    % estimates the sample correlation matrix, showing
 all pairwise correlations.

% Plot of JPM vs BAC
figure;plot(Tsay_data(:,4),Tsay_data(:,5),'+');title('JPM vs BAC');
xlabel('BAC');ylabel('JPM');
```

```
ans =

    1.0000    0.4620    0.4593    0.3384    0.2545
    0.4620    1.0000    0.5495    0.3889    0.2591
    0.4593    0.5495    1.0000    0.3578    0.2521
    0.3384    0.3889    0.3578    1.0000    0.6836
    0.2545    0.2591    0.2521    0.6836    1.0000
```


Stock returns
IBM
HPQ
INTC
JPM
BAC

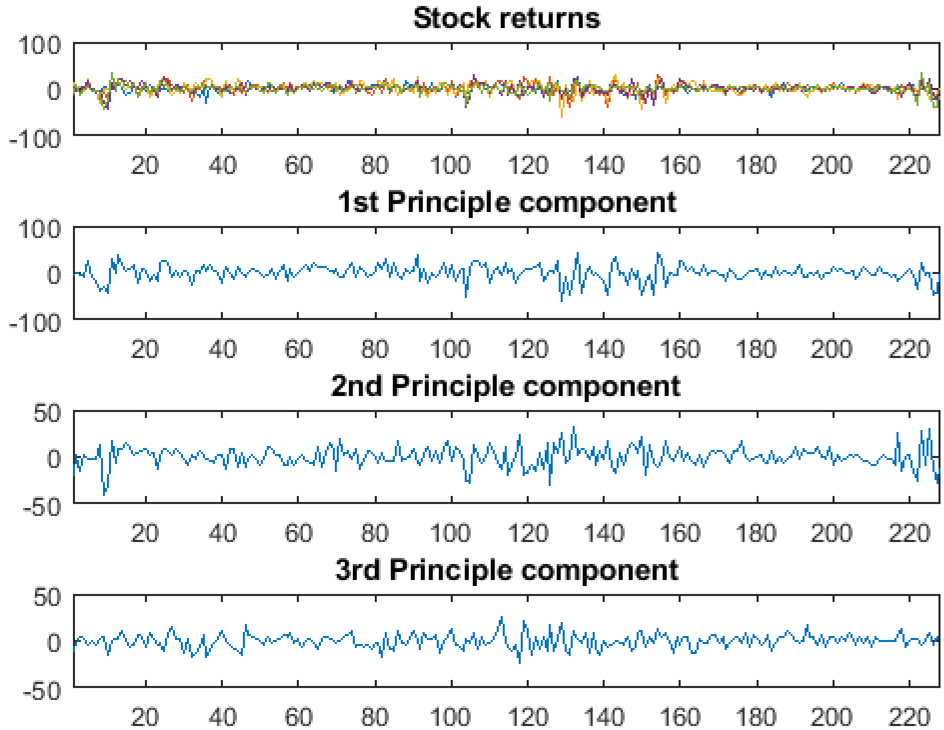

JPM vs BAC
JPM
BAC

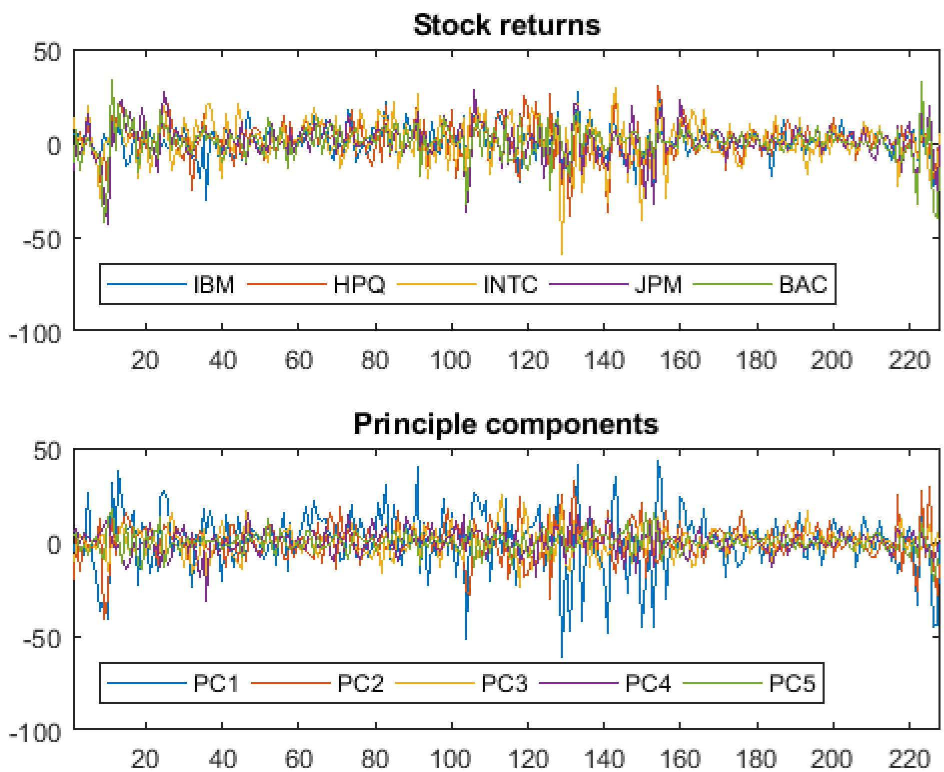
Q2(b) PCA analysis

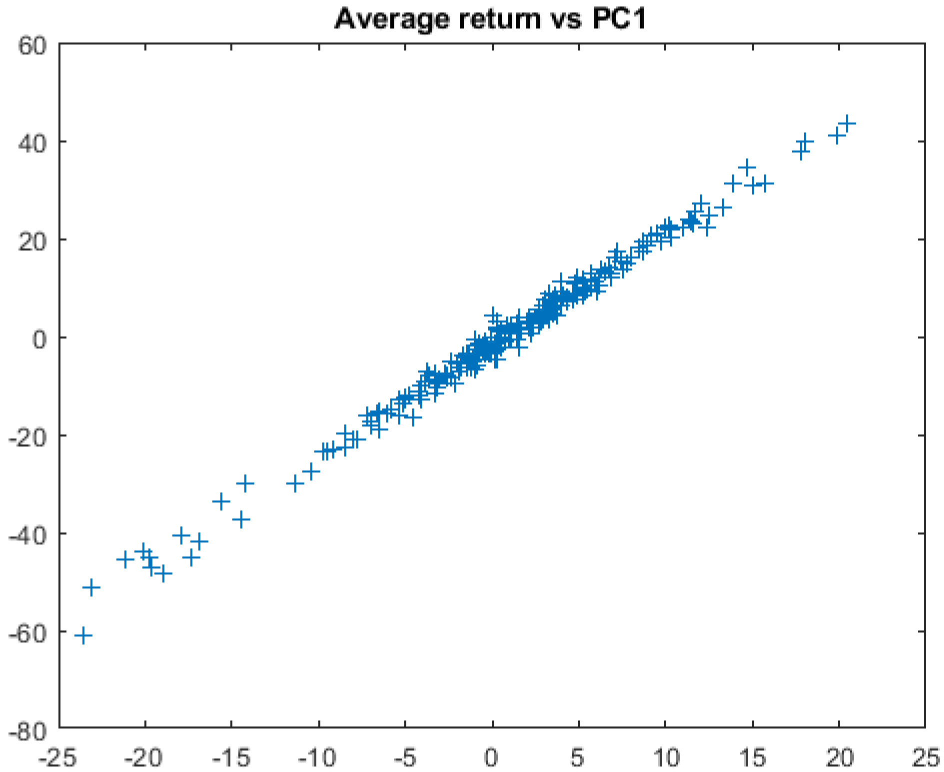
Conduct a principle component analysis of the 5 stock returns in Tsay's data set

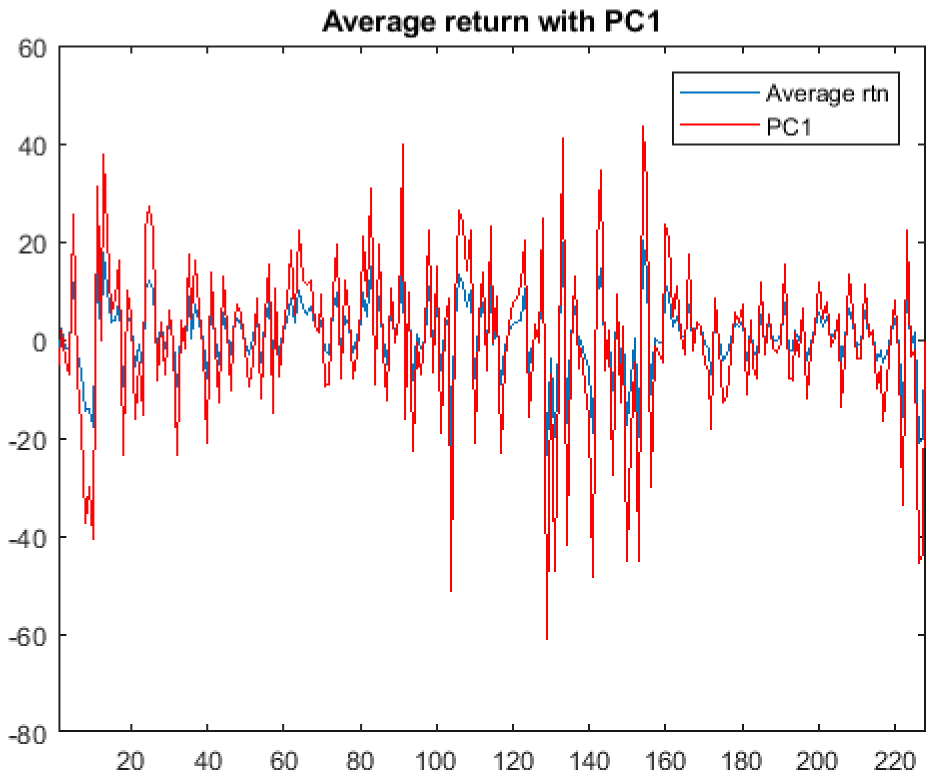
```
[pc_ret,score_ret,latent_ret] = princomp(Tsay_data); % score are
 linearly transformed dataset
pc_ret

% Lambdas, i.e. % of total variance explained per componenet and
 cumulative variance explained per componenet
[latent_ret latent_ret./sum(latent_ret) cumsum(latent_ret)./
sum(latent_ret)]

% Plot 3 componenents below individual stock returns
figure;subplot(4,1,1);plot(Tsay_data);
title('Stock returns');xlim([1 length(Tsay_data)]);
subplot(4,1,2);plot(score_ret(:,1));
title('1st Principle component');xlim([1 length(score_ret)]);
subplot(4,1,3);plot(score_ret(:,2));
title('2nd Principle component');xlim([1 length(score_ret)]);
subplot(4,1,4);plot(score_ret(:,3));
title('3rd Principle component');xlim([1 length(score_ret)]);


% The following demonstrates that the components are uncorrelated
corr(score_ret)

% plot all stock returns in one plot and all components in a second
 plot
% below the stock returns
figure;subplot(2,1,1);plot(Tsay_data);
title('Stock returns');xlim([1 length(Tsay_data)]);
legend('IBM','HPQ','INTC','JPM','BAC','Location','SouthWest','Orientation','horizo
subplot(2,1,2);plot(score_ret);
title('Principle components');xlim([1 length(score_ret)]);
legend('PC1','PC2','PC3','PC4','PC5','Location','SouthWest','Orientation','horizon

% Plot average returns against first componenet
figure;plot(mean(Tsay_data'),score_ret(:,1),'+');title('Average return
 vs PC1');

% Plot average returns with first componenet
figure;plot(mean(Tsay_data'));
hold on;plot(score_ret(:,1),'r');title('Average return with PC1');
legend('Average rtn','PC1');xlim([1 length(Tsay_data)]);

% Calculate correlation of first componenet with average returns
corrcoef(mean(Tsay_data'),score_ret(:,1));

% Bi-plots for components
figure;biplot(pc_ret(:,1:2));
figure;biplot(pc_ret(:,1:3));
```

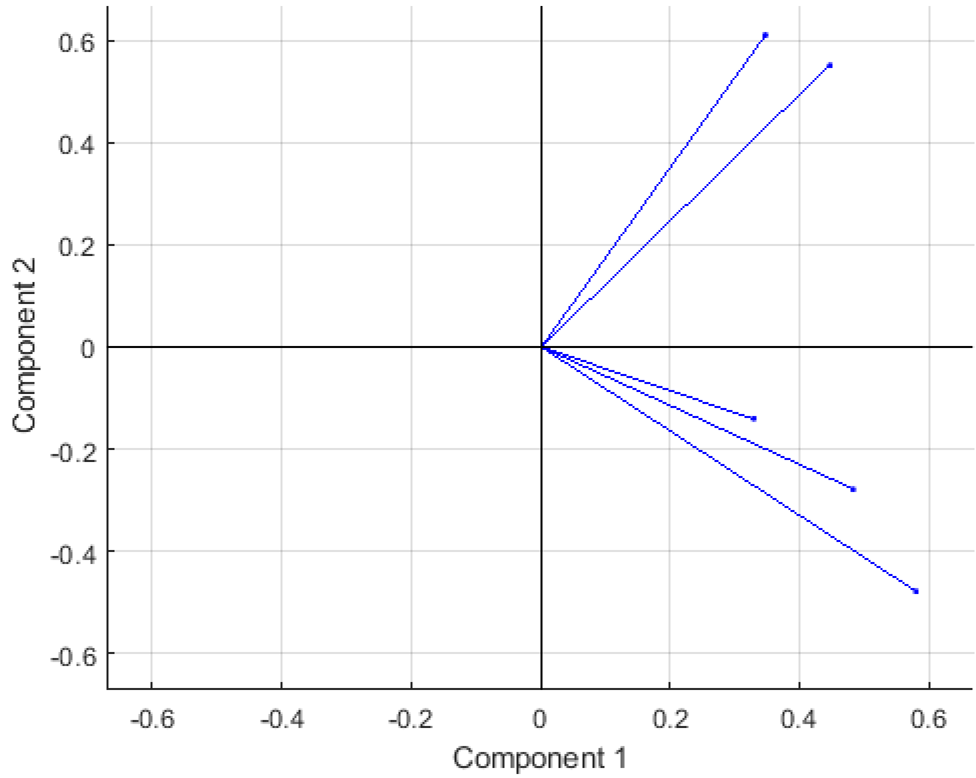
```
Warning: princomp will be removed in a future release. Use pca
 instead.

pc_ret =

    0.3298   -0.1393    0.2643    0.8954    0.0144
    0.4826   -0.2786    0.7009   -0.4298    0.1159
    0.5808   -0.4781   -0.6516   -0.0962    0.0163
    0.4476    0.5502   -0.0128   -0.0642   -0.7019
    0.3474    0.6097   -0.1188   -0.0093    0.7024


ans =

  284.1675    0.5349    0.5349
  112.9316    0.2126    0.7475
   57.4371    0.1081    0.8557
   46.8064    0.0881    0.9438
   29.8740    0.0562    1.0000


ans =

    1.0000    0.0000   -0.0000   -0.0000    0.0000
    0.0000    1.0000   -0.0000    0.0000    0.0000
   -0.0000   -0.0000    1.0000   -0.0000   -0.0000
   -0.0000    0.0000   -0.0000    1.0000    0.0000
    0.0000    0.0000   -0.0000    0.0000    1.0000
```

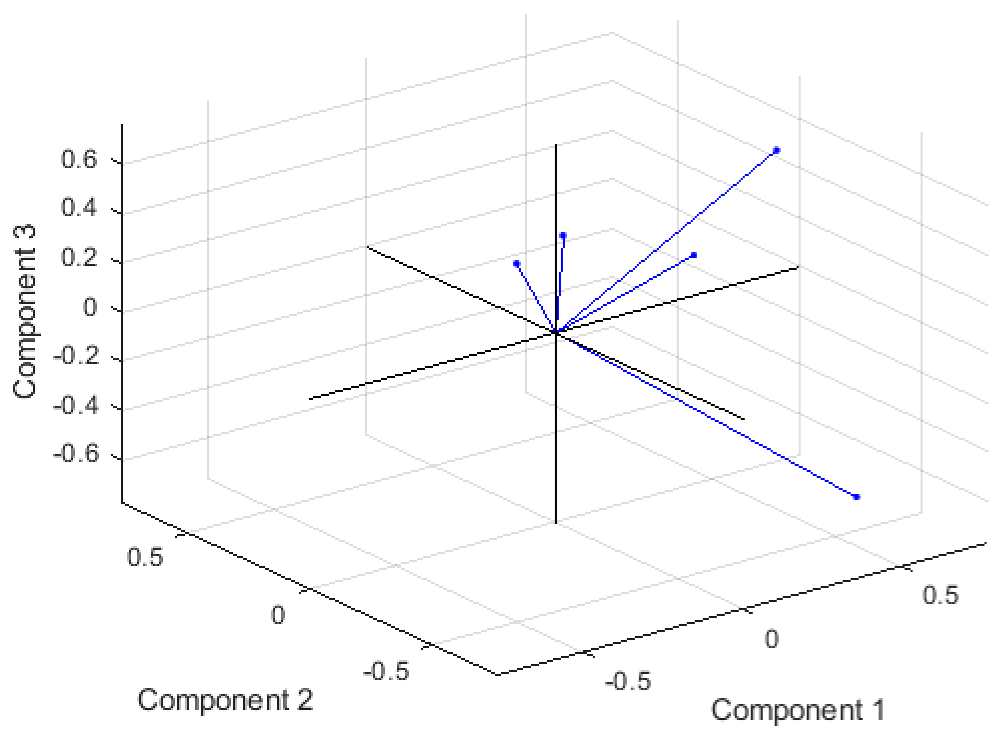

Stock returns
1st Principle component
2nd Principle component
3rd Principle component


Stock returns
IBM
HPQ
INTC
JPM
BAC
Principle components
PC1
PC2
PC3
PC4
PC5


Average return vs PC1
60
40
20
0
-20
-40
-60
-80
-25
-20
-15
-10
-5
0
5
10
15
20
25


Average return with PC1
Average rtn
PC1
60
40
20
0
-20
-40
-60
-80
20
40
60
80
100
120
140
160
180
200
220


0.6
0.4
0.2
0
-0.2
-0.4
-0.6
Component 2
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
Component 1


0.6
0.4
0.2
0
-0.2
-0.4
-0.6
Component 3
0.5
0
-0.5
Component 2
-0.5
0
0.5
Component 1

Published with MATLAB® R2017b