Table of Contents

Lab Sheet 6: Forecasting and forecast accuracy 1
Q1 Forecasting stock returns 1
Q1(a) Conduct exploratory data analysis (EDA) 1
Q1(b) Forecast last 24 months of returns using a variety of methods 3
Q1(c) Assess accuracy of forecasts 11
Q1(d) Calculate returns for investment strategies using forecast methods 16

Lab Sheet 6: Forecasting and forecast accuracy

Import "Tsay_FM_data.txt" as column vectors using the names in the first row for the created vectors save lab6.mat

```
load lab6.mat
```

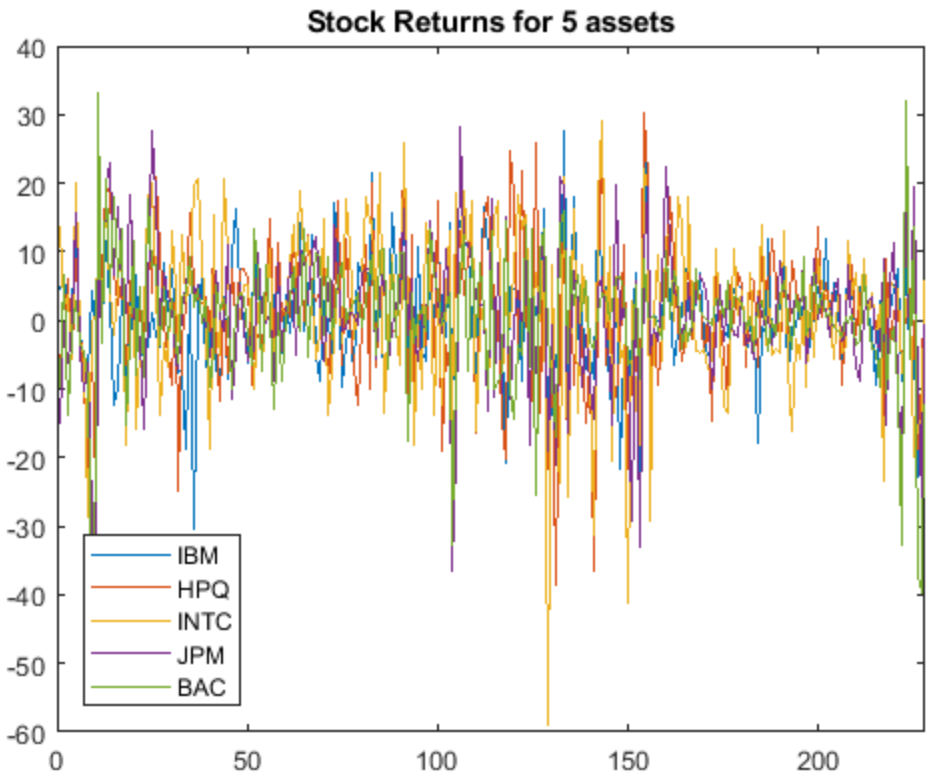
Q1 Forecasting stock returns

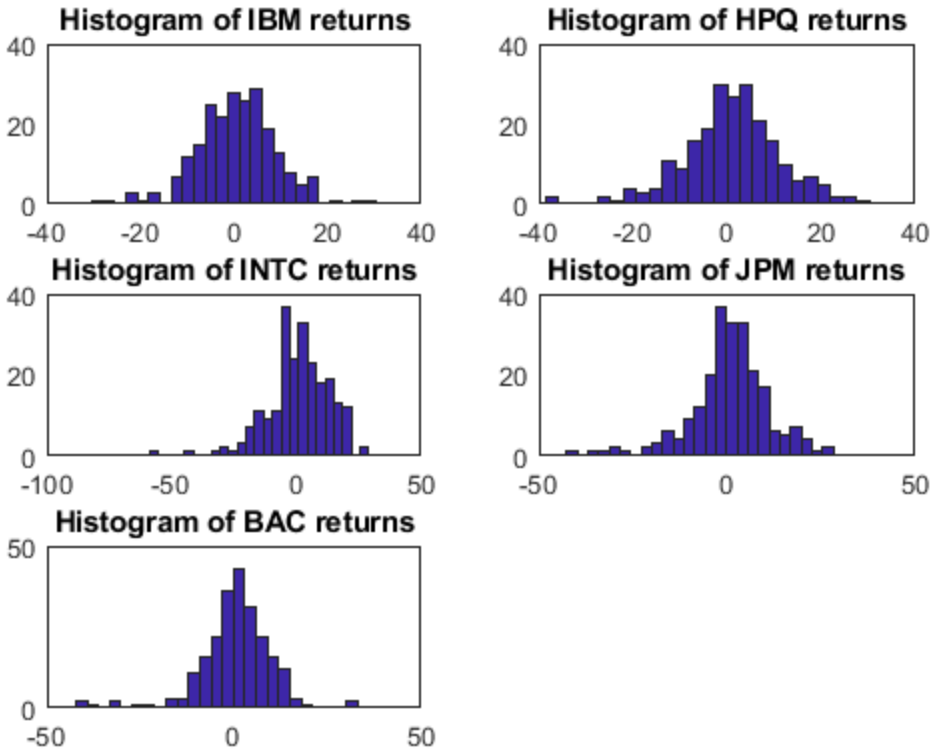
Q1(a) Conduct exploratory data analysis (EDA)

```
Tsay_data=[IBM HPQ INTC JPM BAC];

% Summary Stats
[mean(Tsay_data); median(Tsay_data); std(Tsay_data); min(Tsay_data);
 max(Tsay_data); skewness(Tsay_data); kurtosis(Tsay_data)]

% Plot returns data
figure;plot(Tsay_data);title('Stock Returns for 5 assets');
legend('IBM','HPQ','INTC','JPM','BAC','Location','SouthWest');
xlim([0,length(Tsay_data)]);

% Plot histograms of returns
figure;subplot(3,2,1);hist(IBM,25);title('Histogram of IBM returns');
subplot(3,2,2);hist(HPQ,25);title('Histogram of HPQ returns');
subplot(3,2,3);hist(INTC,25);title('Histogram of INTC returns');
subplot(3,2,4);hist(JPM,25);title('Histogram of JPM returns');
subplot(3,2,5);hist(BAC,25);title('Histogram of BAC returns');
```

```
ans =

    0.7004    0.9860    1.1957    0.8236    0.4105
    1.0071    1.2563    1.7156    0.9876    0.6448
    8.6393   10.5933   12.1036   10.2974    9.5826
  -30.3683  -38.5504  -58.8621  -42.5832  -42.2858
   30.2915   30.2989   29.1361   28.1970   33.0241
```

```
-0.1328   -0.3567   -0.8589   -0.7264   -1.0248
 4.2749    4.1579    5.4647    5.4365    7.6811
```





Q1(b) Forecast last 24 months of returns using a variety of methods

```
% Setup forecast and in-sample data
ret_f = Tsay_data(end-23:end,:); % forecast sample data is last 24
 months.
ret_is = Tsay_data(1:end-24,:);  % in-sample is all days before the
 last 24 months.

% Non-parametrics forecasts (all based on averages)
ret_f1 = mean(ret_is);   % first forecast is long-run mean of in-
sample period for each asset
ret_f2 = mean(ret_is(end-2:end,:));   % second forecast is mean of
 last 3 months of in-sample period
ret_f3 = mean(ret_is(end-11:end,:));   % third forecast is mean of
 last 12 months of in-sample period
ret_f4 = mean(ret_is(end-23:end,:));   % fourth forecast is mean of
 last 24 months of in-sample period
ret_f5 = ret_is(end,:);                % fifth forecast is return of
 the last month of in-sample period
ret_f6 = ones(1,5)*mean(mean(ret_is(end-5:end,:))); % sixth forecast
 is mean of last 6 month's returns over all 5 assets

% Parametric forecast (based on ARMA model)
```

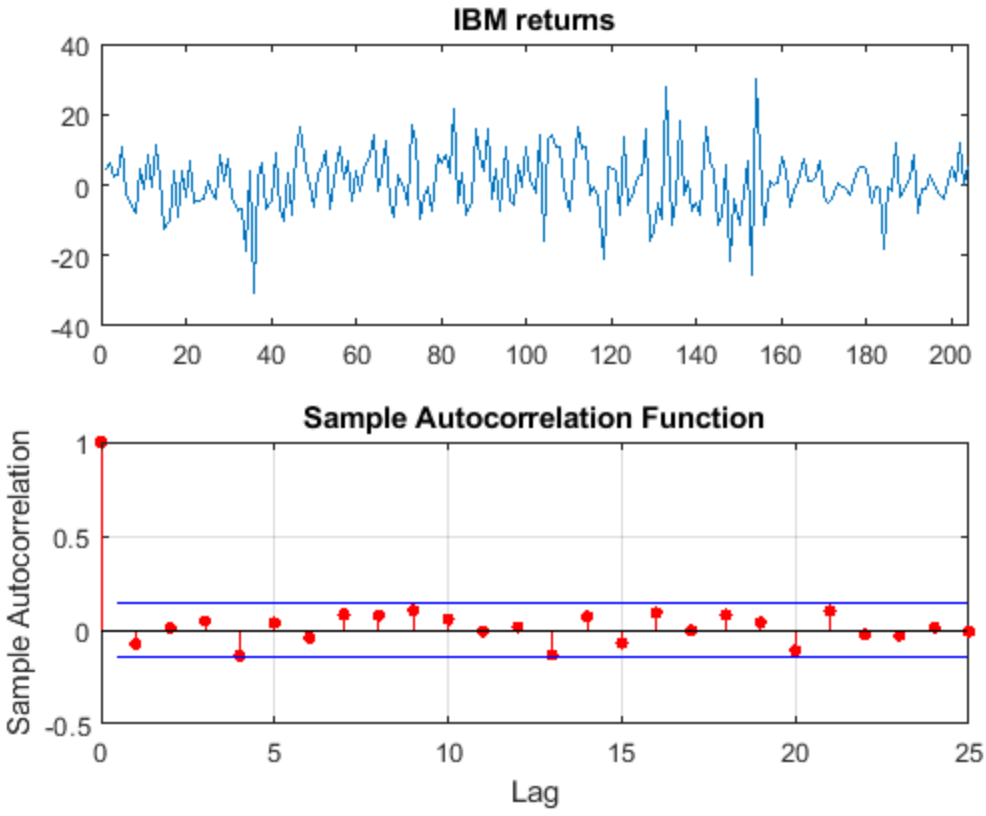
```
ret_f7 = zeros(1,5);  %setting up space for ARMA model forecasts

% 7th method forecasts using an ARMA model chosen for each series.

% Plot IBM series and ACF
figure;subplot(2,1,1);plot(ret_is(:,1));title('IBM
 returns');xlim([0,length(ret_is)]);
subplot(2,1,2);autocorr(ret_is(:,1), 25);
% LB test for AR effects on IBM
[H5, pval5, Qs5, CV5] = lbqtest(ret_is(:,1), 5, 0.05);
[H10, pval10, Qs10, CV10] = lbqtest(ret_is(:,1), 10, 0.05);
[pval5 pval10]
% I choose AR(4) for IBM, now fit it and forecast
Mdl=arima(4,0,0);
[EstMdl,EstParamCov,logL,info] = estimate(Mdl,ret_is(:,1));
[ret_f7(1), FMSE] = forecast(EstMdl,1,'Y0',ret_is(:,1));% 1-period
 forecasts

% Plot HPQ series and ACF
figure; subplot(2,1,1);plot(ret_is(:,2));title('HPQ
 returns');xlim([0,length(ret_is)]);
subplot(2,1,2);autocorr(ret_is(:,2), 25);figure(gcf)
% LB test for AR effects on HPQ
[H5, pval5, Qs5, CV5] = lbqtest(ret_is(:,2), 5, 0.05);
[H10, pval10, Qs10, CV10] = lbqtest(ret_is(:,2), 10, 0.05);
[pval5 pval10]
% I choose AR(7) for HPQ, now fit it and forecast
Mdl=arima(7,0,0);
[EstMdl,EstParamCov,logL,info] = estimate(Mdl,ret_is(:,2));
[ret_f7(2), FMSE] = forecast(EstMdl,1,'Y0',ret_is(:,2));% 1-period
 forecasts

% Plot INTC series and ACF
figure;subplot(2,1,1);plot(ret_is(:,3));title('INTC
 returns');xlim([0,length(ret_is)]);
subplot(2,1,2);autocorr(ret_is(:,3), 25);figure(gcf)
% LB test for AR effects on INTC
[H5, pval5, Qs5, CV5] = lbqtest(ret_is(:,3), 5, 0.05);
[H10, pval10, Qs10, CV10] = lbqtest(ret_is(:,3), 10, 0.05);
[pval5 pval10]
% I choose ARMA(1,1) for INTC, now fit it and forecast
Mdl=arima(1,0,1);
[EstMdl,EstParamCov,logL,info] = estimate(Mdl,ret_is(:,3));
[ret_f7(3), FMSE] = forecast(EstMdl,1,'Y0',ret_is(:,3));% 1-period
 forecasts

% Plot JPM series and ACF
figure;subplot(2,1,1);plot(ret_is(:,4));title('JPM
 returns');xlim([0,length(ret_is)]);
subplot(2,1,2);autocorr(ret_is(:,4), 25);figure(gcf)
% LB test for AR effects on JPM
[H5, pval5, Qs5, CV5] = lbqtest(ret_is(:,4), 5, 0.05);
[H10, pval10, Qs10, CV10] = lbqtest(ret_is(:,4), 10, 0.05);
[pval5 pval10]
```

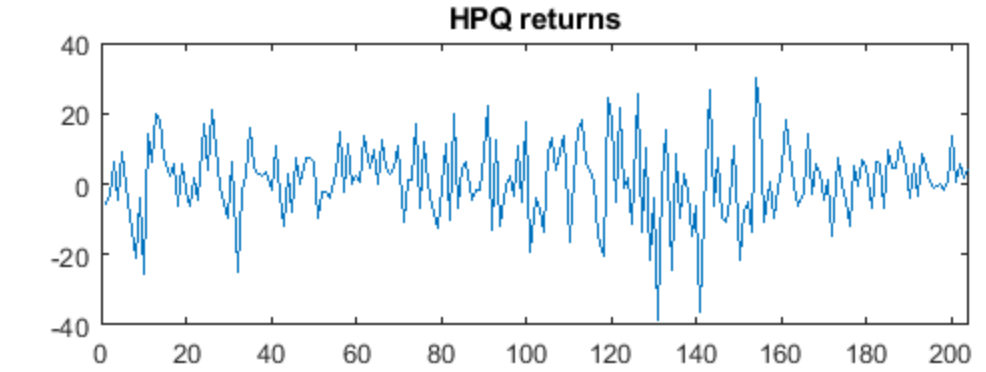
```
% I choose AR(4) for JPM, now fit it and forecast
Mdl=arima(4,0,0);
[EstMdl,EstParamCov,logL,info] = estimate(Mdl,ret_is(:,4));
[ret_f7(4), FMSE] = forecast(EstMdl,1,'Y0',ret_is(:,4));% 1-period
 forecasts

% Plot BAC series and ACF
figure; subplot(2,1,1);plot(ret_is(:,5));title('BAC
 returns');xlim([0,length(ret_is)]);
subplot(2,1,2);autocorr(ret_is(:,5), 25);figure(gcf)
% LB test for AR effects on BAC
[H5, pval5, Qs5, CV5] = lbqtest(ret_is(:,5), 5, 0.05);
[H10, pval10, Qs10, CV10] = lbqtest(ret_is(:,5), 10, 0.05);
[pval5 pval10]
% I choose AR(9) for BAC, now fit it and forecast
Mdl=arima(9,0,0);
[EstMdl,EstParamCov,logL,info] = estimate(Mdl,ret_is(:,5));
[ret_f7(5), FMSE] = forecast(EstMdl,1,'Y0',ret_is(:,5));% 1-period
 forecasts

% Note that these are all vectors, representing the forecasts for all
 5 series.
% These forecasts should be compared with ret_f(1,:), ie actual return
 on day 1 of forecast period

% Create a loop to create forecasts for 24 days in forecast sample
% (Loop starts at 2nd day of forecast sample as day one already
 calculated
% above)
for t=2:24
  ret_is = Tsay_data(t:end-25+t,:); % you need to update in-sample
 data to continuously perform forecast
  ret_f1(t,:) = mean(ret_is);
  ret_f2(t,:) = mean(ret_is(end-2:end,:));
  ret_f3(t,:) = mean(ret_is(end-11:end,:));
  ret_f4(t,:) = mean(ret_is(end-23:end,:));
  ret_f5(t,:) = ret_is(end,:);
  ret_f6(t,:) = ones(1,5)*mean(mean(ret_is(end-5:end,:))); % mean of
 last 6 months across all 5 assets

  %7th method forecasts are from an ARMA model chosen for each series.
  Mdl=arima(4,0,0);
  EstMdl = estimate(Mdl,ret_is(:,1),'Display','Off'); % Note: I
 suppress the display of estimation results here
  [ret_f7(t,1), FMSE] = forecast(EstMdl,1,'Y0',ret_is(:,1)); % 1-
period forecasts

  Mdl=arima(7,0,0);
  EstMdl = estimate(Mdl,ret_is(:,2),'Display','Off');
  [ret_f7(t,2), FMSE] = forecast(EstMdl,1,'Y0',ret_is(:,2)); % 1-
period forecasts

  Mdl=arima(1,0,1);
  EstMdl = estimate(Mdl,ret_is(:,3),'Display','Off');
```

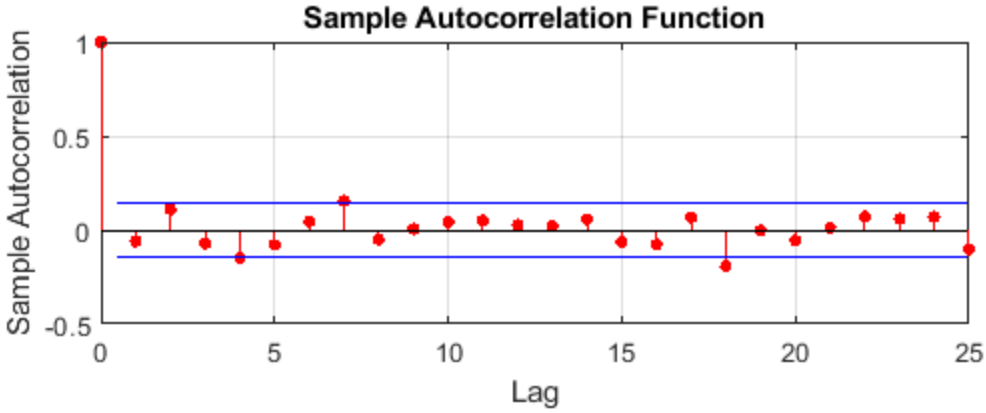
```
  [ret_f7(t,3), FMSE] = forecast(EstMdl,1,'Y0',ret_is(:,3)); % 1-
period forecasts

  Mdl=arima(4,0,0);
  EstMdl = estimate(Mdl,ret_is(:,4),'Display','Off');
  [ret_f7(t,4), FMSE] = forecast(EstMdl,1,'Y0',ret_is(:,4)); % 1-
period forecasts

  Mdl=arima(9,0,0);
  EstMdl = estimate(Mdl,ret_is(:,5),'Display','Off');
  [ret_f7(t,5), FMSE] = forecast(EstMdl,1,'Y0',ret_is(:,5)); % 1-
period forecasts
end
% we now have 24 sets of forecasts one-step-ahead for all five series
 using each of 7 different methods.


ans =

    0.3201    0.2854


    ARIMA(4,0,0) Model:
    --------------------
    Conditional Probability Distribution: Gaussian

                                 Standard          t
     Parameter       Value          Error       Statistic
    -----------   -----------   ------------   -----------
     Constant       0.949798       0.61513        1.54406
        AR{1}     -0.0681231     0.0643003       -1.05945
        AR{2}      0.0089518     0.0689962       0.129743
        AR{3}      0.0385014     0.0678075       0.567805
        AR{4}      -0.132648     0.0727925       -1.82227
     Variance        74.7258       6.90308         10.825

ans =

    0.0671    0.0793


    ARIMA(7,0,0) Model:
    --------------------
    Conditional Probability Distribution: Gaussian

                                 Standard          t
     Parameter       Value          Error       Statistic
    -----------   -----------   ------------   -----------
     Constant         1.1392       0.834785        1.36467
        AR{1}     -0.0781307     0.0676098       -1.15561
        AR{2}        0.14116     0.0617541        2.28583
        AR{3}     -0.0251155     0.0617505      -0.406725
        AR{4}      -0.173265     0.0800552       -2.16431
        AR{5}      -0.103902     0.0740393       -1.40334
```

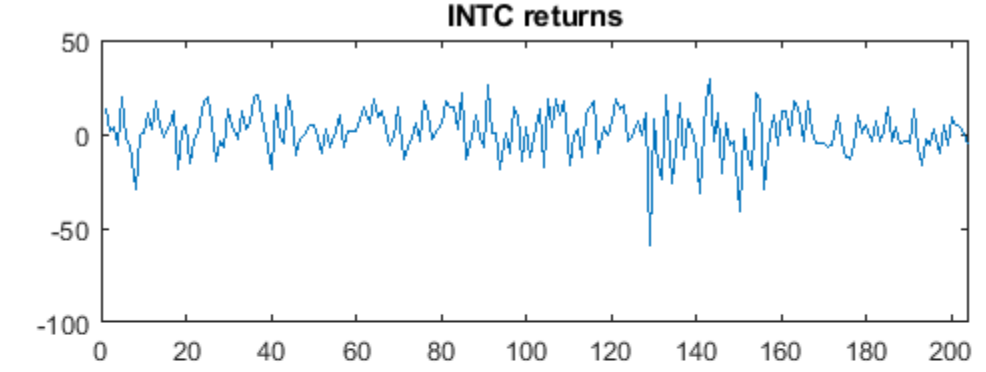
```
        AR{6}       0.080277     0.0815109       0.984862
        AR{7}        0.17388     0.0609287        2.85383
     Variance        108.955       9.13214         11.931

ans =

    0.5742    0.2850


    ARIMA(1,0,1) Model:
    --------------------
    Conditional Probability Distribution: Gaussian

                                  Standard          t
     Parameter       Value          Error       Statistic
    -----------   -----------   ------------   -----------
     Constant        2.97794       1.85707        1.60357
        AR{1}      -0.915635     0.0228203       -40.1236
        MA{1}       0.975308     0.0196014         49.757
     Variance        148.406       10.8915        13.6258

ans =

    0.1724    0.5283


    ARIMA(4,0,0) Model:
    --------------------
    Conditional Probability Distribution: Gaussian

                                  Standard          t
     Parameter       Value          Error       Statistic
    -----------   -----------   ------------   -----------
     Constant        1.31459      0.798901         1.6455
        AR{1}      0.0539292     0.0517241        1.04263
        AR{2}     -0.0264024     0.0518593      -0.509117
        AR{3}     -0.0249335     0.0706593      -0.352869
        AR{4}      -0.175446     0.0673315       -2.60571
     Variance        99.5887       7.00517        14.2164

ans =

    0.8590    0.1643


    ARIMA(9,0,0) Model:
    --------------------
    Conditional Probability Distribution: Gaussian

                                  Standard          t
     Parameter       Value          Error       Statistic
    -----------   -----------   ------------   -----------
     Constant        1.23654      0.656851        1.88252
        AR{1}     -0.0688799     0.0517699        -1.3305
```

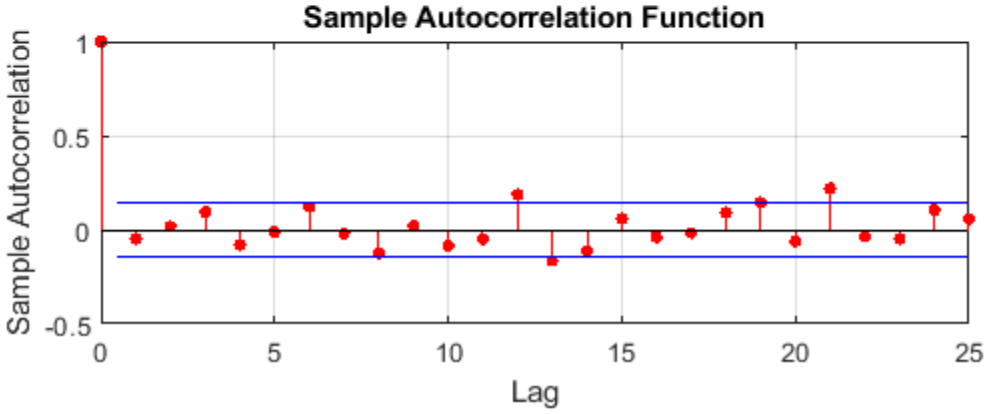
AR{2}	-0.0219241	0.0472072	-0.464423
AR{3}	-0.0264683	0.0799815	-0.330931
AR{4}	-0.0126271	0.0607218	-0.20795
AR{5}	0.00473764	0.0732885	0.0646437
AR{6}	-0.108067	0.0564615	-1.91399
AR{7}	-0.0893957	0.072678	-1.23002
AR{8}	-0.0794686	0.0640872	-1.24001
AR{9}	0.193969	0.0805104	2.40925
Variance	67.6613	4.59443	14.7268

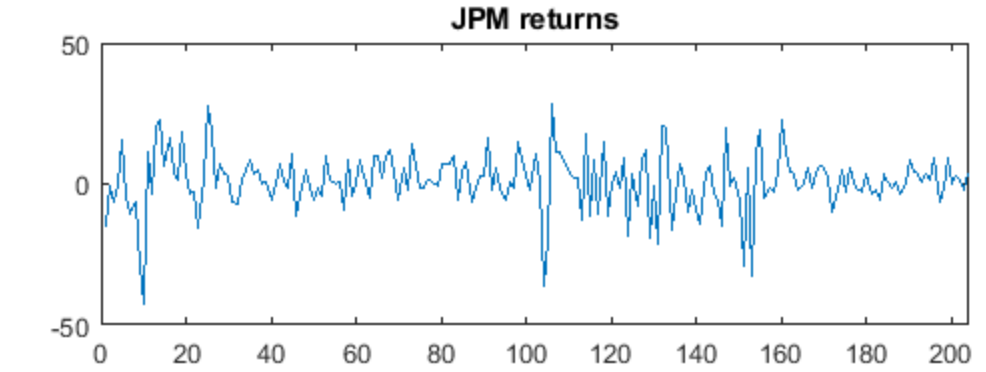


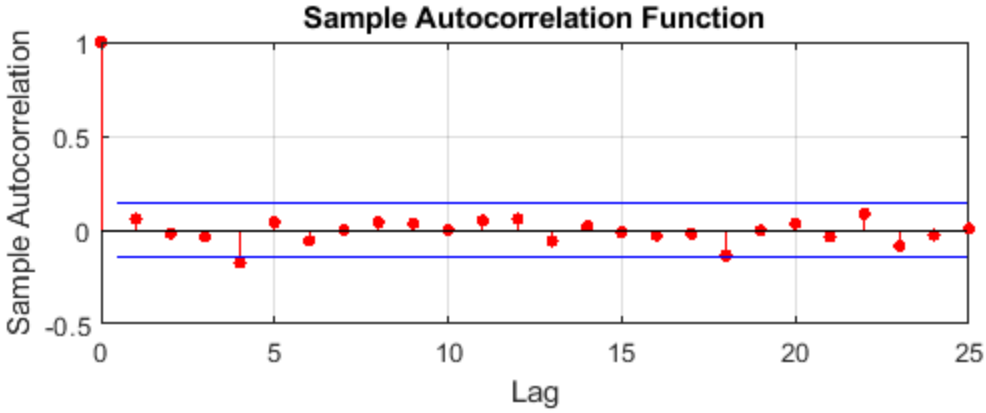

HPQ returns
40
20
0
-20
-40
0
20
40
60
80
100
120
140
160
180
200

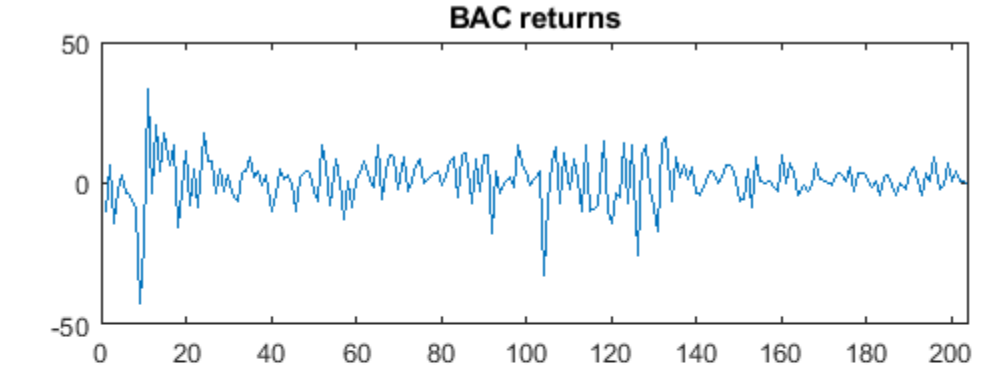

Sample Autocorrelation Function
Sample Autocorrelation
1
0.5
0
-0.5
0
5
10
15
20
25
Lag

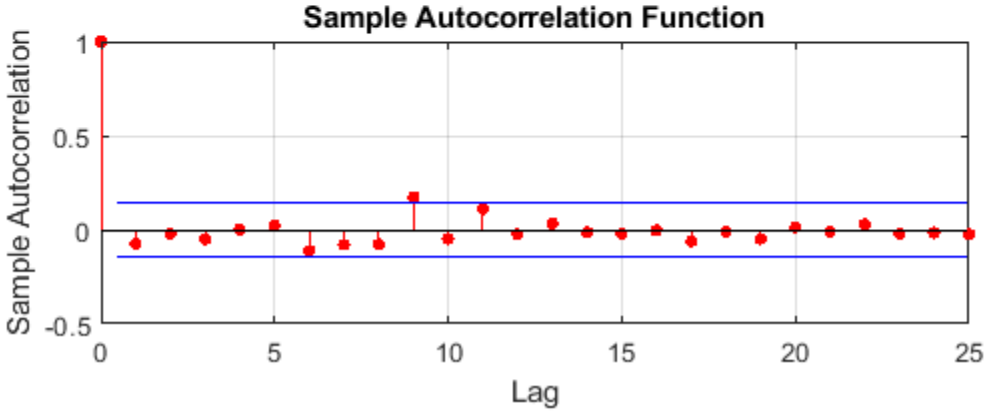

INTC returns
50
0
-50
-100
0
20
40
60
80
100
120
140
160
180
200


Sample Autocorrelation Function
Sample Autocorrelation
1
0.5
0
-0.5
0
5
10
15
20
25
Lag


JPM returns
50
0
-50
0
20
40
60
80
100
120
140
160
180
200


Sample Autocorrelation Function
Sample Autocorrelation
1
0.5
0
-0.5
0
5
10
15
20
25
Lag


BAC returns
50
0
-50
0
20
40
60
80
100
120
140
160
180
200


Sample Autocorrelation Function
Sample Autocorrelation
1
0.5
0
-0.5
0
5
10
15
20
25
Lag

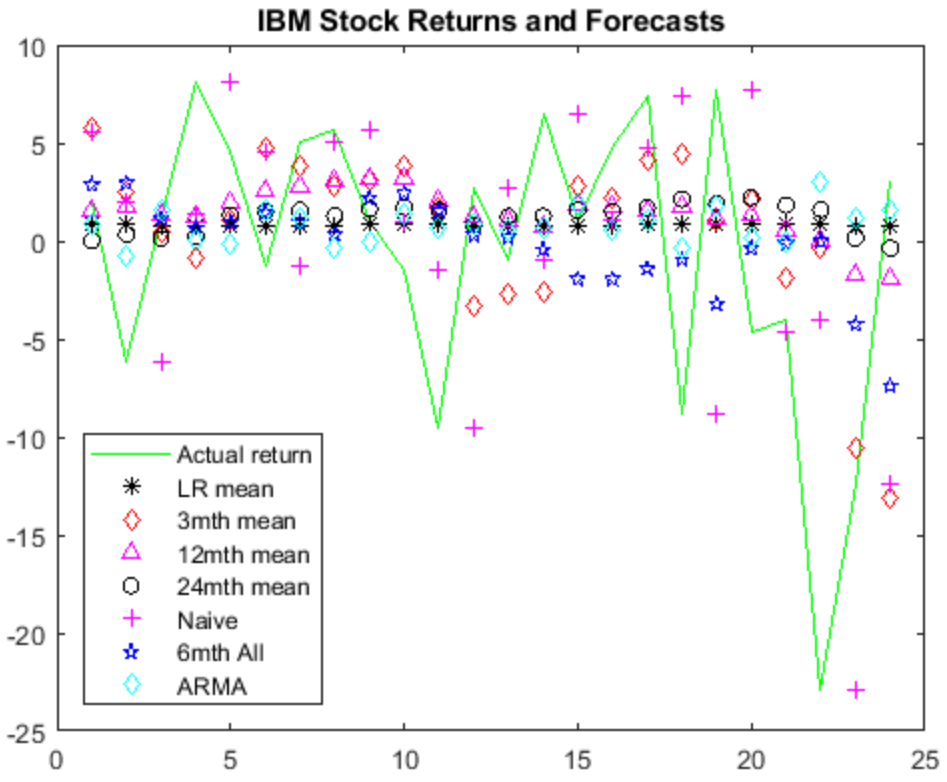
Q1(c) Assess accuracy of forecasts

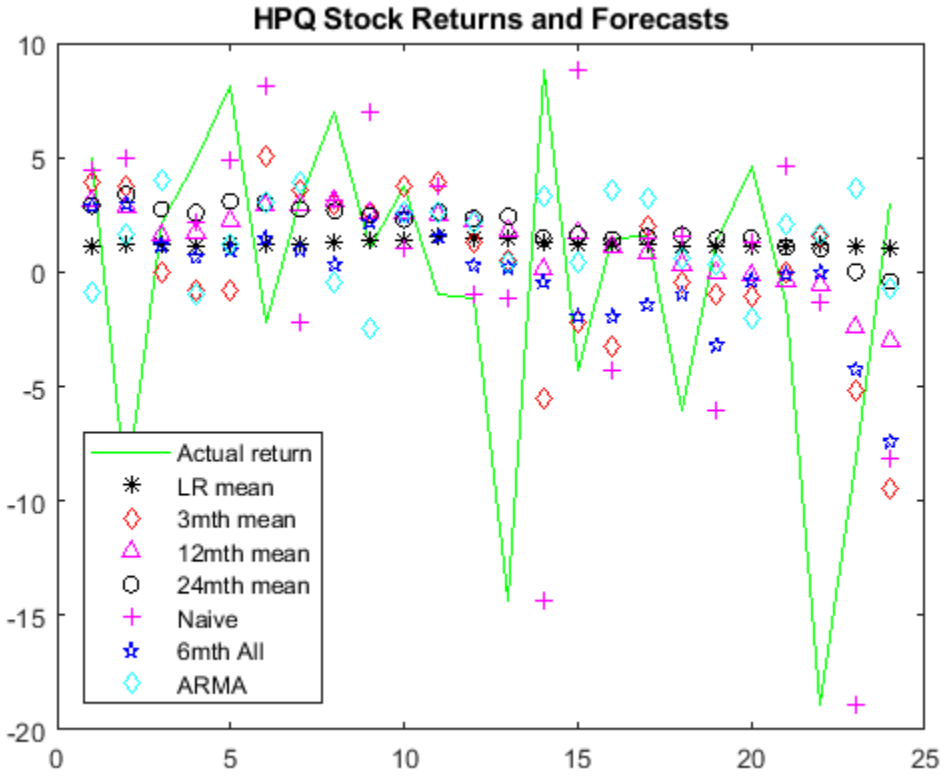
Plot forecasts and measure forecast accuracy for each stock separately and calculate forecast accuracy measures Note that MAPE is undefined if return is zero (can't divide by zero) so won't be used here (getfa()attempts to calculate it and gets an error)

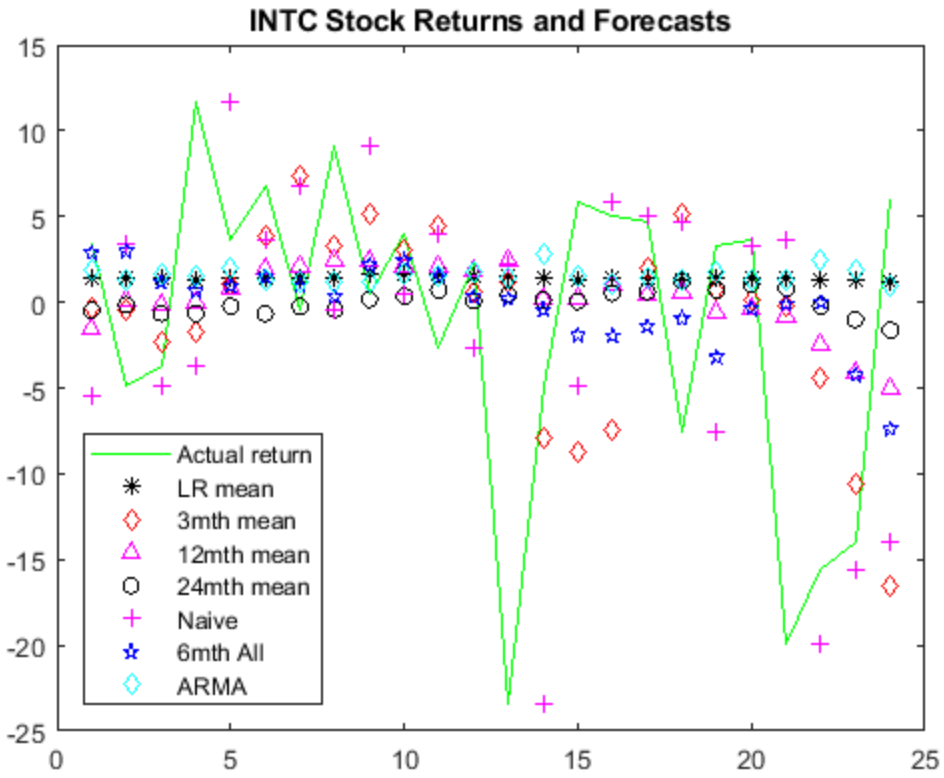
```
% IBM
% Plot forecasts from all 7 methods along with actual returns
figure;plot(ret_f(:,1),'g');
hold on;plot(ret_f1(:,1),'k*');plot(ret_f2(:,1),'rd');plot(ret_f3(:,1),'m^');
plot(ret_f4(:,1),'ko');plot(ret_f5(:,1),'m
+');plot(ret_f6(:,1),'bp');plot(ret_f7(:,1),'cd')
title('IBM Stock Returns and Forecasts');
legend('Actual return','LR mean','3mth mean','12mth mean','24mth
 mean','Naive','6mth All','ARMA','Location','SouthWest');

% Calculate RMSE, MAD, and MAP for all forecast methods using getfa()
[rmse1, mad1, map1]=getfa(ret_f(:,1),ret_f1(:,1));
[rmse2, mad2, map2]=getfa(ret_f(:,1),ret_f2(:,1));
[rmse3, mad3, map3]=getfa(ret_f(:,1),ret_f3(:,1));
[rmse4, mad4, map4]=getfa(ret_f(:,1),ret_f4(:,1));
[rmse5, mad5, map5]=getfa(ret_f(:,1),ret_f5(:,1));
[rmse6, mad6, map6]=getfa(ret_f(:,1),ret_f6(:,1));
[rmse7, mad7, map7]=getfa(ret_f(:,1),ret_f7(:,1));

% Display accuracy measures
[rmse1 rmse2 rmse3 rmse4 rmse5 rmse6 rmse7;mad1 mad2 mad3 mad4 mad5
 mad6 mad7]

% HPQ
% Plot forecasts from all 7 methods along with actual returns
figure;plot(ret_f(:,2),'g')
hold on;plot(ret_f1(:,2),'k*');plot(ret_f2(:,2),'rd');plot(ret_f3(:,2),'m^');
plot(ret_f4(:,2),'ko');plot(ret_f5(:,2),'m
+');plot(ret_f6(:,2),'bp');plot(ret_f7(:,2),'cd')
title('HPQ Stock Returns and Forecasts');
legend('Actual return','LR mean','3mth mean','12mth mean','24mth
 mean','Naive','6mth All','ARMA','Location','SouthWest');

% Calculate RMSE, MAD, and MAP for all forecast methods using getfa()
[rmse1, mad1, map1]=getfa(ret_f(:,2),ret_f1(:,2));
[rmse2, mad2, map2]=getfa(ret_f(:,2),ret_f2(:,2));
[rmse3, mad3, map3]=getfa(ret_f(:,2),ret_f3(:,2));
[rmse4, mad4, map4]=getfa(ret_f(:,2),ret_f4(:,2));
[rmse5, mad5, map5]=getfa(ret_f(:,2),ret_f5(:,2));
[rmse6, mad6, map6]=getfa(ret_f(:,2),ret_f6(:,2));
[rmse7, mad7, map7]=getfa(ret_f(:,2),ret_f7(:,2));

% Display accuracy measures
[rmse1 rmse2 rmse3 rmse4 rmse5 rmse6 rmse7;mad1 mad2 mad3 mad4 mad5
 mad6 mad7]

% INTC
```

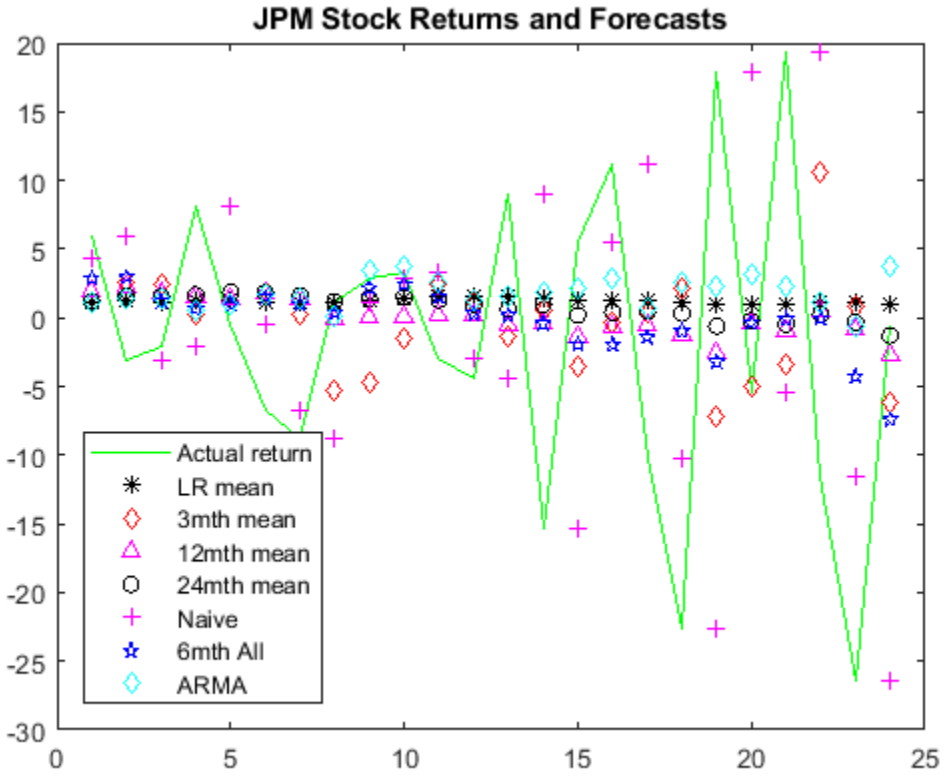
```
% Plot forecasts from all 7 methods along with actual returns
figure;plot(ret_f(:,3),'g')
hold on;plot(ret_f1(:,3),'k*');plot(ret_f2(:,3),'rd');plot(ret_f3(:,3),'m^');
plot(ret_f4(:,3),'ko');plot(ret_f5(:,3),'m
+');plot(ret_f6(:,3),'bp');plot(ret_f7(:,3),'cd')
title('INTC Stock Returns and Forecasts');
legend('Actual return','LR mean','3mth mean','12mth mean','24mth
 mean','Naive','6mth All','ARMA','Location','SouthWest');

% Calculate RMSE, MAD, and MAP for all forecast methods using getfa()
[rmse1, mad1, map1]=getfa(ret_f(:,3),ret_f1(:,3));
[rmse2, mad2, map2]=getfa(ret_f(:,3),ret_f2(:,3));
[rmse3, mad3, map3]=getfa(ret_f(:,3),ret_f3(:,3));
[rmse4, mad4, map4]=getfa(ret_f(:,3),ret_f4(:,3));
[rmse5, mad5, map5]=getfa(ret_f(:,3),ret_f5(:,3));
[rmse6, mad6, map6]=getfa(ret_f(:,3),ret_f6(:,3));
[rmse7, mad7, map7]=getfa(ret_f(:,3),ret_f7(:,3));

% Display accuracy measures
[rmse1 rmse2 rmse3 rmse4 rmse5 rmse6 rmse7;mad1 mad2 mad3 mad4 mad5
 mad6 mad7]

% JPM
% Plot forecasts from all 7 methods along with actual returns
figure;plot(ret_f(:,4),'g')
hold on;plot(ret_f1(:,4),'k*');plot(ret_f2(:,4),'rd');plot(ret_f3(:,4),'m^');
plot(ret_f4(:,4),'ko');plot(ret_f5(:,4),'m
+');plot(ret_f6(:,4),'bp');plot(ret_f7(:,4),'cd')
title('JPM Stock Returns and Forecasts');
legend('Actual return','LR mean','3mth mean','12mth mean','24mth
 mean','Naive','6mth All','ARMA','Location','SouthWest');

% Calculate RMSE, MAD, and MAP for all forecast methods using getfa()
[rmse1, mad1, map1]=getfa(ret_f(:,4),ret_f1(:,4));
[rmse2, mad2, map2]=getfa(ret_f(:,4),ret_f2(:,4));
[rmse3, mad3, map3]=getfa(ret_f(:,4),ret_f3(:,4));
[rmse4, mad4, map4]=getfa(ret_f(:,4),ret_f4(:,4));
[rmse5, mad5, map5]=getfa(ret_f(:,4),ret_f5(:,4));
[rmse6, mad6, map6]=getfa(ret_f(:,4),ret_f6(:,4));
[rmse7, mad7, map7]=getfa(ret_f(:,4),ret_f7(:,4));

% Display accuracy measures
[rmse1 rmse2 rmse3 rmse4 rmse5 rmse6 rmse7;mad1 mad2 mad3 mad4 mad5
 mad6 mad7]

% BAC
% Plot forecasts from all 7 methods along with actual returns
figure;plot(ret_f(:,5),'g')
hold on;plot(ret_f1(:,5),'k*');plot(ret_f2(:,5),'rd');plot(ret_f3(:,5),'m^');
plot(ret_f4(:,5),'ko');plot(ret_f5(:,5),'m
+');plot(ret_f6(:,5),'bp');plot(ret_f7(:,5),'cd')
title('BAC Stock Returns and Forecasts');
legend('Actual return','LR mean','3mth mean','12mth mean','24mth
 mean','Naive','6mth All','ARMA','Location','SouthWest');
```

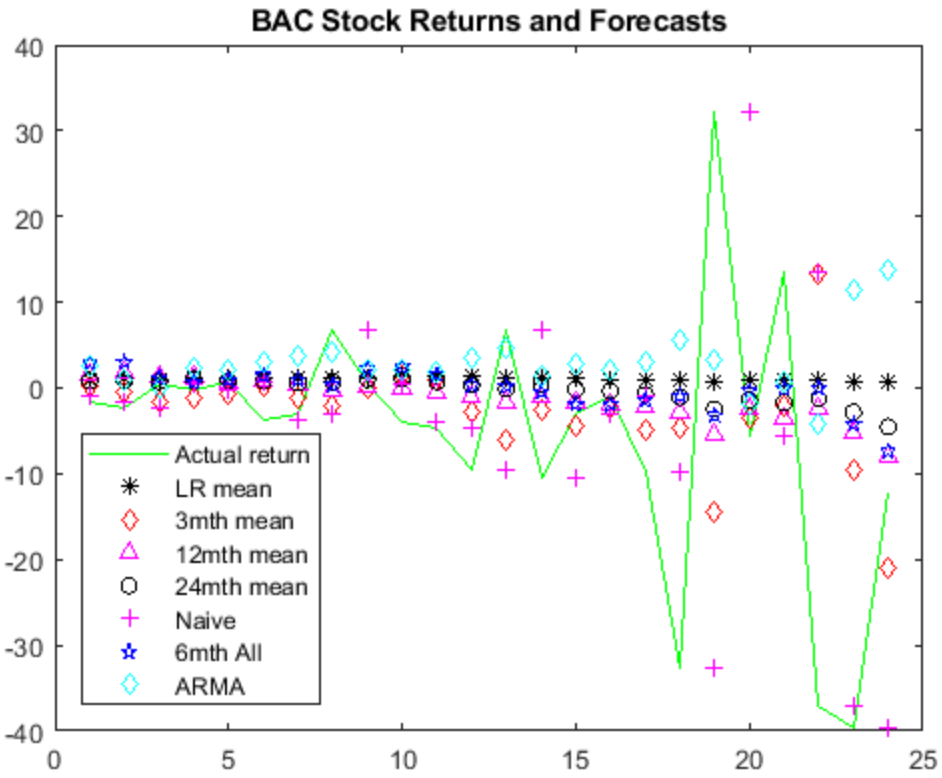
```
% Calculate RMSE, MAD, and MAP for all forecast methods using getfa()
[rmse1, mad1, map1]=getfa(ret_f(:,5),ret_f1(:,5));
[rmse2, mad2, map2]=getfa(ret_f(:,5),ret_f2(:,5));
[rmse3, mad3, map3]=getfa(ret_f(:,5),ret_f3(:,5));
[rmse4, mad4, map4]=getfa(ret_f(:,5),ret_f4(:,5));
[rmse5, mad5, map5]=getfa(ret_f(:,5),ret_f5(:,5));
[rmse6, mad6, map6]=getfa(ret_f(:,5),ret_f6(:,5));
[rmse7, mad7, map7]=getfa(ret_f(:,5),ret_f7(:,5));

% Display accuracy measures
[rmse1 rmse2 rmse3 rmse4 rmse5 rmse6 rmse7;mad1 mad2 mad3 mad4 mad5
 mad6 mad7]


ans =

    7.4037    8.0989    7.2561    7.5729    9.1781    7.7593    7.6178
    5.4269    6.1751    5.4123    5.6249    7.6163    6.1224    5.4494


ans =

    6.9568    7.9224    6.8128    7.0721    9.5984    7.0207    7.4179
    4.9745    5.8034    4.9160    4.9121    7.7031    5.3174    5.8053


ans =

    9.3333   10.4181    8.9583    9.2150   11.4107    9.3153    9.4789
    6.8075    7.9009    6.8789    7.0164    8.9456    7.3503    6.8555


ans =

   11.2844   13.2072   11.2778   11.2487   17.0078   11.0187   11.1621
    8.7453   10.7233    8.9909    8.7634   13.2058    8.8261    8.7477


ans =

   15.9958   17.3274   15.0590   15.3205   21.1524   15.3365   17.7810
   10.6433   10.4629    9.5948    9.9289   13.0770   10.0568   11.8390
```


IBM Stock Returns and Forecasts
Actual return
LR mean
3mth mean
12mth mean
24mth mean
Naive
6mth All
ARMA


HPQ Stock Returns and Forecasts
Actual return
LR mean
3mth mean
12mth mean
24mth mean
Naive
6mth All
ARMA


INTC Stock Returns and Forecasts
Actual return
LR mean
3mth mean
12mth mean
24mth mean
Naive
6mth All
ARMA


JPM Stock Returns and Forecasts
Actual return
LR mean
3mth mean
12mth mean
24mth mean
Naive
6mth All
ARMA



Q1(d) Calculate returns for investment strategies using forecast methods

```
% performance of equally weighted portfolio
ewret=mean(ret_f')';

% Other methods that vary with time
for t=1:24
    % choose max predicted return asset
    [rf1,orf1]=sort(ret_f1(t,:)); % sort forecasts from mthd 1 in
 period t
    ret_inv1(t) =ret_f(t,orf1(5)); % choose actual return in period t
 for asset with highest forecast
    [rf2,orf2]=sort(ret_f2(t,:));
    ret_inv2(t) =ret_f(t,orf2(5));
    [rf3,orf3]=sort(ret_f3(t,:));
    ret_inv3(t) =ret_f(t,orf3(5));
    [rf4,orf4]=sort(ret_f4(t,:));
    ret_inv4(t) =ret_f(t,orf4(5));
    [rf5,orf5]=sort(ret_f5(t,:));
    ret_inv5(t) =ret_f(t,orf5(5));
    [rf6,orf6]=sort(ret_f6(t,:));
    ret_inv6(t) =ret_f(t,orf6(5));
    [rf7,orf7]=sort(ret_f7(t,:));
```

```
    ret_inv7(t) =ret_f(t,orf7(5));

    % weight by predicted returns
    pmwt_r1(t) = sum(ret_f1(t,:).*ret_f(t,:))/sum(ret_f1(t,:));
    pmwt_r2(t) = sum(ret_f2(t,:).*ret_f(t,:))/sum(ret_f2(t,:));
    pmwt_r3(t) = sum(ret_f3(t,:).*ret_f(t,:))/sum(ret_f3(t,:));
    pmwt_r4(t) = sum(ret_f4(t,:).*ret_f(t,:))/sum(ret_f4(t,:));
    pmwt_r5(t) = sum(ret_f5(t,:).*ret_f(t,:))/sum(ret_f5(t,:));
    pmwt_r6(t) = sum(ret_f6(t,:).*ret_f(t,:))/sum(ret_f6(t,:));
    pmwt_r7(t) = sum(ret_f7(t,:).*ret_f(t,:))/sum(ret_f7(t,:));

    % weight by inverse absolute predicted returns
    piamwt_r1(t) = sum(ret_f(t,:)./abs(ret_f1(t,:)))/sum(1./
abs(ret_f1(t,:)));
    piamwt_r2(t) = sum(ret_f(t,:)./abs(ret_f2(t,:)))/sum(1./
abs(ret_f2(t,:)));
    piamwt_r3(t) = sum(ret_f(t,:)./abs(ret_f3(t,:)))/sum(1./
abs(ret_f3(t,:)));
    piamwt_r4(t) = sum(ret_f(t,:)./abs(ret_f4(t,:)))/sum(1./
abs(ret_f4(t,:)));
    piamwt_r5(t) = sum(ret_f(t,:)./abs(ret_f5(t,:)))/sum(1./
abs(ret_f5(t,:)));
    piamwt_r6(t) = sum(ret_f(t,:)./abs(ret_f6(t,:)))/sum(1./
abs(ret_f6(t,:)));
    piamwt_r7(t) = sum(ret_f(t,:)./abs(ret_f7(t,:)))/sum(1./
abs(ret_f7(t,:)));
end

% Mean and standard deviation of portfolio returns over forecast
 period
% using each forecast method and weighting strategy
   [mean(ewret) std(ewret)
    mean(ret_inv1) std(ret_inv1);
    mean(ret_inv2) std(ret_inv2);
    mean(ret_inv3) std(ret_inv3);
    mean(ret_inv4) std(ret_inv4);
    mean(ret_inv5) std(ret_inv5);
    mean(ret_inv6) std(ret_inv6);
    mean(ret_inv7) std(ret_inv7);
    mean(pmwt_r1) std(pmwt_r1);
    mean(pmwt_r2) std(pmwt_r2);
    mean(pmwt_r3) std(pmwt_r3);
    mean(pmwt_r4) std(pmwt_r4);
    mean(pmwt_r5) std(pmwt_r5);
    mean(pmwt_r6) std(pmwt_r6);
    mean(pmwt_r7) std(pmwt_r7);
    mean(piamwt_r1) std(piamwt_r1);
    mean(piamwt_r2) std(piamwt_r2);
    mean(piamwt_r3) std(piamwt_r3);
    mean(piamwt_r4) std(piamwt_r4);
    mean(piamwt_r5) std(piamwt_r5);
    mean(piamwt_r6) std(piamwt_r6);
    mean(piamwt_r7) std(piamwt_r7);]
```

```
% Mean and standard deviation of each stock over forecast period
[mean(ret_f); std(ret_f)]
```

```
ans =

   -1.7170    7.7655
   -1.5834    9.5860
   -3.7795   11.4434
   -1.5329    9.0961
   -0.7817    8.1310
   -2.6841    7.4398
   -4.9962   15.1487
   -2.1898   14.8273
   -1.7315    7.4450
    2.1422   12.3123
   -1.4799   39.8587
   -5.8518   16.8359
   -0.9064   20.1797
   -1.7170    7.7655
   -2.2117   10.0096
   -1.6782    8.1511
   -2.7591    8.1099
   -1.5739    7.1981
   -0.6396    6.9849
   -1.8012    7.1149
   -1.7170    7.7655
   -1.8047    6.7657
```

```
ans =

   -0.4657   -0.4675   -1.1462   -1.5095   -4.9962
    7.4301    6.8777    9.1601   11.1762   15.1487
```

Published with MATLAB® R2017b