Table of Contents

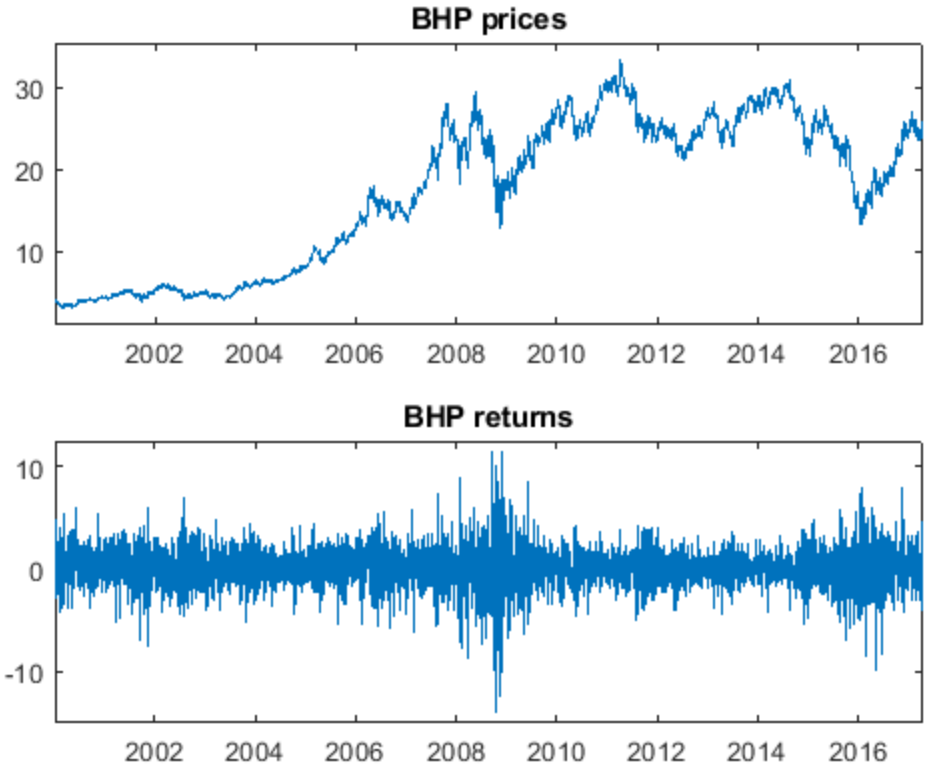
Lab Sheet 8: ARCH Models

Import Data Home-->Import Data for "BHP00-17.csv" and name numeric matrix 'BHPdata' Also import first column separately as a column vector named 'BHPdates' save lab8.mat;

```
load lab8.mat;
```

Q1(a) Plot the price and return series.

```
% Select BHP prices
BHPp=BHPdata(:,7);

% Calculate log-returns
BHPr=100*diff(log((BHPp)));

% Plot the price and return series
figure;subplot(2,1,1);plot(BHPdates,BHPp);title('BHP prices');
ylim([min(BHPp)-2 max(BHPp)+2]);                % set range of y-axis
xlim([BHPdates(1) BHPdates(end)]);              % set range of x-axis
subplot(2,1,2);plot(BHPdates(2:end,1),BHPr);title('BHP returns');
ylim([min(BHPr)-1 max(BHPr)+1]);                % set range of y-axis
xlim([BHPdates(2) BHPdates(end)]);              % set range of x-axis
```

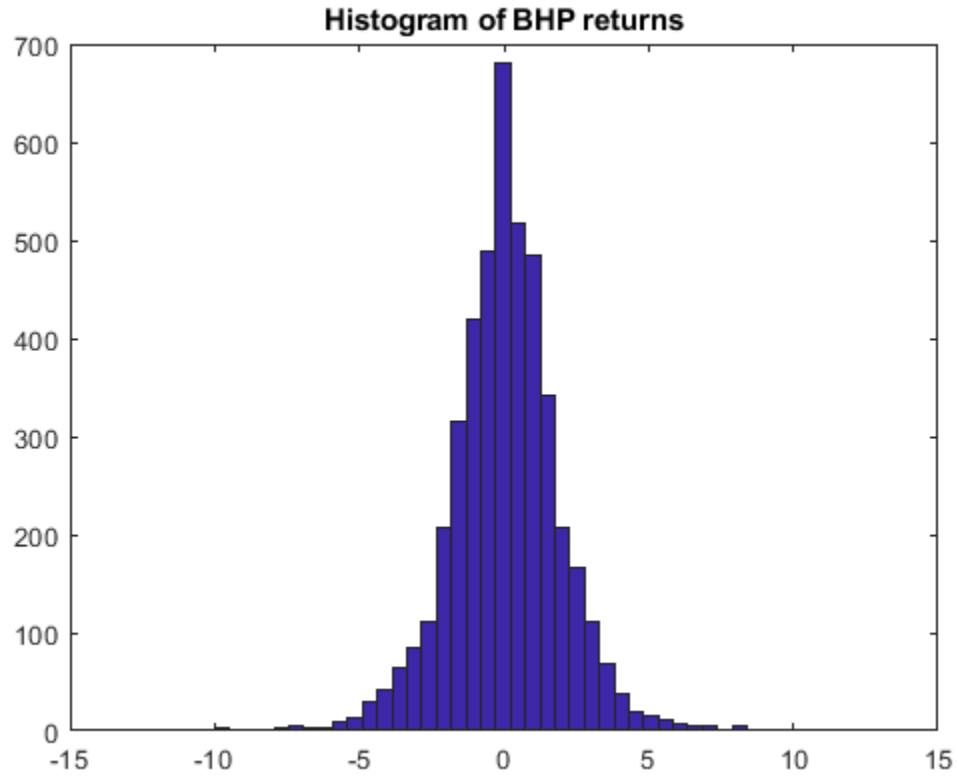


Q1(b) Summary Statistics, histogram and JB test

```
% Summary statistics
[mean(BHPr) median(BHPr) std(BHPr) skewness(BHPr) kurtosis(BHPr)
 min(BHPr) max(BHPr)]
% Selected percentiles
[prctile(BHPr,0.5) prctile(BHPr,1) prctile(BHPr,10)
 prctile(BHPr,25) prctile(BHPr,75) prctile(BHPr,90) prctile(BHPr,99)
 prctile(BHPr,99.5) ]
figure;hist(BHPr,50);title('Histogram of BHP returns');
% JBtest for Gaussianity
[h,p]=jbtest(BHPr)
```

```
ans =

    0.0397         0    1.9350   -0.1562    6.6719  -14.0772   11.4645


ans =

  Columns 1 through 7

   -6.1493   -4.9422   -2.1341   -1.0267    1.1166    2.2877    5.1615
```

```
  Column 8

    6.2152

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

h =

     1


p =

   1.0000e-03
```



Q1(c) ARCH(1) model

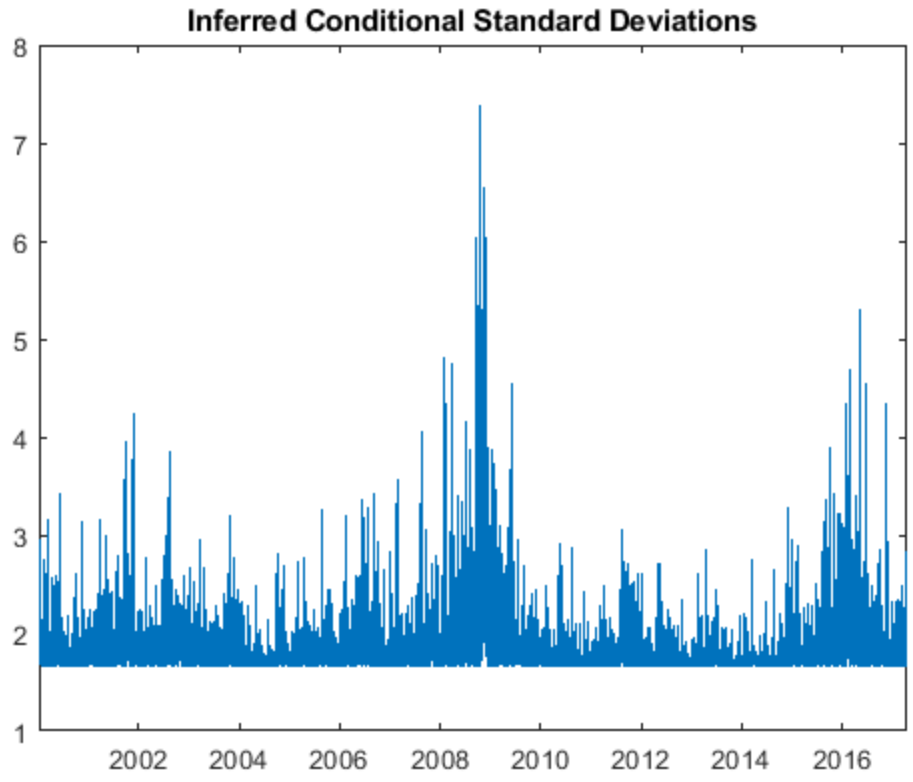
```
% Fit an ARCH(1) model and plot dynamic standard deviations
Mdl = garch('Offset',NaN,'ARCHLags',1); % Specify ARCH(1) model
% Note: the 'Offset' parameter here is set to "NaN" (Not a Number)
 which
% tells the Matlab to estimate it. The default is to set the 'Offset'
 to
```

```
% zero ehich would be appropriate if we had used mean-corrected
 returns
% (mean-corrected returns have zero mean 'Offset' would be zero)

EstMdl=estimate(Mdl,BHPr);              % Estimate ARCH(1) model
v=infer(EstMdl,BHPr);                   % infer the conditional
 variance
% Note: the infer command can have provide additional outputs as you
 shall
% see in future labs. type "help infer" in command window to see
 additional
% options
s=v.^(.5);                              % conditional standard
 deviations

% Plot the estimated conditional standard deviations against time
figure;plot(BHPdates(2:end),s);
xlim([BHPdates(2) BHPdates(end)]);
title('Inferred Conditional Standard Deviations');
```

GARCH(0,1) Conditional Variance Model:

Conditional Probability Distribution: Gaussian

Parameter	Value	Standard Error	t Statistic
Constant	2.76564	0.0544427	50.7991
ARCH{1}	0.25839	0.0159109	16.2398
Offset	0.0508522	0.026067	1.95083



Q1(d) Unconditional variance vs sample variance

```
a1=EstMdl.ARCH{1};  % obtain coefficient of a(t-1)^2 in cond. vol
 equation
a0=EstMdl.Constant; % obtain constant term from conditional vol
 equation
a0/(1-a1)           % the model-based unconditional variance estimate
% the above model-based formula was shown in lectures
var(BHPr)           % this is the sample variance for comparison
% Look at 'EstMdl' in the Workspace (double click on it) and you can
 see
% the various estimated parameters and their data types. It will also
 help
% you understand the commands used to extract them such as a1 and a0
 above
```

```
ans =

    3.7292


ans =
```

```
    3.7441
```

Q1(e) Unconditional kurtosis vs sample kurtosis

```
k=kurtosis(BHPr)      % the sample kurtosis of returns
3*(1-a1^2)/(1-3*a1^2) % the model-based unconditional kurtosis
 estimate
% the above model-based formula was shown in lectures


k =

    6.6719


ans =

    3.5009
```

Q1(f) Least Squares estimates

```
% OLS estimation of an ARCH model
% Setup data
n=length(BHPr);       % sample size
a=BHPr-mean(BHPr);    % demeaned returns (errors in mean equation)
a2=a(2:n).^2;         % demeaned returns squared (y-variable)
x=a(1:n-1).^2;        % lagged demeaned returns squared (x-variable)
xmat=[ones(n-1,1) x]; % X matrix for LS regression of ARCH equation

% OLS regression of demeaned returns squared vs lags
b=regress(a2,xmat) % Coefficients: constant b(1) and arch coefficient
 b(2)

% type the command 'help regress' to see more output options for OLS,
 e.g.
% [B,BINT,R,RINT,STATS] = regress(Y,X) provides more output and was
 used in
% the Lab 7 last week


b =

    2.8819
    0.2299
```

Q1(g) Variance and kurtosis estimates for LS model

```
b(1)/(1-b(2)) % model-based unconditional variance estimate from
 regression
% b(1) is the estimate of constant and b(2) is the estimate of the
% coefficient of a(t-1)^2 in conditional vol equation
3*(1-b(2)^2)/(1-3*b(2)^2) % the model-based unconditional kurtosis
 using
```

```
ans =

    3.7424


ans =

    3.3770
```

Published with MATLAB® R2017b