ARCH/GARCH

1 Importing packages

[4]:
```
#importing packages
import statsmodels.api as sm
from statsmodels.tsa.stattools import adfuller
import pandas as pd
import numpy as np
import statsmodels.formula.api as smf
from sklearn import linear_model
import matplotlib.pyplot as plt
from scipy import stats
```

2 Reading Excel file saved in hard drive

[5]:
```
#reading the file
df = pd.read_excel("C:\\Users\\rluck\\OneDrive\\share.xlsx")
df.head()
```

[5]:
```
   OBS   PRICE
0    1  975.04
1    2  977.07
2    3  966.58
3    4  964.00
4    5  956.05
```

3 Calculating annual return

[6]:
```
#computing the annual return from S&P500
df['R'] = 100*np.log(df['PRICE']/df['PRICE'].shift(1))
df['R_squared']=df['R']**2
df.head()
```

[6]:
```
   OBS   PRICE         R  R_squared
0    1  975.04       NaN        NaN
1    2  977.07  0.207980   0.043256
```

```
2    3  966.58 -1.079423   1.165154
3    4  964.00 -0.267277   0.071437
4    5  956.05 -0.828108   0.685763
```

[7]:
```
df.tail(10)
```

[7]:
```
     OBS    PRICE         R  R_squared
984  985  1149.50 -0.686632   0.471464
985  986  1128.52 -1.842003   3.392973
986  987  1140.21  1.030542   1.062016
987  988  1139.45 -0.066677   0.004446
988  989  1129.90 -0.841656   0.708384
989  990  1144.80  1.310082   1.716314
990  991  1170.35  2.207290   4.872129
991  992  1167.10 -0.278081   0.077329
992  993  1158.31 -0.755999   0.571535
993  994  1139.93 -1.599519   2.558461
```

4 Remove the first row NaN

[8]:
```
#Selecting the sample from
dta =df.iloc[1:993]
dta.head()
```

[8]:
```
   OBS   PRICE         R  R_squared
1    2  977.07  0.207980   0.043256
2    3  966.58 -1.079423   1.165154
3    4  964.00 -0.267277   0.071437
4    5  956.05 -0.828108   0.685763
5    6  927.69 -3.011259   9.067679
```

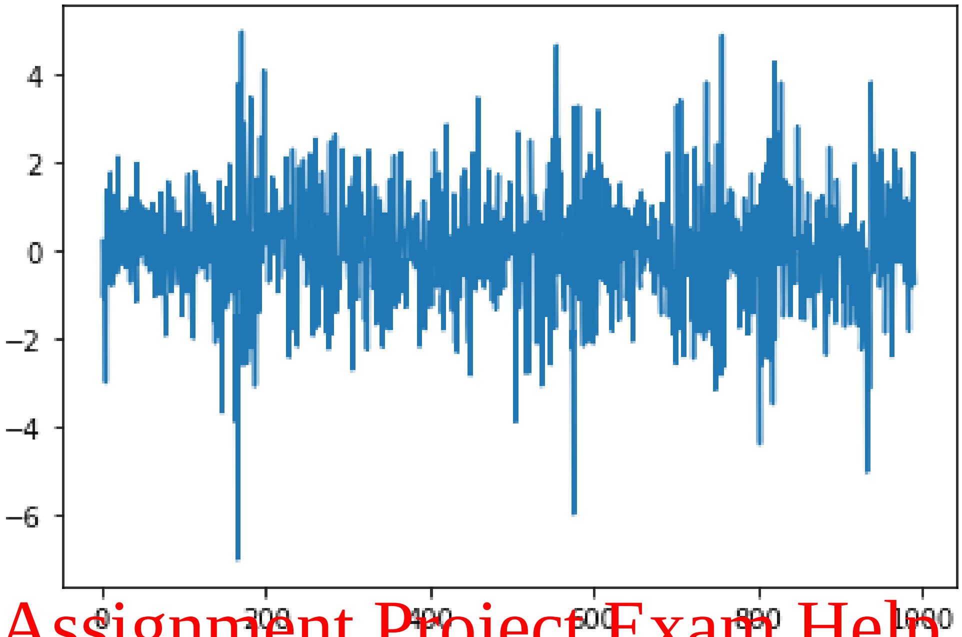
[9]:
```
dta.tail()
```

[9]:
```
     OBS    PRICE         R  R_squared
988  989  1129.90 -0.841656   0.708384
989  990  1144.80  1.310082   1.716314
990  991  1170.35  2.207290   4.872129
991  992  1167.10 -0.278081   0.077329
992  993  1158.31 -0.755999   0.571535
```

5 Plotting the time series: Stock Returns (R) and R_squared

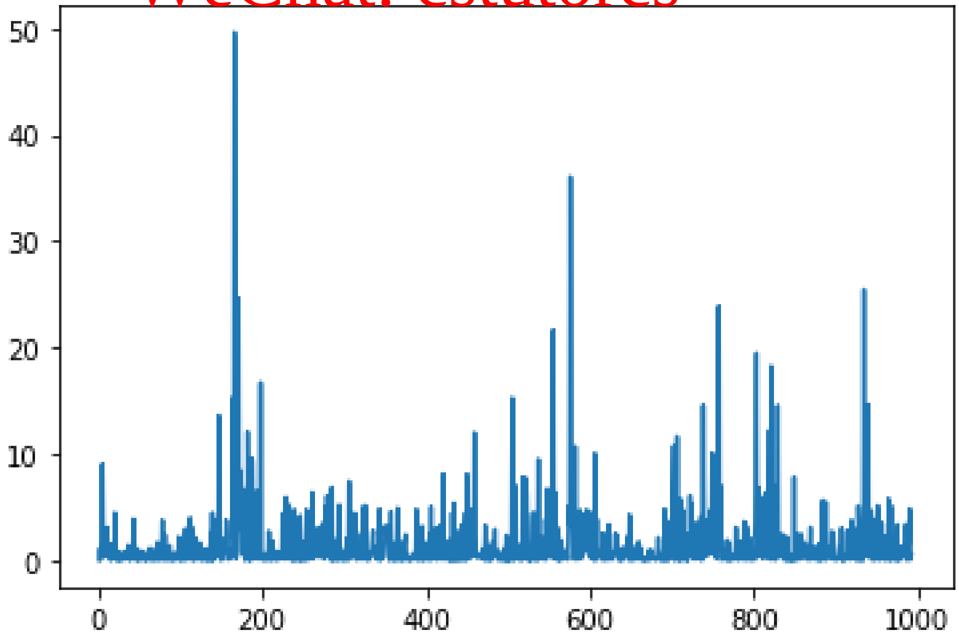
[10]:
```
#plotting the series
plt.plot(dta["R"])
```

```
[10]: [<matplotlib.lines.Line2D at 0x2585011a288>]
```



```
[11]: plt.plot(dta["R_squared"])
```

```
[11]: [<matplotlib.lines.Line2D at 0x25852304c08>]
```



6 Histogram and Descriptive Stats for R and R-squared

[12]:
```
dta.describe()
```

[12]:
```
              OBS        PRICE           R   R_squared
count  992.000000   992.000000  992.000000  992.000000
mean   497.500000  1260.198357    0.017363    1.692547
std    286.510035   146.782611    1.301519    3.393702
min      2.000000   927.690000   -7.043759    0.000000
25%    249.750000  1130.230000   -0.716307    0.127786
50%    497.500000  1277.625000    0.008016    0.565664
75%    745.250000  1378.312500    0.804720    1.887633
max    993.000000  1527.460000    4.964596   49.614541
```

[13]:
```
stats.describe(dta['R'])
```

[13]:
```
DescribeResult(nobs=992, minmax=(-7.043759037302043, 4.964596183505854),
mean=0.01736278522637672, variance=1.6939527052681107,
skewness=-0.14891712656209458, kurtosis=2.0249867442229768)
```

[14]:
```
skewness =-0.14891712656209458
kurtosis =2.024986744222
nobs =992
JB =(skewness**2+0.25*(kurtosis**2))*nobs/6
JB
```

[14]:
```
173.15676433983288
```

[15]:
```
stats.describe(dta['R_squared'])
```

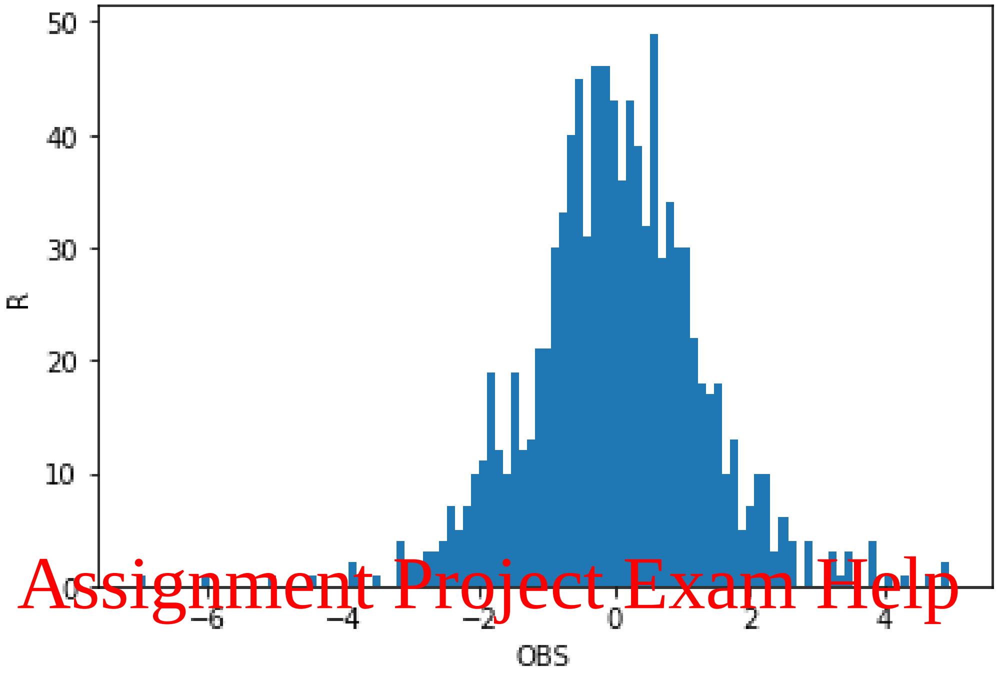
[15]:
```
DescribeResult(nobs=992, minmax=(0.0, 49.614541375574206),
mean=1.6925465579650458, variance=11.51721120039891, skewness=6.16765210396571,
kurtosis=59.59838877039361)
```

[16]:
```
skewness =6.16765210396571
kurtosis =59.59838877039361
nobs =992
JB_R_squared = (skewness**2+0.25*(kurtosis**2))*nobs/6
JB_R_squared
```

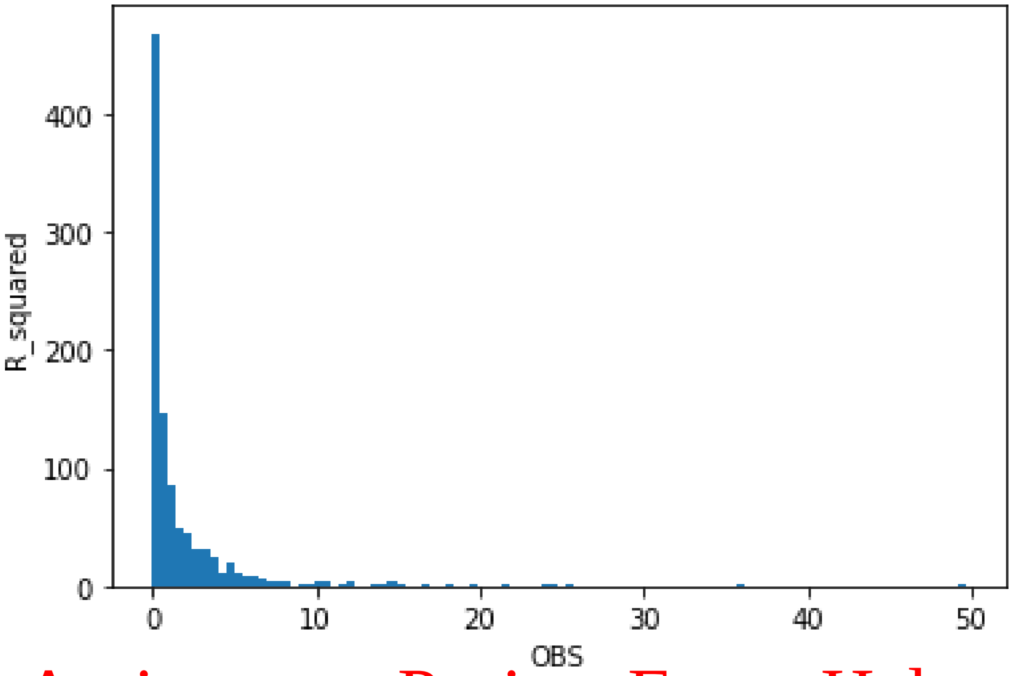
[16]:
```
153103.94385573984
```

[17]:
```
import matplotlib.pyplot as plt
_ = plt.hist(dta['R'],bins=100)
_ = plt.xlabel('OBS')
```

```
_ = plt.ylabel('R')
plt.show()
```

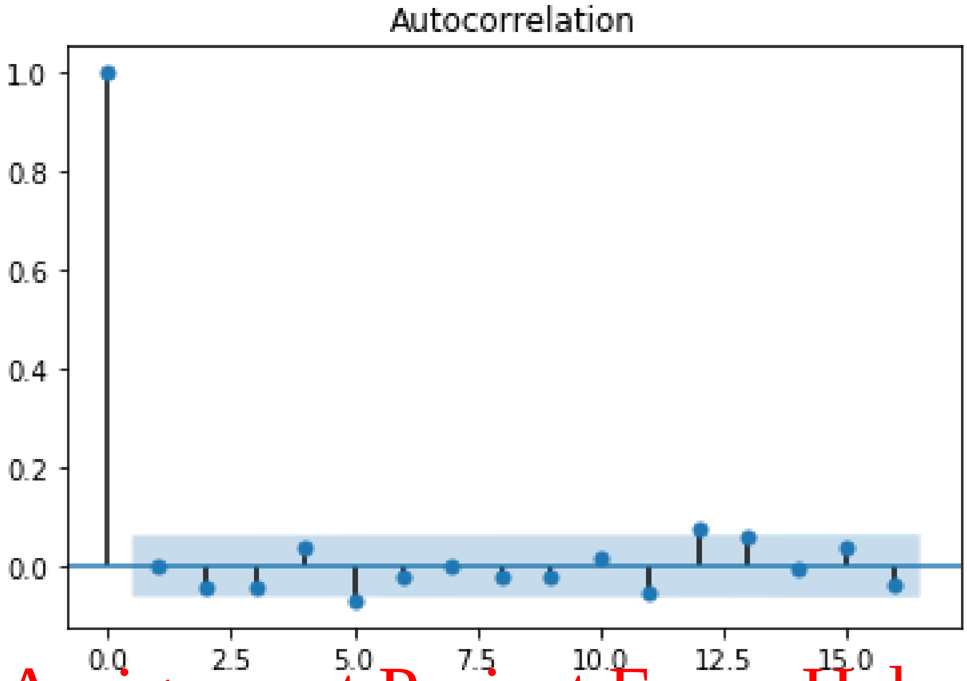


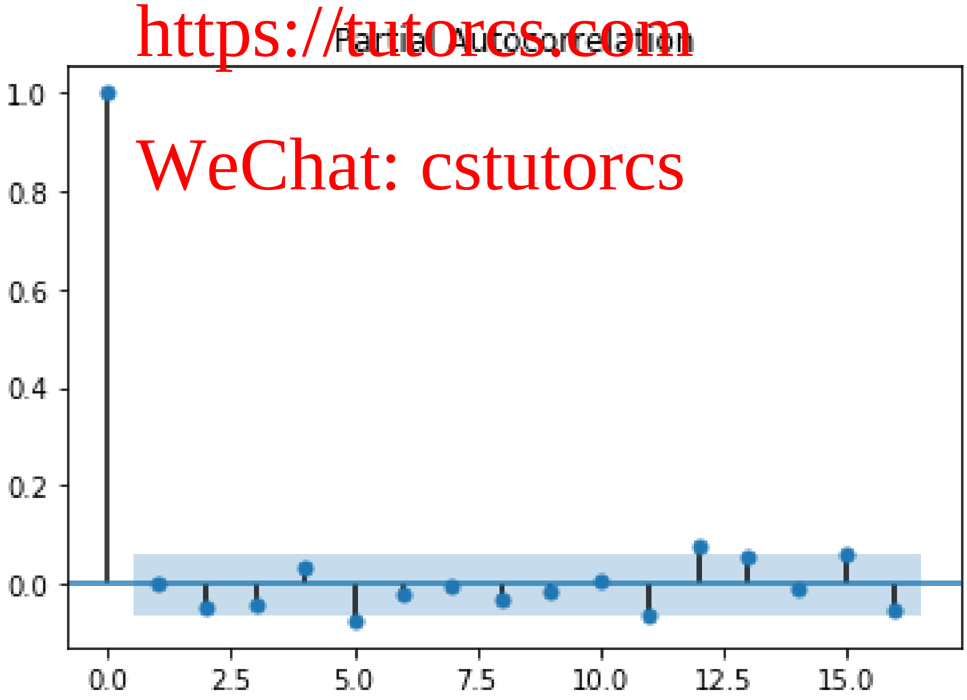
```
[18]: import matplotlib.pyplot as plt
      _ = plt.hist(dta['R_squared'],bins=100)
      _ = plt.xlabel('OBS')
      _ = plt.ylabel('R_squared')
      plt.show()
```



7 Correlogram: ACF and PACF

[19]:
```
#running ACF and PACF for R
dt= dta["R"]
sm.graphics.tsa.plot_acf(dt.values.squeeze(),lags=16)
sm.graphics.tsa.plot_pacf(dt.values.squeeze(),lags=16)
plt.show()
```





[20]:
```
# Generating the Q tables
import numpy as np
```

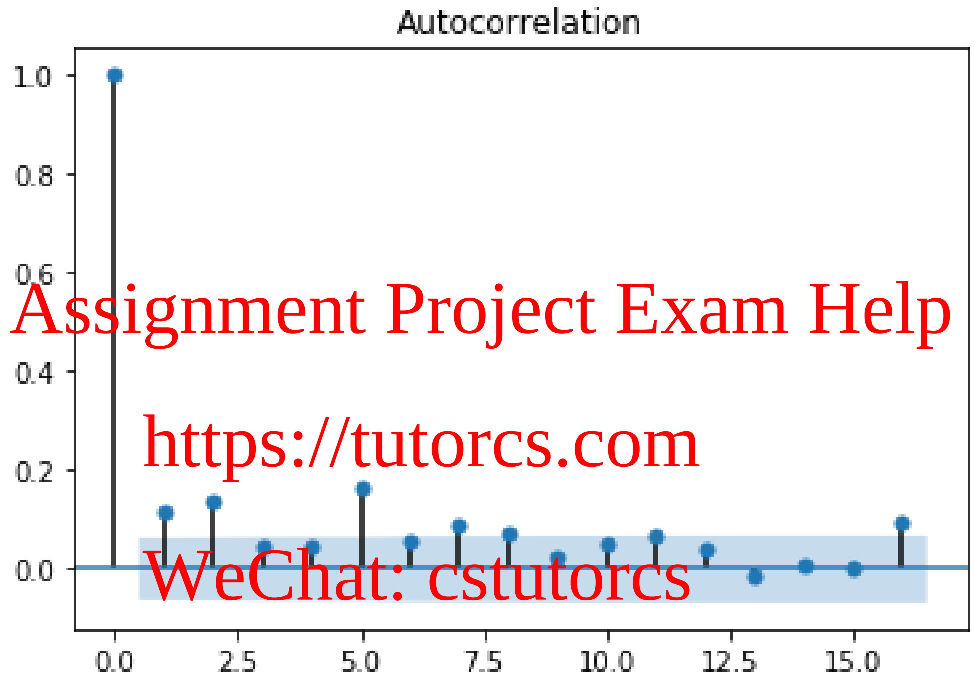
```
r,q,p = sm.tsa.acf(dt.values.squeeze(), qstat=True)
data = np.c_[range(1,41), r[1:], q, p]
table = pd.DataFrame(data, columns=['lag', "AC", "Q", "Prob(>Q)"])
print (table.set_index('lag'))
```

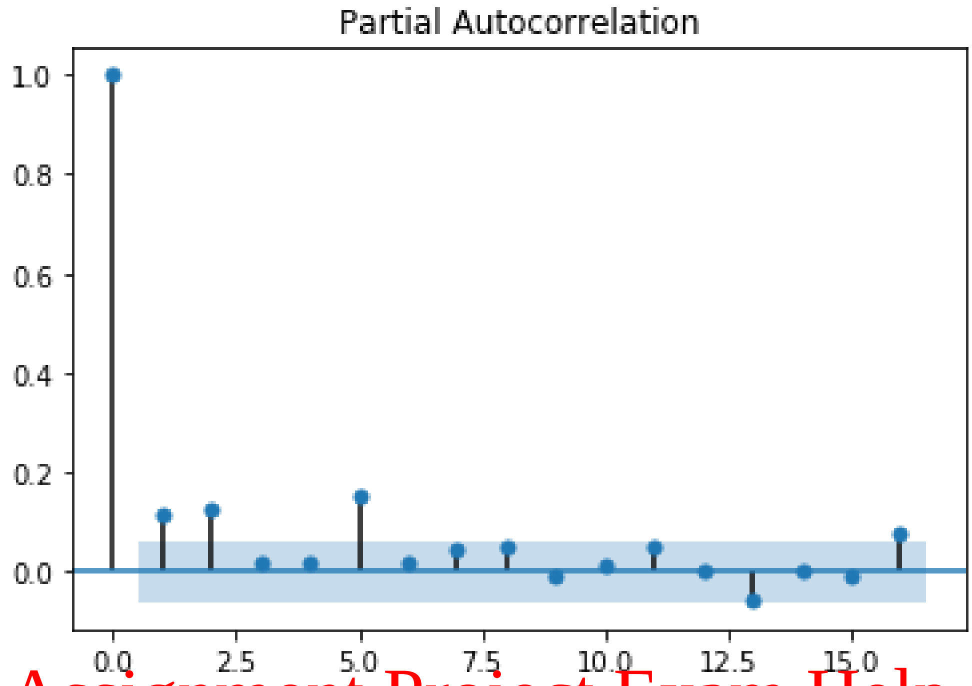
```
            AC          Q  Prob(>Q)
lag
1.0   0.001446   0.002080  0.963620
2.0  -0.044905   2.010474  0.365958
3.0  -0.040718   3.663503  0.300167
4.0   0.037876   5.095256  0.277663
5.0  -0.069898   9.976307  0.075909
6.0  -0.019862  10.370809  0.109880
7.0  -0.000438  10.371001  0.168506
8.0  -0.021094  10.816882  0.212292
9.0  -0.020588  11.242059  0.259481
10.0  0.014929  11.465847  0.322393
11.0 -0.053056  14.295314  0.217081
12.0  0.076836  20.235581  0.062759
13.0  0.058728  23.709332  0.033911
14.0 -0.007241  23.762190  0.048944
15.0  0.039360  25.325770  0.045736
16.0 -0.040193  26.957906  0.041953
17.0  0.014454  27.169196  0.055633
18.0 -0.055402  30.276587  0.034843
19.0  0.021315  30.736999  0.043136
20.0 -0.010348  30.845619  0.057262
21.0 -0.054457  33.857101  0.037536
22.0 -0.015306  34.095254  0.048040
23.0  0.029443  34.977407  0.052285
24.0  0.038375  36.477531  0.049298
25.0 -0.033348  37.611540  0.050457
26.0  0.019285  37.991151  0.060676
27.0  0.082559  44.955799  0.016436
28.0 -0.000481  44.956035  0.022290
29.0  0.014995  45.186276  0.028221
30.0  0.000166  45.186304  0.037088
31.0  0.005648  45.219032  0.047656
32.0 -0.053608  48.170804  0.033175
33.0 -0.041521  49.943454  0.029572
34.0 -0.088390  57.984990  0.006341
35.0 -0.037059  59.400045  0.006170
36.0  0.024673  60.027922  0.007223
37.0 -0.011843  60.172744  0.009385
38.0 -0.000472  60.172975  0.012451
39.0  0.057219  63.560549  0.007761
40.0 -0.073999  69.232214  0.002797
```

```
C:\Users\rluck\anaconda3\lib\site-packages\statsmodels\tsa\stattools.py:572:
FutureWarning: fft=True will become the default in a future version of
statsmodels. To suppress this warning, explicitly set fft=False.
  FutureWarning
```

[21]:

```
#running ACF and PACF for R_squared
dta =dta["R_squared"]
sm.graphics.tsa.plot_acf(dta.values.squeeze(),lags=16)
sm.graphics.tsa.plot_pacf(dta.values.squeeze(),lags=16)
plt.show()
```





[22]:

```
# Generating the Q tables for R squared
import numpy as np
r,q,p = sm.tsa.acf(dta.values.squeeze(), qstat=True)
data = np.c_[range(1,41), r[1:], q, p]
table = pd.DataFrame(data, columns=['lag', "AC", "Q", "Prob(>Q)"])
print (table.set_index('lag'))
```

```
            AC          Q      Prob(>Q)
lag
1.0    0.116494  13.502985  2.381843e-04
2.0    0.137497  32.333061  9.527196e-08
3.0    0.044784  34.332681  1.685331e-07
4.0    0.043315  36.205192  2.625638e-07
5.0    0.163149  62.797123  3.206890e-12
6.0    0.055165  65.840403  2.905376e-12
7.0    0.088482  73.677864  2.660108e-13
8.0    0.072395  78.929848  8.025910e-14
9.0    0.026343  79.625935  1.917629e-13
10.0   0.051302  82.268635  1.800687e-13
11.0   0.065059  86.523037  7.976883e-14
12.0   0.041789  88.280113  1.061870e-13
13.0  -0.014395  88.488810  2.716734e-13
14.0   0.010238  88.594485  6.988975e-13
15.0   0.001890  88.598091  1.809893e-12
16.0   0.092094  97.166798  1.173238e-13
```

```
17.0  0.037119   98.560264  1.643206e-13
18.0  0.072691  103.909558  4.233730e-14
19.0  0.076494  109.839392  8.533981e-15
20.0  0.020480  110.264870  1.757295e-14
21.0  0.111044  122.786785  2.195595e-16
22.0  0.050166  125.345073  1.847860e-16
23.0 -0.001220  125.346586  4.499962e-16
24.0  0.044374  127.352379  4.667582e-16
25.0  0.003717  127.366465  1.090264e-15
26.0  0.028585  128.200510  1.785587e-15
27.0  0.082601  135.172333  2.370793e-16
28.0  0.026304  135.880045  4.053453e-16
29.0 -0.005202  135.907755  8.989330e-16
30.0 -0.007790  135.969957  1.932061e-15
31.0  0.009146  136.055794  4.045732e-15
32.0  0.079309  142.516399  6.755564e-16
33.0  0.006202  142.555947  1.427039e-15
34.0 -0.039068  144.126945  1.632971e-15
35.0 -0.018614  144.483956  2.975030e-15
36.0 -0.038176  145.987149  3.443204e-15
37.0 -0.025142  146.639804  5.487456e-15
38.0 -0.043511  148.596627  5.273067e-15
39.0 -0.014017  148.799929  9.805069e-15
40.0 -0.042392  150.661265  9.670211e-15

C:\Users\rluck\anaconda3\lib\site-packages\statsmodels\tsa\stattools.py:572:
FutureWarning: fft=True will become the default in a future version of
statsmodels. To suppress this warning, explicitly set fft=False.
  FutureWarning
```

8 ARCH(5) Answer to 3(d)

[72]:

```
from arch import arch_model
model = arch_model(dt, mean='Constant', vol='ARCH', p=5)
x =model.fit()
x
```

```
Iteration:      1,   Func. Count:      9,   Neg. LLF: 1641.4320830727252
Iteration:      2,   Func. Count:     21,   Neg. LLF: 1641.117179679314
Iteration:      3,   Func. Count:     32,   Neg. LLF: 1639.8618699736512
Iteration:      4,   Func. Count:     42,   Neg. LLF: 1639.2602976096668
Iteration:      5,   Func. Count:     52,   Neg. LLF: 1638.7165636155444
Iteration:      6,   Func. Count:     63,   Neg. LLF: 1638.5815616395487
Iteration:      7,   Func. Count:     73,   Neg. LLF: 1638.0031576294539
Iteration:      8,   Func. Count:     83,   Neg. LLF: 1637.590224502058
Iteration:      9,   Func. Count:     94,   Neg. LLF: 1637.5781186332947
Iteration:     10,   Func. Count:    104,   Neg. LLF: 1637.4668239040275
```

```
Iteration:     11,   Func. Count:    114,   Neg. LLF: 1637.3821507140146
Iteration:     12,   Func. Count:    123,   Neg. LLF: 1637.3674045891073
Iteration:     13,   Func. Count:    132,   Neg. LLF: 1637.366605883134
Iteration:     14,   Func. Count:    141,   Neg. LLF: 1637.3662139249632
Iteration:     15,   Func. Count:    150,   Neg. LLF: 1637.366197005517
Optimization terminated successfully.    (Exit mode 0)
            Current function value: 1637.3661963353218
            Iterations: 15
            Function evaluations: 151
            Gradient evaluations: 15
```

[72]:

```
                      Constant Mean - ARCH Model Results
==============================================================================
Dep. Variable:                      R   R-squared:                      -0.001
Mean Model:             Constant Mean   Adj. R-squared:                 -0.001
Vol Model:                       ARCH   Log-Likelihood:               -1637.37
Distribution:                  Normal   AIC:                           3288.73
Method:            Maximum Likelihood   BIC:                           3323.03
                                        No. Observations:                  992
Date:                Thu, Aug 20 2020   Df Residuals:                      985
Time:                        17:11:52   Df Model:                            7
                                 Mean Model
===========================================================================
                 coef    std err          t      P>|t|      95.0% Conf. Int.
---------------------------------------------------------------------------
mu             0.0513  3.935e-02      1.303      0.192 [-2.584e-02,  0.128]
                               Volatility Model
=============================================================================
                 coef    std err          t      P>|t|        95.0% Conf. Int.
-----------------------------------------------------------------------------
omega          1.0282      0.152      6.765  1.337e-11    [  0.730,  1.326]
alpha[1]       0.0677  4.075e-02      1.662  9.650e-02 [-1.214e-02,  0.148]
alpha[2]       0.1424  6.086e-02      2.341  1.925e-02  [2.316e-02,  0.262]
alpha[3]       0.0270  2.455e-02      1.100      0.271 [-2.112e-02,7.511e-02]
alpha[4]       0.0493  4.324e-02      1.140      0.254 [-3.547e-02,  0.134]
alpha[5]       0.1040  4.269e-02      2.437  1.481e-02  [2.037e-02,  0.188]
=============================================================================

Covariance estimator: robust
ARCHModelResult, id: 0x258550e2b08
```

9 3e: ARCH test

[62]:

```python
from statsmodels.stats.diagnostic import het_arch
from statsmodels.compat import lzip
```

```
[59]: res = het_arch(dt.values,nlags =5)
      name = ['lm','lmpval','fval','fpval']
      lzip(name,res)
```

```
[59]: [('lm', 52.62649711315813),
       ('lmpval', 4.012739137878328e-10),
       ('fval', 11.050526049487171),
       ('fpval', 2.2665869070632531e-10)]
```

10 4d: ARCH test of standardised residuals

```
[74]: resid = x.resid/x.conditional_volatility
```

```
[75]: #4d: ARCH test
      res = het_arch(resid,nlags =5)
      name = ['lm','lmpval','fval','fpval']
      lzip(name,res)
```

```
[75]: [('lm', 1.841550196914982),
       ('lmpval', 0.8706063058565487),
       ('fval', 0.36675536682138066),
       ('fpval', 0.8714901609636871)]
```

11 GARCH(1,1) Answer to q4e

```
[140]: #GARCH(1,1)
       model = arch_model(dt, mean='Zero', vol='GARCH', p=1, q=1)
       model.fit()
```

```
Iteration:      1,   Func. Count:      5,   Neg. LLF: 1635.7051459253014
Iteration:      2,   Func. Count:     11,   Neg. LLF: 1635.198292821203
Iteration:      3,   Func. Count:     17,   Neg. LLF: 1634.4683889348146
Iteration:      4,   Func. Count:     23,   Neg. LLF: 1634.142726804497
Iteration:      5,   Func. Count:     29,   Neg. LLF: 1633.820026751518
Iteration:      6,   Func. Count:     35,   Neg. LLF: 1633.7942934149069
Iteration:      7,   Func. Count:     41,   Neg. LLF: 1633.5845044893476
Iteration:      8,   Func. Count:     47,   Neg. LLF: 1633.5188557297743
Iteration:      9,   Func. Count:     53,   Neg. LLF: 1633.41918609823
Iteration:     10,   Func. Count:     58,   Neg. LLF: 1633.3859573686977
Iteration:     11,   Func. Count:     63,   Neg. LLF: 1633.380833530281
Iteration:     12,   Func. Count:     68,   Neg. LLF: 1633.3807015892412
Optimization terminated successfully.    (Exit mode 0)
            Current function value: 1633.380701081655
            Iterations: 12
            Function evaluations: 69
```

```
             Gradient evaluations: 12
```

[140]:

```
                       Zero Mean - GARCH Model Results
==============================================================================
Dep. Variable:                      R   R-squared:                       0.000
Mean Model:                 Zero Mean   Adj. R-squared:                  0.001
Vol Model:                      GARCH   Log-Likelihood:               -1633.38
Distribution:                  Normal   AIC:                           3272.76
Method:            Maximum Likelihood   BIC:                           3287.46
                                        No. Observations:                  992
Date:                Sat, Jul 25 2020   Df Residuals:                      989
Time:                        07:57:24   Df Model:                            3
                             Volatility Model
==========================================================================
                 coef    std err          t      P>|t|    95.0% Conf. Int.
--------------------------------------------------------------------------
omega          0.0722  3.410e-02      2.118  3.419e-02 [5.385e-03,  0.139]
alpha[1]       0.0780  2.399e-02      3.252  1.146e-03 [3.100e-02,  0.125]
beta[1]        0.8805  3.299e-02     26.691 5.933e-157   [  0.816,  0.945]
==========================================================================

Covariance estimator: robust
ARCHModelResult, id: 0x1ec7e194488
```

[ ]:

```

```

12 GARC(2,1), GARCH(1,2) and GARCH(2,2) Answer to q4f

[141]:

```python
#GARCH (2,1)
model = arch_model(dt, mean='Zero', vol='GARCH', p=2, q=1)
model.fit()
```

```
Iteration:      1,   Func. Count:      6,   Neg. LLF: 1634.4763048908176
Iteration:      2,   Func. Count:     14,   Neg. LLF: 1634.3114071546615
Iteration:      3,   Func. Count:     21,   Neg. LLF: 1633.898626097601
Iteration:      4,   Func. Count:     28,   Neg. LLF: 1633.6129513377468
Iteration:      5,   Func. Count:     35,   Neg. LLF: 1633.4056492048512
Iteration:      6,   Func. Count:     42,   Neg. LLF: 1633.20362402841
Iteration:      7,   Func. Count:     49,   Neg. LLF: 1633.1525418386232
Iteration:      8,   Func. Count:     56,   Neg. LLF: 1633.0365456637537
Iteration:      9,   Func. Count:     63,   Neg. LLF: 1632.964882807908
Iteration:     10,   Func. Count:     70,   Neg. LLF: 1632.9547557375658
Iteration:     11,   Func. Count:     76,   Neg. LLF: 1632.954485222734
Iteration:     12,   Func. Count:     82,   Neg. LLF: 1632.9543347483414
Optimization terminated successfully.    (Exit mode 0)
            Current function value: 1632.9543338562553
            Iterations: 12
            Function evaluations: 83
            Gradient evaluations: 12
```

[141]:

```
                       Zero Mean - GARCH Model Results
==============================================================================
Dep. Variable:                      R   R-squared:                       0.000
Mean Model:                 Zero Mean   Adj. R-squared:                  0.001
Vol Model:                      GARCH   Log-Likelihood:               -1632.95
Distribution:                  Normal   AIC:                           3273.91
Method:            Maximum Likelihood   BIC:                           3293.51
                                        No. Observations:                  992
Date:                Sat, Jul 25 2020   Df Residuals:                      988
Time:                        07:57:24   Df Model:                            4
                              Volatility Model
===========================================================================
                 coef    std err          t      P>|t|     95.0% Conf. Int.
---------------------------------------------------------------------------
omega          0.0814  4.475e-02      1.819  6.890e-02 [-6.304e-03,  0.169]
alpha[1]       0.0521  3.809e-02      1.367      0.172 [-2.258e-02,  0.127]
alpha[2]       0.0345  4.683e-02      0.736      0.462 [-5.733e-02,  0.126]
beta[1]        0.8668  4.806e-02     18.038  9.771e-73 [  0.773,  0.961]
===========================================================================

Covariance estimator: robust
ARCHModelResult, id: 0x1ec7e0fe388
```

[142]:

```
#GARCH (1,2)
model = arch_model(dt, mean='Zero', vol='GARCH', p=1, q=2)
model.fit()
```

```
Iteration:      1,   Func. Count:      6,   Neg. LLF: 1637.5020935283383
Iteration:      2,   Func. Count:     14,   Neg. LLF: 1637.3630476546887
Iteration:      3,   Func. Count:     21,   Neg. LLF: 1635.5390948880922
Iteration:      4,   Func. Count:     29,   Neg. LLF: 1635.4513712241176
Iteration:      5,   Func. Count:     36,   Neg. LLF: 1634.9403158235568
Iteration:      6,   Func. Count:     43,   Neg. LLF: 1633.881149795298
Iteration:      7,   Func. Count:     49,   Neg. LLF: 1633.7161883728404
Iteration:      8,   Func. Count:     55,   Neg. LLF: 1633.3916902708702
Iteration:      9,   Func. Count:     62,   Neg. LLF: 1633.3817391556056
Iteration:     10,   Func. Count:     69,   Neg. LLF: 1633.38101239148
Iteration:     11,   Func. Count:     76,   Neg. LLF: 1633.3807015808443
Optimization terminated successfully.    (Exit mode 0)
            Current function value: 1633.3807015474256
            Iterations: 11
            Function evaluations: 76
            Gradient evaluations: 11
```

[142]:

```
                           Zero Mean - GARCH Model Results
==============================================================================
Dep. Variable:                      R   R-squared:                       0.000
Mean Model:                 Zero Mean   Adj. R-squared:                  0.001
Vol Model:                      GARCH   Log-Likelihood:               -1633.38
Distribution:                  Normal   AIC:                           3274.76
Method:            Maximum Likelihood   BIC:                           3294.36
                                        No. Observations:                  992
Date:                Sat, Jul 25 2020   Df Residuals:                      988
Time:                        07:57:25   Df Model:                            4
                             Volatility Model
==========================================================================
                 coef    std err          t      P>|t|    95.0% Conf. Int.
--------------------------------------------------------------------------
omega          0.0722  3.493e-02      2.068  3.862e-02 [3.782e-03,  0.141]
alpha[1]       0.0780  2.386e-02      3.271  1.072e-03 [3.127e-02,  0.125]
beta[1]        0.8805      0.175      5.039  4.685e-07   [  0.538,  1.223]
beta[2]    4.5110e-14      0.163  2.775e-13      1.000   [ -0.319,  0.319]
==========================================================================

Covariance estimator: robust
ARCHModelResult, id: 0x1ec7e16f6c8
```

[143]:

```
#GARCH (2,2)
model = arch_model(dt, mean='Zero', vol='GARCH', p=2, q=2)
model.fit()
```

```
Iteration:      1,   Func. Count:      7,   Neg. LLF: 1633.8456883994468
Iteration:      2,   Func. Count:     16,   Neg. LLF: 1632.0020951730494
Iteration:      3,   Func. Count:     26,   Neg. LLF: 1631.9962682958353
Iteration:      4,   Func. Count:     34,   Neg. LLF: 1631.7125855588226
Iteration:      5,   Func. Count:     43,   Neg. LLF: 1631.7070695589769
Iteration:      6,   Func. Count:     51,   Neg. LLF: 1631.5885046883052
Iteration:      7,   Func. Count:     60,   Neg. LLF: 1631.5882871287859
Iteration:      8,   Func. Count:     68,   Neg. LLF: 1631.5727647406475
Iteration:      9,   Func. Count:     75,   Neg. LLF: 1631.5723289793957
Optimization terminated successfully.    (Exit mode 0)
            Current function value: 1631.5723279871504
            Iterations: 9
            Function evaluations: 76
            Gradient evaluations: 9
```

[143]:

```
                           Zero Mean - GARCH Model Results
==============================================================================
Dep. Variable:                      R   R-squared:                       0.000
Mean Model:                 Zero Mean   Adj. R-squared:                  0.001
Vol Model:                      GARCH   Log-Likelihood:               -1631.57
```

```
Distribution:                  Normal   AIC:                           3273.14
Method:            Maximum Likelihood   BIC:                           3297.64
                                        No. Observations:                  992
Date:                Sat, Jul 25 2020   Df Residuals:                      987
Time:                        07:57:25   Df Model:                            5
                              Volatility Model
============================================================================
                 coef    std err          t      P>|t|       95.0% Conf. Int.
----------------------------------------------------------------------------
omega          0.1311  7.046e-02      1.861  6.272e-02   [-6.959e-03,  0.269]
alpha[1]       0.0393  2.745e-02      1.432      0.152 [-1.448e-02,9.311e-02]
alpha[2]       0.0997  4.896e-02      2.036  4.174e-02    [3.730e-03,  0.196]
beta[1]        0.2862      0.166      1.723  8.491e-02   [-3.938e-02,  0.612]
beta[2]        0.4996      0.140      3.577  3.479e-04      [  0.226,  0.773]
============================================================================

Covariance estimator: robust
ARCHModelResult, id: 0x1ec7e1fcd48
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```