ECON423 Assignment 3 - d2tsai 20567329

November 29, 2018

1 ECON 423 Assignment 3

Diana Tsai
20567329

In [65]:
```
# Importing necessary packages
import numpy as np
from arch import arch_model
import matplotlib.pyplot as plt
import pandas as pd
```

1.1 Monte Carlo Simulation

1.1.1 Part 1

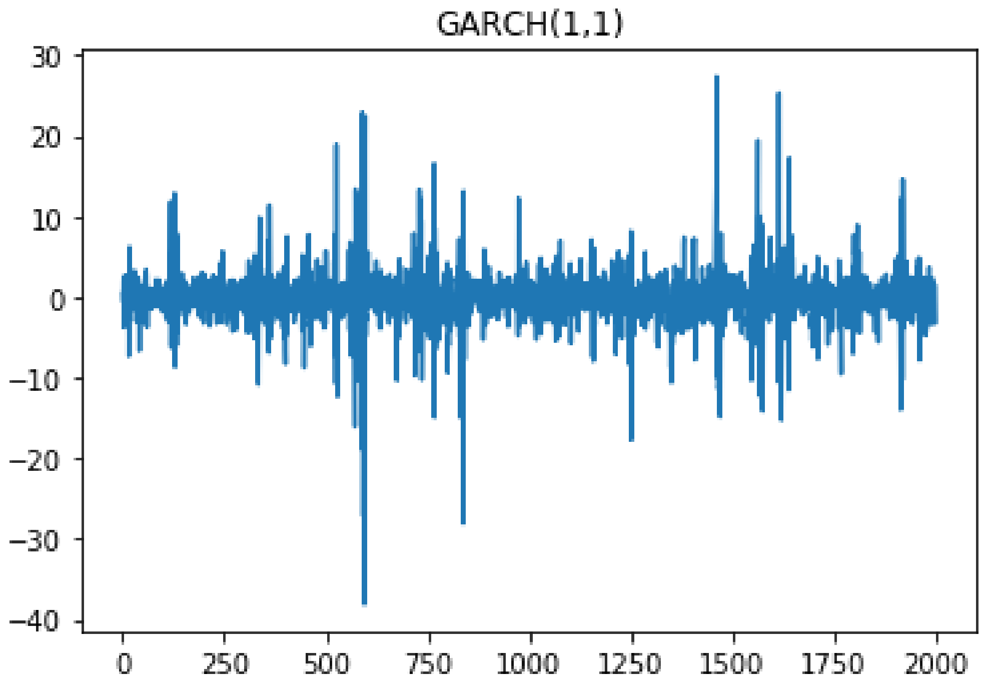
In [66]:
```
n = 2100
a1 = 0.69
b1 = 0.24
mu = 1
errors = np.random.normal(0, 1, n)
```

1.1.2 Part 2

In [67]:
```
np.random.seed(323)
a1 = 0.69
b1 = 0.31

epsilon = np.random.normal(size=n)
x = np.zeros_like(epsilon)
sigma_sqr = np.zeros_like(epsilon)

for i in range(1, n):
    sigma_sqr[i] = mu + a1*(x[i-1]**2) + b1*sigma_sqr[i-1]
    x[i] = epsilon[i] * np.sqrt(sigma_sqr[i])

# drop first 500 observations
sigma_sqr = sigma_sqr[100:]
x = x[100:]
```

```
plt.plot(x)
plt.title('GARCH(1,1)')
```

Out[67]: Text(0.5,1,'GARCH(1,1)')



1.1.3 Part 3

ARCH(1)

In [68]:
```
arch1 = arch_model(x, p=1, q=0)
arch1_resid = arch1.fit(disp='off')
arch1_resid.summary()
```

Out[68]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                      Constant Mean - ARCH Model Results
==============================================================================
Dep. Variable:                      y   R-squared:                      -0.000
Mean Model:             Constant Mean   Adj. R-squared:                 -0.000
Vol Model:                       ARCH   Log-Likelihood:               -4796.20
Distribution:                  Normal   AIC:                           9598.40
Method:            Maximum Likelihood   BIC:                           9615.21
                                        No. Observations:                 2000
Date:                Wed, Nov 28 2018   Df Residuals:                     1997
Time:                        21:21:41   Df Model:                            3
```

```
                            Mean Model
==========================================================================
                 coef    std err          t      P>|t|    95.0% Conf. Int.
--------------------------------------------------------------------------
mu            -0.0129  4.818e-02     -0.268      0.789 [ -0.107,8.152e-02]
                        Volatility Model
========================================================================
                 coef    std err          t      P>|t|  95.0% Conf. Int.
------------------------------------------------------------------------
omega          2.9117      0.224     13.003  1.181e-38 [  2.473,  3.351]
alpha[1]       0.9003  6.028e-02     14.936  1.923e-50 [  0.782,  1.018]
========================================================================

Covariance estimator: robust
"""
```

GARCH(1,1)

In [69]:
```python
garch11 = arch_model(x, p=1, q=1)
garch11_resid = garch11.fit(disp='off')
garch11_resid.summary()
```

Out[69]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                     Constant Mean - GARCH Model Results
==============================================================================
Dep. Variable:                      y   R-squared:                      -0.000
Mean Model:             Constant Mean   Adj. R-squared:                 -0.000
Vol Model:                      GARCH   Log-Likelihood:               -4692.36
Distribution:                  Normal   AIC:                           9392.72
Method:            Maximum Likelihood   BIC:                           9415.12
                                        No. Observations:                 2000
Date:                Wed, Nov 28 2018   Df Residuals:                     1996
Time:                        21:21:41   Df Model:                            4
                               Mean Model
=============================================================================
                 coef    std err          t      P>|t|       95.0% Conf. Int.
-----------------------------------------------------------------------------
mu            -0.0187  3.913e-02     -0.477      0.633 [-9.536e-02,5.802e-02]
                        Volatility Model
========================================================================
                 coef    std err          t      P>|t|  95.0% Conf. Int.
------------------------------------------------------------------------
omega          1.1233      0.124      9.024  1.816e-19 [  0.879,  1.367]
alpha[1]       0.7213  5.111e-02     14.115  3.088e-45 [  0.621,  0.822]
beta[1]        0.2787  3.078e-02      9.054  1.380e-19 [  0.218,  0.339]
========================================================================
```

```
Covariance estimator: robust
"""
```

GARCH(2,2)

In [70]:
```
garch22 = arch_model(x, p=2, q=2)
garch22_resid = garch22.fit(disp='off')
garch22_resid.summary()
```

Out[70]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                     Constant Mean - GARCH Model Results
==============================================================================
Dep. Variable:                      y   R-squared:                      -0.000
Mean Model:             Constant Mean   Adj. R-squared:                 -0.000
Vol Model:                      GARCH   Log-Likelihood:               -4691.69
Distribution:                  Normal   AIC:                           9395.38
Method:            Maximum Likelihood   BIC:                           9428.99
                                        No. Observations:                 2000
Date:                Wed, Nov 28 2018   Df Residuals:                     1994
Time:                        21:21:41   Df Model:                            6
                                 Mean Model
=============================================================================
                 coef    std err          t      P>|t|       95.0% Conf. Int.
-----------------------------------------------------------------------------
mu            -0.0202  3.933e-02     -0.513      0.608 [-9.729e-02,5.690e-02]
                              Volatility Model
========================================================================
                 coef    std err          t      P>|t|  95.0% Conf. Int.
------------------------------------------------------------------------
omega          1.1415      0.388      2.945  3.234e-03 [  0.382,  1.901]
alpha[1]       0.7379  5.377e-02     13.722  7.488e-43 [  0.632,  0.843]
alpha[2]       0.0000      0.252      0.000      1.000 [ -0.494,  0.494]
beta[1]        0.2312      0.307      0.754      0.451 [ -0.370,  0.832]
beta[2]        0.0310  7.332e-02      0.422      0.673 [ -0.113,  0.175]
========================================================================

Covariance estimator: robust
"""
```

1.1.4 Part 4

(i)

In [71]:
```
x_half1 = x[:1000]
x_half2 = x[1000:]
```

(ii)

```
In [72]: arch1p4 = arch_model(x_half1, p=1, q=0)
         arch1_residp4 = arch1p4.fit(disp='off')
         arch1_paramp4 = arch1_residp4.params
         arch1_residp4.summary()
```

```
Out[72]: <class 'statsmodels.iolib.summary.Summary'>
         """
                               Constant Mean - ARCH Model Results
         ==============================================================================
         Dep. Variable:                      y   R-squared:                      -0.000
         Mean Model:             Constant Mean   Adj. R-squared:                 -0.000
         Vol Model:                       ARCH   Log-Likelihood:               -2373.51
         Distribution:                  Normal   AIC:                           4753.03
         Method:            Maximum Likelihood   BIC:                           4767.75
                                                 No. Observations:                 1000
         Date:                Wed, Nov 28 2018   Df Residuals:                      997
         Time:                        21:21:41   Df Model:                            3
                                      Mean Model
         ==========================================================================
                          coef    std err          t      P>|t|    95.0% Conf. Int.
         --------------------------------------------------------------------------
         mu            -0.1023  6.262e-02     -1.634      0.102 [ -0.225,2.040e-02]
                                  Volatility Model
         ========================================================================
                          coef    std err          t      P>|t|  95.0% Conf. Int.
         ------------------------------------------------------------------------
         omega          2.3566      0.238      9.884  4.906e-23 [  1.889,  2.824]
         alpha[1]       1.0000  8.528e-02     11.726  9.367e-32 [  0.833,  1.167]
         ========================================================================

         Covariance estimator: robust
         """
```

```
In [73]: garch11p4 = arch_model(x_half1, p=1, q=1)
         garch11_residp4 = garch11p4.fit(disp='off')
         garch11_paramp4 = garch11_residp4.params
         garch11_residp4.summary()
```

```
Out[73]: <class 'statsmodels.iolib.summary.Summary'>
         """
                              Constant Mean - GARCH Model Results
         ==============================================================================
         Dep. Variable:                      y   R-squared:                      -0.000
         Mean Model:             Constant Mean   Adj. R-squared:                 -0.000
         Vol Model:                      GARCH   Log-Likelihood:               -2332.16
         Distribution:                  Normal   AIC:                           4672.33
         Method:            Maximum Likelihood   BIC:                           4691.96
                                                 No. Observations:                 1000
```

```
Date:                Wed, Nov 28 2018   Df Residuals:                     996
Time:                        21:21:42   Df Model:                           4
                                 Mean Model
============================================================================
                 coef    std err          t      P>|t|    95.0% Conf. Int.
----------------------------------------------------------------------------
mu            -0.0624  5.462e-02     -1.143      0.253 [ -0.169,4.462e-02]
                              Volatility Model
==========================================================================
                 coef    std err          t      P>|t|  95.0% Conf. Int.
--------------------------------------------------------------------------
omega          1.0029      0.169      5.950  2.678e-09 [  0.673,  1.333]
alpha[1]       0.7146  7.091e-02     10.077  6.965e-24 [  0.576,  0.854]
beta[1]        0.2854  4.684e-02      6.094  1.104e-09 [  0.194,  0.377]
==========================================================================

Covariance estimator: robust
"""
```

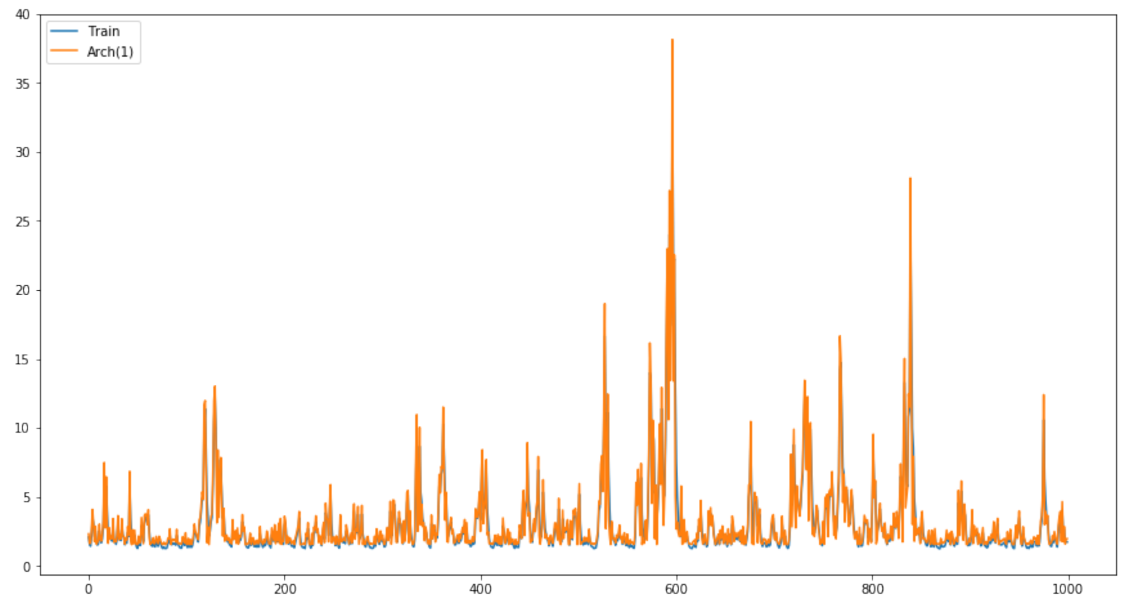
In [74]:
```
garch22p4 = arch_model(x_half1, p=2, q=2)
garch22_residp4 = garch22p4.fit(disp='off')
garch22_paramp4 = garch22_residp4.params
garch22_residp4.summary()
```

Out[74]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                     Constant Mean - GARCH Model Results
==============================================================================
Dep. Variable:                      y   R-squared:                      -0.000
Mean Model:             Constant Mean   Adj. R-squared:                 -0.000
Vol Model:                      GARCH   Log-Likelihood:               -2331.75
Distribution:                  Normal   AIC:                           4675.50
Method:            Maximum Likelihood   BIC:                           4704.94
                                        No. Observations:                 1000
Date:                Wed, Nov 28 2018   Df Residuals:                      994
Time:                        21:21:42   Df Model:                            6
                                 Mean Model
============================================================================
                 coef    std err          t      P>|t|    95.0% Conf. Int.
----------------------------------------------------------------------------
mu            -0.0622  5.476e-02     -1.136      0.256 [ -0.170,4.514e-02]
                               Volatility Model
=============================================================================
                 coef    std err          t      P>|t|     95.0% Conf. Int.
-----------------------------------------------------------------------------
omega          1.0499      0.290      3.617  2.979e-04    [  0.481,  1.619]
alpha[1]       0.7371  7.690e-02      9.584  9.303e-22    [  0.586,  0.888]
alpha[2]       0.0170      0.171  9.973e-02      0.921    [ -0.318,  0.352]
```

```
beta[1]        0.1976      0.214      0.922      0.357    [ -0.222,  0.618]
beta[2]        0.0484  6.430e-02      0.752      0.452 [-7.767e-02,  0.174]
==============================================================================

Covariance estimator: robust
"""
```

(iii)

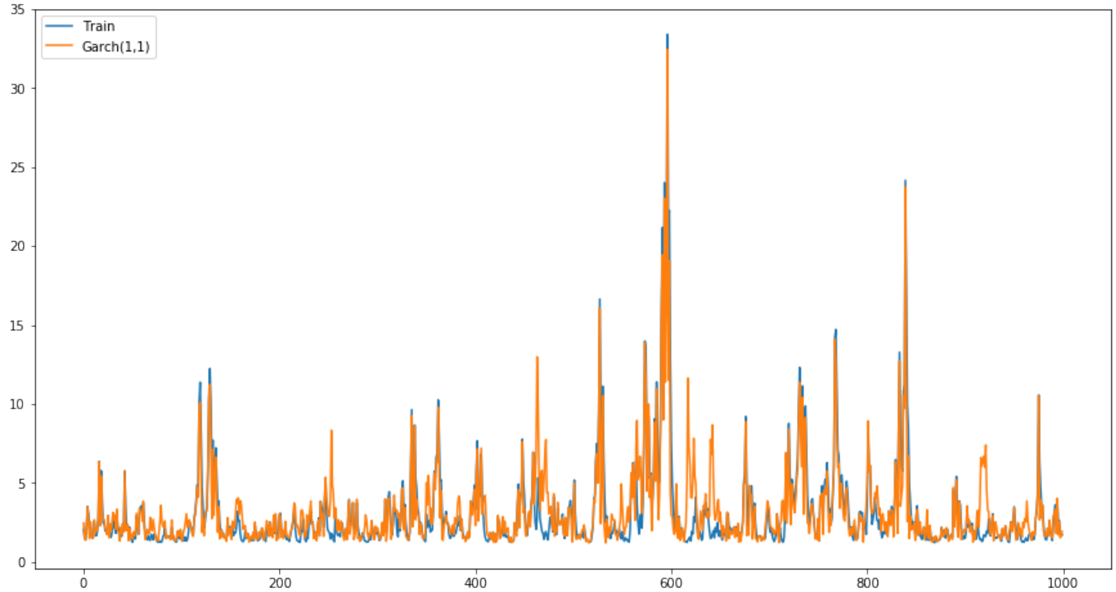
In [124]:
```
omegaarch1=arch1_paramp4[1]
alphaarch1=arch1_paramp4[2]
n=1000

e=np.random.normal(loc=0,scale=1,size=n)
rtAR=np.zeros_like(e)
xtAR=np.zeros_like(e)
sigstAR=np.zeros_like(e)

for x in range(0,n):
    sigstAR[x]=omegaarch1+alphaarch1*(x_half1[x-1]**2)

testdfarch1=pd.DataFrame(np.sqrt(sigma_sqr[:1000]))
ardfarch1=pd.DataFrame(np.sqrt(sigstAR))

plt.figure(figsize=(15,8))
plt.plot(testdfarch1)
plt.plot(ardfarch1)
plt.legend(['Train', 'Arch(1)'], loc='upper left')
plt.show()
rmse=1/n*(ardfarch1-testdfarch1)**2
rmse=rmse.values.sum()
print('RMSE:',rmse)
```




```
RMSE: 0.7888461682561507
```

```
In [125]: omegagarch11=garch11_paramp4[1]
          alphagarch11=garch11_paramp4[2]
          betagarch11=garch11_paramp4[3]
          n=1000
          sigsqrhalf1=sigma_sqr[1000:]

          e=np.random.normal(loc=0,scale=1,size=n)
          rtAR=np.zeros_like(e)
          xtAR=np.zeros_like(e)
          sigstGARCH11=np.zeros_like(e)

          for x in range(0,n):
              sigstGARCH11[x]=omegagarch11+alphagarch11*
              (x_half1[x-1]**2)+betagarch11*(sigsqrhalf1[x-1])

          testdfgarch11=pd.DataFrame(np.sqrt(sigma_sqr[:1000]))
          ardfgarch11=pd.DataFrame(np.sqrt(sigstGARCH11))

          plt.figure(figsize=(15,8))
          plt.plot(testdfgarch11)
          plt.plot(ardfgarch11)
          plt.legend(['Train', 'Garch(1,1)'], loc='upper left')
          plt.show()
          rmse=1/n*(ardfgarch11-testdfgarch11)**2
```

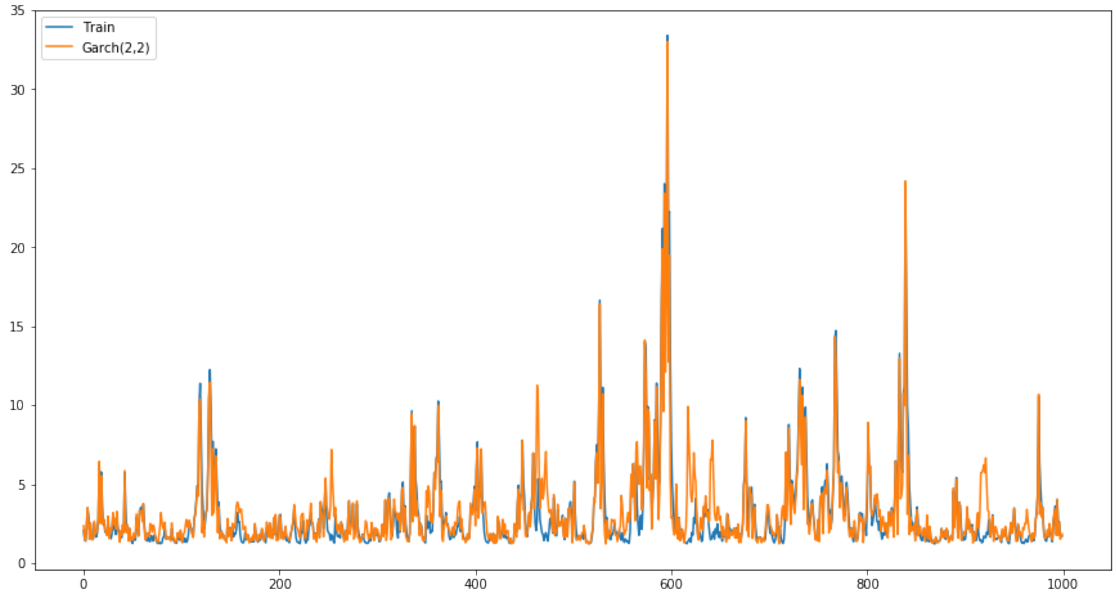
```
rmse=rmse.values.sum()
print('RMSE:',rmse)
```




```
RMSE: 1.9855503140086266
```

```
In [153]: omegagarch22=garch22_paramp4[1]
          alphagarch22=garch22_paramp4[2]
          alpha2garch22=garch22_paramp4[3]
          betagarch22=garch22_paramp4[4]
          beta2garch22=garch22_paramp4[5]
          n=1000

          e=np.random.normal(loc=0,scale=1,size=n)
          sigstGARCH22=np.zeros_like(e)

          for x in range(0,n):
              sigstGARCH22[x]=omegagarch22+alphagarch22*(x_half1[x-1]**2)
              +alpha2garch22*(x_half1[x-2]**2)+betagarch22
              *(sigsqrhalf1[x-1])+beta2garch22*(sigsqrhalf1[x-2])

          testdfgarch22=pd.DataFrame(np.sqrt(sigma_sqr[:1000]))
          ardfgarch22=pd.DataFrame(np.sqrt(sigstGARCH22))

          plt.figure(figsize=(15,8))
          plt.plot(testdfgarch22)
          plt.plot(ardfgarch22)
          plt.legend(['Train', 'Garch(2,2)'], loc='upper left')
```

```
plt.show()
rmse=1/n*(ardfgarch22-testdfgarch22)**2
rmse=rmse.values.sum()
print('RMSE:',rmse)
```




```
RMSE: 1.585339355839273
```

1.2 Empirical Application

1.2.1 Part 1

```
In [113]: # Import .csv files
          # Coke
          cok = pd.read_csv("C:/Users/Diana/Documents/University of Waterloo/" + \
                            "4th Year/ECON 423/COKE.csv")
          cok["Return"] = 100*(np.log(cok["Adj Close"]) - \
                                  np.log(cok["Adj Close"].shift(1)))
          cok = cok.iloc[1:]
          cok_train=cok["Return"][:-2263]
          cok_test=cok["Return"][-2264:]
```

```
In [227]: arch1cok = arch_model(cok["Return"], p=1, q=0)
          arch1cok_res = arch1cok.fit(disp='off')
          arch1cokparam = arch1cok_res.params
          arch1cok_res.summary()
```

```
Out[227]: <class 'statsmodels.iolib.summary.Summary'>
          """
```

```
                   Constant Mean - ARCH Model Results
==============================================================================
Dep. Variable:                    Return   R-squared:                      -0.000
Mean Model:                Constant Mean   Adj. R-squared:                 -0.000
Vol Model:                          ARCH   Log-Likelihood:               -9253.98
Distribution:                     Normal   AIC:                           18514.0
Method:               Maximum Likelihood   BIC:                           18533.2
                                           No. Observations:                 4527
Date:                   Thu, Nov 29 2018   Df Residuals:                     4524
Time:                           03:36:38   Df Model:                            3
                                 Mean Model
=============================================================================
                 coef    std err          t      P>|t|       95.0% Conf. Int.
-----------------------------------------------------------------------------
mu             0.0260  2.675e-02      0.973      0.330 [-2.640e-02,7.846e-02]
                            Volatility Model
========================================================================
                 coef    std err          t      P>|t|  95.0% Conf. Int.
------------------------------------------------------------------------
omega          2.7574      0.195     14.142  2.092e-45 [  2.375,  3.140]
alpha[1]       0.3037  5.280e-02      5.752  8.799e-09 [  0.200,  0.407]
========================================================================

Covariance estimator: robust
"""
```

In [104]:
```python
arch2cok = arch_model(cok["Return"], p=2, q=0)
arch2cok_res = arch2cok.fit(disp='off')
arch2cokparam = arch2cok_res.params
arch2cok_res.summary()
```

Out[104]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                   Constant Mean - ARCH Model Results
==============================================================================
Dep. Variable:                    Return   R-squared:                      -0.000
Mean Model:                Constant Mean   Adj. R-squared:                 -0.000
Vol Model:                          ARCH   Log-Likelihood:               -9202.10
Distribution:                     Normal   AIC:                           18412.2
Method:               Maximum Likelihood   BIC:                           18437.9
                                           No. Observations:                 4527
Date:                   Wed, Nov 28 2018   Df Residuals:                     4523
Time:                           23:35:30   Df Model:                            4
                                 Mean Model
=============================================================================
                 coef    std err          t      P>|t|       95.0% Conf. Int.
-----------------------------------------------------------------------------
mu             0.0313  2.763e-02      1.133      0.257 [-2.286e-02,8.544e-02]
```

```
                         Volatility Model
==========================================================================
                 coef    std err          t      P>|t|     95.0% Conf. Int.
--------------------------------------------------------------------------
omega          2.4340      0.210     11.566  6.151e-31   [  2.021,  2.846]
alpha[1]       0.2449  4.305e-02      5.689  1.274e-08   [  0.161,  0.329]
alpha[2]       0.1420  3.777e-02      3.760  1.700e-04 [6.799e-02,  0.216]
==========================================================================

Covariance estimator: robust
"""
```

In [105]:
```python
garch11cok = arch_model(cok["Return"], p=1, q=1)
garch11cok_res = garch11cok.fit(disp='off')
garch11cokparam = garch11cok_res.params
garch11cok_res.summary()
```

Out[105]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                     Constant Mean - GARCH Model Results
==============================================================================
Dep. Variable:                 Return   R-squared:                      -0.000
Mean Model:             Constant Mean   Adj. R-squared:                 -0.000
Vol Model:                      GARCH   Log-Likelihood:               -9100.95
Distribution:                  Normal   AIC:                           18209.9
Method:            Maximum Likelihood   BIC:                           18235.6
                                        No. Observations:                 4527
Date:                Wed, Nov 28 2018   Df Residuals:                     4523
Time:                        23:35:33   Df Model:                            4
                                 Mean Model
=============================================================================
                 coef    std err          t      P>|t|        95.0% Conf. Int.
-----------------------------------------------------------------------------
mu             0.0412  2.734e-02      1.506      0.132 [-1.242e-02,9.476e-02]
                              Volatility Model
=============================================================================
                 coef    std err          t      P>|t|        95.0% Conf. Int.
-----------------------------------------------------------------------------
omega          0.0214  2.677e-02      0.800      0.424 [-3.105e-02,7.387e-02]
alpha[1]       0.0263  1.615e-02      1.629      0.103 [-5.337e-03,5.796e-02]
beta[1]        0.9683  2.312e-02     41.888      0.000      [  0.923,  1.014]
=============================================================================

Covariance estimator: robust
"""
```

In [106]:
```python
garch12cok = arch_model(cok["Return"], p=1, q=2)
garch12cok_res = garch12cok.fit(disp='off')
```

```
garch12cokparam = garch12cok_res.params
garch12cok_res.summary()
```

Out[106]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                     Constant Mean - GARCH Model Results
==============================================================================
Dep. Variable:                 Return   R-squared:                      -0.000
Mean Model:             Constant Mean   Adj. R-squared:                 -0.000
Vol Model:                      GARCH   Log-Likelihood:               -9094.15
Distribution:                  Normal   AIC:                           18198.3
Method:            Maximum Likelihood   BIC:                           18230.4
                                        No. Observations:                 4527
Date:                Wed, Nov 28 2018   Df Residuals:                     4522
Time:                        23:35:34   Df Model:                            5
                                 Mean Model
=============================================================================
                 coef    std err          t      P>|t|       95.0% Conf. Int.
-----------------------------------------------------------------------------
mu             0.0396  2.694e-02      1.469      0.142 [-1.322e-02,9.240e-02]
                              Volatility Model
=============================================================================
                 coef    std err          t      P>|t|       95.0% Conf. Int.
-----------------------------------------------------------------------------
omega          0.0325  3.709e-02      0.877      0.380   [-4.016e-02,  0.105]
alpha[1]       0.0416  2.168e-02      1.920  5.483e-02 [-8.616e-04,8.413e-02]
beta[1]        0.3774      0.110      3.435  5.916e-04        [  0.162,  0.593]
beta[2]        0.5728      0.123      4.660  3.164e-06        [  0.332,  0.814]
=============================================================================

Covariance estimator: robust
"""
```

In [107]:
```
garch22cok = arch_model(cok["Return"], p=2, q=2)
garch22cok_res = garch22cok.fit(disp='off')
garch22cokparam = garch22cok_res.params
garch22cok_res.summary()
```

Out[107]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                     Constant Mean - GARCH Model Results
==============================================================================
Dep. Variable:                 Return   R-squared:                      -0.000
Mean Model:             Constant Mean   Adj. R-squared:                 -0.000
Vol Model:                      GARCH   Log-Likelihood:               -9098.98
Distribution:                  Normal   AIC:                           18210.0
Method:            Maximum Likelihood   BIC:                           18248.5
                                        No. Observations:                 4527
```

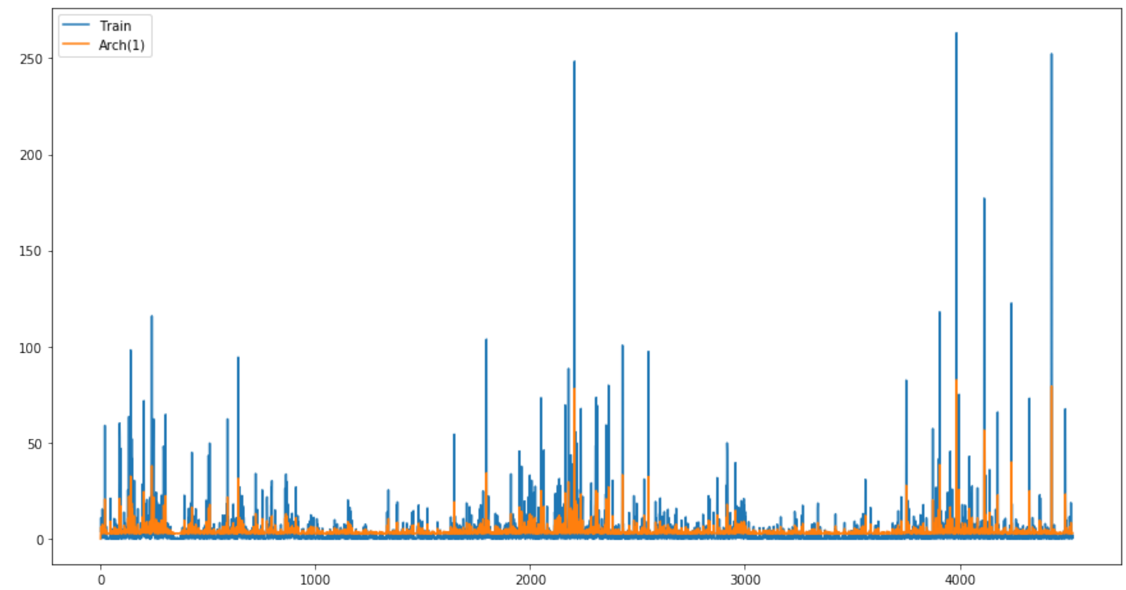
```
Date:                   Wed, Nov 28 2018   Df Residuals:                   4521
Time:                           23:35:36   Df Model:                          6
                                 Mean Model
=============================================================================
                  coef    std err          t      P>|t|       95.0% Conf. Int.
-----------------------------------------------------------------------------
mu              0.0426  2.664e-02      1.600      0.110 [-9.602e-03,9.483e-02]
                             Volatility Model
===========================================================================
                  coef    std err          t      P>|t|     95.0% Conf. Int.
---------------------------------------------------------------------------
omega           0.2226      0.108      2.053  4.010e-02  [1.005e-02,  0.435]
alpha[1]        0.1024  7.672e-02      1.335      0.182 [-4.794e-02,  0.253]
alpha[2]    3.5277e-12  9.539e-02  3.698e-11      1.000     [ -0.187,  0.187]
beta[1]         0.5059      0.845      0.599      0.549     [ -1.151,  2.163]
beta[2]         0.3336      0.829      0.402      0.687     [ -1.291,  1.958]
===========================================================================

Covariance estimator: robust
"""
```

1.2.2 Part 2

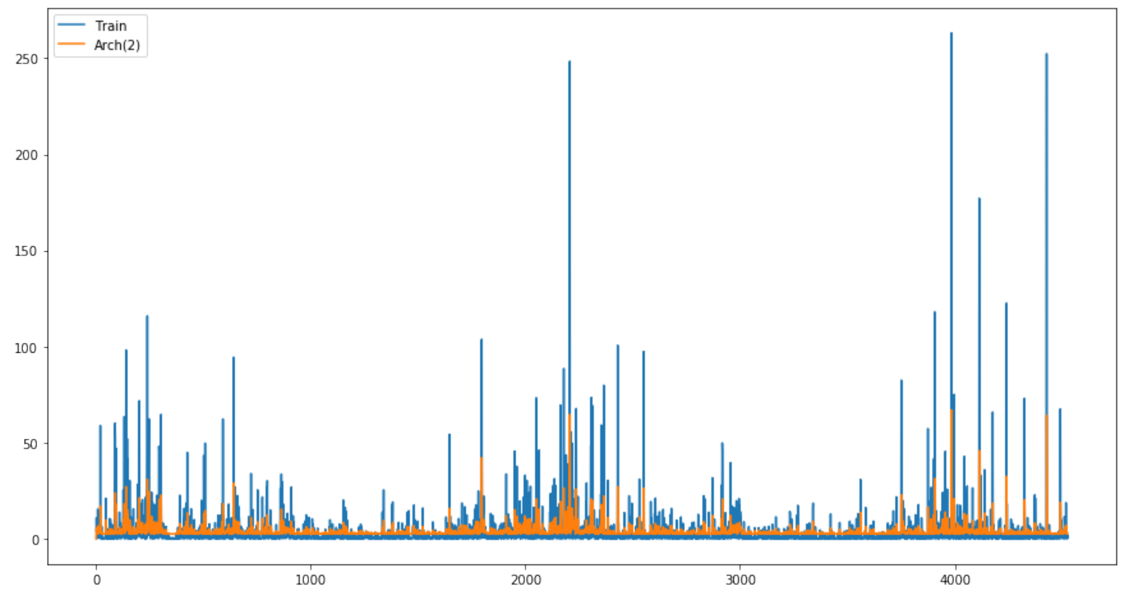
We can look at the AIC (Akaike's Information Criteria) and BIC(Bayesian Information Criteria) of the models to determine which one is the "best" model for the sample data. AIC measures the quality of the model based on the amount of information lost. BIC attempts to select the "true" model from a set of candidate models. GARCH(1,2) has the lowest BIC and AIC and hence is the "best" model.

1.2.3 Part 3

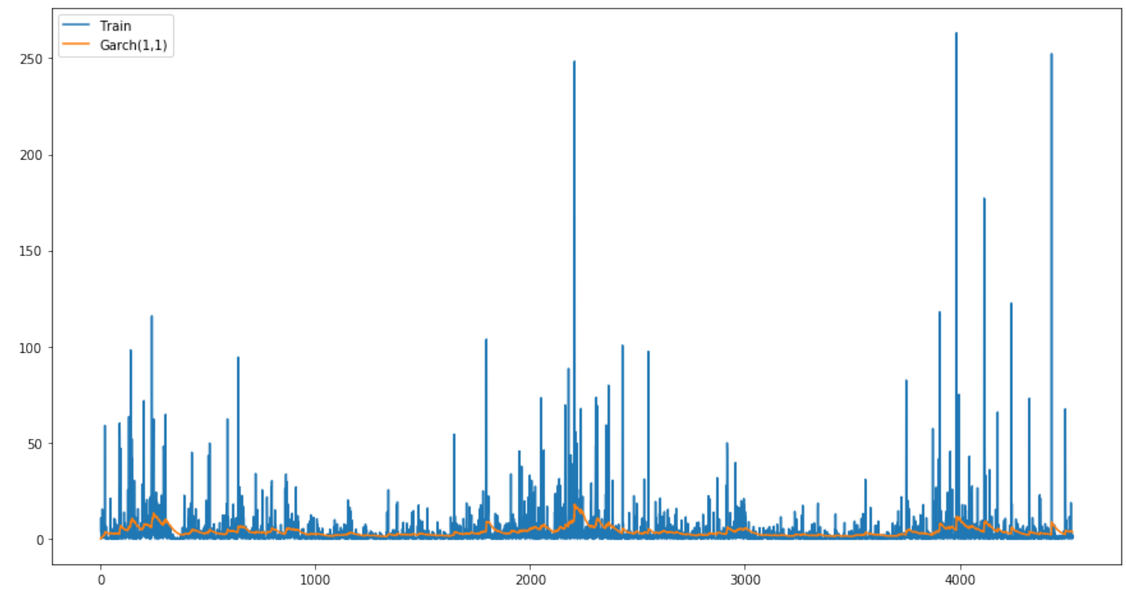
In [228]:
```
omegaarch1cok=arch1cokparam[1]
alphaarch1cok=arch1cokparam[2]
n=len(cok["Return"])

e=np.random.normal(loc=0,scale=1,size=n-1)
varARCH1cok=np.zeros_like(e)

for x in range(1,n-1):
    varARCH1cok[x]=omegaarch1cok+alphaarch1cok*(list(cok["Return"])[x-1]**2)

plt.figure(figsize=(15,8))
plt.plot(np.square(cok["Return"]))
plt.plot(varARCH1cok)
plt.legend(['Train', 'Arch(1)'], loc='upper left')
plt.show()
```



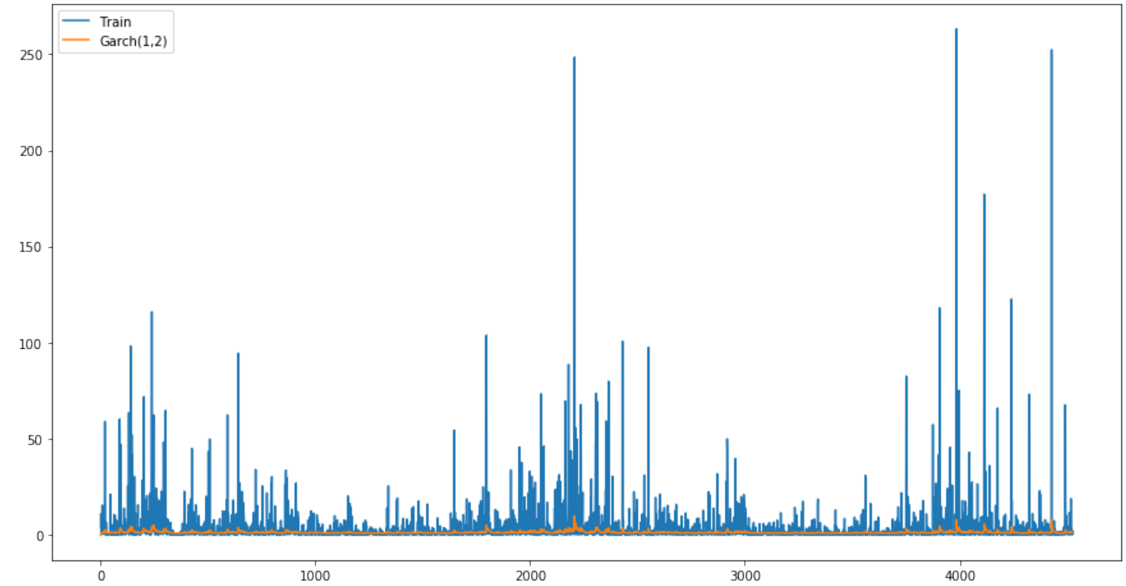

```
In [235]: omegaarch2cok=arch2cokparam[1]
          alphaarch2cok=arch2cokparam[2]
          alpha2arch2cok=arch2cokparam[3]
          n=len(cok["Return"])

          e=np.random.normal(loc=0,scale=1,size=n-1)
          varARCH2cok=np.zeros_like(e)

          for x in range(1,n-1):
              varARCH2cok[x]=omegaarch2cok+alphaarch2cok* \
              (list(cok["Return"])[x-1]**2)+alpha2arch2cok*(list(cok["Return"])[x-2]**2)

          plt.figure(figsize=(15,8))
          plt.plot(np.square(cok["Return"]))
          plt.plot(varARCH2cok)
          plt.legend(['Train', 'Arch(2)'], loc='upper left')
          plt.show()
```



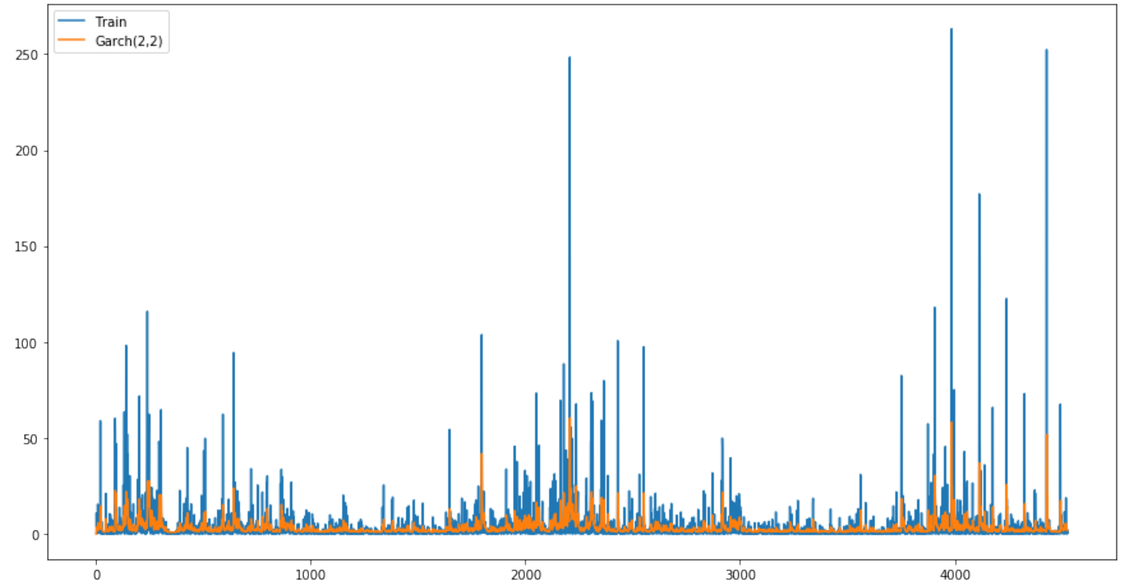

```
In [236]: omegagarch11cok=garch11cokparam[1]
          alphagarch11cok=garch11cokparam[2]
          betagarch11cok=garch11cokparam[3]
          n=len(cok["Return"])

          e=np.random.normal(loc=0,scale=1,size=n-1)
          vargarch11cok=np.zeros_like(e)

          for x in range(1,n-1):
              vargarch11cok[x]=omegagarch11cok+alphagarch11cok* \
              (list(cok["Return"])[x-1]**2)+betagarch11cok*vargarch11cok[x-1]

          plt.figure(figsize=(15,8))
          plt.plot(np.square(cok["Return"]))
          plt.plot(vargarch11cok)
          plt.legend(['Train', 'Garch(1,1)'], loc='upper left')
          plt.show()
```




```
In [237]: omegagarch12cok=garch12cokparam[1]
          alphagarch12cok=garch12cokparam[2]
          betagarch12cok=garch12cokparam[3]
          beta2garch12cok=garch12cokparam[4]
          n=len(cok["Return"])

          e=np.random.normal(loc=0,scale=1,size=n-1)
          vargarch12cok=np.zeros_like(e)

          for x in range(1,n-1):
              vargarch12cok[x]=omegagarch12cok+alphagarch11cok* \
              (list(cok["Return"])[x-1]**2)+betagarch12cok*vargarch12cok[x-1] \
              +beta2garch12cok*vargarch12cok[x-2]

          plt.figure(figsize=(15,8))
          plt.plot(np.square(cok["Return"]))
          plt.plot(vargarch12cok)
          plt.legend(['Train', 'Garch(1,2)'], loc='upper left')
          plt.show()
```



In [238]:
```
omegagarch22cok=garch22cokparam[1]
alphagarch22cok=garch22cokparam[2]
alpha2garch22cok=garch22cokparam[3]
betagarch22cok=garch22cokparam[4]
beta2garch22cok=garch22cokparam[5]
n=len(cok["Return"])

e=np.random.normal(loc=0,scale=1,size=n-1)
vargarch22cok=np.zeros_like(e)

for x in range(1,n-1):
    vargarch22cok[x]=omegagarch22cok+alphagarch22cok \
    *(list(cok["Return"])[x-1]**2)+alphagarch22cok \
    *(list(cok["Return"])[x-2]**2)+betagarch22cok \
    *vargarch12cok[x-1]+beta2garch22cok*vargarch22cok[x-2]

plt.figure(figsize=(15,8))
plt.plot(np.square(cok["Return"]))
plt.plot(vargarch22cok)
plt.legend(['Train', 'Garch(2,2)'], loc='upper left')
plt.show()
```



1.2.4 Part 4

```
In [239]: varARCH1cokpred60 = arch1cok_res.forecast(horizon=60)
          varARCH1cokpred60 = varARCH1cokpred60.variance.dropna().head().T

          varARCH2cokpred60 = arch2cok_res.forecast(horizon=60)
          varARCH2cokpred60 = varARCH2cokpred60.variance.dropna().head().T

          vargarch11cokpred60 = garch11cok_res.forecast(horizon=60)
          vargarch11cokpred60 = vargarch11cokpred60.variance.dropna().head().T

          vargarch12cokpred60 = garch12cok_res.forecast(horizon=60)
          vargarch12cokpred60 = vargarch12cokpred60.variance.dropna().head().T

          vargarch22cokpred60 = garch22cok_res.forecast(horizon=60)
          vargarch22cokpred60 = vargarch22cokpred60.variance.dropna().head().T
```

```
In [274]: def realized_variance(returns, k):
              """
              Returns is a numpy array. This is for the out-of-sample section.
              Returns should include the k-1 data points before the prediction range.

              Then if want to predict after time T, should use
                  realized_variance(returns[T-k-+1:], k)
              """
              sq_returns = returns**2
              realized_variance = pd.Series(sq_returns).rolling(k).sum()[k-1:]
              c = np.var(returns) / np.mean(realized_variance)
```

```
        sigmasq_t = c * realized_variance
        return sigmasq_t

    def MSE_1(rv, preds):
        return np.mean((np.sqrt(rv) - preds)**2)

    MSE_1(realized_variance(cok["Return"][-60-5+1:], 5), varARCH1cokpred60[varARCH1cokpre
```

```
Out[274]: 7.22890897875777
```

```
In [275]: MSE_1(realized_variance(cok["Return"][-60-5+1:], 5), varARCH2cokpred60[varARCH2cokpre
```

```
Out[275]: 7.167660262570629
```

```
In [276]: MSE_1(realized_variance(cok["Return"][-60-5+1:], 5), vargarch11cokpred60[vargarch11co
```

```
Out[276]: 5.121325224938479
```

```
In [277]: MSE_1(realized_variance(cok["Return"][-60-5+1:], 5), vargarch12cokpred60[vargarch12co
```

```
Out[277]: 8.253181411946402
```

```
In [278]: MSE_1(realized_variance(cok["Return"][-60-5+1:], 5), vargarch22cokpred60[vargarch22co
```

```
Out[278]: 8.252735646950587
```

Looking at these MSE results, GARCH(1,1) has the smallest number and hence is better.

```
In [279]: def MAE_1(rv, preds):
              return np.mean(np.abs(np.sqrt(rv) - np.sqrt(preds)))

          MAE_1(realized_variance(cok["Return"][-60-5+1:], 5), varARCH1cokpred60[varARCH1cokpre
```

```
Out[279]: 0.953969511653309
```

```
In [281]: MAE_1(realized_variance(cok["Return"][-60-5+1:], 5), varARCH2cokpred60[varARCH2cokpre
```

```
Out[281]: 0.9497613961086594
```

```
In [282]: MAE_1(realized_variance(cok["Return"][-60-5+1:], 5), vargarch11cokpred60[vargarch11co
```

```
Out[282]: 0.9042371759517865
```

```
In [283]: MAE_1(realized_variance(cok["Return"][-60-5+1:], 5), vargarch12cokpred60[vargarch12co
```

```
Out[283]: 0.9575990138183529
```

```
In [284]: MAE_1(realized_variance(cok["Return"][-60-5+1:], 5), vargarch22cokpred60[vargarch22co
```

```
Out[284]: 0.957586258811187
```

Looking at these MAE Results, GARCH(1,1) has the smallest number and hence is better.