

BÀI KIỂM TRA PHÂN TÍCH CHUỖI THỜI GIAN

Đề 2 : PHÂN TÍCH ARIMA,GARCH

Họ và tên : Trịnh Đạt

Mã sinh viên : 2151264649

Lớp : 63TTNT

Bài làm

1.ARIMA

Mô hình ARIMA là một phương pháp dựa trên giả thuyết chuỗi dừng và phương sai sai số không đổi, sử dụng các tín hiệu quá khứ để dự báo giá trị tương lai của chuỗi thời gian. Ví dụ : ARIMA áp dụng hiệu quả trong việc dự báo các hiện tượng kinh tế, chẳng hạn như dự báo lạm phát của Việt Nam. Mô hình ARIMA kết hợp các tín hiệu từ chuỗi tự hồi quy (AR) với chuỗi trung bình trượt (MA), và được xác định bởi ba tham số: $ARIMA(p,d,q)$, trong đó **p** là bậc hồi quy tự động, **d** là số lần lấy sai phân cần thiết để biến chuỗi thời gian **không dừng** thành chuỗi **dừng**, và **q** là bậc trung bình trượt. Ưu điểm của ARIMA là không cần nhiều dữ liệu, hiệu quả với chuỗi thời gian dừng hoặc có thể làm cho dừng bằng cách lấy sai phân, và cấu trúc đơn giản dễ hiểu(chỉ cần 3 chỉ số tạo nên). Tuy nhiên, ARIMA không xử lý tốt các chuỗi thời gian vụ rõ ràng (trong trường hợp này chúng ta nên dùng SARIMA) và gặp khó khăn khi xử lý dữ liệu bị mất.

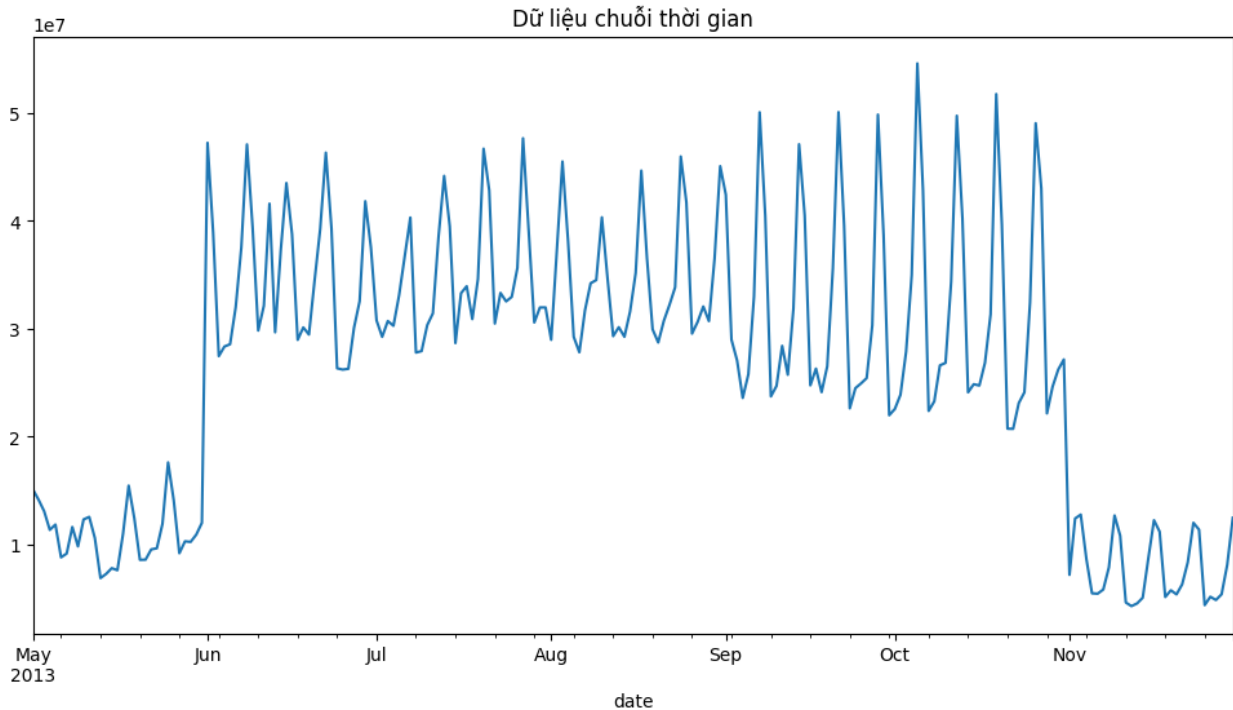
2.GARCH

Mô hình GARCH là một phương pháp giúp dự đoán phương sai của sai số (hay biến động) dựa trên các giá trị sai số trong quá khứ. Mô hình này dựa trên giả thuyết phương sai của sai số thay đổi theo thời gian và phụ thuộc vào các giá trị sai số trong quá khứ. Mô hình GARCH là sự kết hợp bậc thành phần ARCH với bậc thành phần GARCH, và được xác định bởi 2 tham số $GARCH(p,q)$, trong đó **p** là sai số trong quá khứ, **q** là phương sai trong quá khứ. Ưu điểm của GARCH là được ứng dụng rộng rãi trong nhiều lĩnh vực như tài chính, kinh tế, khí hậu... và phản ánh tốt với các đặc điểm thực tế của dữ liệu. Tuy nhiên, GARCH phức tạp hơn so với các mô hình đơn giản như ARIMA, đòi hỏi kiến thức chuyên sâu về thống kê và kinh tế lượng thì mới làm được mô hình.

Bài 2 : Huấn luyện mô hình

MÔ HÌNH ARIMA:

- Trực quan hóa dữ liệu của “truong_3”:



- Dữ liệu của “truong_3” từ tháng 5 đến tháng 11 trong năm 2013 có xu hướng giảm vào tháng 5 và tháng 11 duy trì trong mức từ 0-2, các tháng còn lại duy trì ở cao từ 2-5.

- Sau đó, kiểm tra tính dừng:

```
# Kiểm tra tính dừng bằng kiểm tra Dickey-Fuller mở rộng
result = adfuller(df['truong_3'])
print(f'Thống kê ADF: {result[0]}')
print(f'Giá trị p: {result[1]}')
```

[23] ✓ 0.0s Python

... Thống kê ADF: -1.8800479847552338
Giá trị p: 0.3415045885718311

- Lấy sai phân nếu dữ liệu không dừng:

```
# Áp dụng lấy sai phân nếu dữ liệu không dừng
df['truong_3_diff'] = df['truong_3'].diff().dropna()
```

[97] ✓ 0.0s Python

- Tìm mô hình ARIMA tốt nhất bằng auto_arima:

```
# Tìm mô hình ARIMA tốt nhất bằng auto_arima
model = auto_arima(df['truong_3'], seasonal=False, trace=True)

[98] ✓ 1.5s Python

... Performing stepwise search to minimize aic
ARIMA(2,1,2)(0,0,0)[0] intercept : AIC=7291.994, Time=0.11 sec
ARIMA(0,1,0)(0,0,0)[0] intercept : AIC=7355.125, Time=0.00 sec
ARIMA(1,1,0)(0,0,0)[0] intercept : AIC=7356.124, Time=0.02 sec
ARIMA(0,1,1)(0,0,0)[0] intercept : AIC=7352.697, Time=0.02 sec
ARIMA(0,1,0)(0,0,0)[0] intercept : AIC=7353.128, Time=0.02 sec
ARIMA(1,1,2)(0,0,0)[0] intercept : AIC=7300.303, Time=0.26 sec
ARIMA(2,1,1)(0,0,0)[0] intercept : AIC=7288.000, Time=0.05 sec
ARIMA(1,1,1)(0,0,0)[0] intercept : AIC=7345.726, Time=0.05 sec
ARIMA(2,1,0)(0,0,0)[0] intercept : AIC=7315.067, Time=0.03 sec
ARIMA(3,1,1)(0,0,0)[0] intercept : AIC=7289.985, Time=0.13 sec
ARIMA(3,1,0)(0,0,0)[0] intercept : AIC=7315.762, Time=0.03 sec
ARIMA(3,1,2)(0,0,0)[0] intercept : AIC=7310.356, Time=0.14 sec
ARIMA(2,1,1)(0,0,0)[0] intercept : AIC=7285.995, Time=0.16 sec
ARIMA(1,1,1)(0,0,0)[0] intercept : AIC=7343.726, Time=0.07 sec
ARIMA(2,1,0)(0,0,0)[0] intercept : AIC=7313.065, Time=0.02 sec
ARIMA(3,1,1)(0,0,0)[0] intercept : AIC=7287.980, Time=0.08 sec
ARIMA(2,1,2)(0,0,0)[0] intercept : AIC=7289.735, Time=0.12 sec
ARIMA(1,1,0)(0,0,0)[0] intercept : AIC=7354.125, Time=0.02 sec
ARIMA(1,1,2)(0,0,0)[0] intercept : AIC=7298.009, Time=0.06 sec
ARIMA(3,1,0)(0,0,0)[0] intercept : AIC=7313.761, Time=0.03 sec
ARIMA(3,1,2)(0,0,0)[0] intercept : AIC=7308.110, Time=0.15 sec

Best model: ARIMA(2,1,1)(0,0,0)[0]
Total fit time: 1.560 seconds
```

- Huấn luyện mô hình:

```
# Thực hiện việc fit mô hình vào dữ liệu
model.fit(df['truong_3'])

[99] ✓ 0.1s Python

... * ARIMA
ARIMA(2,1,1)(0,0,0)[0]
```

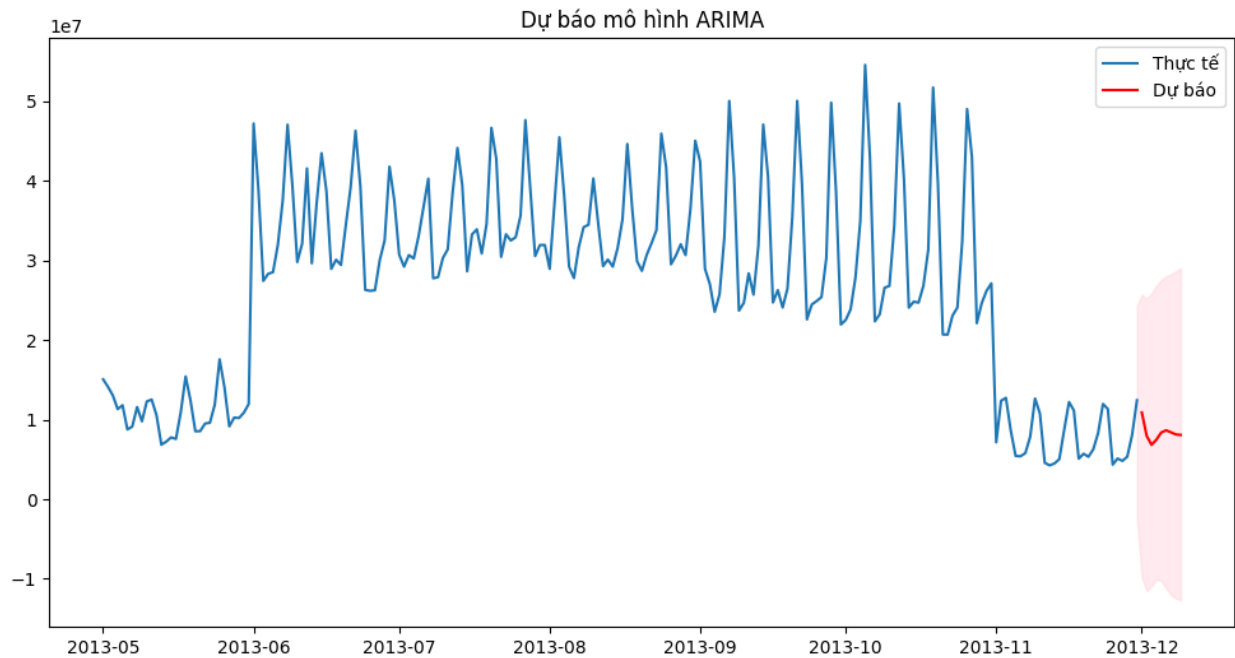
- Tóm tắt mô hình:

```
# Tóm tắt mô hình
print(model.summary())

[100] ✓ 0.0s Python

... SARIMAX Results
=====
Dep. Variable: y No. Observations: 214
Model: SARIMAX(2, 1, 1) Log Likelihood: -3638.998
Date: Tue, 04 Jun 2024 AIC: 7285.995
Time: 12:06:40 BIC: 7299.440
Sample: 05-01-2013 HQIC: 7291.429
- 11-30-2013
Covariance Type: opg
=====
coef std err z P>|z| [0.025 0.975]
-----
ar.L1 0.6197 0.098 6.310 0.000 0.427 0.812
ar.L2 -0.4543 0.076 -5.976 0.000 -0.603 -0.305
ma.L1 -0.7478 0.078 -9.642 0.000 -0.900 -0.596
sigma2 4.675e+13 1.21e-15 3.86e+28 0.000 4.68e+13 4.68e+13
=====
Ljung-Box (L1) (Q): 0.01 Jarque-Bera (JB): 286.92
Prob(Q): 0.92 Prob(JB): 0.00
Heteroskedasticity (H): 1.44 Skew: 1.26
Prob(H) (two-sided): 0.12 Kurtosis: 8.09
=====
```

- Dự báo 10 kỳ:



Nhìn chung, giá trị thực tế thấp hơn so với giá trị dự đoán bởi mô hình ARIMA trong hầu hết các thời điểm được thể hiện trong biểu đồ. Điều này cho thấy mô hình ARIMA có thể chưa hoàn toàn chính xác trong việc dự đoán.

MÔ HÌNH GARCH:

- Dự đoán cho cột "truong_1" và cho $p=1, q=1$:

```
# Giả sử bạn muốn dự đoán cột 'truong_1'
returns = df['truong_1'].pct_change().dropna() # Tính toán tỷ lệ lợi nhuận

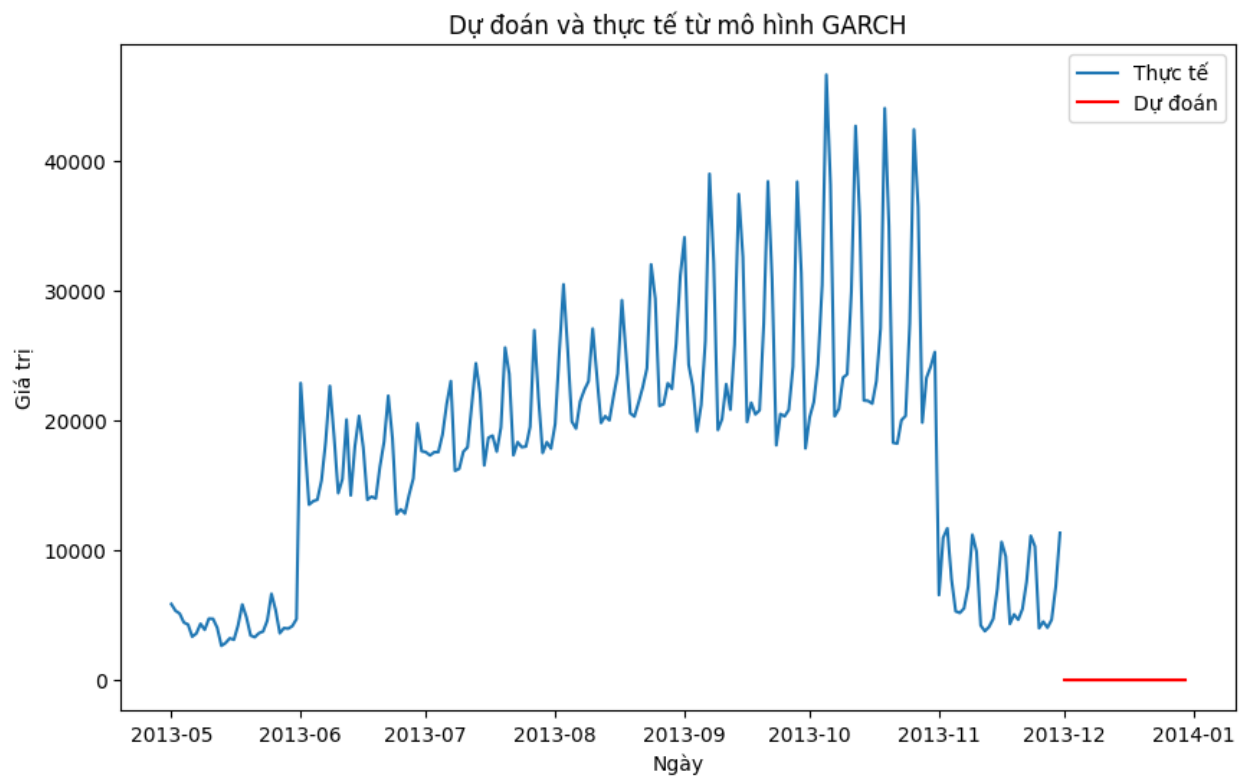
[103] ✓ 0.0s Python

# Khớp mô hình GARCH(1,1)
model = arch_model(returns, mean='Zero', vol='GARCH', p=1, q=1)
result = model.fit()

[104] ✓ 0.0s Python

... Iteration: 1, Func. Count: 5, Neg. LLF: 689.1643743270356
Iteration: 2, Func. Count: 12, Neg. LLF: 83.5158620460978
Iteration: 3, Func. Count: 16, Neg. LLF: 83.51411722192245
Iteration: 4, Func. Count: 20, Neg. LLF: 83.5120421180132
Iteration: 5, Func. Count: 24, Neg. LLF: 83.51204154206158
Optimization terminated successfully (Exit mode 0)
Current function value: 83.51204154206158
Iterations: 5
Function evaluations: 24
Gradient evaluations: 5
```

- Dự đoán:



- Kết quả dự đoán thấp hơn so với kết quả thực tế.

Link github: https://github.com/trinhdat24/KiemTraGiuaKy_TimeSeries