

SỬ DỤNG MÔ HÌNH ARIMA ĐỂ DỰ ĐOÁN THỜI TIẾT VÀ PHÂN TÍCH KẾT QUẢ

Đông Dương - Anh Hào - Nga Nhi

Ngày 18 tháng 7 năm 2019

MỞ ĐẦU: Thời tiết là một yếu tố có ảnh hưởng lớn đến hoạt động của con người. Vì vậy việc dự báo thời tiết là yêu cầu trọng yếu và luôn được quan tâm.

Đề tài giới thiệu về dạng dữ liệu chuỗi thời gian và mô hình ARIMA. Mô hình ARIMA sẽ được sử dụng để phân tích các đặc trưng của chuỗi thời gian, ước lượng các tham số, và dựa vào đó đưa ra dự báo về biến đổi của chuỗi thời gian trong tương lai sự biến động trong tương lai của thời tiết mà không dùng các phương pháp dự báo truyền thống. Từ kết quả của mô hình ARIMA, đề tài sẽ đưa ra đánh giá về sự chuẩn xác và những ưu nhược điểm khi dự báo thời tiết bằng mô hình này.

1 Bài toán Dự báo thời tiết

Một mô hình Machine Learning gồm 5 thành phần: **TEFPA**

- **Task:** Là nhiệm vụ mà mô hình phải thực hiện; mô hình sẽ nhận vào thông tin gì và cho ra thông tin gì.
- **Experience:** Những dữ liệu, kinh nghiệm mà mô hình học được;
- **Function:** Không gian hàm số để miêu tả mô hình;
- **Performance:** Độ đo thể hiện mức độ phù hợp của dữ liệu đầu ra (output) so với dữ liệu thực tế (ground truth);

- **Algorithm:** thuật toán được sử dụng để tối ưu hóa mức độ phù hợp của output so với ground truth.

Trong bài toán Dự báo thời tiết sử dụng mô hình ARIMA:

- **Task** là dự báo thông số thời tiết (nhiệt độ, lượng mưa, độ ẩm) trung bình tháng ở Hà Nội trong 2 năm tới. Đầu vào (input) của mô hình là thông số thời tiết ở Hà Nội từ tháng 1 năm 2010 đến tháng 6 năm 2019; đầu ra (output) của mô hình là thông số thời tiết ở Hà Nội từ tháng 7 năm 2019 đến tháng 12 năm 2021;
- **Experience** là bộ dữ liệu (dataset) thu thập được về nhiệt độ, lượng mưa, và độ ẩm trung bình tháng ở Hà Nội từ tháng 1 năm 2010 đến tháng 6 năm 2019 (nói kĩ hơn ở phần 1.2 Dữ liệu chuỗi thời gian);
- **Function** là các hàm số trong thuật toán phân ARIMA, như hàm tự hồi quy (Autoregression - AR), hàm sai phân (Integrated), hàm trung bình động (Moving Averages - MA) (nói kĩ hơn ở phần 2. Mô hình ARIMA);
- **Performance** là hàm đánh giá Akaike Information Criterion - AIC (nói kĩ hơn ở phần 2. Mô hình ARIMA);
- **Algorithm** là thuật toán Grid Search (nói kĩ hơn ở phần 2. Mô hình ARIMA).

2 Dữ liệu chuỗi thời gian (Time Series)

2.1 Một số khái niệm

Dữ liệu chuỗi thời gian (Time Series) là chuỗi các giá trị của một đại lượng nào đó được ghi lại theo các khoảng thời gian liên nhau với một tần suất thống nhất. Các thông số thời tiết là ví dụ điển hình về dữ liệu chuỗi thời gian.

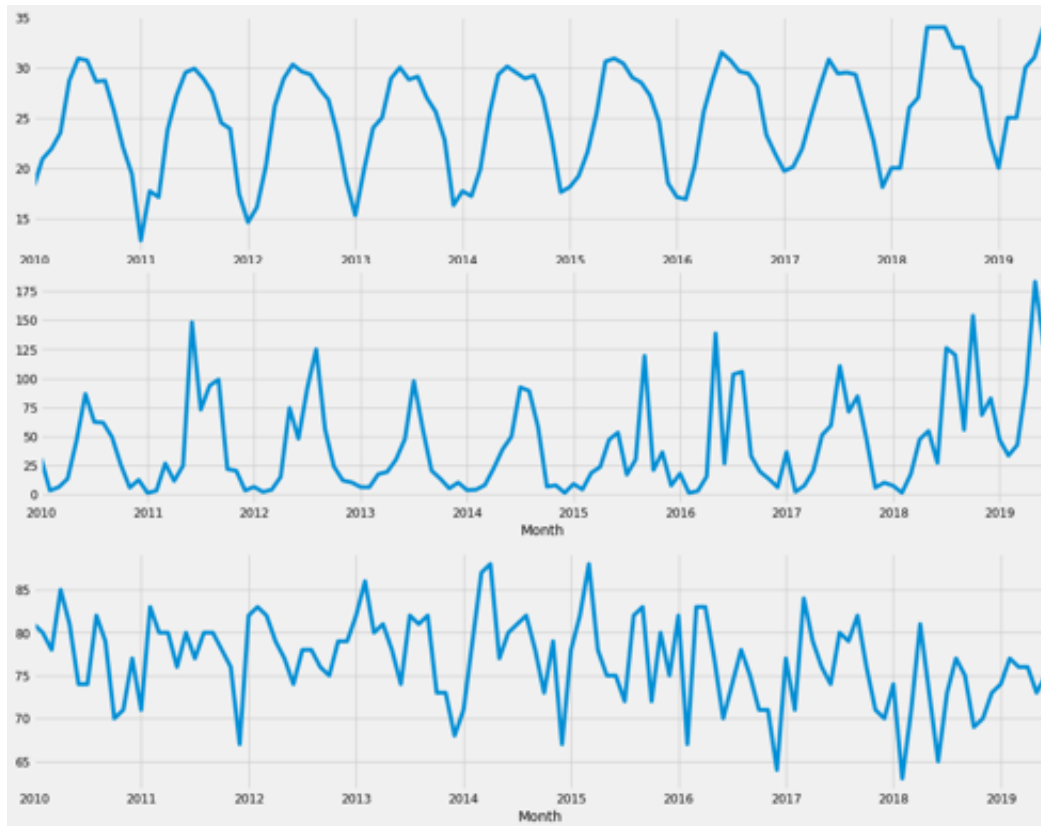
Phân tích chuỗi thời gian (Time series analysis) là việc trích xuất những thuộc tính thống kê đặc trưng có ý nghĩa. Xu hướng đi lên của nhiệt độ trung bình toàn cầu qua các năm là một ví dụ về output của phân tích chuỗi thời gian.

Dự đoán chuỗi thời gian (Time series forecasting) là việc sử dụng mô hình để dự đoán các điểm dữ liệu trong tương lai dựa vào thông tin dữ liệu đã biết trong quá khứ; ví dụ như dự

báo nhiệt độ trung bình các tháng của năm sau dựa vào thông tin nhiệt độ trung bình các tháng của 5 năm gần nhất.

2.2 Dataset Time Series của đề tài

Đề tài sử dụng dataset là nhiệt độ, lượng mưa, và độ ẩm trung bình tháng ở Hà Nội từ tháng 1 năm 2010 đến tháng 6 năm 2019.[1][2]



Hình 1: Từ trên xuống dưới: Biểu đồ nhiệt độ, lượng mưa, và độ ẩm trung bình tháng ở Hà Nội từ tháng 1 năm 2010 đến tháng 6 năm 2019.

2.3 Phân tích thành phần Time Series (Decomposition of Time Series)

2.3.1 Các thành phần của Time Series

Time Series gồm 4 thành phần:

1. Thành phần xu hướng (Trend Component)

Thể hiện xu hướng biến đổi dài hạn của chuỗi thời gian. Đồ thị thể thành phần xu hướng thường mịn do đã loại bỏ đi những biến động ngắn hạn.

Ví dụ: xu hướng biến đổi đi lên của nhiệt độ trung bình toàn cầu qua các năm.

2. Thành phần mùa (Seasonal Component)

Thể hiện những biến đổi tuần hoàn ngắn hạn có tính lặp lại với tần suất cố định.

Ví dụ: nhiệt độ trung bình tháng ở Hà Nội có xu hướng tăng vào giữa năm, giảm vào đầu và cuối năm.

3. Thành phần chu kỳ (Cyclical Component)

Thể hiện những biến đổi lặp lại nhưng không có tần suất cố định; thường do ảnh hưởng của yếu tố ngoại cảnh.

Ví dụ: tỉ lệ thất nghiệp thường tăng đột biến mỗi khi có khủng hoảng kinh tế.

4. Thành phần bất thường (Irregular Component / Residual)

Thể hiện những biến đổi ngẫu nhiên của dữ liệu.

Trong bài toán Dự báo thời tiết của đề tài, do input chỉ có thông số thời tiết trung trong quá khứ, không có thêm thông tin về các yếu tố bên ngoài, nên mô hình của đề tài sẽ chỉ xét đến 3 thành phần của Time Series là Trend, Seasonal, và Residual.

2.3.2 Phương pháp phân tích các thành phần của Time Series

Có nhiều phương pháp phân tích các thành phần của Time Series với các ưu, nhược điểm khác nhau. Đề tài sẽ giới thiệu phương pháp phân tích cổ điển nhất (Classical Decomposition) và cách phân tích các thành phần của Time Series tự động bằng thư viện *statsmodel* trong *python*.

Gọi y_t là giá trị của Time Series data tại thời điểm t ; T_t, S_t, R_t lần lượt là giá trị các thành phần Trend, Seasonal, và Residual của Time Series data tại thời điểm t . y_t có thể được biểu diễn dưới dạng tổng hoặc tích của các hàm thành phần. Trong trường hợp S_t không tăng về giá trị qua các chu kỳ, phương pháp phân tích thành tổng thường được sử dụng.

Các bước phân tích các thành phần của Time Series:

- *Bước 1:* Tính thành phần Trend bằng phương pháp Làm mịn Trung bình động (Moving Average smoothing)

Thành phần Trend của time series tại thời điểm t được tính bằng trung bình cộng của $2k+1$ điểm dữ liệu gần thời điểm t nhất.

$$T_t = \sum_{i=-k}^k y_{t+i}$$

Đây là phương pháp cơ bản nhất để loại bỏ đi những biến động ngắn hạn của dữ liệu time series.

- *Bước 2:* Tính hiệu số giữa dữ liệu thực tế và thành phần Trend của nó (detrended series) tại thời điểm t .

$$y_t - T_t$$

- *Bước 3:* Thành phần Seasonal của dữ liệu time series được tính bằng trung bình cộng của detrended series và chuẩn hóa.

Trong Classical Decomposition, độ lớn của thành phần Seasonal được cho là giống hệt nhau trong mỗi chu kỳ, không bị ảnh hưởng bởi thời gian. Tại mỗi thời điểm trong chu kỳ, thành phần Seasonal được tính bằng trung bình cộng của các giá trị detrended tại các thời điểm tương ứng trong mỗi chu kỳ đã ghi nhận.

Ví dụ, với dataset là nhiệt độ trung bình tháng ở Hà Nội từ năm 2010 đến năm 2018, thành phần Seasonal của tháng 1 được tính bằng:

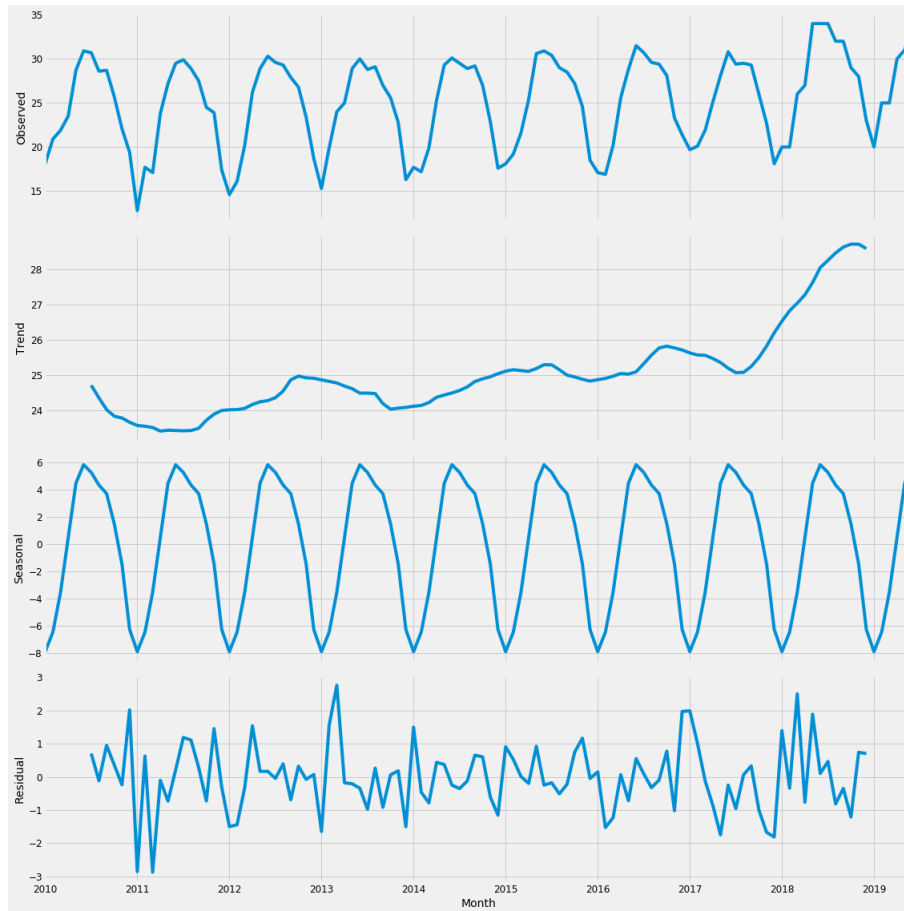
$$S_{Jan} = \frac{1}{8} \sum_{i=Jan,2010}^{Jan,2018} (y_i - T_i)$$

Thành phần Seasonal sẽ được chuẩn hóa sao cho tổng các giá trị S_t trong một chu kỳ bằng 0

- *Bước 4:* Thành phần Residual là phần sai khác còn lại giữa dữ liệu thực, thành phần Trend, và thành phần Seasonal.

$$R_t = y_t - T_t - S_t$$

Có thể sử dụng lệnh *seasonal decomposition* trong thư viện *statsmodel* của *python* để tự động phân tích time series thành 3 thành phần.



Hình 2: Từ trên xuống dưới: Biểu đồ dữ liệu Thực, thành phần Trend, thành phần Seasonal, và thành phần Residual của nhiệt độ trung bình tháng ở Hà Nội từ tháng 1 năm 2010 đến tháng 6 năm 2019.

3 Mô hình ARIMA

Mô hình ARIMA, tên đầy đủ là Auto Regressive Integrated Moving Average, được sử dụng để khai phá dữ liệu chuỗi thời gian, giúp hiểu rõ hơn các đặc trưng của dữ liệu và đưa ra dự đoán phù hợp nhất trong tương lai.

Mô hình ARIMA(p,d,q) gồm 3 mô hình kết hợp với nhau: mô hình AR(p), mô hình MA(q), và mô hình Sai phân I(d)

3.1 Mô hình AR(p)

Mô hình Auto Regressive - AR(p) hồi quy trên chính số liệu quá khứ của nó ở những chu kỳ trước.

$$y_t = a_0 + a_1 \cdot y_{t-1} + a_2 \cdot y_{t-2} + a_p \cdot y_{t-p} + e_t$$

Trong đó:

- y_t là dữ liệu hiện tại;
- y_{t-i} là dữ liệu trong quá khứ;
- a_i là các tham số phân tích hồi quy;
- e_t là sai số dự báo ngẫu nhiên của giai đoạn hiện tại. Giá trị trung bình hi vọng là 0.

Nói cách khác, khi sử dụng phân tích hồi quy y_t theo các giá trị time series có độ trễ, chúng ta sẽ có mô hình AR. Mô hình AR tách yếu tố Trend ra khỏi yếu tố thời gian, tập trung miêu tả thành phần Seasonal của dữ liệu Time Series.

Số dữ liệu quá khứ được sử dụng trong mô hình là bậc p của mô hình AR. p càng lớn thì mô hình càng xét nhiều dữ liệu trong quá khứ.

Mô hình AR(1) : $y_t = a_0 + a_1 \cdot y_{t-1} + e_t$

Mô hình AR(2) : $y_t = a_0 + a_1 \cdot y_{t-1} + a_2 \cdot y_{t-2} + e_t$

3.2 Mô hình MA(q)

Mô hình bình quân di động là một trung bình trọng số của những sai số mới nhất.

$$y_t = b_0 + e_t + b_1 \cdot e_{t-1} + b_2 \cdot e_{t-2} + b_p \cdot e_{t-p}$$

Trong đó:

- y_t là dữ liệu hiện tại;
- e_t là sai số dự báo ngẫu nhiên;
- e_{t-i} là sai số dự báo trong quá khứ;
- b_i là giá trị trung bình của y_t và các hệ số bình quân di động.

Số sai số quá khứ được sử dụng trong mô hình bình quân di động là bậc q của mô hình.

Mô hình $MA(1)$: $y_t = b_0 + e_t + b_1 \cdot e_{t-1}$

Mô hình $MA(2)$: $y_t = b_0 + e_t + b_1 \cdot e_{t-1} + b_2 \cdot e_{t-2}$

3.3 Mô hình Sai phân I(d)

Chuỗi thời gian được coi là *chuỗi dừng* nếu như trung bình và phương sai của nó không đổi theo thời gian và giá trị của đồng phương sai giữa hai thời đoạn chỉ phụ thuộc vào khoảng cách và độ trễ về thời gian giữa hai thời đoạn này chứ không phụ thuộc vào thời điểm thực tế mà đồng phương sai được tính.

Sai phân chỉ sự khác nhau giữa giá trị hiện tại và giá trị trước đó. Phân tích sai phân nhằm làm cho ổn định giá trị trung bình của chuỗi dữ liệu, giúp cho việc chuyển đổi chuỗi thành một chuỗi dừng.

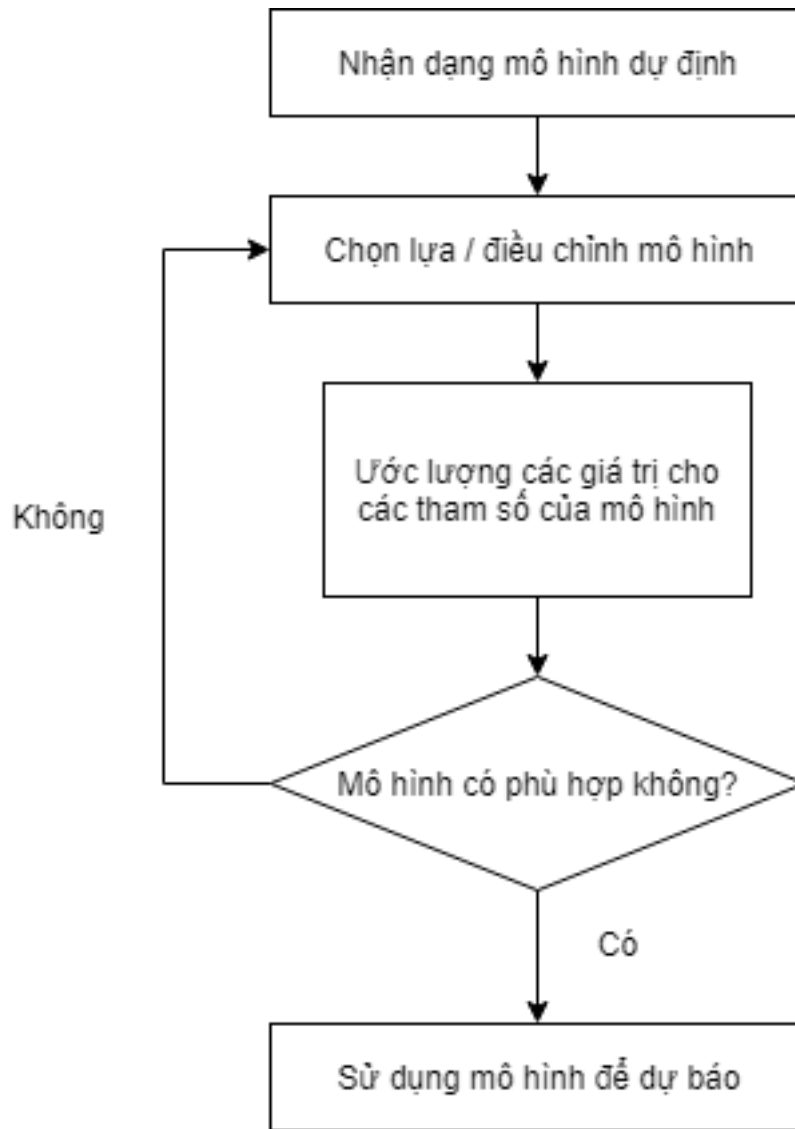
Sai phân lần 1 $I(1)$: $z_t = y_t - y_{t-1}$

Sai phân lần 2 $I(2)$: $h_t = z_t - z_{t-1}$

3.4 Mô hình ARIMA

Mô hình ARIMA là sự kết hợp của 3 mô hình trên.

Mô hình ARIMA được thiết lập dựa trên mô hình Box-Jenkins.



Hình 3: Sơ đồ mô phỏng mô hình Box-Jenkins

Phương pháp Box – Jenkins bao gồm các bước chung:

1. Xác nhận mô hình thử nghiệm
2. Ước lượng tham số
3. Kiểm nghiệm bằng chuẩn đoán
4. Dự đoán

3.4.1 Xác nhận mô hình thử nghiệm

Mô hình ARIMA gồm ba thông số p, d, q ; được thể hiện dưới dạng $ARIMA(p,d,q)$. Trong đó:

- $AR(p)$ là phần auto-regressive (đặc trưng cho tính thời vụ *seasonality* của data) là không gian hàm tuyến tính hồi quy trên chính số liệu quá khứ ở những chu kì trước.
- $I(d)$ là phần integrated (sai phân) (đặc trưng cho xu hướng *trend* của data), với mỗi giá trị chính là hiệu của giá trị hiện tại và giá trị trước của data.
- $MA(q)$ là phần moving-average (bình quân động) (đặc trưng cho tính nhiễu *noise* của data), là trung bình trọng số của những sai số mới nhất đến thời điểm hiện tại.

Do đặc điểm của dữ liệu thời tiết, chúng ta sẽ dùng mô hình cải tiến và chuyên dụng hơn đối với dữ liệu mang tính seasonal, đó là mô hình SARIMAX với bộ thông số $SARIMAX(p,d,q)(P,D,Q)s$. Trong đó bộ (P,D,Q) giống như (p,d,q) nhưng là các đặc trưng của seasonal data, s là chu kì thời gian.

3.4.2 Ước lượng tham số

Sử dụng *grid search* để liệt kê tất cả các tổ hợp thông số (p,d,q) và (P,D,Q) dưới dạng list (trong mô hình thử nghiệm, mỗi thông số sẽ được sinh giá trị 0 hoặc 1).

Với mỗi bộ thông số vừa sinh ra, ta dùng model SARIMAX để tính giá trị AIC. Giá trị này sẽ đánh giá mức độ fit (độ bám sát) của mô hình với dữ liệu. Bộ thông số tối ưu nhất sẽ cho ra giá trị AIC nhỏ nhất.

3.4.3 Kiểm định bằng chẩn đoán

Trong bài toán dự báo thời tiết của đề tài, phần dữ liệu từ tháng 1 năm 2018 sẽ được tách ra để sử dụng làm test data. Sau khi các tham số của mô hình đã được xây dựng, mô hình sẽ dựa vào dữ liệu training (từ tháng 1 năm 2010 đến tháng 12 năm 2017) và đưa ra chẩn đoán bằng cách so sánh dữ liệu dự đoán với dữ liệu thực trên test data, và kiểm tra một lần nữa độ chính xác bằng chuẩn đánh giá MSE hoặc RMSE.

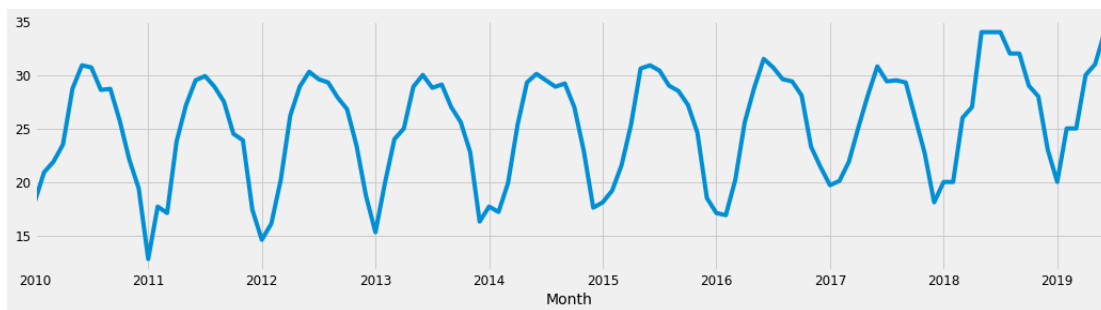
3.4.4 Dự báo

Khi chẩn đoán của mô hình trở nên chính xác, mô hình sẽ tiến hành đưa ra dự báo các giá trị tương lai bắt đầu tại thời điểm cuối của data.

4 Áp dụng mô hình SARIMAX cho bài toán Dự báo thời tiết

4.1 Dự đoán Nhiệt độ

- Số liệu thua thập được



Hình 4: Nhiệt độ trung bình tháng ở Hà Nội từ tháng 1 năm 2010 đến tháng 6 năm 2019

- Khai báo mô hình SARIMAX trong *python*

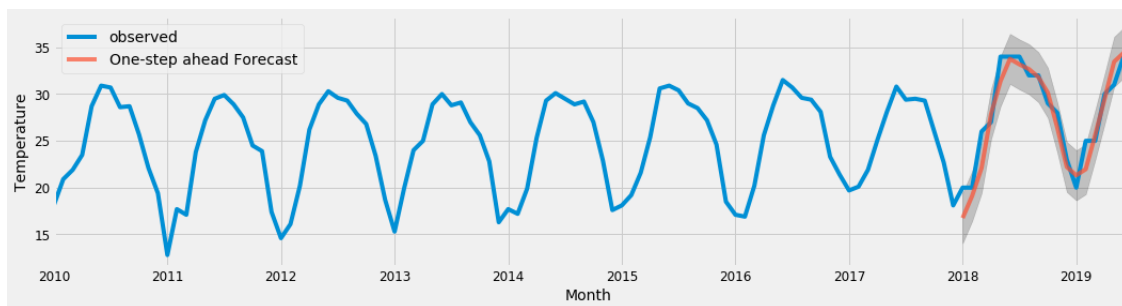
```
1 p = d = q = range(0, 2)
2 pdq = list(itertools.product(p, d, q))
3 period = 12
4 seasonal_pdq = [(x[0], x[1], x[2], period)
5                  for x in list(itertools.product(p, d, q))]
6
7 optimal_result = 10**5
8 for param in pdq:
9     for param_seasonal in seasonal_pdq:
10         try:
```

```

11         mod = sm.tsa.statespace.SARIMAX(y,
12                                         order = param,
13                                         seasonal_order = param_seasonal,
14                                         enforce_stationarity=False,
15                                         enforce_invertibility=False)
16
17     results = mod.fit()
18
19     print('ARIMA{x}{y} - AIC:{z}'
20           .format(param, param_seasonal,
21                   period, results.aic))
22
23     if results.aic < optimal_result:
24         optimal_result = results.aic
25         optimal_order = param
26         optimal_seasonal_order = param_seasonal
27
28     except:
29         continue

```

- Số liệu dự đoán trên test data



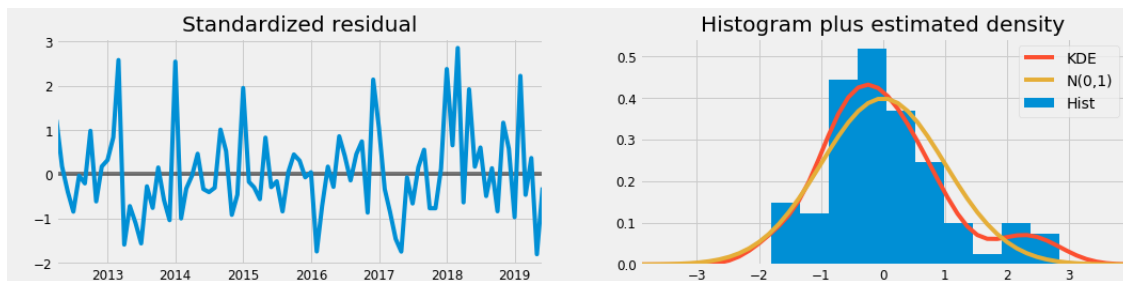
Hình 5: Dự đoán Nhiệt độ trung bình tháng ở Hà Nội từ tháng 1 năm 2018 đến tháng 6 năm 2019

- Nhận xét

Quan sát chủ quan: Dữ liệu dự đoán bám khá sát dữ liệu thực tế cả về độ lớn và chiều hướng thay đổi tại mỗi thời điểm.

Kiểm tra bằng hàm MSE, RMSE: Mô hình dự đoán cho ra giá trị $MSE = 3.2$ và $RMSE = 1.79$ đều nhỏ, cho thấy dữ liệu dự đoán bám khá sát dữ liệu thực tế.

Kiểm tra giả thiết: Trong mô hình ARIMA hay SARIMAX, một giả thiết quan trọng được đặt ra đó là thành phần Residual là thành phần ngẫu nhiên, không có quy luật. Vì vậy, kiểm tra sự ngẫu nhiên của thành phần Residual cũng là một cách để kiểm tra xem mô hình SARIMAX có phải là một mô hình dự đoán hợp lý cho dữ liệu time series của bài toán hay không.



Hình 6

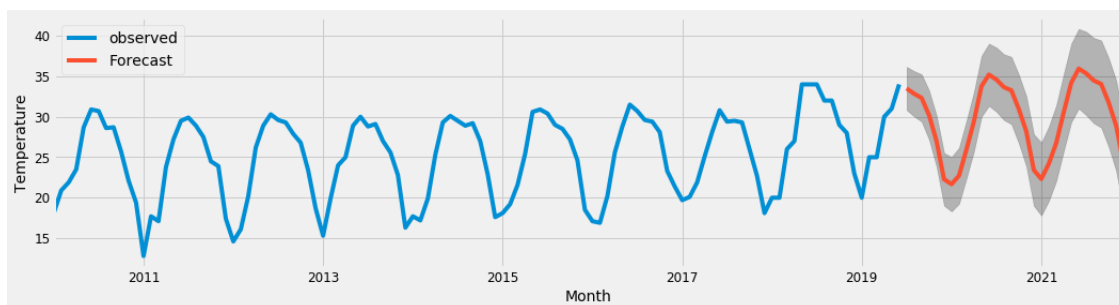
Hình vẽ bên trái biểu diễn thành phần Residual của test data sau khi được chuẩn hóa với tổng tất cả các giá trị bằng 0.

Ta không nhìn thấy bất kỳ quy luật cụ thể nào với các giá trị Residual này.

Hình vẽ bên phải biểu diễn phân bố của thành phần Residual theo giá trị tăng dần. Biểu đồ Histogram (được biểu diễn bằng các cột màu xanh) thể hiện phân bố theo từng khoảng $((-3, -2), (-2, -1), \dots)$. Đường cong KDE (được biểu diễn bằng đường màu đỏ) thể hiện phân bố theo các giá trị liên tục.

Một chuỗi các giá trị ngẫu nhiên sau khi được chuẩn hóa thường có xu hướng phân bố theo normal distribution (biểu diễn bằng đường cong $N(0,1)$ màu vàng). Đường cong KDE không có sự khác biệt quá lớn so với đường cong $N(0,1)$ nên biểu đồ này không phát hiện ra quy luật bất thường nào của các giá trị Residual

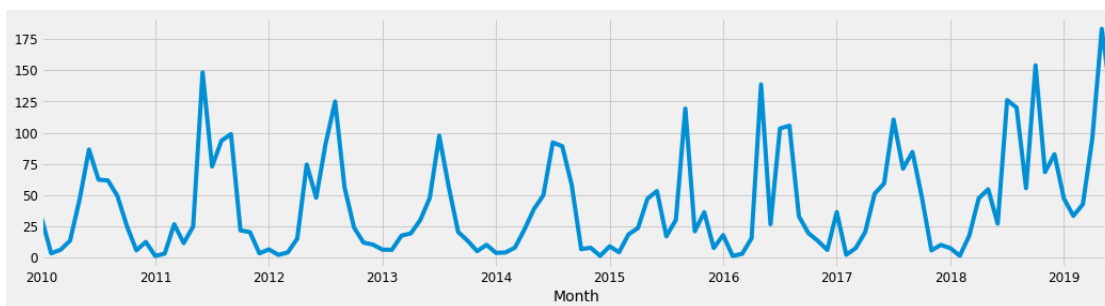
- Dự đoán



Hình 7: Dự đoán nhiệt độ trung bình tháng ở Hà Nội trong 2 năm tới

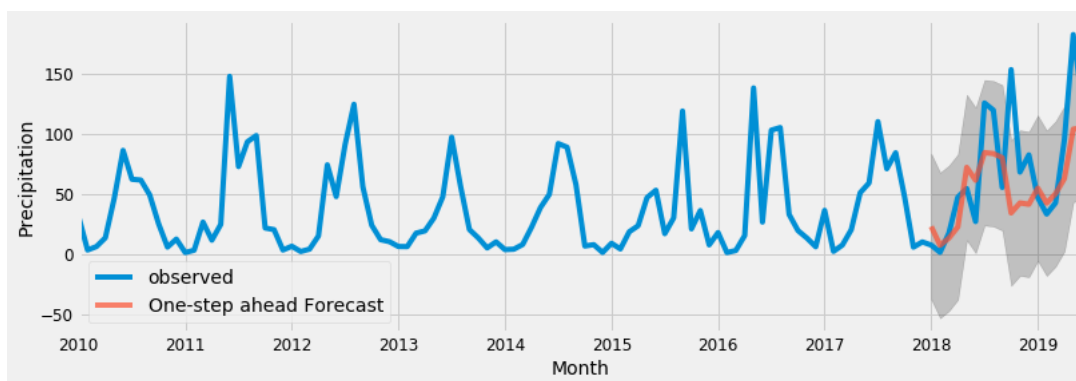
4.2 Dự đoán Lượng mưa

- Số liệu thua thập được



Hình 8: Lượng mưa trung bình tháng ở Hà Nội từ tháng 1 năm 2010 đến tháng 6 năm 2019

- Số liệu dự đoán trên test data



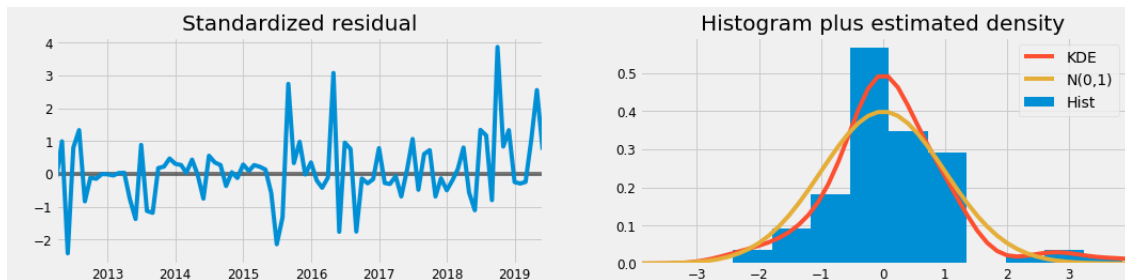
Hình 9: Dự đoán Lượng mưa trung bình tháng ở Hà Nội từ tháng 1 năm 2018 đến tháng 6 năm 2019

- Nhận xét

Quan sát chủ quan: Dữ liệu dự đoán chưa bám sát dữ liệu thực tế. Trong khoảng tháng 8 năm 2018, số liệu dự báo lượng mưa sẽ giảm giống như quy luật các năm trước nhưng thực tế lượng mưa lại đạt đỉnh điểm trong năm 2018.

Kiểm tra bằng hàm MSE, RMSE: Mô hình dự đoán cho ra giá trị $MSE = 1701.63$ và $RMSE = 41.25$ khá lớn. cho thấy dữ liệu dự đoán chưa sát dữ liệu thực tế.

Kiểm tra giả thiết:



Hình 10

Hình vẽ bên trái biểu diễn thành phần Residual của test data sau khi được chuẩn hóa với tổng tất cả các giá trị bằng 0.

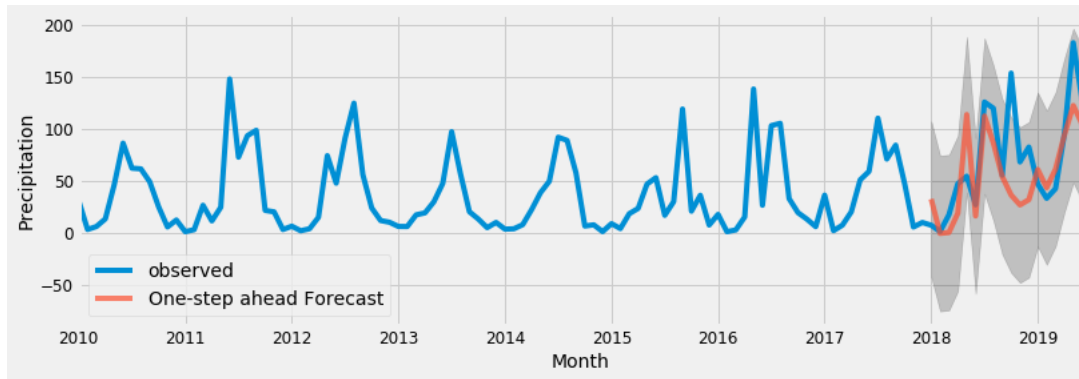
Qua thời gian, các giá trị Residual thể hiện xu hướng lệch lên giá trị lớn khá rõ.

Hình vẽ bên phải biểu diễn phần bố của thành phần Residual theo giá trị tăng dần.

Dựa vào đường biểu đồ histogram, đường cong KDE, đường cong $N(0,1)$, các giá trị Residual khá đối xứng qua giá trị trung bình bằng 0, tuy nhiên số lượng giá trị Residual sát 0 lớn hơn khác biệt so với phân bố tiêu chuẩn $N(0,1)$. Đây có thể là biểu hiện bất thường cho thấy các giá trị Residual trong mô hình này không có tính ngẫu nhiên.

- Phương án cải thiện

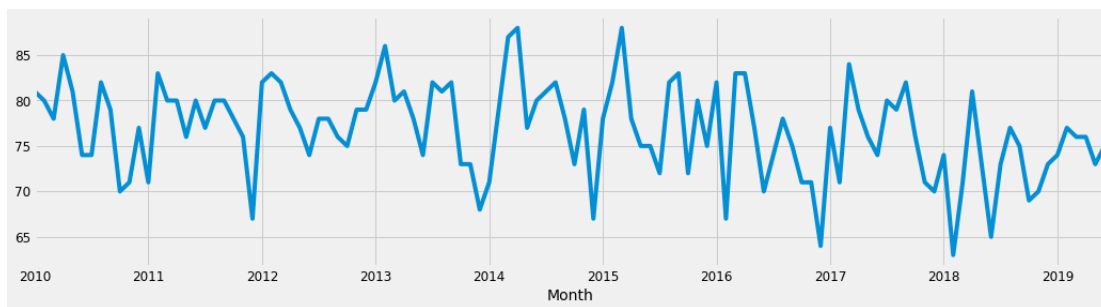
Một phương án cải thiện mô hình có thể xem xét đến là tăng khoảng giá trị cho các tham số p, d, q và P, D, Q , làm tăng số bộ tham số có thể lựa chọn. Phương án này sẽ làm tăng thời gian grid search nhưng có thể giúp chọn được mô hình sát với số liệu thực hơn.



Hình 11: Dự đoán Lượng mưa trung bình tháng ở Hà Nội trên test data khi tăng giá trị của các tham số lên 0, 1, và 2

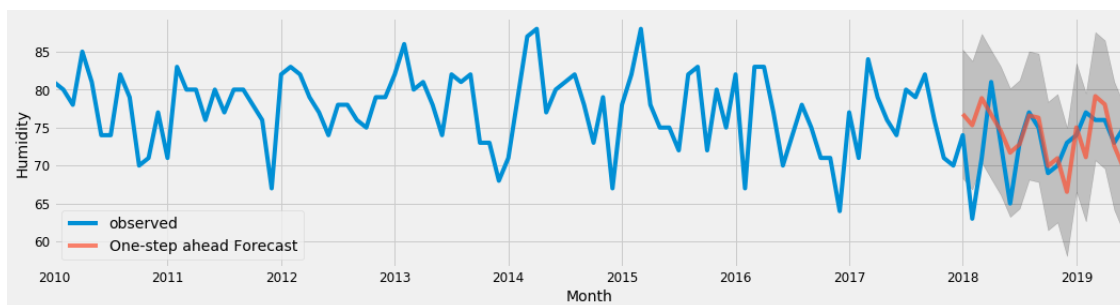
4.3 Dự đoán Độ ẩm

- Số liệu thua thập được



Hình 12: Độ ẩm trung bình tháng ở Hà Nội từ tháng 1 năm 2010 đến tháng 6 năm 2019

- Số liệu dự đoán trên test data



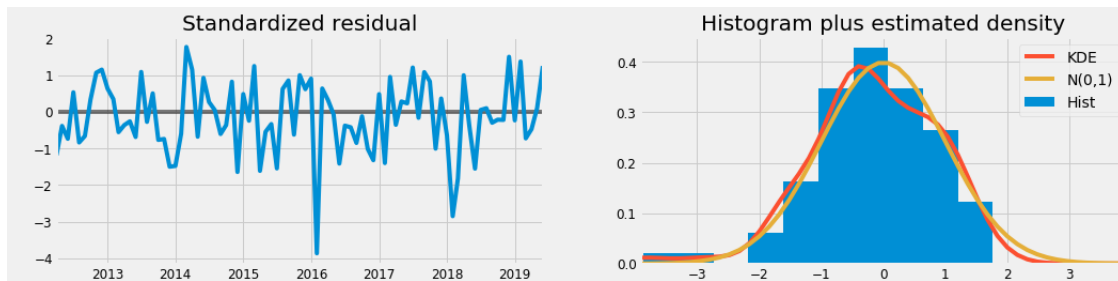
Hình 13: Dự đoán Độ ẩm trung bình tháng ở Hà Nội từ tháng 1 năm 2018 đến tháng 6 năm 2019

- Nhận xét

Quan sát chủ quan: Dữ liệu dự đoán bám khá sát so với giá trị thực tế về chiều hướng thay đổi, nhưng không thực sự sát về mặt giá trị.

Kiểm tra bằng hàm MSE, RMSE: Mô hình dự đoán cho ra giá trị $MSE = 22.87$ và $RMSE = 4.78$, cho thấy dữ liệu dự đoán có thể chưa sát dữ liệu thực tế.

Kiểm tra giả thiết:



Hình 14

Hình vẽ bên trái biểu diễn thành phần Residual của test data sau khi được chuẩn hóa với tổng tất cả các giá trị bằng 0.

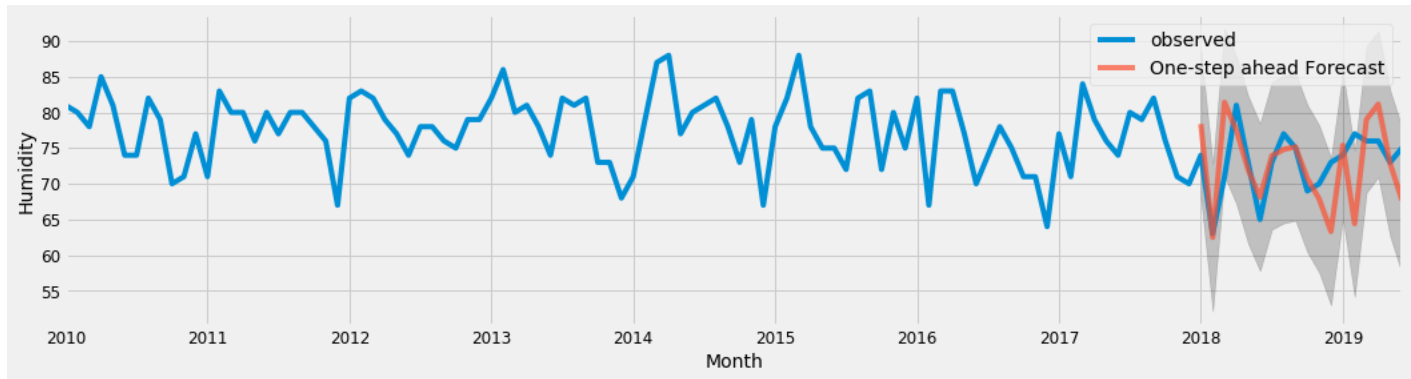
Qua thời gian, các giá trị Residual không thể hiện xu hướng nào rõ ràng.

Hình vẽ bên phải biểu diễn phần bố của thành phần Residual theo giá trị tăng dần.

Dựa vào đường biểu đồ histogram, đường cong KDE, đường cong $N(0,1)$, các giá trị Residual phân bố khá giống với phân bố chuẩn ở những giá trị xa trung bình, nhưng với những giá trị gần 0 lại phân bố khá khác biệt so với phân bố chuẩn. Đây có thể là một biểu hiện bất thường.

- Phương án cải thiện

Một phương án cải thiện mô hình có thể xem xét đến là tăng khoảng giá trị cho các tham số p, d, q và P, D, Q , làm tăng số bộ tham số có thể lựa chọn. Phương án này sẽ làm tăng thời gian grid search nhưng có thể giúp chọn được mô hình sát với số liệu thực hơn.



Hình 15: Dự đoán Độ ẩm trung bình tháng ở Hà Nội trên test data khi tăng giá trị của các tham số lên 0, 1, và 2

5 Kết luận

Đề tài đã tìm hiểu khái niệm và thành phần của dữ liệu time series và quy trình xây dựng mô hình ARIMA cho dữ liệu thời tiết và áp dụng mô hình này vào bài toán thực tế - bài toán dự báo nhiệt độ, lượng mưa, độ ẩm trung bình tháng ở Hà Nội, cũng như đánh giá kết quả của những dự đoán này.

Bên cạnh những kết quả đã đạt được, còn có những vấn đề mà thời điểm này, đề tài chưa giải quyết được như phương án lựa chọn khoảng thích hợp cho tham số, hay thêm những yếu tố bên ngoài có thể gây ảnh hưởng đến các thành phần của dữ liệu chuỗi thời gian.

6 Tài liệu tham khảo

- [1]. Hyndman, Rob J, and George Athanasopoulos. "Forecasting: Principles and Practice." OTexts, Monash University, Australia, <otexts.com/fpp2/>.
- [2]. "Số liệu thống kê, 01. Đơn vị hành chính, đất đai và khí hậu". Tổng cục thống kê, Việt Nam, <www.gso.gov.vn/default.aspx?tabid=713>.