ỨNG DỤNG MÔ HÌNH SARIMA VÀ NGÔN NGỮ LẬP TRÌNH PYTHON ĐỂ DỰ BÁO LƯỢNG MƯA CỦA TỈNH NAM ĐỊNH TRONG NĂM 2024

APPLYING SARIMA MODEL AND PYTHON PROGRAMMING LANGUAGE TO FORECAST RAINFALL IN NAM DINH PROVINCE FOR THE YEAR 2024

Ngô Thị Toán

Khoa Khoa học ứng dụng, Trường Đại học Kinh tế - Kỹ thuật Công nghiệp

Đến Tòa soạn ngày 15/04/2024, chấp nhận đăng ngày 16/05/2024

Tóm tắt: Lượng mưa đóng vai trò rất quan trọng đối với đời sống sản xuất sinh hoạt của con người, đặc biệt với một tỉnh thiên về sản xuất nông ngư nghiệp như tỉnh Nam Định. Nghiên cứu tập trung vào việc vận dụng mô hình dự báo chuỗi thời gian SARIMA (dạng mở rộng của mô hình ARIMA) vào dự báo lượng mưa cho năm 2024 ở tỉnh Nam Định. Đồng thời sử dụng ngôn ngữ lập trình Python hỗ trợ trong quá trình tính toán của dự báo để thu được một mô hình tốt nhất, hiệu quả nhất.

Từ khóa: Chuỗi thời gian, mô hình SARIMA, Ngôn ngữ lập trình Python.

Abstract: Rainfall plays a critical role in the life, livelihoods and production of people, especially in a province focused on agriculture and fisheries like Nam Dinh. The study focuses on applying the SARIMA time series forecasting model to predict rainfall for the year 2024 in Nam Dinh province. Additionally, Python programming language will be utilized to assist in the computational aspects of forecasting, aiming to develop the most effective and efficient forecasting model.

Keywords: Time series, SARIMA model, Python programming language.

1. MỞ ĐẦU

Lượng mưa đóng vai trò quan trọng trong hệ sinh thái và cuộc sống hằng ngày của con người. Nó cung cấp nước cho cây cỏ, hỗ trợ sự phát triển của động vật và thực vật, đồng thời cung cấp nguồn nước cho các hệ thống cung cấp nước và năng lượng thủy điện. Đặc biệt lượng mưa có ảnh hưởng rất lớn đến sản xuất nông lâm ngư nghiệp. Nó quyết định sự phát triển của cây trồng và thú nuôi, tác động lớn đến môi trường sống của các loài thủy hải sản… Vì vậy, biết được lượng mưa, người dân và chính quyền địa phương sẽ có biện pháp tối ưu hóa vai trò của nước mưa và hạn chế được những tác động tiêu cực của nó tới sản xuất và đời sống của người dân. Do đó việc dự đoán lượng mưa theo các tháng có vai trò vô cùng quan trọng. Đặc biệt đối với Nam Định là một tỉnh thiên về sản xuất nông nghiệp và ngư nghiệp nên việc dự đoán được lượng mưa có tác động rất lớn đến việc thúc đẩy phát triển kinh tế ổn định sản xuất cho người dân. Nghiên cứu này tập trung vào việc dự báo lượng mưa của tỉnh Nam Định cho các tháng của các năm 2024 dựa vào số liệu lượng mưa của tỉnh Nam Định trong các tháng từ tháng 1 năm 2018 đến tháng 12 năm 2023, số liệu được lấy từ Trung tâm Khí tượng thủy văn của tỉnh Nam Định cung cấp.

Hiện nay, có rất nhiều mô hình khác nhau để dự báo, mỗi phương pháp đều có ưu nhược điểm riêng. Trong bài báo này, tác giả lựa

chọn mô hình phân tích chuỗi thời gian SARIMA (dạng mở rộng của mô hình ARIMA) để dự đoán lượng mưa của tỉnh Nam Định cho các tháng của năm 2024. Mô hình này được lựa chọn vì khả năng điều chỉnh xu thế cục bộ trong dữ liệu, dữ liệu ở giai đoạn trước có thể dùng để dự đoán cho tương lai. Tác giả cũng sử dụng ngôn ngữ lập trình Python để kiểm tra các bước trong mô hình dự báo của mình.

Bảng 1. Lượng mưa từ tháng 1/2018 đến tháng 12/2023 của tỉnh Nam Định

Year	Month	Luong mua (mm)
2018	1	14,1
2018	2	10,7
2018	3	40,9
2018	4	102
2018	5	146,9
2018	6	86,6
2018	7	510,9
2018	8	375,3
2018	9	189,7
2018	10	225,9
2018	11	6
2018	12	91,1
2019	1	16,9
2019	2	17,5
2019	3	38,6
2019	4	98,7
2019	5	159,9
2019	6	101,3
2019	7	73,5
2019	8	421,2
2019	9	142,8
2019	10	150,8
2019	11	42,6
2019	12	1,2
2020	1	142
2020	2	23,1
2020	3	86,5
2020	4	45,2
2020	5	68,4
2020	6	43,7
2020	7	92,3
2020	8	408,2
2020	9	247,6
2020	10	404,2
2020	11	72,2
2020	12	8,1
2021	1	0,5
2021	2	39,3
2021	3	27,1
2021	4	148,8
2021	5	195,9
2021	6	223,8
2021	7	355,7
2021	8	148,2
2021	9	713,2
2021	10	324,4
2021	11	47,4
2021	12	3
2022	1	88,1
2022	2	101,1
2022	3	62,7
2022	4	102,2
2022	5	172,2
2022	6	136,2
2022	7	329
2022	8	515
2022	9	653
2022	10	285,2
2022	11	87
2022	12	10,7
2023	1	42,0
2023	2	26,9
2023	3	28,2
2023	4	80,7
2023	5	97,9
2023	6	255,6
2023	7	71,5
2023	8	261,3
2023	9	602,3
2023	10	5,9
2023	11	16,4
2023	12	22,7

2. MÔ HÌNH VÀ PHƯƠNG PHÁP NGHIÊN CỨU

2.1. Mô hình nghiên cứu

Mô hình SARIMA là dạng mở rộng của mô hình ARIMA. Mô hình ARIMA lần đầu tiên được đưa ra bởi Box - Jenkins, nó được hình thành bởi sự kết hợp ba thành phần chính: AR (Augmented Reality - tự hồi quy); I (Intergrated - tính dừng của chuỗi thời gian); MA (Moving Average - trung bình trượt). Quá trình tự hồi quy AR: Mô hình tự tương quan bậc p (viết tắt là AR(p)) có dạng như sau:

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + ... + \phi_p Y_{t-p} + \varepsilon_t. \quad (1)$$

trong đó Y_t là quan sát dừng hiện tại, Y_{t-1}; Y_{t-2}; … là các quan sát dừng trong quá khứ, c là hằng số giúp điều chỉnh mức độ trung bình của chuỗi dữ liệu, $\phi_1;\phi_2;....$ là các tham số hồi quy biểu thị mức độ ảnh hưởng của các giá trị quan sát trước đó lên giá trị hiện tại; ε_t là nhiễu trắng (thành phần ngẫu nhiên). Giá trị của p được xác định thông qua biểu đồ ACF và PACF. Nếu chuỗi có dạng AR(p) thì biểu đồ ACF sẽ có các hệ số tự tương quan giảm dần và giảm đột ngột sau giá trị p trong khi đó các giá trị PACF chỉ giảm dần và trở nên không đáng kể sau giá trị p. Điều kiện để quá trình AR(p) có tính dừng là $-1 < \phi_i < 1$ với mọi i = 1; 2; …; p tức là nghiệm của (1) nằm trong vòng tròn đơn vị.

Quá trình trung bình trượt MA: Mô hình trung bình trượt bậc q, viết tắt là MA(q) là quá trình có dạng:

$$Y_t = c + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + ... + \theta_q \varepsilon_{t-q} \quad (2)$$

trong đó Y_t là quan sát hiện tại, $\varepsilon_{t-1};\varepsilon_{t-2};...$ là các nhiễu trong quá khứ, c là hằng số giúp điều chỉnh mức độ trung bình của chuỗi dữ liệu, $\theta_1;\theta_2;...$ là hệ số trọng số của quá trình trung bình trượt biểu thị mức độ ảnh hưởng của các giá trị nhiễu trong quá khứ lên giá trị hiện tại, ε_t là nhiễu trắng (thành phần ngẫu nhiên).

Quá trình trung bình trượt và tự hồi quy ARMA: Khi kết hợp cả hai yếu tố AR và MA ta được quá trình trung bình trượt và tự hồi quy ARMA(p;q). Quá trình này có dạng:

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + ... + \phi_p Y_{t-p} + \theta_0 \varepsilon_t + \\ + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + ... + \theta_q \varepsilon_{t-q}. \quad (3)$$

Quá trình trung bình trượt, tích hợp và tự hồi quy ARIMA: Chuỗi được gọi là tích hợp bậc d, kí hiệu là I(d) nếu sai phân bậc d là chuỗi dừng. Khi d = 0 thì chuỗi xuất phát là chuỗi dừng. Áp dụng mô hình ARMA(p;q) cho chuỗi Y_t tích hợp bậc d ta được mô hình ARIMA(p;d;q) và nó có dạng biểu diễn như sau:

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + ... + \phi_p Y_{t-p} + \\ + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + ... + \theta_q \varepsilon_{t-q} + \varepsilon_t. \quad (4)$$

Quá trình trung bình trượt, tích hợp, tự hồi quy và có yếu tố mùa vụ SARIMA: Ta viết lại phương trinh (4) dưới dạng toán tử trễ như sau:

$$(1 - \phi_1 L - \phi_2 L^2 - ... - \phi_p L^p) Y_t = \\ \phi_0 + (1 + \theta_1 L + \theta_2 L^2 + ... + \theta_q L^q) \varepsilon_t. \\ \Leftrightarrow \phi_p(L) Y_t = \phi_0 + \theta_q(L) \varepsilon_t. \quad (5)$$

Bây giờ ta sẽ mở rộng (5) trong trường hợp có yếu tố mùa vụ. Nếu chỉ có Y_t có yếu tố mùa vụ thì phương trình (5) có dạng:

$$\phi_p(L).(1 - L^s) Y_t = \phi_0 + \theta_q(L) \varepsilon_t. \quad (6)$$

trong đó $(1 - L^s) Y_t$ tạo ra chuỗi đã hiệu chỉnh yếu tố mùa vụ đối với Y_t. Nếu chỉ có ε_t có tính mùa vụ thì phương trình (5) có dạng:

$$\phi_p(L) Y_t = \phi_0 + \theta_q(L).(1 - L^s) \varepsilon_t. \quad (7)$$

trong đó $(1 - L^s) \varepsilon_t$ tạo ra chuỗi đã hiệu chỉnh

yếu tố mùa vụ đối với ε_t. Nếu cả Y_t và ε_t cùng có yếu tố mùa vụ thì phương trình (5) có dạng:

$$\phi_p(L).(1-L^s)Y_t = \phi_0 + \theta_q(L).(1-L^s)\varepsilon_t. \quad (8)$$

Mô hình trong trường hợp này kí hiệu là SARIMA(p;d;q)(P;D;Q)$_s$, trong đó P và Q lần lượt là bậc của thành phần mùa AR và MA; D là bậc của sai phân có tính mùa; s là số thời đoạn trong một chu kỳ. Chu kỳ mùa vụ s = 12 hoặc 4 nếu chuỗi thời gian quan sát theo tháng hoặc quý. Mô hình SARIMA đã thể hiện sự vượt trội của mình về độ chính xác trong việc dự đoán độ trễ tiếp theo của chuỗi thời gian có yếu tố mùa vụ.

2.2. Phương pháp nghiên cứu

Để xây dựng hàm dự báo, nghiên cứu sử dụng số liệu là lượng mưa của tỉnh Nam Định trong các tháng từ tháng 1 năm 2018 đến tháng 12 năm 2023 (bảng 1). Nghiên cứu xây dựng kết quả dự báo thông qua các bước sau:

Bước 1: Nhận dạng mô hình: Xác định các giá trị d; D; p; P; q; Q trong đó trước hết cần xác định bậc sai phân thường d và bậc sai phân theo mùa vụ D và biến đổi chuỗi thành chuỗi dừng. Sau đó kiểm tra biểu đồ của hàm tự tương quan ACF và hàm tự tương quan riêng phần PACF tại các trễ thường và trễ mùa vụ; thực hiện kiểm định nghiệm đơn vị để xác định bậc tự hồi quy p và bậc tự hồi quy mùa vụ P; bậc trung bình trượt q và trung bình trượt mùa vụ Q.

*Bước 2: Ước lượng mô hình: Ướ*c lượng các tham số của mô hình.

Bước 3: Kiểm định: Kiểm định tính hợp lý của mô hình SARIMA được lựa chọn. Nếu kiểm định mô hình được chọn không thỏa mãn thì quay lại bước 1.

Bước 4: Dự báo: Dựa trên mô hình đã lựa chọn tiến hành dự báo cho giá trị tương lai của chuỗi mùa vụ.

Trong nghiên cứu này, tác giả sử dụng sự hỗ trợ của ngôn ngữ lập trình Python trong các bước dự đoán.

3. KẾT QUẢ VÀ THẢO LUẬN

Nghiên cứu đã lấy số liệu là lượng mưa của tỉnh Nam Định trong các tháng từ tháng 1 năm 2018 đến tháng 12 năm 2023.

Ở bước 1: Trước hết cần kiểm tra tính mùa vụ của chuỗi dữ liệu. Nếu chuỗi dữ liệu có tính mùa vụ thì sẽ dùng thuật toán SARIMA(p;d;q)(P;D;Q)$_s$; còn nếu chuỗi không có tính mùa vụ thì lựa chọn mô hình ARIMA(p;d;q). Bằng ngôn ngữ lập trình Python, tác giả kiểm tra tính mùa vụ của chuỗi dữ liệu bằng kiểm định ADF như sau:

```
import pandas as pd
from statsmodels.tsa.stattools import adfuller
# Đọc dữ liệu từ file Excel
df = pd.read_excel("C:¥¥Users¥¥Administrator¥¥Desktop¥¥luongmua_ND.xls")
# Kiểm định ADF kiểm tra tính mùa vụ
result = adfuller(df['Luong mua (mm)'])
# Trích xuất và in kết quả
print('ADF Statistic:', result[0])
print('p-value:', result[1])
print('Critical Values:')
for key, value in result[4].items():
print('¥t%s: %.3f' % (key, value))
```

Kết quả thu được như sau: ADF Statistic: -1.1475629942579546;p-value: 0.69573668469; Critical Values: 1%: -3.5443688564; 5%: -2.91107314814; 10%: -2.59319027777.

Với kết quả này thì dữ liệu lượng mưa tại tỉnh Nam Định có yếu tố mùa vụ nên tác giả sẽ dùng mô hình SARIMA(p;d;q)(P;D;Q)$_s$. Để xác định bậc sai phân d, trước hết cần kiểm tra tính dừng của chuỗi dữ liệu.

```
import pandas as pd
from statsmodels.tsa.stattools import adfuller
# Đọc dữ liệu từ file Excel
df = pd.read_excel("C:¥¥Users¥¥Administrator¥¥Desktop¥¥luongmua_ND.xls")
# Chọn cột chứa dữ liệu lượng mưa
rainfall_data = df['Luong mua (mm)']
# Kiểm tra tính dừng của chuỗi dữ liệu
result = adfuller(rainfall_data)
# Trích xuất kết quả và in ra màn hình
print('ADF Statistic:', result[0])
print('p-value:', result[1])
print('Critical Values:')
for key, value in result[4].items():
print('¥t%s: %.3f' % (key, value))
```

Kết quả thu được: ADF Statistic: -1.243058763379;p-value: 0.65480423421; Critical Values: 1%: -3.575; 5%: -2.924; 10%: -2.600. Chuỗi dữ liệu không phải là chuỗi dừng với mức ý nghĩa 0,05. Lấy sai phân bậc 1 và kiểm tra tính dừng của chuỗi sau khi lấy sai phân bậc 1.

```
import pandas as pd
from statsmodels.tsa.stattools import adfuller
df = pd.read_excel("C:¥¥Users¥¥Administrator¥¥Desktop¥¥luongmua_ND.xls")
# Lấy sai phân bậc 1 của chuỗi dữ liệu 'Luong mua (mm)'
df_diff = df['Luong mua (mm)'].diff().dropna()
# Tạo DataFrame từ chuỗi dữ liệu sau khi lấy sai phân
df_diff_df = pd.DataFrame(df_diff, columns=['Luong mua (mm)'])
# Lưu chuỗi sau khi lấy sai phân vào file Excel
df_diff_df.to_excel("C:¥¥Users¥¥Administrator¥¥Desktop¥¥luongmua_diff.xls", index=False)
# Kiểm định ADF cho chuỗi sau khi lấy sai phân bậc 1
result = adfuller(df_diff)
# Trích xuất và in kết quả kiểm định ADF
print('ADF Statistic:', result[0])
print('p-value:', result[1])
print('Critical Values:')
for key, value in result[4].items():
print('¥t%s: %.3f' % (key, value))
```

Kết quả thu được: ADF Statistic: -7.17913833884; p-value: 2.678231170564385e-10; Critical Values: 1%: -3.575; 5%: -2.924; 10%: -2.600. Nên chuỗi sau khi lấy sai phân là dừng. Do đó d =1.

Ở bước 2 và bước 3: Vẽ biểu đồ ACF và PACF để xác định khoảng giá trị của p và q.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.stattools import acf, pacf
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
df = pd.read_excel("C:¥¥Users¥¥Administrator¥¥Desktop¥¥luongmua_diff.xls")
# Xác định lại cột 'Month' là số nguyên (int) nếu cần thiết
df['Month'] = df['Month'].astype(int)
```

```
# Sử dụng dữ liệu trong cột 'Luong mua (mm)' cho tính toán ACF và PACF

luong_mua = df['Luong mua (mm)']

# Tạo đồ thị ACF và PACF của chuỗi lượng mưa

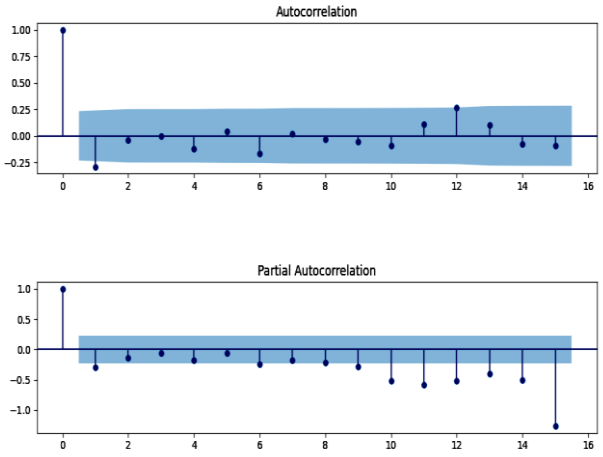
fig, ax = plt.subplots(2, 1, figsize=(10, 8))

plot_acf(luong_mua, lags=15, ax=ax[0]) # Đồ thị ACF

plot_pacf(luong_mua, lags=15, ax=ax[1]) # Đồ thị PACF

plt.show()
```

Kết quả thu được:



Hình 1. Biểu đồ ACF và PACF

Dựa vào biểu đồ, ta lựa chọn bậc p và q nhỏ hơn hoặc bằng 5. Chuỗi mùa vụ s =12, bậc sai phân là d = 1. Sử dụng chiến lược stepwise (Chiến lược này gồm nhiều bước, mỗi bước tìm cách thêm hoặc bớt đi một biến giải thích nhằm tạo ra một mô hình mới sao cho sai số giảm so với mô hình gốc và thu được mô hình tốt nhất). Sử dụng ngôn ngữ lập trình Python cho chiến lược stepwise như sau để xác định một số mô hình phù hợp:

```
import pandas as pd

df = pd.read_excel("C:¥¥Users¥¥Administrator¥¥Desktop¥¥luongmua_ND.xls")

# Chọn cột 'luong mua (mm)' làm chuỗi dữ liệu thời gian

mua_seri = df['Luong mua (mm)']

import pmdarima as pm

# Sử dụng chiến lược stepwise để tự động tìm kiếm mô hình SARIMA tốt nhất

model = pm.auto_arima(mua_seri, start_p=0, start_q=0,

max_p=5, max_q=5, m=12,  # Thử các giá trị p, q từ 1 đến 5

start_P=0, seasonal=True,

d=1, D=1, trace=True,

error_action='ignore',  # Bỏ qua các lỗi nếu có suppress_warnings=True,  # Tắt cảnh báo stepwise=True) # Sử dụng chiến lược stepwise

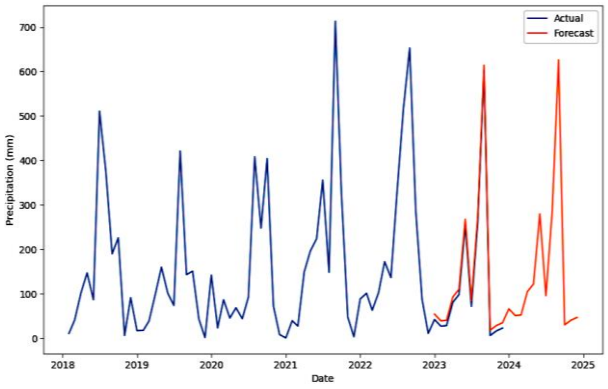
# Hiển thị thông tin về mô hình tốt nhất

print(model.summary())
```

Kết quả thu được một số dạng mô hình SARIMA như sau: ARIMA$(0,1,0)(0,1,1)_{12}$; ARIMA$(0,1,0)(0,1,0)_{12}$; ARIMA$(1,1,0)(1,1,0)_{12}$; ARIMA$(0,1,1)(0,1,1)_{12}$; ARIMA$(0,1,1)(0,1,0)_{12}$; ARIMA$(0,1,1)(1,1,1)_{12}$; ARIMA$(0,1,1)(0,1,2)_{12}$; ARIMA$(0,1,1)(1,1,0)_{12}$; ARIMA$(0,1,1)(1,1,2)_{12}$; ARIMA$(1,1,1)(0,1,1)_{12}$; ARIMA$(0,1,2)(0,1,1)_{12}$; ARIMA$(1,1,0)(0,1,1)_{12}$; ARIMA$(1,1,2)(0,1,1)_{12}$.

Thử nghiệm và so sánh các mô hình trên, lựa chọn mô hình tốt nhất là SARIMA$(0,1,0)(0,1,0)_{12}$. Mô hình cho kết quả khá tốt khi các hệ số hồi quy đều có ý nghĩa thống kê. Đồng thời các chỉ số RMSE

(biên độ giao động của giá trị dự báo xung quanh giá trị thực tế) và MAE (trung bình sai số giữa giá trị thực tế và giá trị dự báo) đều tương đối thấp (đều bằng 12.00). Thêm nữa việc trực quan biểu đồ so sánh lượng mưa thực tế và lượng mưa dự báo càng khẳng định tính tối ưu của mô hình.



Hình 2. Biểu đồ lượng mưa thực tế và dự đoán

Theo kết quả ở hình 3 thì các giá trị dự báo và giá trị thực tế chênh lệch nhau khá ít.

Ở bước 4: Sử dụng mô hình SARIMA$(0,1,0)(0,1,0)_{12}$ để dự báo lượng mưa của tỉnh Nam Định cho năm 2024.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.statespace.sarimax import SARIMAX

# Đọc dữ liệu từ file Excel

cols = ['YEAR', 'MONTH', 'Luong mua (mm)']
df = pd.read_excel("C:¥¥Users¥¥Administrator¥¥Desktop¥¥luongmua_ND.xls",names = cols,header=1)

#Kết hợp cột Year và Month thành cột Date làm index

df['Date'] = pd.to_datetime(df[['YEAR', 'MONTH']].assign(day=1))

df.set_index('Date', inplace=True)

# Lấy chuỗi dữ liệu 'luong mua (mm)'

mua_seri = df['Luong mua (mm)']

# Huấn luyện mô hình SARIMA(0,101)(0,1,0)_12

model = SARIMAX(mua_seri, order=(0, 1, 0), seasonal_order=(0, 1, 0, 12))

results = model.fit()

print(results.summary())

# Dự báo lượng mưa cho năm 2023 và 2024

forecast_start = pd.to_datetime('2023-01-01')

forecast_end = pd.to_datetime('2024-12-01')

forecast = results.get_forecast(steps=len(mua_seri) + (forecast_end - forecast_start).days // 30)

# Lấy kết quả dự báo

forecast_values = forecast.predicted_mean

forecast_index = pd.date_range(start=forecast_start, end=forecast_end, freq='MS')

# Hiển thị kết quả dự báo

plt.figure(figsize=(10, 6))

plt.plot(mua_seri.index, mua_seri, label='Actual')

plt.plot(forecast_index, forecast_values[:len(forecast_index)], label='Forecast')  # Đảm bảo số lượng điểm dữ liệu dự báo khớp với số lượng thời điểm trong forecast_index

plt.title('SARIMA Forecast for Precipitation')

plt.xlabel('Date')

plt.ylabel('Precipitation (mm)')

plt.legend()

plt.show()

# In kết quả dự báo

print("Forecasted Precipitation for 2023 and 2024:")

print(pd.DataFrame({'Date': forecast_index, 'Forecast':
```

forecast_values[:len(forecast_index)]}).set_index('Date'))

Kết quả dự báo lượng mưa cho năm 2024 như sau:

2024-01-01	66.0	2024-05-01	121.9	2024-09-01	626.3
2024-02-01	50.9	2024-06-01	279.6	2024-10-01	29.9
2024-03-01	52.2	2024-07-01	95.5	2024-11-01	40.4
2024-04-01	104.7	2024-08-01	285.3	2024-12-01	46.7

Hình 3. Kết quả lượng mưa dự báo năm 2024

Với kết quả dự báo này, trong năm 2024 tháng có lượng mưa tương đối lớn là tháng 6, tháng 8 và tháng 9 (đặc biệt tháng 9). Người dân và chính quyền cần chú trọng khơi thông dòng chảy, tích trữ nước cho sản xuất và sinh hoạt để phục vụ cho các tháng có lượng mưa ít hơn là tháng 10, 11, 12.

4. KẾT LUẬN

- Như vậy, bài báo đã dùng ngôn ngữ lập trình Python trong các bước dự báo và chỉ ra mô hình SARIMA$(0,1,0)(0,1,0)_{12}$ là phù hợp nhất cho việc dự báo lượng mưa của tỉnh Nam Định năm 2024.
- Kết quả dự báo có thể làm số liệu tham khảo phục vụ cho việc dự báo ở các trung tâm khí tượng thủy văn tỉnh Nam Định cũng như để định hướng quy hoạch sản suất của người dân Nam Định năm 2024.
- Kết quả nghiên cứu có thể ứng dụng cho một số lĩnh vực dự báo khác tương tự như dự báo nhiệt độ, độ ẩm, dự báo kinh tế…

TÀI LIỆU THAM KHẢO

[1] Nguyễn Văn Huân, Phạm Viết Bình, Giáo trình Phân tích và dự báo trong kinh tế, trang 98-112, Nhà xuất bản Khoa học và Kỹ thuật, lần 1, năm 2011.

[2] ThS. Bùi Thị Minh Nguyệt, ThS. Nguyễn Thị Quỳnh Nga, ThS. Nguyễn Thị Quỳnh Châm, Sử dụng mô hình ARIMA trong dự báo giá trị xuất khẩu của Việt Nam, Tạp chí Nghiên cứu tài chính kế toán, tập 186, số 01, trang 58-62 (2019).

[3] Nguyễn Thị Hồng Thảo, Đào Minh Châu, Vũ Thanh Nguyên, Phù Đức Huy, Áp dụng một vài phương pháp máy học vào bài toán dự báo theo chuỗi thời gian, Tạp chí Khoa học Trường Đại học Sư phạm Thành phố Hồ Chí Minh, tập 19, số 12, trang 2064-2075 (2022).

Thông tin liên hệ: **Ngô Thị Toán**

Điện thoại: 0978 392 745 - Email: nttoan@uneti.edu.vn

Khoa Khoa học ứng dụng, Trường Đại học Kinh tế - Kỹ thuật Công nghiệp.

-
-