ĐỒ ÁN GIỮA KỲ NĂM HỌC 2025-2026

ứng dụng PHÂN TÍCH DỮ LIỆU đánh giá Ô NHIỄM TP.HCM

GVHD: Trương Quang Nhật
Mã lớp HP: 251BMA202503

Nhóm thực hiện **FIREFLY**

Nhóm sinh viên thực hiện

MSSV	Họ và tên	Chữ ký
K244161769	Hồng Trang Anh	
K244161770	Nguyễn Vũ Tường Anh	
K244161776	Nguyễn Quỳnh Duyên	
K244161778	Nguyễn Trần Yên Đan	
K244161798	Đinh Hải Ngân	

Mục lục

LỜI CẢM ƠN 4

DANH MỤC HÌNH ẢNH 5

DANH MỤC BẢNG BIỂU 6

CHƯƠNG 1 7-21

CHƯƠNG 2 22-32

CHƯƠNG 3 33-81

CHƯƠNG 4 82-119

CHƯƠNG 5 120 - 133

TÀI LIỆU THAM KHẢO 134

Lời cảm ơn

Lời đầu tiên, chúng em xin chân thành cảm ơn thầy ***Trương Quang Nhật***, giảng viên môn *Phân tích dữ liệu*, đã tận tình giảng dạy và hướng dẫn chúng em trong suốt học kỳ vừa qua. Qua môn học này, chúng em không chỉ được củng cố kiến thức chuyên ngành mà còn rèn luyện tư duy phân tích và kỹ năng thực hành dữ liệu – những hành trang quý báu cho con đường học tập và nghề nghiệp sau này.

Trong quá trình thực hiện đồ án nhóm này, do kiến thức và kinh nghiệm còn hạn chế, bài làm của chúng em khó tránh khỏi những thiếu sót. Kính mong thầy xem xét và góp ý để chúng em có thể hoàn thiện hơn trong tương lai.

Cuối cùng, chúng em xin kính chúc thầy luôn mạnh khỏe, hạnh phúc và thành công trong sự nghiệp giảng dạy.

Danh mục hình ảnh

Hình 3.1.	Các cột đánh dấu các outlier được tìm bằng phương pháp IQR	52
Hình 3.2.	Cột iso_flag trong dataframe	54
Hình 3.3.	Cột is_outlier trong dataframe	55
Hình 3.4.	So sánh Boxplot các chỉ tiêu môi trường tại trạm 6 trước và sau khi xử lý ngoại lệ	56
Hình 3.5.	Thống kê mô tả sau khi chuẩn hóa	59
Hình 3.6.	Ma trận tương quan Pearson giữa các biến đo ô nhiễm và khí tượng	71
Hình 3.7.	Ma trận tương quan Spearman giữa các biến đo ô nhiễm và khí tượng	72
Hình 3.8.	Các cặp biến có tương quan tuyệt đối cao ($\|r\| \geqslant 0,8$)	74
Hình 3.9.	Tương quan Pearson sau khi loại bỏ mùa vụ và xu hướng thời gian.	76
Hình 3.10.	Tương quan Spearman sau khi loại bỏ mùa vụ và xu hướng thời gian.	76
Hình 3.11.	So sánh tương quan trước và sau khi loại bỏ mùa vụ / xu hướng.	77
Hình 3.12.	Bảng kiểm tra đa cộng tuyến: VIF theo biến và Condition Index toàn hệ	80

Danh mục bảng biểu

Bảng 3.1. Bảng các giá trị của BPi đối với các thông số trong Quyết định 1459/QĐ-TCMT 64

Biểu đồ 4.1. Phân bổ giá trị AQI hàng ngày theo tháng 85

Biểu đồ 4.2. Mức AQI trung bình 6 tháng đầu năm giữa năm 2021 và 2022 89

Biểu đồ 4.3. Tỷ lệ đóng góp trung bình của các chất ô nhiễm 93

Biểu đồ 4.4. Mức độ chất ô nhiễm theo mùa 97

Biểu đồ 4.5. Xu hướng AQI trung bình theo tháng 100

Biểu đồ 4.6. So sánh mức độ ô nhiễm không khí giữa 2 mùa 104

Biểu đồ 4.7. Phân bố & xu hướng AQI giữa 2 mùa theo quý 107

Biểu đồ 4.8. Tương quan giữa nhiệt độ và AQI 113

Biểu đồ 4.9. Tương quan độ ẩm và AQI 117

Chương 1: Giới thiệu

1.1. Bối cảnh nghiên cứu

1.2. Mục tiêu nghiên cứu

1.3. Công cụ sử dụng

1.4. Phương pháp nghiên cứu

1.1. Bối cảnh nghiên cứu

Ô nhiễm không khí hiện được xem là một trong những thách thức nghiêm trọng nhất đối với sức khỏe cộng đồng và sự phát triển bền vững toàn cầu. Theo Tổ chức Y tế Thế giới (WHO), hơn 99% dân số thế giới đang phải hít thở bầu không khí có nồng độ bụi và bụi mịn ($PM_{2.5}$, PM_{10}) vượt ngưỡng khuyến nghị, gây ra khoảng 6,7 triệu ca tử vong sớm mỗi năm [1]. Tại Việt Nam, tình trạng này ngày càng đáng lo ngại khi các đô thị lớn như Hà Nội và Thành phố Hồ Chí Minh thường xuyên ghi nhận chỉ số chất lượng không khí (AQI) ở mức kém, đặc biệt trong mùa khô và các thời điểm giao thông cao điểm [2].

Cùng với xu hướng gia tăng mức độ ô nhiễm là sự mở rộng nhanh chóng của các hoạt động quan trắc môi trường. Tuy nhiên, dữ liệu thu thập trong lĩnh vực này vẫn còn nhiều hạn chế. Mạng lưới trạm quan trắc cố định hiện nay chưa đạt mật độ đủ dày để phản ánh chính xác sự biến động không gian của ô nhiễm ở cấp độ đô thị vi mô [3]. Trong khi đó, dữ liệu từ các cảm biến di động hoặc nguồn cộng đồng tuy giúp mở rộng phạm vi thu thập, nhưng lại dễ gặp phải những vấn đề như độ ổn định thấp, nhiễu phép đo hoặc thiếu thống nhất về định dạng và tiêu chuẩn. Sự đa dạng, khối lượng lớn và mức độ phức tạp của dữ liệu đã khiến quá trình khai thác, xử lý và phân tích trở thành một thách thức đáng kể đối với các nhà nghiên cứu và quản lý môi trường.

1.1. Bối cảnh nghiên cứu

Trong bối cảnh đó, phân tích và trực quan dữ liệu đóng vai trò trung tâm, là cầu nối giữa dữ liệu thô và tri thức khoa học có giá trị ứng dụng. Thông qua các quy trình xử lý và chuẩn hóa, PTDL không chỉ khắc phục những sai lệch kỹ thuật, mà còn tạo điều kiện cho việc khám phá các quy luật tiềm ẩn trong biến động của các chất ô nhiễm. Việc phân tích mối tương quan giữa các yếu tố khí tượng như nhiệt độ, độ ẩm, tốc độ gió và nồng độ các chất như $PM_{2.5}$, NO_2 hay O_3 cho phép làm sáng tỏ cơ chế phát tán và tích tụ ô nhiễm trong điều kiện khí hậu cụ thể. Đồng thời, trực quan hóa dữ liệu đóng vai trò quan trọng trong việc chuyển tải kết quả phân tích một cách trực quan, giúp các nhà hoạch định chính sách, chuyên gia và cộng đồng dễ dàng tiếp cận và hiểu rõ hơn về hiện trạng môi trường.

Sự phát triển của công nghệ cảm biến giá rẻ, mạng IoT và các nền tảng dữ liệu mở đã làm thay đổi căn bản phương thức tiếp cận dữ liệu môi trường. Lượng dữ liệu lớn (big data) thu thập từ nhiều nguồn khác nhau đòi hỏi các phương pháp xử lý tiên tiến cùng khả năng phân tích linh hoạt. Trong bối cảnh này, ngôn ngữ lập trình Python và các thư viện chuyên biệt như Pandas, Scikit-learn, Seaborn và Matplotlib đã trở thành những công cụ quan trọng, cho phép triển khai toàn bộ quy trình xử lý, khai phá, mô hình hóa và trực quan hóa dữ liệu một cách hiệu quả. Việc ứng dụng phân tích dữ liệu trong nghiên cứu ô nhiễm không khí không chỉ nâng cao độ tin cậy và tính chính xác của kết quả mà còn mở rộng năng lực dự báo, đánh giá và hỗ trợ ra quyết định trong quản lý môi trường đô thị.

1.1. Bối cảnh nghiên cứu

Như vậy, phân tích và trực quan dữ liệu không chỉ đóng vai trò như một công cụ hỗ trợ kỹ thuật, mà đã trở thành nền tảng cốt lõi trong việc chuyển hóa dữ liệu thành tri thức và tri thức thành hành động. PTDL góp phần hình thành một hệ thống tri thức toàn diện về chất lượng không khí, từ đó cung cấp cơ sở khoa học cho việc hoạch định chính sách, phát triển đô thị bền vững và bảo vệ sức khỏe cộng đồng.

1.2. Mục tiêu nghiên cứu

Trên cơ sở ứng dụng các kỹ thuật phân tích dữ liệu vào xử lý và diễn giải thông tin quan trắc môi trường, nghiên cứu hướng đến việc đánh giá mức độ ô nhiễm không khí và xu hướng biến động chất lượng không khí tại TP. Hồ Chí Minh, qua đó đề xuất giải pháp quản lý và cải thiện. Cụ thể, nghiên cứu tập trung vào các mục tiêu sau:

1. Xác định mức độ và đặc điểm ô nhiễm không khí tại các khu vực khác nhau của Thành phố Hồ Chí Minh thông qua các chỉ số quan trắc như TSP, PM2.5, O_3, CO, NO_2, SO_2, nhiệt độ và độ ẩm.

2. Đánh giá chất lượng không khí tổng hợp bằng Chỉ số Chất lượng Không khí (AQI) để phản ánh tình trạng ô nhiễm trong từng giai đoạn và khu vực.

3. Phân tích xu hướng biến động của AQI theo thời gian và điều kiện khí tượng, nhằm nhận diện các mô hình thay đổi và yếu tố chi phối chính đến chất lượng không khí.

4. Đề xuất hàm ý quản lý và giải pháp cải thiện chất lượng không khí, góp phần hỗ trợ các cơ quan chức năng trong việc giám sát và hoạch định chính sách môi trường đô thị.

1.2. Mục tiêu nghiên cứu

Tóm lại, thông qua việc xác định rõ các mục tiêu trên, nghiên cứu kỳ vọng sẽ làm sáng tỏ hiện trạng và xu hướng biến động chất lượng không khí tại Thành phố Hồ Chí Minh trong giai đoạn khảo sát. Kết quả phân tích không chỉ cung cấp cái nhìn định lượng và toàn diện về mức độ ô nhiễm không khí, mà còn góp phần đề xuất những định hướng chính sách và giải pháp thực tiễn nhằm nâng cao hiệu quả quản lý môi trường đô thị. Qua đó, nghiên cứu khẳng định vai trò thiết yếu của phân tích dữ liệu như một công cụ hỗ trợ ra quyết định trong lĩnh vực bảo vệ môi trường và phát triển bền vững.

1.3. Công cụ sử dụng

Trong quá trình thực hiện, nhóm lựa chọn ngôn ngữ lập trình Python làm công cụ chính để xử lý và phân tích dữ liệu. Python được xem là một trong những ngôn ngữ phổ biến và mạnh mẽ nhất trong lĩnh vực khoa học dữ liệu nhờ cú pháp đơn giản, dễ đọc, cùng hệ sinh thái thư viện phong phú, hỗ trợ đa dạng các tác vụ từ tiền xử lý dữ liệu, thống kê mô tả đến xây dựng mô hình học máy.

Cụ thể, nhóm sử dụng một số thư viện sau:

- **Pandas**: Đây là thư viện nền tảng trong Python dùng để đọc, xử lý, và thao tác dữ liệu ở dạng bảng (DataFrame). Pandas cung cấp nhiều công cụ mạnh mẽ giúp kiểm tra dữ liệu rỗng, trùng lặp, chuyển đổi định dạng, tính toán thống kê mô tả, cũng như hỗ trợ kết hợp và trích xuất thông tin từ tập dữ liệu lớn một cách hiệu quả.

- **NumPy**: Là thư viện cơ bản trong Python được sử dụng để xử lý dữ liệu dạng mảng và thực hiện các phép tính toán số học có hiệu suất cao. Thư viện này cung cấp nhiều hàm toán học tối ưu, giúp giảm thời gian xử lý và tăng độ chính xác trong tính toán. Trong đề tài, NumPy đóng vai trò nền tảng cho việc chuẩn hóa, biến đổi và tính toán dữ liệu phục vụ cho các bước phân tích tiếp theo.

1.3. Công cụ sử dụng

- **SciPy**: Là thư viện mở rộng của NumPy, chuyên cung cấp các công cụ và hàm phục vụ cho phân tích khoa học và thống kê. Với các module hỗ trợ xử lý tín hiệu, tối ưu hóa, kiểm định giả thuyết, và phân phối xác suất, SciPy được sử dụng trong nghiên cứu này để thực hiện các phép kiểm định thống kê và phân tích chuyên sâu, đảm bảo độ tin cậy của các kết quả suy luận.

- **OpenPyXL**: là thư viện Python cho phép đọc, ghi và chỉnh sửa các tệp dữ liệu ở định dạng Excel (.xlsx). Trong đồ án, thư viện được sử dụng để xuất kết quả xử lý và mô hình hóa ra tệp Excel, giúp lưu trữ, tổng hợp và chia sẻ dữ liệu một cách thuận tiện và có hệ thống.

- **IPython.display**: Là thư viện cung cấp các công cụ hiển thị trực quan trong môi trường notebook, cho phép chèn và trình bày nội dung đa phương tiện như hình ảnh, bảng, hoặc biểu đồ một cách linh hoạt. Trong đồ án này, IPython.display được sử dụng để hỗ trợ việc trình bày kết quả trực quan, giúp quá trình quan sát và phân tích dữ liệu trở nên trực quan và sinh động hơn.

1.3. Công cụ sử dụng

- **Scikit-learn**: Là thư viện học máy nổi tiếng, hỗ trợ nhiều thuật toán phân tích và mô hình hóa dữ liệu. Trong đề tài này, nhóm đặc biệt sử dụng mô hình Isolation Forest – một thuật toán chuyên dụng trong việc phát hiện ngoại lệ (outliers) và giá trị bất thường (anomalies) trong tập dữ liệu. Scikit-learn cung cấp các công cụ giúp huấn luyện, đánh giá và trực quan hóa kết quả của mô hình, đảm bảo tính chính xác và khả năng tái lập trong quá trình nghiên cứu.

- **Seaborn và Matplotlib**: Hai thư viện này được sử dụng nhằm mục đích trực quan hóa dữ liệu. Matplotlib cho phép tạo ra nhiều loại biểu đồ khác nhau (biểu đồ cột, đường, phân tán, hộp, v.v.) giúp thể hiện xu hướng và mối quan hệ giữa các biến. Trong khi đó, Seaborn được xây dựng trên nền Matplotlib nhưng hỗ trợ trực quan hóa dữ liệu ở mức cao hơn, với giao diện thân thiện và phong cách hiển thị trực quan, dễ quan sát. Nhờ đó, nhóm có thể biểu diễn dữ liệu thông qua biểu đồ hộp (boxplot) và các dạng biểu đồ so sánh khác, giúp phát hiện nhanh các mẫu hình và bất thường trong tập dữ liệu.

1.3. Công cụ sử dụng

Nhờ sự kết hợp giữa Python và các thư viện mạnh mẽ trên, quá trình nghiên cứu đạt được hiệu quả cao trong cả giai đoạn tiền xử lý, phân tích, mô hình hóa và trình bày kết quả, đảm bảo tính khoa học, minh bạch và dễ tái lập.

Các đoạn mã và quy trình xử lý được thực thi trên Google Colab, một môi trường tính toán đám mây do Google cung cấp. Việc sử dụng Google Colab mang lại nhiều ưu điểm vượt trội như: không cần cài đặt phần mềm phức tạp, có thể chạy mã nguồn Python trực tiếp trên trình duyệt, đồng thời cho phép kết hợp giữa mã lệnh (code), kết quả đầu ra, và biểu đồ trực quan trong cùng một giao diện. Ngoài ra, Colab hỗ trợ thao tác với tệp dữ liệu từ Google Drive, giúp nhóm dễ dàng chia sẻ, cộng tác và theo dõi tiến độ nghiên cứu một cách linh hoạt và thuận tiện.

1.4. Phương pháp nghiên cứu

QUY TRÌNH

XỬ LÝ VÀ PHÂN TÍCH DỮ LIỆU

1.4. Phương pháp nghiên cứu

Phương pháp nghiên cứu được xây dựng dựa trên quy trình xử lý và phân tích dữ liệu định lượng, kết hợp thống kê mô tả suy diễn, gồm sáu giai đoạn: **(1) Thu thập và khảo sát dữ liệu, (2) Tiền xử lý dữ liệu, (3) Phát hiện và xử lý ngoại lệ, (4) Chuẩn hóa biến định lượng, (5) Tạo biến mới và tính toán chỉ số, (6) Trực quan hóa dữ liệu.** Quy trình này nhằm đảm bảo dữ liệu được làm sạch, chuẩn hóa và diễn giải một cách khoa học, khách quan, đồng thời có tính ứng dụng thực tiễn trong đánh giá chất lượng môi trường không khí.

(1) Thu thập và khảo sát dữ liệu:

Dữ liệu gốc được thu thập từ các tệp Excel và nhập vào môi trường Python để tiến hành khảo sát sơ bộ. Ở giai đoạn này, tập dữ liệu được kiểm tra về kích thước, cấu trúc và danh mục biến, đồng thời phân loại thành ba nhóm chính: biến môi trường, biến khí tượng và biến thời gian. Các thống kê mô tả cơ bản được thực hiện nhằm đánh giá tính đầy đủ, độ hợp lệ và đặc điểm phân bố của dữ liệu, từ đó xác định hướng xử lý phù hợp cho các bước tiếp theo.

1.4. Phương pháp nghiên cứu

2) Tiền xử lý dữ liệu:

Đây là giai đoạn trọng tâm trong toàn bộ quy trình. Dữ liệu được chuẩn hóa về định dạng tên biến (chuyển sang chữ thường, không dấu, dùng dấu gạch dưới thay cho khoảng trắng) và chuẩn hóa thời gian về dạng dd-mm-yyyy hh:mm. Các giá trị khuyết được nhận diện và thay thế bằng giá trị trung bình tại cùng thời điểm của các trạm khác, trong khi các bản ghi trùng lặp được loại bỏ nhằm đảm bảo tính chính xác của tập dữ liệu.

(3) Phát hiện và xử lý ngoại lệ:

Hai phương pháp được áp dụng để phát hiện và xử lý ngoại lệ là:

- IQR (Interquartile Range): xác định các giá trị vượt ngưỡng thống kê thông thường.
- Isolation Forest: thuật toán học máy không giám sát giúp phát hiện điểm dữ liệu bất thường tiềm ẩn.

Việc kết hợp hai phương pháp này giúp tăng độ tin cậy trong việc loại bỏ nhiễu và đảm bảo chất lượng dữ liệu đầu vào.

1.4. Phương pháp nghiên cứu

(4) Chuẩn hóa biến định lượng:

Sau khi dữ liệu được làm sạch, các biến định lượng được chuẩn hóa bằng công cụ StandardScaler trong thư viện Scikit-learn, đưa chúng về cùng thang đo (trung bình = 0, độ lệch chuẩn = 1). Bước này giúp giảm thiểu sai lệch do khác biệt đơn vị đo lường, đồng thời tạo điều kiện thuận lợi cho các phân tích tương quan và mô hình hóa sau này.

(5) Tạo biến mới và tính toán chỉ số:

Để tăng tính đa chiều trong phân tích, một số biến mới được xây dựng, cụ thể:

- Biến Season (mùa) được xác định dựa trên tháng quan trắc, chia dữ liệu thành bốn nhóm mùa trong năm.

- Chỉ số AQI (Air Quality Index) được tính toán theo Quyết định 1459/QĐ-TCMT năm 2019 của Tổng cục Môi trường, nhằm lượng hóa mức độ ô nhiễm không khí theo chuẩn quốc gia.

Các biến này được dùng trong phân tích tương quan để xác định mức độ ảnh hưởng giữa các yếu tố khí tượng và nồng độ chất ô nhiễm.

1.4. Phương pháp nghiên cứu

(6) Trực quan hóa dữ liệu:

Quá trình trực quan hóa được thực hiện bằng các thư viện Seaborn và Matplotlib, thông qua các biểu đồ hộp (boxplot), tán xạ (scatter plot), đường (Line chart), cột (Bar chart), tròn (Pie chart), cột nhóm (Clustered Bar chart), vĩ cầm (Violin plot) và Hexbin plot Các biểu đồ này thể hiện hoặc sự thay đổi của dữ liệu trước và sau xử lý hoặc làm rõ xu hướng và mối quan hệ giữa các biến.

Kết quả trực quan không chỉ khẳng định tính hiệu quả của quy trình tiền xử lý, mà còn hỗ trợ rút ra các nhận định tổng quát về đặc điểm chất lượng không khí tại các trạm quan trắc, góp phần tạo nền tảng cho các nghiên cứu chuyên sâu và các đề xuất chính sách quản lý môi trường trong tương lai.

Quy trình nghiên cứu được thiết kế một cách hệ thống, đảm bảo tính nhất quán và độ tin cậy trong toàn bộ quá trình xử lý và phân tích dữ liệu. Việc kết hợp các phương pháp định lượng cùng thống kê mô tả và suy diễn giúp mô tả khách quan hiện trạng chất lượng không khí, đồng thời làm rõ các xu hướng và mối quan hệ giữa các yếu tố môi trường. Cách tiếp cận này góp phần bảo đảm tính khoa học, minh bạch và giá trị ứng dụng thực tiễn của nghiên cứu.

Chương 2: Dữ liệu

2.1. Mô tả dữ liệu

2.2. Thống kê mô tả

2.1. Mô tả dữ liệu

2.1.1. Đặc điểm dữ liệu

Bộ dữ liệu được sử dụng trong nghiên cứu này bao gồm các thông tin về chất lượng không khí và các yếu tố khí tượng được thu thập từ Mạng lưới Giám sát Chất lượng Không khí (Air Quality Monitoring Network) bao gồm sáu trạm quan trắc ô nhiễm không khí trên khắp Thành phố Hồ Chí Minh (TP.HCM), Việt Nam; được thu thập bởi Đại học Quốc gia Việt Nam, Việt Nam và Đại học Dublin, Ireland. Dữ liệu phản ánh sự biến động của các thông số môi trường theo thời gian và không gian, phục vụ cho việc phân tích, phát hiện ngoại lệ và đánh giá chất lượng không khí.

Cấu trúc dữ liệu bao gồm bốn nhóm biến chính:

- **Nhóm biến môi trường:** gồm các chỉ tiêu ô nhiễm không khí quan trọng như $PM_{2.5}$, PM_{10}, SO_2, NO_2, CO, và O_3. Đây là các thông số đặc trưng thể hiện nồng độ của bụi mịn và các khí gây ô nhiễm trong không khí.
    - ***$PM_{2.5}$ (µg/m³)***: Bụi mịn có đường kính nhỏ hơn 2.5 µm, ảnh hưởng trực tiếp đến hệ hô hấp và được xem là chỉ số quan trọng nhất trong đánh giá ô nhiễm không khí.
    - ***PM_{10} (µg/m³)***: Bụi có đường kính nhỏ hơn 10 µm, phản ánh tổng lượng bụi thô trong không khí.
    - ***SO_2 (µg/m³)***: Khí lưu huỳnh dioxide, phát sinh từ hoạt động đốt nhiên liệu hóa thạch, có thể gây kích ứng đường hô hấp.

2.1. Mô tả dữ liệu

2.1.1. Đặc điểm dữ liệu

- **NO_2 ($\mu g/m^3$)**: Khí nitơ dioxide, sản phẩm phụ của quá trình đốt cháy, đặc trưng cho ô nhiễm do giao thông và công nghiệp.

- **CO ($\mu g/m^3$)**: Khí carbon monoxide, sinh ra từ quá trình đốt cháy không hoàn toàn, ảnh hưởng đến khả năng vận chuyển oxy trong máu.

- **O_3 ($\mu g/m^3$)**: Ozon tầng thấp, hình thành do phản ứng quang hóa, là tác nhân chính gây ra hiện tượng sương mù quang hóa ở đô thị.

- **Nhóm biến khí tượng**: bao gồm temperature - nhiệt độ (°C), humidity - độ ẩm (%). Các yếu tố này có ảnh hưởng trực tiếp đến sự khuếch tán và biến động của các chất ô nhiễm trong khí quyển.

- **Nhóm biến thời gian**: gồm ngày, tháng, năm, và giờ, dùng để xác định chu kỳ, xu hướng và sự thay đổi theo thời gian của các thông số môi trường.

- **Nhóm biến không gian**: thể hiện thông tin về trạm quan trắc, là vị trí địa lý nơi thu thập dữ liệu.

2.1. Mô tả dữ liệu

2.1.1. Đặc điểm dữ liệu

Bộ dữ liệu có tính đa chiều, kết hợp giữa yếu tố thời gian, không gian và môi trường, cho phép phân tích tổng hợp và phát hiện các mẫu hình bất thường trong chất lượng không khí. Trong đó, năm trạm đầu tiên (Trạm 1, 2, 3, 4, 5) được sử dụng với mục đích chuẩn hóa dữ liệu và hiệu chỉnh mô hình, đảm bảo độ tin cậy cho các bước phân tích tiếp theo. Trạm 6 là trạm hoàn chỉnh, được sử dụng làm dữ liệu chính để kiểm định và đánh giá kết quả sau khi mô hình và quy trình xử lý được thiết lập.

2.1. Mô tả dữ liệu

2.1.2. Kích thước dữ liệu

Bộ dữ liệu gốc bao gồm 10 biến (cột) và 52.548 quan sát (hàng). Bộ dữ liệu của trạm 6 bao gồm 10 biến (cột) và 9499 quan sát (hàng). Mỗi hàng đại diện cho một bản ghi đo lường tại một thời điểm cụ thể ở một trạm quan trắc nhất định.

Kích thước này cho phép thực hiện các phân tích thống kê và mô hình học máy ở quy mô đủ lớn để đảm bảo tính tin cậy và đại diện cho hiện trạng chất lượng không khí trong khu vực nghiên cứu. Tuy nhiên, do dữ liệu được thu thập theo thời gian thực trong một giai đoạn dài, nên việc xuất hiện các giá trị khuyết (NaN), trùng lặp hoặc ngoại lệ (outlier) là điều khó tránh khỏi. Những vấn đề này được xử lý chi tiết thông qua các bước xử lý dữ liệu nhằm đảm bảo chất lượng và độ chính xác của phân tích.

2.2. Thống kê mô tả

```
data6 = pd.read_excel("DataSetSt6.xlsx")
data6.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9499 entries, 0 to 9498
Data columns (total 10 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   date          9499 non-null   object
 1   Station_No    9499 non-null   int64
 2   TSP           9499 non-null   float64
 3   PM2.5         9499 non-null   float64
 4   O3            9499 non-null   float64
 5   CO            9499 non-null   float64
 6   NO2           9499 non-null   float64
 7   SO2           9196 non-null   float64
 8   Temperature   9499 non-null   float64
 9   Humidity      9499 non-null   float64
dtypes: float64(8), int64(1), object(1)
memory usage: 742.2+ KB
```

Đoạn code trên thực hiện bước đầu tiên của quá trình thống kê mô tả dữ liệu, cụ thể là đọc và kiểm tra thông tin tổng quan của bộ dữ liệu trạm 6. Lệnh pd.read_excel("DataSetSt6.xlsx") dùng để tải dữ liệu từ file Excel, sau đó hàm data6.info() giúp hiển thị cấu trúc của bảng dữ liệu, bao gồm số lượng dòng, tên cột, kiểu dữ liệu và số lượng giá trị không bị khuyết (non-null).

Kết quả cho thấy tập dữ liệu gồm 9.499 bản ghi và 10 cột, trong đó có 8 cột dạng số thực (float64), 1 cột số nguyên (int64) và 1 cột dạng chuỗi (object). Đáng chú ý, cột SO_2 có 303 giá trị bị thiếu, trong khi các cột còn lại đầy đủ dữ liệu. Điều này cho thấy dữ liệu tương đối đầy đủ và có thể tiếp tục sử dụng cho bước xử lý và phân tích thống kê, tuy nhiên cần chú ý xử lý phần thiếu của SO_2 để đảm bảo tính chính xác trong các phân tích sau.

2.2. Thống kê mô tả

```
data6.drop(columns=['Station_No']).describe()
```

	TSP	PM2.5	O3	CO	NO2	SO2	Temperature	Humidity
count	9499.000000	9499.000000	9499.000000	9499.000000	9499.000000	9196.000000	9499.000000	9499.000000
mean	47.889637	20.198337	97.526608	1362.109842	112.046341	249.267907	29.380564	74.437878
std	33.673675	13.466133	31.586527	926.389186	37.625368	139.021951	2.674149	12.020361
min	6.031667	5.533333	0.000000	213.538000	0.000000	4.803333	21.621667	35.235000
25%	26.658333	12.555833	77.658461	715.963500	94.727333	137.550000	27.403333	65.878851
50%	38.753333	16.758333	91.410367	1089.846000	117.342700	216.150000	28.876667	75.983333
75%	58.667500	23.230000	110.287883	1692.451000	137.260533	323.133333	31.130000	83.796667
max	657.968333	310.400000	300.502583	21092.570770	251.152900	699.976667	39.256667	98.275000

Sau khi thực hiện lệnh data6.drop(columns=['Station_No']).describe(), nhóm thu được bảng thống kê mô tả cho 8 biến định lượng trong bộ dữ liệu của trạm quan trắc số 6 tại Thành phố Hồ Chí Minh. Mục đích của bước này là nhằm đánh giá sơ bộ đặc điểm phân bố, mức độ biến động, và tính hoàn chỉnh của các biến đo lường trước khi tiến hành các bước xử lý và phân tích chuyên sâu tiếp theo.

Về số lượng quan sát, hầu hết các biến có 9.499 giá trị, đảm bảo độ đầy đủ của dữ liệu. Riêng biến SO_2 chỉ có 9.196 giá trị, tức là còn thiếu khoảng 3,2% dữ liệu. Điều này cho thấy trong quá trình thu thập, đã có một số thời điểm thiết bị đo khí SO_2 bị lỗi hoặc không ghi nhận được kết quả. Việc này đòi hỏi nhóm phải thực hiện các biện pháp xử lý dữ liệu thiếu (imputation) ở bước kế tiếp để tránh làm sai lệch kết quả mô hình.

2.2. Thống kê mô tả

```
data6.drop(columns=['Station_No']).describe()
```

	TSP	PM2.5	O3	CO	NO2	SO2	Temperature	Humidity
count	9499.000000	9499.000000	9499.000000	9499.000000	9499.000000	9196.000000	9499.000000	9499.000000
mean	47.889637	20.198337	97.526608	1362.109842	112.046341	249.267907	29.380564	74.437878
std	33.673675	13.466133	31.586527	926.389186	37.625368	139.021951	2.674149	12.020361
min	6.031667	5.533333	0.000000	213.538000	0.000000	4.803333	21.621667	35.235000
25%	26.658333	12.555833	77.658461	715.963500	94.727333	137.550000	27.403333	65.878851
50%	38.753333	16.758333	91.410367	1089.846000	117.342700	216.150000	28.876667	75.983333
75%	58.667500	23.230000	110.287883	1692.451000	137.260533	323.133333	31.130000	83.796667
max	657.968333	310.400000	300.502583	21092.570770	251.152900	699.976667	39.256667	98.275000

Về giá trị trung bình (mean), có thể nhận thấy rằng nồng độ TSP (47,89 µg/m³) cao hơn rõ rệt so với $PM_{2.5}$ (20,20 µg/m³), phản ánh tình trạng bụi tổng thể tại khu vực này vẫn khá đáng lo ngại, đặc biệt trong bối cảnh mật độ giao thông và hoạt động xây dựng tại TP.HCM luôn ở mức cao. Các chất khí ô nhiễm như NO_2 (112,05 µg/m³) và SO_2 (249,27 µg/m³) đều có giá trị trung bình ở mức đáng chú ý, vượt ngưỡng khuyến nghị an toàn của Tổ chức Y tế Thế giới (WHO) trong một số thời điểm, cho thấy chất lượng không khí tại khu vực này có thể thường xuyên vượt ngưỡng cho phép.

2.2. Thống kê mô tả

```
data6.drop(columns=['Station_No']).describe()
```

	TSP	PM2.5	O3	CO	NO2	SO2	Temperature	Humidity
count	9499.000000	9499.000000	9499.000000	9499.000000	9499.000000	9196.000000	9499.000000	9499.000000
mean	47.889637	20.198337	97.526608	1362.109842	112.046341	249.267907	29.380564	74.437878
std	33.673675	13.466133	31.586527	926.389186	37.625368	139.021951	2.674149	12.020361
min	6.031667	5.533333	0.000000	213.538000	0.000000	4.803333	21.621667	35.235000
25%	26.658333	12.555833	77.658461	715.963500	94.727333	137.550000	27.403333	65.878851
50%	38.753333	16.758333	91.410367	1089.846000	117.342700	216.150000	28.876667	75.983333
75%	58.667500	23.230000	110.287883	1692.451000	137.260533	323.133333	31.130000	83.796667
max	657.968333	310.400000	300.502583	21092.570770	251.152900	699.976667	39.256667	98.275000

Về mức độ dao động, độ lệch chuẩn (std) của các chất ô nhiễm cho thấy sự biến thiên khá lớn: TSP (33,67), $PM_{2.5}$ (13,47), và đặc biệt là CO (926,39) cùng SO_2 (139,02) – hai biến có biên độ thay đổi mạnh nhất trong tập dữ liệu. Sự biến động này có thể do ảnh hưởng của các yếu tố thời gian (giờ cao điểm giao thông, ban ngày – ban đêm), điều kiện khí tượng, hoặc do sự cố kỹ thuật của thiết bị đo.

Đáng chú ý, CO có giá trị cực đại lên đến 21.092,57 $\mu g/m^3$, cao hơn hàng chục lần so với trung bình, và SO_2 đạt mức 699,97 $\mu g/m^3$, điều này cho thấy khả năng tồn tại nhiều giá trị ngoại lai (outliers) trong bộ dữ liệu. Các giá trị này nếu không được xử lý cẩn thận có thể gây sai lệch đáng kể cho kết quả phân tích tương quan hoặc mô hình hồi quy sau này.

2.2. Thống kê mô tả

```
data6.drop(columns=['Station_No']).describe()
```

	TSP	PM2.5	O3	CO	NO2	SO2	Temperature	Humidity
count	9499.000000	9499.000000	9499.000000	9499.000000	9499.000000	9196.000000	9499.000000	9499.000000
mean	47.889637	20.198337	97.526608	1362.109842	112.046341	249.267907	29.380564	74.437878
std	33.673675	13.466133	31.586527	926.389186	37.625368	139.021951	2.674149	12.020361
min	6.031667	5.533333	0.000000	213.538000	0.000000	4.803333	21.621667	35.235000
25%	26.658333	12.555833	77.658461	715.963500	94.727333	137.550000	27.403333	65.878851
50%	38.753333	16.758333	91.410367	1089.846000	117.342700	216.150000	28.876667	75.983333
75%	58.667500	23.230000	110.287883	1692.451000	137.260533	323.133333	31.130000	83.796667
max	657.968333	310.400000	300.502583	21092.570770	251.152900	699.976667	39.256667	98.275000

Về các yếu tố khí tượng, nhiệt độ trung bình tại trạm đo là 29,38°C (dao động từ 21,62°C đến 39,26°C) và độ ẩm trung bình là 74,44%. Các giá trị này phản ánh điều kiện khí hậu nhiệt đới đặc trưng của khu vực TP.HCM – nóng ẩm quanh năm, ít biến động theo mùa. Độ lệch chuẩn thấp của hai biến này (lần lượt là 2,67 và 12,02) cho thấy dữ liệu ổn định và có thể được dùng làm biến kiểm soát trong các mô hình phân tích ô nhiễm không khí. Tổng hợp lại, bảng thống kê mô tả cho thấy bộ dữ liệu có chất lượng tương đối tốt, tuy nhiên tồn tại ba vấn đề cần lưu ý: (1) dữ liệu thiếu ở biến SO_2, (2) khả năng xuất hiện giá trị ngoại lai ở CO và SO_2, và (3) sự phân tán cao giữa các chất ô nhiễm, phản ánh đặc điểm phức tạp của môi trường đô thị. Những phát hiện này là cơ sở quan trọng để nhóm tiến hành chuẩn hóa dữ liệu, kiểm tra tương quan, và xây dựng mô hình hồi quy hoặc dự đoán nồng độ SO_2 ở các bước tiếp theo.

2.2. Thống kê mô tả

```
data6.drop(columns=['Station_No']).describe()
```

	TSP	PM2.5	O3	CO	NO2	SO2	Temperature	Humidity
count	9499.000000	9499.000000	9499.000000	9499.000000	9499.000000	9196.000000	9499.000000	9499.000000
mean	47.889637	20.198337	97.526608	1362.109842	112.046341	249.267907	29.380564	74.437878
std	33.673675	13.466133	31.586527	926.389186	37.625368	139.021951	2.674149	12.020361
min	6.031667	5.533333	0.000000	213.538000	0.000000	4.803333	21.621667	35.235000
25%	26.658333	12.555833	77.658461	715.963500	94.727333	137.550000	27.403333	65.878851
50%	38.753333	16.758333	91.410367	1089.846000	117.342700	216.150000	28.876667	75.983333
75%	58.667500	23.230000	110.287883	1692.451000	137.260533	323.133333	31.130000	83.796667
max	657.968333	310.400000	300.502583	21092.570770	251.152900	699.976667	39.256667	98.275000

Nhìn chung, chất lượng không khí tại trạm 6 trong giai đoạn khảo sát được đánh giá là ô nhiễm ở mức trung bình đến cao, đặc biệt đối với các thông số bụi mịn ($PM_{2.5}$, TSP) và khí SO_2. Sự hiện diện thường xuyên của các giá trị vượt chuẩn WHO cho thấy tình trạng ô nhiễm không khí tại khu vực đô thị TP.HCM đang ở mức đáng lo ngại, có thể ảnh hưởng đến sức khỏe cộng đồng, nhất là nhóm dân cư sống gần các tuyến giao thông lớn và khu công nghiệp.

Chương 3: Xử lý dữ liệu

3.1. Chuẩn hóa định dạng

3.2. Dữ liệu khuyết

3.3. Dữ liệu trùng lặp

3.4. Phát hiện & Xử lý ngoại lệ

3.5. Chuẩn hóa dữ liệu

3.6. Loại bỏ biến không hữu ích & tạo biến mới

3.7. Kiểm tra tương quan

3.1. Chuẩn hóa định dạng

3.1.1. Chuẩn hóa tên cột

```
import pandas as pd
# Hàm chuẩn hóa tên cột
def clean_columns(df):
 df.columns = (
 df.columns
 # Xóa khoảng trắng đầu/cuối
 .str.strip()
 # Xóa ký tự BOM ẩn
 .str.replace('\ufeff', '', regex=False)
 # Thay dấu gạch dưới bằng khoảng trắng
 .str.replace('_', ' ', regex=False)
 # Chuyển chữ hoa → chữ thường
 .str.lower()
 )
 return df
# Danh sách các file cần xử lý
files = {
 "1": "DataSetSt1.xlsx",
 "2": "DataSetSt2.xlsx",
 "3": "DataSetSt3.xlsx",
 "4": "DataSetSt4.xlsx",
 "5": "DataSetSt5.xlsx",
 "6": "DataSetSt6.xlsx"
}
# Áp dụng hàm và lưu file mới
for st, filename in files.items():
 df = pd.read_excel(filename)
 df = clean_columns(df)
 clean_name = f"DataSetSt{st}Clean1.xlsx"
 df.to_excel(clean_name, index=False)
 print(f"Đã lưu {clean_name}")
```

3.1. Chuẩn hóa định dạng

3.1.1. Chuẩn hóa tên cột

Trong quá trình tổng hợp dữ liệu quan trắc không khí từ nhiều trạm khác nhau, nhóm nghiên cứu nhận thấy tên các cột dữ liệu không đồng nhất. Cụ thể, giữa các file có sự khác biệt về cách viết hoa – viết thường, ký tự đặc biệt (như dấu gạch dưới "_"), hoặc khoảng trắng thừa và ký tự BOM ẩn (\ufeff). Những sai lệch nhỏ này có thể gây ra lỗi khi truy cập cột trong Python, làm ảnh hưởng đến quá trình xử lý và phân tích dữ liệu. Vì vậy, việc chuẩn hóa tên cột được thực hiện nhằm đảm bảo tính nhất quán giữa các file dữ liệu, giúp cho việc thao tác, tổng hợp và trực quan hóa sau này diễn ra thuận lợi, chính xác hơn.

Đoạn mã trên được sử dụng để tự động hóa quy trình chuẩn hóa. Đầu tiên, hàm clean_columns(df) được định nghĩa để xử lý toàn bộ các lỗi định dạng trong tên cột. Hàm này thực hiện các bước: xóa khoảng trắng thừa, loại bỏ ký tự BOM ẩn, thay dấu gạch dưới bằng khoảng trắng và chuyển toàn bộ tên cột sang chữ thường. Sau đó, chương trình áp dụng hàm này lần lượt cho sáu file dữ liệu tương ứng với sáu trạm quan trắc, và lưu lại kết quả với tên mới có hậu tố Clean1 để dễ dàng quản lý. Khi kiểm tra lại file trạm 6, kết quả cho thấy các tên cột đã được chuẩn hóa đồng nhất như date, station no, tsp, pm2.5, o3, co, no2, so2, temperature, humidity. Điều này chứng minh việc chuẩn hóa đã được thực hiện thành công, đảm bảo dữ liệu sạch, đồng bộ và sẵn sàng cho các bước phân tích tiếp theo.

3.1. Chuẩn hóa định dạng

3.1.2. Chuẩn hóa cột date

```
import pandas as pd
# Hàm chuẩn hóa cột ngày
def clean_date_column(df, column_name='date'):
    # Chuẩn hóa chuỗi
    df[column_name] = df[column_name].astype(str).str.strip()
    # Thử các định dạng khác nhau
      df['date_clean'] = pd.to_datetime(df[column_name], format='%d-
%m-%Y %H:%M', errors='coerce')
    mask = df['date_clean'].isna()
         df.loc[mask, 'date_clean'] = pd.to_datetime(df.loc[mask,
column_name], format='%Y-%m-%d %H:%M:%S', errors='coerce')
    mask = df['date_clean'].isna()
         df.loc[mask, 'date_clean'] = pd.to_datetime(df.loc[mask,
column_name], errors='coerce', dayfirst=True)
    # Gán lại và xóa cột tạm
    df[column_name] = df['date_clean']
    df.drop(columns=['date_clean'], inplace=True)
    return df
# Danh sách các file cần xử lý
files1 = {
    "1": "DataSetSt1Clean1.xlsx",
    "2": "DataSetSt2Clean1.xlsx",
    "3": "DataSetSt3Clean1.xlsx",
    "4": "DataSetSt4Clean1.xlsx",
    "5": "DataSetSt5Clean1.xlsx",
    "6": "DataSetSt6Clean1.xlsx"
}
# Xử lý từng file
for st, filename in files1.items():
    df = pd.read_excel(filename)
    df = clean_date_column(df, column_name='date')
    clean_name = f"DataSetSt{st}Clean2.xlsx"
    df.to_excel(clean_name, index=False)
    valid_dates = df['date'].notna().sum()
    print(f"Đã lưu {clean_name}")
```

3.1. Chuẩn hóa định dạng

3.1.2. Chuẩn hóa cột date

Trong các tệp dữ liệu ban đầu, cột "date" tồn tại nhiều định dạng thời gian khác nhau, chẳng hạn như ngày-tháng-năm (dd-mm-yyyy), năm-tháng-ngày (yyyy-mm-dd), hoặc có sự khác biệt trong cách hiển thị giờ, phút, giây. Sự không nhất quán này gây khó khăn trong quá trình xử lý và phân tích dữ liệu, đặc biệt khi cần thực hiện các thao tác liên quan đến chuỗi thời gian như gộp, lọc hoặc tính trung bình theo ngày. Vì vậy, việc chuẩn hóa định dạng của cột ngày là bước quan trọng nhằm đảm bảo tính thống nhất trong toàn bộ tập dữ liệu, tạo điều kiện thuận lợi cho các bước phân tích, trực quan hóa và đối chiếu dữ liệu ở các giai đoạn tiếp theo.

Đoạn mã trên định nghĩa hàm clean_date_column() với mục đích chuẩn hóa cột thời gian trong từng tệp dữ liệu. Hàm này trước hết loại bỏ các ký tự dư thừa và khoảng trắng, sau đó tuần tự chuyển đổi dữ liệu ngày sang kiểu datetime theo ba định dạng phổ biến: %d-%m-%Y %H:%M, %Y-%m-%d %H:%M:%S, và định dạng tự động có tham số dayfirst=True để bao quát các trường hợp còn lại. Sau khi chuyển đổi thành công, dữ liệu ngày được gán trở lại vào cột gốc, đảm bảo tất cả các giá trị đều hợp lệ và thống nhất. Vòng lặp tiếp theo áp dụng hàm này cho sáu tệp dữ liệu tương ứng với sáu trạm quan trắc, sau đó lưu kết quả dưới dạng các tệp mới mang tên DataSetStxClean2.xlsx. Việc chương trình hiển thị thông báo "Đã lưu DataSetStxClean2.xlsx" cho từng trạm cho thấy quá trình chuẩn hóa đã được thực hiện thành công, giúp toàn bộ dữ liệu thời gian đạt định dạng chuẩn, sẵn sàng phục vụ cho quá trình xử lý và phân tích chuyên sâu ở các bước kế tiếp.

3.1. Chuẩn hóa định dạng

3.1.3. Dữ liệu khuyết

```
import pandas as pd
import numpy as np
# Hàm xử lý giá trị 0 và chuỗi rỗng
def clean_missing_values(df):
    numeric_cols = df.select_dtypes(include='number').columns
     df[numeric_cols] = df[numeric_cols].replace(0, np.nan) # 0 →
NaN
    df.replace("", np.nan, inplace=True)          # "" → NaN
    return df
# Danh sách các file cần xử lý
files2 = {
    "1": "DataSetSt1Clean2.xlsx",
    "2": "DataSetSt2Clean2.xlsx",
    "3": "DataSetSt3Clean2.xlsx",
    "4": "DataSetSt4Clean2.xlsx",
    "5": "DataSetSt5Clean2.xlsx",
    "6": "DataSetSt6Clean2.xlsx"
}
# Xử lý từng file
for st, filename in files2.items():
    df = pd.read_excel(filename)
    df = clean_missing_values(df)
    output_name = f"DataSetSt{st}Clean3.xlsx"
    df.to_excel(output_name, index=False)
    print(f"Đã lưu {output_name}")
```

3.1. Chuẩn hóa định dạng

3.1.3. Chuẩn hóa dữ liệu khuyết

Trong bộ dữ liệu gốc, một số giá trị được ghi nhận là 0 tại các cột chứa chỉ số quan trắc. Tuy nhiên, các giá trị này không phản ánh giá trị thực của phép đo, mà thường biểu thị tình trạng thiếu dữ liệu hoặc lỗi cảm biến trong quá trình thu thập. Nếu giữ nguyên, các giá trị này có thể gây sai lệch cho kết quả phân tích thống kê hoặc mô hình hóa dữ liệu. Vì vậy, nhóm đã tiến hành chuẩn hóa các giá trị 0 thành giá trị rỗng (NaN) để giúp chương trình nhận diện đúng các trường hợp thiếu dữ liệu. Đồng thời, các ô trống trong tập dữ liệu cũng được chuyển đổi thành NaN, nhằm đảm bảo tính thống nhất về định dạng và giúp quá trình xử lý dữ liệu sau này được thuận lợi và chính xác hơn.

Đoạn mã trên định nghĩa hàm clean_missing_values() nhằm xử lý các giá trị thiếu và không hợp lệ trong dữ liệu. Cụ thể, hàm đầu tiên xác định các cột có kiểu dữ liệu số, sau đó thay thế tất cả các giá trị bằng 0 trong những cột này bằng NaN. Tiếp đến, toàn bộ các ô chứa chuỗi rỗng trong DataFrame được thay thế bằng NaN để đồng bộ định dạng dữ liệu. Sau khi hàm được định nghĩa, một vòng lặp được thực hiện trên sáu tệp dữ liệu tương ứng với sáu trạm quan trắc. Kết quả sau khi xử lý được lưu thành các tệp mới mang tên DataSetStxClean3.xlsx. Thông báo "Đã lưu DataSetStxClean3.xlsx" cho thấy rằng việc thay thế đã hoàn tất thành công, giữ nguyên kích thước dữ liệu nhưng giúp loại bỏ những giá trị sai lệch, đảm bảo dữ liệu sẵn sàng cho giai đoạn phân tích và điền giá trị thiếu tiếp theo.

3.2. Dữ liệu khuyết

3.2.1. Chuẩn bị

```
date              0
station no        0
tsp               0
pm2.5             0
o3                1
co                0
no2              31
so2             303
temperature       0
humidity          0
dtype: int64
```

Kết quả cho thấy các cột O_3, NO_2 và SO_2 vẫn còn dữ liệu bị khuyết, trong đó SO_2 có số lượng thiếu lớn nhất (303 giá trị), tiếp đến là NO_2 (31 giá trị) và O_3 (1 giá trị). Điều này cho thấy trong quá trình thu thập, một số phép đo có thể bị gián đoạn hoặc lỗi cảm biến. Vì bộ dữ liệu mang tính chuỗi thời gian, các giá trị bị thiếu ở mức nhỏ như O_3 và NO_2 có thể được nội suy dựa trên các mốc thời gian lân cận, trong khi cột SO_2 — với tỉ lệ khuyết lớn — cần được xử lý bằng phương pháp hồi quy hoặc sử dụng dữ liệu từ trạm phụ trợ để ước lượng lại, đảm bảo tính đầy đủ và độ tin cậy cho phân tích sau này.

Sau khi xử lý dữ liệu thiếu và chuẩn hóa ở các bước trước, nhóm tiến hành xác định mối quan hệ giữa các trạm quan trắc khác nhau nhằm hỗ trợ quá trình điền dữ liệu bị thiếu của trạm 6. Trong tập dữ liệu gốc, các thông số ô nhiễm như SO_2, NO_2 và O_3 của trạm 6 có những thời điểm bị thiếu, trong khi các trạm lân cận có thể vẫn ghi nhận đầy đủ dữ liệu ở cùng thời điểm đó. Do đó, việc tìm ra cặp trạm có mức độ tương đồng cao và xuất hiện đồng thời trong cùng các thời điểm thiếu dữ liệu sẽ giúp xác định nguồn thay thế phù hợp khi tiến hành nội suy hoặc ước lượng giá trị bị thiếu cho trạm 6.

3.2. Dữ liệu khuyết

3.2.1. Chuẩn bị

```
import pandas as pd
from itertools import combinations
from collections import Counter
# Đọc trạm 6
df6 = pd.read_excel("DataSetSt6Clean3.xlsx")
cols_check = ['so2','no2','o3']
# Cột missing_vars liệt kê các biến còn thiếu trên từng dòng
def vars_missing(row):
    return [c for c in cols_check if pd.isna(row[c])]
df6['missing_vars'] = df6.apply(vars_missing, axis=1)
df6_missing = df6[df6['missing_vars'].apply(len) > 0].copy()
# Đọc trạm 1-5
files3 = {
    "1": "DataSetSt1Clean3.xlsx",
    "2": "DataSetSt2Clean3.xlsx",
    "3": "DataSetSt3Clean3.xlsx",
    "4": "DataSetSt4Clean3.xlsx",
    "5": "DataSetSt5Clean3.xlsx"
}
dfs = {}
for name, f in files3.items():
    df = pd.read_excel(f)
    df['date'] = pd.to_datetime(df['date'], dayfirst=True)
    # Chỉ giữ những dòng có date xuất hiện trong trạm 6
    df = df[df['date'].isin(df6['date'])]
    # Đảm bảo các cột cần check tồn tại
    for c in cols_check:
        if c not in df.columns:
            df[c] = pd.NA
    # Đổi tên các cột
        df = df.rename(columns={c: f"{c}_{name}" for c in
cols_check})
        dfs[name] = df[['date'] + [f"{c}_{name}" for c in
cols_check]]
```

3.2. Dữ liệu khuyết

3.2.1. Chuẩn bị

```
# Merge trạm 6 còn thiếu với các trạm 1-5 theo date (chỉ các
date của trạm 6)
merged = df6_missing[['date','missing_vars']].copy()
for name, df in dfs.items():
    merged = pd.merge(merged, df, on='date', how='left')
# Đếm cặp trạm xuất hiện cùng nhau nhiều nhất
trams = ["1","2","3","4","5"]
pair_counter = Counter()
for idx, row in merged.iterrows():
    missing_vars = row['missing_vars']
    present_trams = []
    for t in trams:
              if  all(pd.notna(row[f"{var}_{t}"])  for  var  in
missing_vars):
            present_trams.append(t)
    for pair in combinations(sorted(present_trams), 2):
        pair_counter[pair] += 1
# In cặp trạm xuất hiện nhiều nhất
most_common_pairs = pair_counter.most_common(1)
print("Cặp trạm xuất hiện cùng nhau nhiều nhất trên các dòng
thiếu của trạm 6:")
for pair, count in most_common_pairs:
    print(f"Cặp {pair} - xuất hiện {count} dòng")
```

3.2. Dữ liệu khuyết

3.2.1. Chuẩn bị

Đoạn mã trên thực hiện quy trình xác định cặp trạm quan trắc có dữ liệu tương đồng nhất đối với các trường hợp trạm 6 bị thiếu. Đầu tiên, hàm vars_missing() được sử dụng để liệt kê các biến bị thiếu (SO_2, NO_2, O_3) trong từng dòng dữ liệu của trạm 6. Sau đó, dữ liệu của năm trạm còn lại (1–5) được đọc vào, chuẩn hóa định dạng thời gian và lọc để chỉ giữ lại những dòng có cùng thời điểm với trạm 6. Các cột chỉ số được đổi tên tương ứng (ví dụ: so2_1, no2_1) nhằm phân biệt dữ liệu giữa các trạm. Toàn bộ dữ liệu này được gộp lại (merge) dựa trên cột date để so sánh.

Bằng cách sử dụng thư viện Counter, chương trình đếm tần suất xuất hiện của các cặp trạm có đủ dữ liệu cho cùng loại biến bị thiếu trong cùng một thời điểm. Kết quả cho thấy cặp trạm (2, 4) xuất hiện đồng thời nhiều nhất với 240 lần, cho thấy hai trạm này có độ đầy đủ và tương quan dữ liệu cao nhất so với trạm 6. Từ đó, nhóm có thể ưu tiên sử dụng dữ liệu của trạm 2 và trạm 4 để hỗ trợ điền giá trị thiếu cho trạm 6 trong các bước xử lý tiếp theo.

3.2. Dữ liệu khuyết

3.2.2. Xử lý

```
import pandas as pd
import numpy as np
from google.colab import files
# Đọc dữ liệu
df6 = pd.read_excel("DataSetSt6Clean3.xlsx")
df2 = pd.read_excel("DataSetSt2Clean3.xlsx")
df4 = pd.read_excel("DataSetSt4Clean3.xlsx")
for df in [df6, df2, df4]:
    df['date'] = pd.to_datetime(df['date'], dayfirst=True)
cols = ['so2','no2','o3']
# Merge chỉ giữ các ngày gốc của trạm 6
merged = df6[['date'] + cols].copy()
merged = pd.merge(merged, df2[['date'] + cols], on='date',
how='left', suffixes=('', '_2'))
merged = pd.merge(merged, df4[['date'] + cols], on='date',
how='left', suffixes=('', '_4'))
# Nếu trạm 2 & 4 đều có dữ liệu → mean
for col in cols:
    col2, col4 = f"{col}_2", f"{col}_4"
   mask = merged[col].isna() & merged[col2].notna() &
merged[col4].notna()
      merged.loc[mask, col] = (merged.loc[mask, col2] +
merged.loc[mask, col4]) / 2
# Điền missing theo mean ngày trạm 6
daily_mean_6 = merged.groupby('date')
[cols].transform('mean')
merged[cols] = merged[cols].fillna(daily_mean_6)
# Điền missing theo mean ngày trạm 2 & 4
for col in cols:
    col2, col4 = f"{col}_2", f"{col}_4"
                merged[f'{col}_mean24'] = merged[[col2,
col4]].mean(axis=1)
                daily_mean_24 = merged.groupby('date')
[f'{col}_mean24'].transform('mean')
    merged[col] = merged[col].fillna(daily_mean_24)
    merged.drop(columns=f'{col}_mean24', inplace=True)
```

3.2. Dữ liệu khuyết

3.2.2. Xử lý

```
# Điền missing theo thời gian nếu còn
merged.sort_values('date', inplace=True)
merged.set_index('date', inplace=True)
merged[cols] = merged[cols].interpolate(method='time')
merged.reset_index(inplace=True)
# Kiểm tra và xuất file
print("Số dòng còn thiếu:",
merged[cols].isna().any(axis=1).sum())
df6_filled = merged[['date'] + cols].copy()
# Lưu và tải file
output_name = "DataSetSt6Clean4.xlsx"
df6_filled.to_excel(output_name, index=False)
print(f"Đã lưu file {output_name}")
```

Đầu tiên, ba bộ dữ liệu từ trạm 6, trạm 2 và trạm 4 được đọc vào và chuẩn hóa cột date về cùng định dạng thời gian. Các bộ dữ liệu này được gộp (merge) theo cột date, chỉ giữ lại những ngày có trong trạm 6 để đảm bảo tính tương ứng theo thời gian. Sau đó, chương trình tiến hành ba cấp độ điền dữ liệu:

1. Nếu cả trạm 2 và 4 đều có dữ liệu ở cùng thời điểm mà trạm 6 bị thiếu, giá trị được điền bằng trung bình của hai trạm.

2. Nếu vẫn còn thiếu, chương trình sử dụng giá trị trung bình theo ngày của trạm 6 để thay thế.

3. Cuối cùng, với những giá trị còn thiếu sau hai bước trên, dữ liệu được nội suy tuyến tính theo thời gian (time interpolation) để làm mượt chuỗi.

3.2. Dữ liệu khuyết

3.2.2. Xử lý

Kết quả cho thấy toàn bộ dữ liệu bị thiếu đã được xử lý hoàn chỉnh, với số dòng còn thiếu bằng 0, và bộ dữ liệu mới được lưu thành công dưới tên “DataSetSt6Clean4.xlsx”. Điều này chứng tỏ quy trình điền dữ liệu đã hoạt động hiệu quả, đảm bảo bộ dữ liệu trạm 6 đạt được mức độ hoàn chỉnh và nhất quán cao, sẵn sàng cho các phân tích tiếp theo.

3.2. Dữ liệu khuyết

3.2.3. Ghép file

```
import pandas as pd
from openpyxl import load_workbook
from google.colab import files
# Đọc file chưa điền missing values
df6_original = pd.read_excel("DataSetSt6Clean3.xlsx")
# Chỉ lấy các cột khác ngoài so2, no2, o3
other_cols = [c for c in df6_original.columns if c not in
['so2','no2','o3']]
# Đọc file đã điền missing values
df6_filled = pd.read_excel("DataSetSt6Clean4.xlsx") # chỉ
chứa 'date','so2','no2','o3'
# Ghép lại: chỉ giữ các ngày gốc của trạm 6
df6_complete = pd.merge(
    df6_original[other_cols],    # các cột gốc trạm 6, bao
gồm cột date
    df6_filled,           # các cột đã điền missing
    on='date',           # merge theo ngày gốc của trạm 6
    how='left'
)
# Lưu file hoàn chỉnh
file_name = "DataSetSt6Clean5.xlsx"
df6_complete.to_excel(file_name, index=False)
# Chỉnh cột 'date' tự dãn đủ dài
wb = load_workbook(file_name)
ws = wb.active
# Cột 'date' thường là cột đầu tiên
date_col = 1
max_length =
max(df6_complete['date'].astype(str).apply(len)) + 2
ws.column_dimensions[ws.cell(row=1,
column=date_col).column_letter].width = max_length
wb.save(file_name)
print(f"Đã lưu file hoàn chỉnh: {file_name}")
```

3.2. Dữ liệu khuyết

3.2.3. Ghép file

Đầu tiên, chương trình đọc hai bộ dữ liệu: (1) DataSetSt6Clean3.xlsx – bản gốc của trạm 6 trước khi điền dữ liệu thiếu, và (2) DataSetSt6Clean4.xlsx – bản đã điền đầy đủ các giá trị cho ba cột SO_2, NO_2, O_3. Sau đó, chương trình tách riêng các cột khác ngoài ba biến này để tránh trùng lặp, rồi ghép (merge) hai bảng theo cột date nhằm khôi phục đầy đủ cấu trúc dữ liệu ban đầu.

Cuối cùng, bộ dữ liệu hoàn chỉnh được lưu thành file "DataSetSt6Clean5.xlsx", đồng thời điều chỉnh độ rộng của cột date trong Excel để hiển thị toàn bộ giá trị thời gian rõ ràng và chuyên nghiệp hơn. Kết quả cho thấy toàn bộ dữ liệu của trạm 6 đã được khôi phục thành công, với các giá trị thiếu được điền chính xác, đảm bảo bộ dữ liệu hoàn chỉnh, đồng bộ và sẵn sàng cho phân tích thống kê mô tả trong các bước tiếp theo.

3.3. Dữ liệu trùng lặp

```
import pandas as pd
# Đọc file hoàn chỉnh
df6_complete = pd.read_excel("DataSetSt6Clean5.xlsx")
# Kiểm tra số lượng dòng trùng nhau hoàn toàn
num_duplicates = df6_complete.duplicated().sum()
print(f"Số dòng trùng nhau hoàn toàn: {num_duplicates}")
```

Đoạn mã trên được sử dụng để kiểm tra xem trong bộ dữ liệu có dòng nào bị trùng lặp hoàn toàn hay không. Kết quả trả về 0 cho thấy không có bản ghi trùng lặp, chứng tỏ dữ liệu của trạm 6 được ghi nhận duy nhất cho từng mốc thời gian, đảm bảo tính toàn vẹn và độ tin cậy của bộ dữ liệu.

3.4. Phát hiện & xử lý ngoại lệ

Trong quá trình thu thập dữ liệu thực tế từ các trạm quan trắc, dữ liệu thường chứa nhiều điểm bất thường (outliers) do lỗi cảm biến, điều kiện khí tượng cực đoan, hoặc sự cố kỹ thuật trong hệ thống truyền nhận dữ liệu. Những ngoại lệ này có thể ảnh hưởng đáng kể đến độ chính xác của các phân tích và mô hình hóa sau này. Do đó, việc phát hiện và xử lý ngoại lệ là bước không thể thiếu trong giai đoạn tiền xử lý dữ liệu.

Trong nghiên cứu này, nhóm áp dụng hai phương pháp kết hợp để phát hiện và hiệu chỉnh ngoại lệ:

- Phương pháp thống kê truyền thống: Dựa trên IQR (Interquartile Range).
- Phương pháp học máy không giám sát: Dựa trên mô hình Isolation Forest.

Việc kết hợp cả hai phương pháp giúp đảm bảo quy trình phát hiện ngoại lệ vừa mang tính chặt chẽ về mặt thống kê, vừa có khả năng thích ứng với các dạng phân bố dữ liệu phức tạp.

3.4.1. Xác định các biến định lượng cần kiểm tra

Các biến được lựa chọn để kiểm tra ngoại lệ bao gồm: TSP, PM2.5, O_3, CO, NO_2, SO_2, Temperature, Humidity. Đây là những biến định lượng có khả năng chịu ảnh hưởng mạnh của yếu tố môi trường, và thường dễ phát sinh sai số hoặc giá trị lệch.

3.4. Phát hiện & xử lý ngoại lệ

3.4.2. Phát hiện ngoại lệ bằng phương pháp IQR

Phương pháp IQR (Interquartile Range) là kỹ thuật thống kê phổ biến để phát hiện giá trị nằm ngoài phạm vi phân bố dữ liệu thông thường. Khoảng IQR được tính bằng:

IQR = Q3 - Q1, trong đó Q1 và Q3 lần lượt là tứ phân vị thứ nhất và thứ ba của dữ liệu.

Một giá trị được xem là ngoại lệ nếu:

$x < Q1 - 1.5 \times IQR$ hoặc $x > Q3 + 1.5 \times IQR$

Đoạn mã thực hiện:

```
def detect_outliers_iqr(series, k=1.5):
    q1 = series.quantile(0.25)
    q3 = series.quantile(0.75)
    iqr = q3 - q1
    lower = q1 - k * iqr
    upper = q3 + k * iqr
    return (series < lower) | (series > upper), lower, upper
for col in quant_cols:
    df6[f'flag_iqr_{col}'] = detect_outliers_iqr(df6[col])
df6['flag_iqr'] = df6[[f'flag_iqr_{c}' for c in
quant_cols]].any(axis=1)
```

Kết quả trả về là các vị trí có giá trị nằm ngoài khoảng giới hạn được đánh dấu bằng chữ TRUE, giúp nhận diện nhanh các cột hoặc thời điểm có dữ liệu bất thường.

3.4. Phát hiện & xử lý ngoại lệ

3.4.2. Phát hiện ngoại lệ bằng phương pháp IQR

Đoạn mã trên thực hiện logic phát hiện và gắn cờ các giá trị outlier trong dữ liệu bằng phương pháp IQR. Cụ thể, hàm detect_outliers_iqr() tính các tứ phân vị Q1 và Q3 của từng biến định lượng, sau đó xác định khoảng giá trị hợp lệ trong khoảng (Q1 – 1.5 × IQR, Q3 + 1.5 × IQR). Những điểm dữ liệu nằm ngoài khoảng này được xem là ngoại lai. Tiếp đó, vòng lặp áp dụng hàm này cho từng biến trong danh sách quant_cols để tạo các cột cờ (flag_iqr_<tên từng biến>) thể hiện vị trí các giá trị bất thường. Cuối cùng, mã tổng hợp các cột này thành một cột duy nhất flag_iqr, đánh dấu những bản ghi có ít nhất một giá trị ngoại lai ở bất kỳ biến nào. Logic này giúp nhanh chóng nhận diện và cô lập các quan sát bất thường, phục vụ cho bước làm sạch và chuẩn hóa dữ liệu trước khi phân tích.

date	station_no	tsp	pm2.5	o3	co	no2	so2	emperatur	humidity	lag_iqr_ts	g_iqr_pm2	lag_iqr_o3	lag_iqr_co	ag_iqr_no	lag_iqr_so	qr_temper	_iqr_humi
2021-02-24 16:00:00	6	27.24545	14.38182	49.61036	1243.41	92.47464	327.5	25.72727	62.8	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
2021-02-24 17:00:00	6	30.54	15.22	59.2826	1155.168	106.8976	324.88	26.7	62.1	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
2021-03-05 10:00:00	6	39.9204	22.10079	77.57916	782.909	8.317544	299.0231	34.15874	56.78036	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE
2021-03-05 11:00:00	6	41.1912	23.38633	110.335	765.719	15.05224	309.6141	34.88493	54.476	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE
2021-03-05 14:00:00	6	39.68524	22.33616	112.2419	814.5673	11.27578	371.1485	36.21538	46.86865	FALSE	FALSE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE
2021-03-05 15:00:00	6	39.03251	20.56384	112.5657	859.1075	21.17409	384.4339	35.68671	47.80421	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
2021-03-05 16:00:00	6	39.78467	20.67033	100.8393	1005.042	29.98653	382.6073	34.71267	47.00733	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
2021-03-11 16:00:00	6	45.52433	24.47033	154.5753	974.0332	114.1355	214.4033	29.41	47.9075	FALSE	FALSE	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE
2021-03-11 17:00:00	6	29.54	13.76667	53.91707	1290.396	107.0231	349.3333	23.72	60.24	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
2021-03-12 10:00:00	6	57.67433	29.764	112.9379	807.8536	84.1254	96.94	27.90083	45.76083	FALSE	FALSE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE
2021-03-12 14:00:00	6	42.504	21.40133	120.3973	642.8439	84.22734	115.1344	28.765	41.7075	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	TRUE
2021-03-12 15:00:00	6	41.71667	21.068	114.7155	685.2173	89.78708	114.2611	28.5275	42.02708	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE
2021-03-12 16:00:00	6	64.8	22.765	75.15018	1632.095	45.23073	355.9726	32.55333	51.50833	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
2021-03-12 17:00:00	6	50.53167	30.495	106.6672	1062.609	108.8188	211.7833	27.00542	46.22667	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE
2021-03-12 18:00:00	6	60.96833	21.595	89.05477	3027.541	92.6885	355.9726	30.76333	53.12833	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
2021-03-12 19:00:00	6	64.0463	35.46606	116.3605	994.3693	141.2615	167.3404	25.11278	50.90944	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE
2021-03-12 20:00:00	6	42.71708	23.31875	85.75038	897.1461	135.2322	163.095	24.63333	47.84222	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE

Hình 3.1. Các cột đánh dấu các outlier được tìm bằng phương pháp IQR

3.4. Phát hiện & xử lý ngoại lệ

3.4.3. Phát hiện ngoại lệ bằng mô hình Isolation Forest

Bên cạnh phương pháp thống kê, nhóm nghiên cứu áp dụng thêm mô hình Isolation Forest, một thuật toán học máy không giám sát chuyên dùng cho phát hiện điểm bất thường (anomaly detection).

Nguyên lý hoạt động của mô hình Isolation Forest: Isolation Forest được đề xuất bởi Liu et al. (2008), dựa trên ý tưởng rằng các điểm bất thường thường "dễ bị cô lập" hơn so với các điểm bình thường. Thuật toán xây dựng nhiều cây nhị phân ngẫu nhiên (isolation trees) bằng cách chia dữ liệu theo các thuộc tính và ngưỡng được chọn ngẫu nhiên. Những điểm dữ liệu bình thường cần qua nhiều lần chia mới bị cô lập. Trong khi đó, ngoại lệ thường nằm xa trung tâm phân bố và chỉ cần ít lần chia để cô lập hoàn toàn.

Độ sâu trung bình của việc cô lập được dùng làm thước đo bất thường, từ đó mô hình xác định các điểm có độ sâu nhỏ là các outliers.

3.4. Phát hiện & xử lý ngoại lệ

3.4.3. Phát hiện ngoại lệ bằng mô hình Isolation Forest

Cấu hình mô hình trong nghiên cứu:

```
from sklearn.ensemble import IsolationForest
iso = IsolationForest(contamination=0.05, random_state=42)
```

- Tham số contamination=0.05 giả định rằng khoảng 5% dữ liệu có thể là ngoại lệ.
- Tham số random_state=42 đảm bảo tính tái lập của kết quả.

Sau khi huấn luyện mô hình (iso.fit(X)), các điểm dữ liệu được gán nhãn bằng 1 → Dữ liệu bình thường và -1 → Dữ liệu bất thường, bằng câu lệnh bên dưới:

```
df6.loc[X.index, 'iso_flag'] = iso.predict(X)
```

Kết quả này được lưu vào một cột mới (iso_flag) trong data frame, giúp xác định nhanh các dòng bị nghi ngờ có lỗi đo hoặc giá trị bất thường. Kết quả tương tự như hình dưới.

	A	B	C	D	E	F	G	H	I	J	K
1	date	station_no	tsp	pm2.5	o3	co	no2	so2	emperatur	humidity	iso_flag
2	2021-02-24 16:00:00	6	27.24545	14.38182	49.61036	1243.41	92.47464	327.5	25.72727	62.8	1
3	2021-02-24 17:00:00	6	30.54	15.22	59.2826	1155.168	106.8976	324.88	26.7	62.1	1
4	2021-03-05 10:00:00	6	39.9204	22.10079	77.57916	782.909	8.317544	299.0231	34.15874	56.78036	-1
5	2021-03-05 11:00:00	6	41.1912	23.38633	110.335	765.719	15.05224	309.6141	34.88493	54.476	-1
6	2021-03-05 14:00:00	6	39.68524	22.33616	112.2419	814.5673	11.27578	371.1485	36.21538	46.86865	-1
7	2021-03-05 15:00:00	6	39.03251	20.56384	112.5657	859.1075	21.17409	384.4339	35.68671	47.80421	-1
8	2021-03-05 16:00:00	6	39.78467	20.67033	100.8393	1005.042	29.98653	382.6073	34.71267	47.00733	-1
9	2021-03-11 16:00:00	6	45.52433	24.47033	154.5753	974.0332	114.1355	214.4033	29.41	47.9075	-1
10	2021-03-11 17:00:00	6	29.54	13.76667	53.91707	1290.396	107.0231	349.3333	23.72	60.24	1
11	2021-03-12 10:00:00	6	57.67433	29.764	112.9379	807.8536	84.1254	96.94	27.90083	45.76083	-1
12	2021-03-12 14:00:00	6	42.504	21.40133	120.3973	642.8439	84.22734	115.1344	28.765	41.7075	-1
13	2021-03-12 15:00:00	6	65.7	23.75667	37.297	1352.089	3.136667	355.9726	33.93333	48.23167	1
14	2021-03-12 16:00:00	6	64.8	22.765	75.15018	1632.095	45.23073	355.9726	32.55333	51.50833	1
15	2021-03-12 17:00:00	6	63.23	21.50667	85.29235	2039.88	73.61757	695.61	31.55167	52.79	1
16	2021-03-12 18:00:00	6	60.96833	21.595	89.05477	3027.541	92.6885	355.9726	30.76333	53.12833	1

Hình 3.2. Cột iso_flag trong dataframe

3.4. Phát hiện & xử lý ngoại lệ

3.4.4. Kết hợp hai phương pháp và xử lý ngoại lệ

Sau khi có kết quả từ cả hai phương pháp, nhóm kết hợp lại để tạo một cờ tổng hợp:

```
st6['is_outlier'] = (st6['flag_iqr'] == True) |
(st6['iso_flag'] == -1)
```

Nếu một dòng bị đánh dấu là ngoại lệ bởi IQR hoặc Isolation Forest, dòng đó được xác định là ngoại lệ cần xử lý.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	date	station_nc	tsp	pm2.5	o3	co	no2	so2	emperatur	humidity	iso_flag	flag_iqr	is_outlier
2	2021-02-24 16:00:00	6	27.24545	14.38182	49.61036	1243.41	92.47464	327.5	25.72727	62.8	1	FALSE	FALSE
3	2021-02-24 17:00:00	6	30.54	15.22	59.2826	1155.168	106.8976	324.88	26.7	62.1	1	FALSE	FALSE
4	2021-03-05 10:00:00	6	39.9204	22.10079	77.57916	782.909	8.317544	299.0231	34.15874	56.78036	-1	FALSE	TRUE
5	2021-03-05 11:00:00	6	41.1912	23.38633	110.335	765.719	15.05224	309.6141	34.88493	54.476	-1	FALSE	TRUE
6	2021-03-05 14:00:00	6	39.68524	22.33616	112.2419	814.5673	11.27578	371.1485	36.21538	46.86865	-1	FALSE	TRUE
7	2021-03-05 15:00:00	6	39.03251	20.56384	112.5657	859.1075	21.17409	384.4339	35.68671	47.80421	-1	FALSE	TRUE
8	2021-03-05 16:00:00	6	39.78467	20.67033	100.8393	1005.042	29.98653	382.6073	34.71267	47.00733	-1	FALSE	TRUE
9	2021-03-11 16:00:00	6	45.52433	24.47033	154.5753	974.0332	114.1355	214.4033	29.41	47.9075	-1	FALSE	TRUE
10	2021-03-11 17:00:00	6	29.54	13.76667	53.91707	1290.396	107.0231	349.3333	23.72	60.24	1	FALSE	FALSE
11	2021-03-12 10:00:00	6	57.67433	29.764	112.9379	807.8536	84.1254	96.94	27.90083	45.76083	-1	FALSE	TRUE
12	2021-03-12 14:00:00	6	42.504	21.40133	120.3973	642.8439	84.22734	115.1344	28.765	41.7075	-1	FALSE	TRUE
13	2021-03-12 15:00:00	6	65.7	23.75667	37.297	1352.089	3.136667	355.9726	33.93333	48.23167	1	FALSE	FALSE
14	2021-03-12 16:00:00	6	64.8	22.765	75.15018	1632.095	45.23073	355.9726	32.55333	51.50833	1	FALSE	FALSE
15	2021-03-12 17:00:00	6	63.23	21.50667	85.29235	2039.88	73.61757	695.61	31.55167	52.79	1	FALSE	FALSE
16	2021-03-12 18:00:00	6	60.96833	21.595	89.05477	3027.541	92.6885	355.9726	30.76333	53.12833	1	FALSE	FALSE
17	2021-03-12 19:00:00	6	64.00833	22.44	87.05905	3187.408	96.8289	355.9726	30.49833	53.375	1	FALSE	FALSE
18	2021-03-12 20:00:00	6	62.24	22.51333	76.1644	2504.583	86.54063	667.2267	30.335	52.01	1	FALSE	FALSE
19	2021-03-12 21:00:00	6	62.36667	22.92833	80.02497	1766.941	90.39873	534.0433	29.47333	55.40667	1	FALSE	FALSE
20	2021-03-12 22:00:00	6	63.73	23.89	89.41465	1004.66	104.5765	364.6167	28.605	59.35667	1	FALSE	FALSE

Hình 3.3. Cột is_outlier trong dataframe

Đối với trạm 6, nhóm thay thế các giá trị ngoại lệ bằng giá trị trung bình của cùng thời điểm tính từ 5 trạm còn lại. Cách làm này giúp bảo toàn cấu trúc thời gian và tính tương quan giữa các trạm:

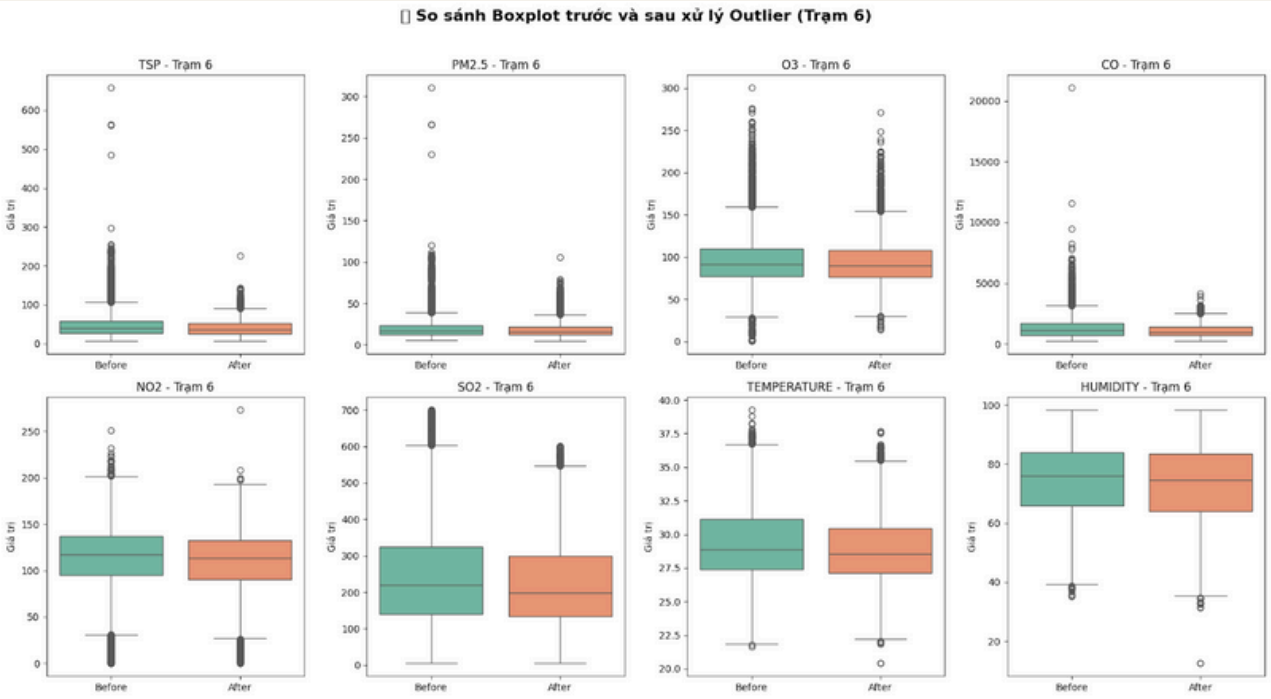
```
for col in quant_cols:
    mask = st6['is_outlier']
st6.loc[mask, col] = st6.loc[mask, 'date'].map(mean_by_time[col])
```

3.4. Phát hiện & xử lý ngoại lệ

3.4.5. Kiểm tra sau xử lý

Sau khi thực hiện các bước xử lý ngoại lệ, nhóm tiến hành trực quan hóa dữ liệu trước và sau khi làm sạch bằng biểu đồ Boxplot nhằm đánh giá trực quan hiệu quả của quy trình.

Boxplot cho phép quan sát sự thay đổi trong phân bố dữ liệu, giá trị trung vị, độ phân tán và số lượng điểm ngoại lệ giữa hai giai đoạn Before (trước xử lý) và After (sau xử lý). Sau khi thay thế, nhóm tiến hành chạy lại hai thuật toán trên để đánh giá hiệu quả làm sạch. Kết quả cho thấy số lượng điểm bất thường giảm rõ rệt, chứng tỏ quy trình xử lý hoạt động hiệu quả.



Hình 3.4. So sánh Boxplot các chỉ tiêu môi trường tại trạm 6 trước và sau khi xử lý ngoại lệ

3.4. Phát hiện & xử lý ngoại lệ

3.4.5. Kiểm tra sau xử lý

Hình 1 thể hiện rõ sự thay đổi giữa dữ liệu trước và sau khi áp dụng các phương pháp xử lý ngoại lệ bằng IQR và Isolation Forest. Nhìn chung, các hộp dữ liệu sau khi xử lý trở nên gọn hơn và cân đối hơn, cho thấy độ biến động của dữ liệu đã giảm và phân bố trở nên ổn định hơn.

Số lượng điểm ngoại lệ giảm đáng kể ở hầu hết các chỉ tiêu, đặc biệt là TSP, PM2.5, CO và SO_2, phản ánh hiệu quả của quá trình làm sạch dữ liệu. Ở hai biến Temperature và Humidity, tuy mức thay đổi không quá lớn nhưng vẫn có sự điều chỉnh rõ ràng: hộp của Temperature dịch nhẹ xuống và thu hẹp lại, còn Humidity có xu hướng nâng phần trung vị và giảm các giá trị thấp bất thường.

Đối với NO_2 và O_3, hình dạng phân bố trung tâm vẫn được giữ nguyên, nhưng các giá trị cực trị đã được loại bỏ, giúp dữ liệu trở nên nhất quán và đại diện hơn cho thực tế quan trắc.

Từ biểu đồ có thể kết luận rằng quá trình xử lý đã loại bỏ thành công các điểm dị thường mà vẫn bảo toàn cấu trúc thống kê cốt lõi của dữ liệu. Dữ liệu sau khi làm sạch trở nên ổn định hơn, phản ánh đúng thực tế hơn, và sẵn sàng sử dụng cho các bước chuẩn hóa, phân tích tương quan, cũng như mô hình hóa trong các phần tiếp theo.

3.6. Chuẩn hóa dữ liệu (Scaling)

Trước khi đưa dữ liệu vào mô hình học máy, nhóm tiến hành chuẩn hóa (scaling) nhằm đưa các biến định lượng về cùng một thang đo chuẩn, giúp quá trình huấn luyện mô hình trở nên ổn định và chính xác hơn.

Trong bộ dữ liệu nghiên cứu, có 8 biến định lượng được dùng: TSP, PM2.5, O3, CO, NO2, SO2, Temperature, Humidity.

Các biến này có đơn vị đo lường khác nhau. Nồng độ bụi TSP, PM2.5, khí CO, SO_2, NO_2 tính bằng $\mu g/m^3$, còn nhiệt độ tính bằng °C, độ ẩm tính bằng %. Việc giữ nguyên các đơn vị này khiến một số biến có độ lớn vượt trội chi phối kết quả tính toán của mô hình, dẫn đến các sai lệch không đáng có.

Để khắc phục điều này, nhóm sử dụng công cụ StandardScaler từ thư viện sklearn.preprocessing của Python. Phương pháp này chuẩn hóa dữ liệu theo công thức:

$$z = (x - \text{mean}(x)) / \text{std}(x)$$

Trong đó:

- x: giá trị gốc của biến,
- mean(x): giá trị trung bình,
- std(x): độ lệch chuẩn.

Dòng lệnh chuẩn hóa dữ liệu:

```
st6_scaled[quant_cols] =
scaler.fit_transform(st6[quant_cols])
```

3.6. Chuẩn hóa dữ liệu (Scaling)

Sau khi chuẩn hóa, mỗi biến có:

- Giá trị trung bình (mean) ≈ 0
- Độ lệch chuẩn (standard deviation) ≈ 1

📊 Thống kê mô tả sau khi chuẩn hóa:

	tsp_scaled	pm2.5_scaled	o3_scaled	co_scaled	no2_scaled	so2_scaled	temperature_scaled	humidity_scaled
count	8382.00	8382.00	8381.00	8382.00	8382.00	8381.00	8382.00	8382.00
mean	-0.00	-0.00	-0.00	0.00	0.00	-0.00	-0.00	-0.00
std	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
min	-1.71	-1.56	-3.03	-1.61	-3.17	-1.92	-3.36	-4.58
25%	-0.75	-0.68	-0.67	-0.77	-0.52	-0.81	-0.71	-0.68
50%	-0.23	-0.26	-0.17	-0.29	0.16	-0.24	-0.14	0.12
75%	0.54	0.40	0.51	0.51	0.72	0.62	0.62	0.79
max	9.03	9.66	6.67	5.30	4.82	3.24	3.47	1.92

Hình 3.5. Thống kê mô tả sau khi chuẩn hóa

Điều này giúp tất cả các biến đóng góp tương đương vào mô hình, không bị ảnh hưởng bởi sự khác biệt về đơn vị đo hoặc quy mô giá trị ban đầu.

Ngoài ra, chuẩn hóa còn giúp tăng tốc độ hội tụ trong quá trình huấn luyện và cải thiện độ ổn định số học của các thuật toán. Đặc biệt trong mô hình Isolation Forest, một mô hình vốn dựa trên việc chia nhỏ không gian dữ liệu ngẫu nhiên để cô lập các điểm bất thường, việc các đặc trưng có cùng thang đo giúp mô hình dễ dàng xác định ranh giới hợp lý giữa điểm bình thường và bất thường, tránh sai lệch do biến có giá trị tuyệt đối quá lớn.

Nhờ bước chuẩn hóa này, dữ liệu đầu vào đạt được tính cân bằng và so sánh được giữa các đặc trưng, tạo nền tảng cho các bước phân tích và mô hình hóa tiếp theo.

3.6. Loại bỏ biến không hữu ích & tạo biến mới

3.6.1. Kiểm tra và loại bỏ biến không hữu ích

Trước khi tiến hành các bước xử lý sâu hơn, nhóm đã thực hiện rà soát toàn bộ các biến trong bộ dữ liệu nhằm phát hiện những cột không mang giá trị phân tích. Tuy nhiên, kết quả kiểm tra cho thấy toàn bộ các biến trong bộ dữ liệu đều có ý nghĩa thực tiễn, phản ánh trực tiếp hoặc gián tiếp đến chất lượng không khí.

Do đó, không có biến nào bị loại bỏ ở bước này. Tất cả các cột được giữ lại để phục vụ cho việc chuẩn hóa, mô hình hóa và phân tích sau này.

3.6.2. Tạo biến mới

Nhằm bổ sung thêm thông tin bối cảnh và nâng cao khả năng phân tích xu hướng, nhóm tiến hành tạo mới hai biến đặc trưng gồm Season (Mùa) và AQI (Air Quality Index – Chỉ số chất lượng không khí).

a) Biến "Season" (Mùa)

Từ thông tin tháng trong cột date, nhóm xây dựng biến Season nhằm phản ánh đặc trưng khí hậu theo mùa của TP.HCM, với hai giai đoạn chính:

- Mùa mưa: Tháng 5 – 11;
- Mùa khô: Tháng 12 – 4.

3.6. Loại bỏ biến không hữu ích & tạo biến mới

3.6.2. Tạo biến mới

Hàm xử lý thêm biến "Season":

```
def get_season(month):
 if 5 <= month <= 11:
 return 'Rainy'
else:
      return 'Dry'
df['Season'] = df['date'].dt.month.apply(get_season)
```

Hàm xử lý trên sẽ tạo thêm 1 cột mới trong dataframe tên "Season", nếu tháng trong cột date là từ tháng 5 đến tháng 11 thì sẽ trả về kết quá là Rainy (mùa mưa), các tháng còn lại là Dry (mùa khô).

Việc bổ sung biến Season như thế này cho phép nhóm phân tích xu hướng biến động nồng độ bụi và khí độc theo mùa cũng như đánh giá xem liệu ô nhiễm tăng cao trong mùa khô hay giảm trong mùa mưa. Bên cạnh đó, việc này cũng giúp người đọc nhận biết các yếu tố thời tiết ảnh hưởng đến chất lượng không khí.

3.6. Loại bỏ biến không hữu ích & tạo biến mới

3.6.2. Tạo biến mới

b) Các biến "AQI" (Air Quality Index – Chỉ số chất lượng không khí)

Để đánh giá tổng hợp mức độ ô nhiễm tại mỗi thời điểm, nhóm tính toán biến AQI dựa theo Quyết định 1459/QĐ-TCMT (2019) của Bộ Tài nguyên & Môi trường. Theo Quyết định 1459/QĐ-TCMT (2019) của Việt Nam, chỉ số AQI chính thức không tính cho TSP, mà chỉ cho: PM2.5, PM10, SO_2, NO_2, CO, O_3 [4].

Cụ thể:

- **AQI giờ** được tính riêng cho từng chất ô nhiễm chính gồm: aqi_pm2.5, aqi_pm10, aqi_so2, aqi_no2, aqi_co, và aqi_o3.
- **AQI giờ tổng** ký hiệu là aqi_h_total.
- **AQI ngày** ký hiệu lần lượt là: pm2.5_aqi_day, pm10_aqi_day, so2_aqi_day, no2_aqi_day, co_aqi_day, o3_aqi_day.
- **AQI ngày tổng** ký hiệu là aqi_daily_total.

Công thức tính dựa trên nội suy tuyến tính giữa các khoảng giá trị chuẩn của từng chất. Sau đó, AQI tổng được xác định bằng giá trị lớn nhất trong sáu chỉ số thành phần – tức là chỉ số của chất ô nhiễm nghiêm trọng nhất tại thời điểm đó.

3.6. Loại bỏ biến không hữu ích & tạo biến mới

3.6.2. Tạo biến mới

b) Các biến "AQI" (Air Quality Index – Chỉ số chất lượng không khí)

Với mỗi chất ô nhiễm x (PM2.5, SO_2, NO_2, CO, O_3), khi nồng độ C nằm trong khoảng [BPi; BPi + 1] tương ứng với mức i, giá trị AQI thành phần được tính bằng nội suy tuyến tính:

$$AQI_x = \frac{I_{i+1} - I_i}{BP_{i+1} - BP_i} \times (C_x - BP_i) + I_i$$

Trong đó:

- **AQIx**: Giá trị AQI thông số của thông số x,
- **BPi**: Nồng độ giới hạn dưới của giá trị thông số quan trắc được quy định trong Bảng 2 tương ứng với mức i,
- **BPi + 1**: Nồng độ giới hạn trên của giá trị thông số quan trắc được quy định trong Bảng 2 tương ứng với mức i+1,
- **Ii**: Giá trị AQI ở mức i đã cho trong bảng tương ứng với giá trị BPi,
- **Ii + 1**: Giá trị AQI ở mức i+1 cho trong bảng tương ứng với giá trị BPi + 1,
- **Cx**: Giá trị quan trắc trung bình 1 giờ của thông số x.

3.6. Loại bỏ biến không hữu ích & tạo biến mới

3.6.2. Tạo biến mới

b) Các biến "AQI" (Air Quality Index – Chỉ số chất lượng không khí)

i	I_i	Giá trị BP_i quy định đối với từng thông số (*Đơn vị: $\mu g/m^3$*)						
		O_3(1h)	O_3(8h)	CO	SO_2	NO_2	PM_{10}	$PM_{2.5}$
1	0	0	0	0	0	0	0	0
2	50	160	100	10.000	125	100	50	25
3	100	200	120	30.000	350	200	150	50
4	150	300	170	45.000	550	700	250	80
5	200	400	210	60.000	800	1.200	350	150
6	300	800	400	90.000	1.600	2.350	420	250
7	400	1.000	-	120.000	2.100	3.100	500	350
8	500	≥1.200	-	≥150.000	≥2.630	≥3.850	≥600	≥500

Ghi chú: - Tính toán AQI giờ (AQI^h) của O_3 sử dụng giá trị BPi cho O_3 (1h)

Bảng 3.1. Bảng các giá trị của BPi đối với các thông số trong Quyết định 1459/QĐ-TCMT

3.6. Loại bỏ biến không hữu ích & tạo biến mới

3.6.2. Tạo biến mới

b) Các biến "AQI" (Air Quality Index – Chỉ số chất lượng không khí)

Cách tính AQI giờ (AQI^H): Với dữ liệu đo theo giờ, ta có thể tính AQI giờ cho từng chất bằng cách áp công thức nội suy trên giá trị nồng độ quan trắc trong từng giờ. Theo đó, ta có xử lý theo công thức nội suy:

```
BP = {
    'pm2.5': [0, 25, 50, 80, 150, 250, 350, 500],
    'so2': [0, 125, 350, 550, 800, 1600, 2100, 2630],
    'no2': [0, 100, 200, 700, 1200, 2350, 3100, 3850],
    'co':  [0, 10000, 30000, 45000, 60000, 90000, 120000, 150000],
    'o3':  [0, 160, 200, 300, 400, 800, 1000, 1200]
}
I = [0, 50, 100, 150, 200, 300, 400, 500]
def calc_aqi_substance(Cp, pollutant):
    bp = BP[pollutant]
    for i in range(len(bp) - 1):
        if bp[i] <= Cp <= bp[i + 1]:
            Il, Ih = I[i], I[i + 1]
            BPl, BPh = bp[i], bp[i + 1]
            Ia = (Ih - Il) / (BPh - BPl) * (Cp - BPl) + Il
            return round(Ia, 0)
    if Cp > bp[-1]:
        return 500
    return np.nan
for p in available_cols:
      df[f'aqi_{p}'] = df[p].apply(lambda x: calc_aqi_substance(x,
p))
```

Hàm xử lý này được sử dụng để chuyển đổi giá trị nồng độ đo được của từng chất ô nhiễm sang chỉ số AQI tương ứng dựa trên Quyết định 1459. Hàm thực hiện kiểm tra dữ liệu, xác định khoảng giá trị chuẩn, và áp dụng phép nội suy tuyến tính để tính toán AQI chi tiết cho từng chất. Kết quả được làm tròn và giới hạn trong khoảng từ 0 – 500.

3.6. Loại bỏ biến không hữu ích & tạo biến mới

3.6.2. Tạo biến mới

b) Các biến "AQI" (Air Quality Index – Chỉ số chất lượng không khí)

Cách tính AQI giờ tổng: Sau khi đã có giá trị AQIx của mỗi thông số, chọn giá trị AQI lớn nhất của các thông số để lấy làm giá trị AQI giờ tổng hợp. Theo đó, ta có lần lượt công thức và xử lý như bên dưới:

$$AQI^{H} = \max(AQIx)$$

```
df['aqi_h_total'] = df[[f'aqi_{p}' for p in
available_cols]].max(axis=1)
```

Lệnh trên được sử dụng để xác định chỉ số AQI tổng theo giờ. Cụ thể, chương trình lấy giá trị AQI của từng chất ô nhiễm (PM2.5, PM10, O_3, CO, NO_2, v.v.) và chọn giá trị lớn nhất làm AQI tổng hợp. Điều này phản ánh đúng nguyên tắc tính AQI trong Quyết định 1459, khi mức ô nhiễm cao nhất của một chất sẽ đại diện cho chất lượng không khí chung tại thời điểm đó.

3.6. Loại bỏ biến không hữu ích & tạo biến mới

3.6.2. Tạo biến mới

b) Các biến "AQI" (Air Quality Index – Chỉ số chất lượng không khí)

Cách tính AQI ngày (AQI^d):

- **PM2.5, PM10**: dùng giá trị trung bình 24 giờ (24-hour mean) để tính AQI^d cho thành phần này.
- **O_3, SO_2, NO_2, CO**: thường sử dụng giá trị cực đại 1 giờ trong 24 giờ (max 1-h) để làm căn cứ tính AQI^d cho các chất này.
- **Điều kiện số liệu tối thiểu**: nếu ngày đó không có đủ số liệu (ví dụ < 18 bản ghi trong 24 giờ cho dạng đo giờ), thì giá trị AQI ngày cho chất đó bị coi là không đủ điều kiện để tính, do đó kết quả AQI ngày trong các ô đó sẽ bị trả về là ô trống.

Theo đó, ta có hàm xử lý AQI ngày như sau:

```
available_cols = [c for c in cols if c in df.columns]
...
for col in available_cols:
    if col == 'pm2.5':
        Cp_day = daily_aqi[f"{col}_mean"]
    else:
        Cp_day = daily_aqi[f"{col}_max"]
    valid = daily_aqi[f"{col}_count"] >= 18
            daily_aqi[f"aqi_{col}_day"]   =   np.where(valid,
Cp_day.apply(lambda x: calc_aqi(x, col)), np.nan)
```

Hàm này hoạt động bằng cách xác định khoảng mà giá trị nồng độ C nằm trong, sau đó nội suy tuyến tính để tính ra chỉ số AQI cụ thể của chất đó. Nếu nồng độ vượt quá giới hạn cao nhất, chỉ số được giới hạn ở mức tối đa 500.

3.6. Loại bỏ biến không hữu ích & tạo biến mới

3.6.2. Tạo biến mới

b) Các biến "AQI" (Air Quality Index – Chỉ số chất lượng không khí)

Cách tính AQI ngày tổng: Sau khi đã có giá trị AQIx ngày của mỗi thông số, chọn giá trị AQI lớn nhất của các thông số để lấy làm giá trị AQI ngày tổng hợp. Dưới lần lượt là công thức tính của AQI ngày tổng và code xử lý.

AQI^d = max(AQIx), giá trị AQI ngày được làm tròn thành số nguyên.

```
aqi_day_cols = [f'aqi_{c}_day' for c in available_cols]
daily_aqi['aqi_daily_total'] =
daly_aqi[aqi_day_cols].max(axis=1)
```

Những dòng lệnh trên đầu tiên sẽ tạo ra danh sách tên các cột AQI ngày của từng chất ô nhiễm, sau đó sẽ lấy giá trị AQI lớn nhất trong ngày giữa các chất ô nhiễm để xác định AQI đại diện cho toàn bộ khu vực trong ngày đó.

Việc tạo các biến AQI mang lại nhiều lợi ích, một trong số đó là cung cấp một chỉ tiêu tổng hợp, giúp đánh giá chất lượng không khí nhanh và trực quan hơn. Bên cạnh đó, nó cũng cho phép so sánh giữa các thời điểm, các khu vực hoặc giữa các mùa khí hậu.

3.7. Kiểm tra tương quan

3.7.1. Dữ liệu và phương pháp

Bộ dữ liệu dùng để phân tích tương quan là tập sau tiền xử lý cuối cùng **Station6_AQI_Daily_with_Season.xlsx**, đã chuẩn hoá thời gian, gán mùa, đánh dấu ngoại lệ đo đạc (iso_flag) và tổng hợp chỉ số AQI theo ngày. Tập này gồm **9.499 quan sát và 43 biến (30 biến định lượng)**, thu thập trong giai đoạn **04/01/2021–06/12/2022** tại **một trạm duy nhất (trạm 6)**.

Kiểm tra giá trị khuyết cho thấy các biến đo trực tiếp từ cảm biến **(TSP, PM2.5, CO, SO_2, NO_2, O_3, nhiệt độ, độ ẩm)** gần như không thiếu dữ liệu **(≈0,01–0,03%)**, trong khi các chỉ số AQI theo ngày **(aqi_pm2.5_day, aqi_no2_day, aqi_daily_total,...)** thiếu khoảng **7,32%**. Vì vậy, các phân tích tương quan và đa cộng tuyến tập trung vào nhóm biến đo gốc này.

Các biến đo trực tiếp từ cảm biến (**TSP, PM2.5, CO, SO_2, NO_2, O_3, nhiệt độ, độ ẩm, iso_flag)** được tính hệ số tương quan Spearman và Pearson, sau đó biểu diễn dưới dạng bản đồ nhiệt (heatmap) để quan sát nhanh mức độ cùng biến động giữa các cặp biến.

3.7. Kiểm tra tương quan

Mở đầu

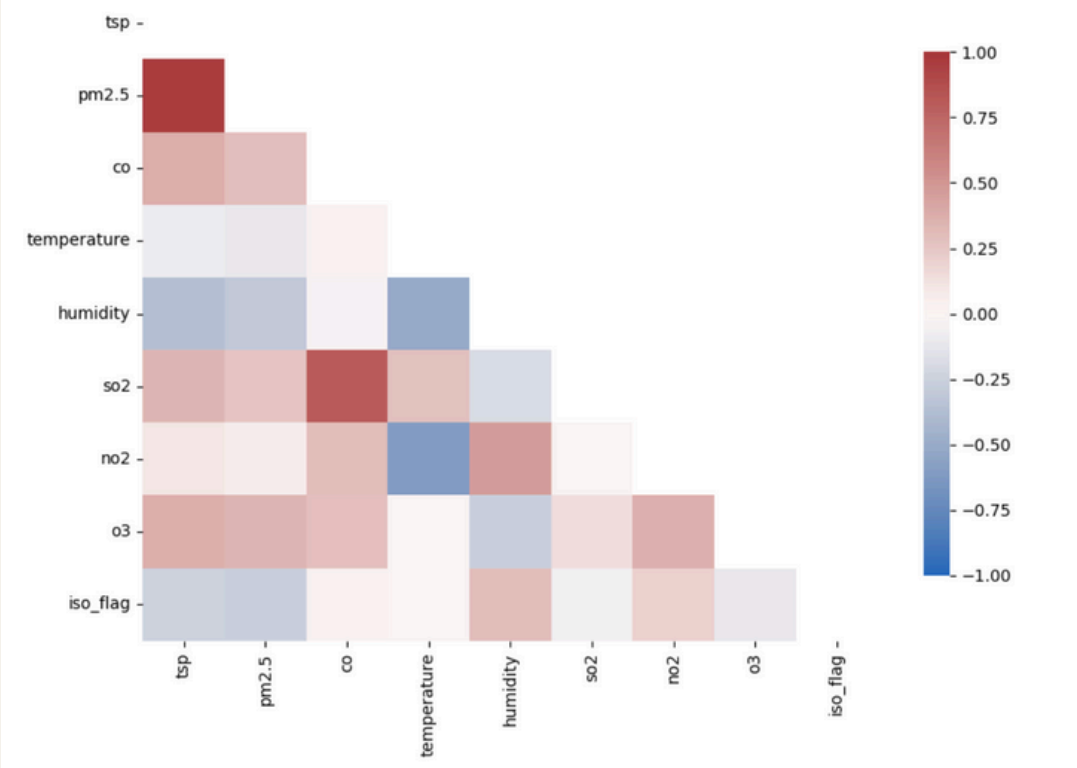
Chất lượng không khí tại trạm chịu tác động đồng thời của bụi **(PM2.5, TSP)**, các khí ô nhiễm **(CO, SO_2, NO_2)**, ozone thứ cấp **(O_3)** và điều kiện khí tượng **(nhiệt độ, độ ẩm)**. Các biến này thường biến động cùng nhau theo nguồn phát thải và thời tiết, nên cần kiểm tra chúng phụ thuộc lẫn nhau đến mức nào trước khi phân tích mô hình.

Phần 4.7 tập trung vào hai mục tiêu chính: (1) nhận diện các cặp biến gần như mang cùng tín hiệu (như TSP và PM2.5) dễ gây **đa cộng tuyến**; (2) kiểm tra xem các quan hệ quan sát được là quan hệ thực chất hay chỉ do **mùa vụ và xu hướng thời gian**.

Dữ liệu sử dụng là các biến đo trực tiếp sau tiền xử lý **(TSP, PM2.5, CO, SO_2, NO_2, O_3, nhiệt độ, độ ẩm, iso_flag)**, được dùng để tính ma trận tương quan, lặp lại tương quan sau khi loại mùa vụ và xu hướng, và cuối cùng đánh giá đa cộng tuyến bằng hệ số phóng đại phương sai **(VIF)**.

3.7. Kiểm tra tương quan

3.7.2. Ma trận tương quan (Heatmap)

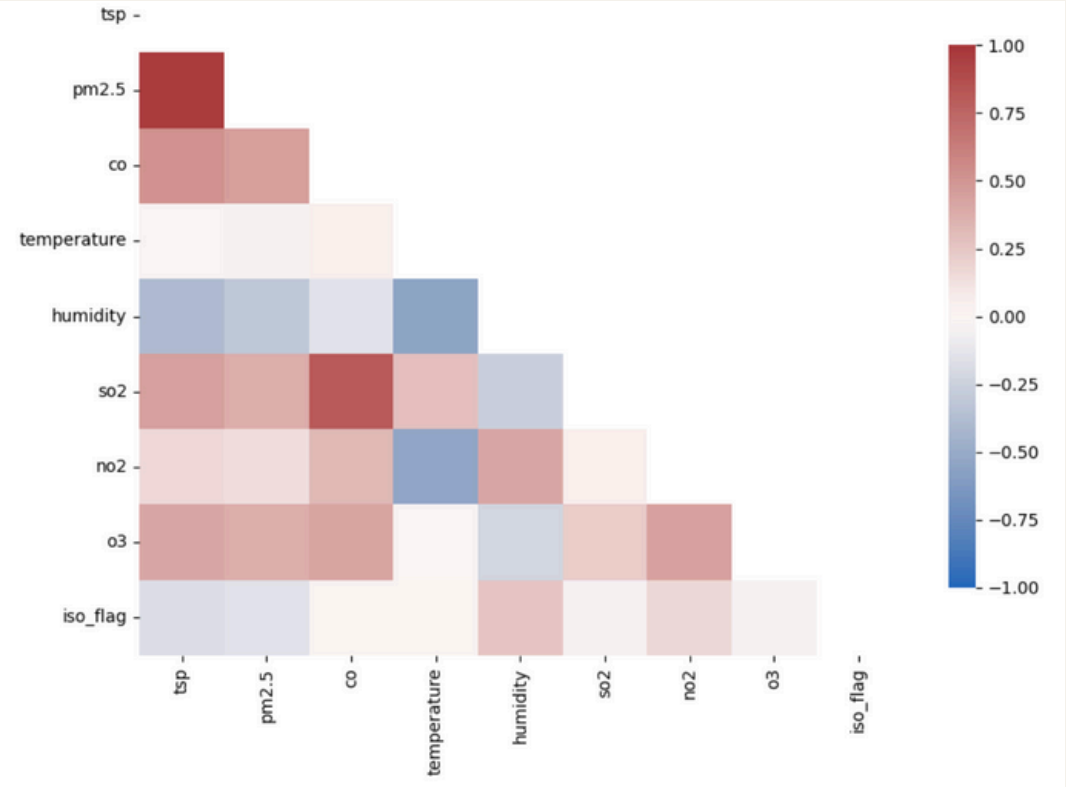


Hình 3.6. Ma trận tương quan Pearson giữa các biến đo ô nhiễm và khí tượng

3.7. Kiểm tra tương quan

3.7.2. Ma trận tương quan (Heatmap)

Để đánh giá mối quan hệ tuyến tính và đơn điệu giữa các biến đo lường chất lượng không khí **(bụi TSP, PM2.5; khí CO, SO_2, NO_2, O_3)** và các yếu tố khí tượng **(nhiệt độ, độ ẩm)**, chúng tôi tiến hành phân tích ma trận tương quan. Phân tích này sử dụng cả hai hệ số tương quan **Pearson** và tương quan **Spearman** nhằm kiểm tra cả mối quan hệ tuyến tính lẫn mối quan hệ đơn điệu tổng quát giữa các cặp biến. Kết quả tương quan được trực quan hóa bằng biểu đồ bản đồ nhiệt **(Heatmap)** chỉ hiển thị tam giác dưới, giúp dễ dàng nhận biết cường độ và chiều hướng của các mối tương quan, làm cơ sở quan trọng cho các bước phân tích mô hình tiếp theo.



Hình 3.7. Ma trận tương quan Spearman giữa các biến đo ô nhiễm và khí tượng

3.7. Kiểm tra tương quan

3.7.2. Ma trận tương quan (Heatmap)

Kết quả cho thấy:

TSP và PM2.5 có tương quan dương rất mạnh → hai biến này gần như mang cùng một tín hiệu bụi trong không khí: khi bụi tổng tăng thì bụi mịn PM2.5 cũng tăng theo.

CO và SO_2 cũng có tương quan dương rõ rệt → gợi ý chúng thường cùng xuất hiện ở mức cao tại cùng thời điểm, phù hợp với giả thuyết cùng đến từ các nguồn đốt nhiên liệu (giao thông, công nghiệp…).

Nhiệt độ và độ ẩm tương quan âm đáng kể: nhiệt độ cao thì độ ẩm thường giảm; mối quan hệ này cũng xuất hiện giữa độ ẩm và NO_2, cho thấy NO_2 có xu hướng cao hơn trong điều kiện không khí khô.

Kết quả **Pearson và Spearman** cho cấu trúc rất giống nhau (các cặp mạnh/yếu tương tự), nghĩa là các quan hệ trên nhất quán và không chỉ do giả định tuyến tính; các cặp biến đồng biến/nghịch biến này không phải chủ yếu do nhiễu của mùa vụ hay xu hướng dài hạn.

3.7. Kiểm tra tương quan

3.7.3. Các cặp biến tương quan cao (|r| ⩾ 0,8)

Trong giai đoạn tiền xử lý mô hình, mục tiêu là tìm các cặp biến có tương quan tuyệt đối rất cao để nhận diện những biến gần như mang cùng một tín hiệu. Nhóm đặt ngưỡng **|r| ≥ 0,80** trên ma trận tương quan Pearson và kiểm tra lại với Spearman, đồng thời chỉ lấy một phía tam giác trên của ma trận để tránh đếm trùng cặp A–B và B–A.

```
[HIGH-CORR] số cặp |r| ≥ 0.8: 2
```

	var1	var2	r_pearson	r_spearman
0	tsp	pm2.5	0.962	0.971
1	co	so2	0.807	0.813

Hình 3.8. Các cặp biến có tương quan tuyệt đối cao (|r| ≥ 0,8)

Kết quả: chỉ có **2 cặp biến vượt ngưỡng |r| ≥ 0,8**

TSP – PM2.5: r_P ≈ **0,962**, r_S ≈ **0,971** → hai biến gần như cùng mô tả bụi trong không khí; khi bụi tổng tăng thì bụi mịn PM2.5 cũng tăng gần như đồng thời.

CO – SO_2: **r_P ≈ 0,807**, **r_S ≈ 0,813** → thường tăng cùng lúc, gợi ý khả năng chung nguồn phát thải (đốt nhiên liệu, giao thông, công nghiệp…).

Những cặp biến này mang thông tin gần trùng nhau; nếu đưa cả hai biến vào cùng một mô hình hồi quy đa biến rất dễ gây **đa cộng tuyến mạnh**, nên cần xem xét lựa chọn bớt hoặc biến đổi biến.

3.7. Kiểm tra tương quan

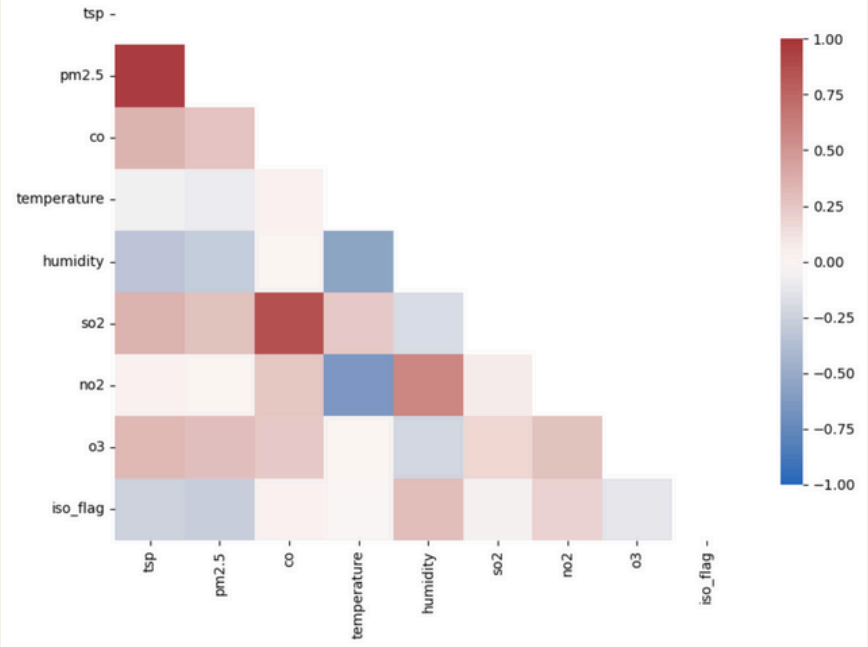
3.7.4. Tương quan sau khi loại bỏ yếu tố mùa vụ và xu hướng thời gian

Trong quá trình phân tích chuỗi thời gian, mỗi biến đo được tách thành hai phần: (1) **thành phần mùa vụ/tháng và xu hướng theo thời gian** – ví dụ chu kỳ mưa/khô lặp lại hằng năm hoặc xu thế tăng/giảm dài hạn; và (2) phần dư sau khi loại bỏ hai thành phần đó. Nói cách khác, ta "khử" bớt ảnh hưởng của thời tiết và thời gian để chỉ còn lại phần biến động ngắn hạn mang tính "thuần" của từng biến ô nhiễm.

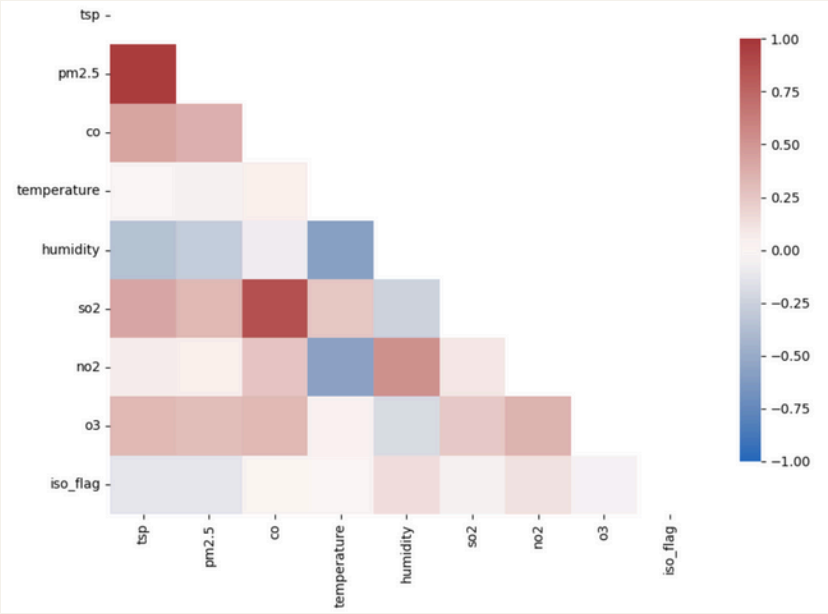
Trên phần dư này, ma trận tương quan **Pearson** và **Spearman** được tính lại, giúp so sánh trực tiếp cấu trúc quan hệ giữa các biến **trước khi khử** và **sau khi khử**. Nếu sau khi loại bỏ mùa vụ/xu hướng mà cặp biến vẫn còn tương quan rất cao thì có thể xem đó là **quan hệ thực chất giữa các biến**, không chỉ do cùng chịu tác động của mùa mưa/khô hay xu thế thời gian.

3.7. Kiểm tra tương quan

3.7.4. Tương quan sau khi loại bỏ yếu tố mùa vụ và xu hướng thời gian



Hình 3.9. Tương quan Pearson sau khi loại bỏ mùa vụ và xu hướng thời gian.



Hình 3.10. Tương quan Spearman sau khi loại bỏ mùa vụ và xu hướng thời gian.

3.7. Kiểm tra tương quan

3.7.4. Tương quan sau khi loại bỏ yếu tố mùa vụ và xu hướng thời gian

```
[SUMMARY] |r| ≥ 0.8: TRƯỚC=2  |  SAU=2
[VẪN ≥ THR SAU KHỬ] (ưu tiên xử lý/giữ 1 đại diện nếu cần)
```

	var1	var2	r_after	r_before	delta
0	tsp	pm2.5	0.96	0.962	-0.002
1	co	so2	0.86	0.807	0.053

```
[TOP GIẢM MẠNH SAU KHỬ]
```

	var1	var2	r_before	r_after	delta
0	no2	o3	0.366	0.277	-0.089
1	tsp	no2	0.106	0.036	-0.070
2	pm2.5	no2	0.070	0.007	-0.062
3	tsp	o3	0.373	0.326	-0.047
4	co	o3	0.289	0.244	-0.045
5	pm2.5	o3	0.339	0.297	-0.042
6	co	no2	0.298	0.257	-0.041
7	co	humidity	-0.035	0.002	-0.033
8	tsp	humidity	-0.364	-0.332	-0.032
9	humidity	o3	-0.265	-0.233	-0.032

Hình 3.11. So sánh tương quan trước và sau khi loại bỏ mùa vụ / xu hướng.

3.7. Kiểm tra tương quan

3.7.4. Tương quan sau khi loại bỏ yếu tố mùa vụ và xu hướng thời gian

Kết luận:

Hai cặp **TSP – PM2.5** và **CO – SO_2** vẫn giữ tương quan rất cao sau khi khử mùa vụ/xu hướng (khoảng **0,96** và **0,86**) → đây là **quan hệ thật giữa các biến**, không chỉ do mùa mưa/khô hay xu hướng thời gian.

Các cặp khác (ví dụ **NO_2 – O_3, độ ẩm với các chất ô nhiễm**) giảm tương quan đáng kể sau khi khử, cho thấy phần lớn tương quan ban đầu chủ yếu phản ánh **yếu tố thời tiết/mùa** hơn là mối quan hệ trực tiếp giữa các biến.

Vì vậy, **TSP – PM2.5** và **CO – SO_2** là những cặp dễ gây **đa cộng tuyến mạnh** nếu đưa cả hai biến vào cùng một mô hình; khi xây dựng mô hình nên ưu tiên chọn **một biến đại diện** cho mỗi cặp.

3.7. Kiểm tra tương quan

3.7.5. Đa cộng tuyến (VIF & Condition Index)

Nhằm đảm bảo tính ổn định và độ tin cậy của mô hình hồi quy, nhóm đánh giá xem các biến đầu vào có "trùng thông tin" với nhau nhiều hay không, tức là kiểm tra **đa cộng tuyến** trong mô hình. Nếu nhiều biến gần như mang cùng một tín hiệu, mô hình hồi quy sẽ khó ước lượng chính xác, hệ số dễ dao động mạnh và suy luận trở nên kém tin cậy.

Để đo lường, nhóm sử dụng hai thước đo bổ sung cho nhau: **VIF (Variance Inflation Factor)** cho từng biến riêng lẻ và **Condition Index (CI)** cho cả hệ biến. VIF phản ánh mức độ mà phương sai của hệ số hồi quy bị "thổi phồng" do biến đó tương quan với các biến còn lại (VIF càng cao → biến càng trùng lặp thông tin); thường **VIF > 5** được xem là **dấu hiệu cảnh báo**, **VIF > 10** là **đa cộng tuyến mạnh**. Condition Index thì tổng quát hơn, dựa trên các eigenvalue của ma trận tương quan; **CI > 30** thường được xem là mức **đa cộng tuyến nghiêm trọng**, còn CI trong khoảng **15–30** là **cần lưu ý.**

3.7. Kiểm tra tương quan

3.7.5. Đa cộng tuyến (VIF & Condition Index)

```
=== VIF ( >5 cảnh báo, >10 nghiêm trọng ) ===
```

	variable	VIF	R2
0	tsp	17.939	0.944
1	pm2.5	15.840	0.937
2	co	4.221	0.763
5	so2	3.944	0.746
6	no2	3.017	0.669
4	humidity	2.460	0.594
3	temperature	2.397	0.583
7	o3	1.758	0.431
8	iso_flag	1.245	0.197

```
=== Condition Index ( >30 nghiêm trọng; 15–30 lưu ý ) ===
```

	Eigenvalue	ConditionIndex
0	2.9699	1.00
1	2.1643	1.17
2	1.4855	1.41
3	0.9126	1.80
4	0.7848	1.95
6	0.3051	3.12
7	0.2172	3.70
8	0.1307	4.77
5	0.0299	9.97

```
[SUMMARY] CI_max=9.97 | VIF_max=17.94
```

Hình 3.12. Bảng kiểm tra đa cộng tuyến: VIF theo biến và Condition Index toàn hệ

3.7. Kiểm tra tương quan

3.7.5. Đa cộng tuyến (VIF & Condition Index)

Kết quả:

TSP và PM2.5 có VIF rất cao (**≈17,9 và ≈15,8**, lớn hơn 10) → hai biến gần như trùng tín hiệu bụi, nếu đưa cùng lúc vào mô hình sẽ dễ gây **đa cộng tuyến mạnh.**

Các biến còn lại có VIF thấp hơn rõ rệt (đa số < 5) → mức độ chấp nhận được, ít trùng lặp thông tin.

Condition Index tối đa chỉ khoảng 10 (< 30) → cấu trúc đa cộng tuyến tổng thể của mô hình không nghiêm trọng, vấn đề chủ yếu nằm ở cặp **TSP – PM2.5.**

Khi xây dựng mô hình, nên **chọn một biến đại diện** cho tín hiệu bụi (TSP hoặc PM2.5), tránh đưa cả hai biến này vào cùng một phương trình.

Chương 4:

4.1. Đánh giá tổng quan

4.2. Đánh giá các thành phần gây ô nhiễm không khí

4.3. Xu hướng AQI theo thời gian

4.4. Tương quan

4.1. Đánh giá tổng quan

4.1.1. Đánh giá chất lượng không khí tổng thể (theo giá trị AQI trung bình hàng ngày)

Đặt vấn đề: Một trong những cách tiếp cận cơ bản khi phân tích chất lượng không khí là theo dõi sự biến động của chỉ số AQI (Air Quality Index) theo thời gian. Giá trị AQI trung bình hằng ngày phản ánh mức độ ô nhiễm tổng thể của môi trường không khí tại khu vực nghiên cứu, đồng thời giúp xác định các giai đoạn ô nhiễm tăng cao hoặc cải thiện đáng kể. Câu hỏi được đặt ra là: "*Chỉ số AQI trung bình hằng ngày tại TP.HCM thay đổi như thế nào trong giai đoạn 2021-2022? Chất lượng không khí có thường xuyên vượt ngưỡng trung bình hoặc kém hay không?*"

Để trả lời câu hỏi đó, nhóm sử dụng biểu đồ boxplot. Nhờ đó, nhóm có thể đánh giá được mức độ ổn định của chất lượng không khí, và phát hiện những thời điểm có ô nhiễm tăng đột biến. Quy trình xử lý và trực quan hóa dữ liệu gồm 3 bước chính sau: (1) Tạo cột month, (2) Trực quan hóa dữ liệu bằng biểu đồ Boxplot.

(1) Tạo cột month

```
df['month'] = df['date_only'].dt.to_period('M').astype(str)
df = df.dropna(subset=['aqi_daily_total'])
df = df.sort_values('date_only')
```

Các dòng lệnh này tạo thêm một cột mới tên là month, trích xuất phần tháng-năm từ cột date_only bằng phương thức .dt.to_period('M'). Việc chuyển sang dạng chuỗi (astype(str)) giúp dễ dàng hiển thị hoặc nhóm dữ liệu theo từng tháng khi vẽ biểu đồ. Ngoài ra, còn giúp loại bỏ các giá trị AQI bị thiếu để tránh sai lệch trong thống kê và sắp xếp dữ liệu theo thứ tự thời gian, giúp biểu đồ hiển thị các tháng đúng trình tự. Nhờ đó, dữ liệu đầu vào được đảm bảo đầy đủ, chính xác và có cấu trúc phù hợp cho việc phân tích.

4.1. Đánh giá tổng quan

4.1.1. Đánh giá chất lượng không khí tổng thể (theo giá trị AQI trung bình hàng ngày)

(2) Trực quan hóa bằng biểu đồ

Sau khi xử lý dữ liệu, nhóm tiến hành vẽ biểu đồ thể hiện xu hướng biến động AQI hàng ngày theo tháng:

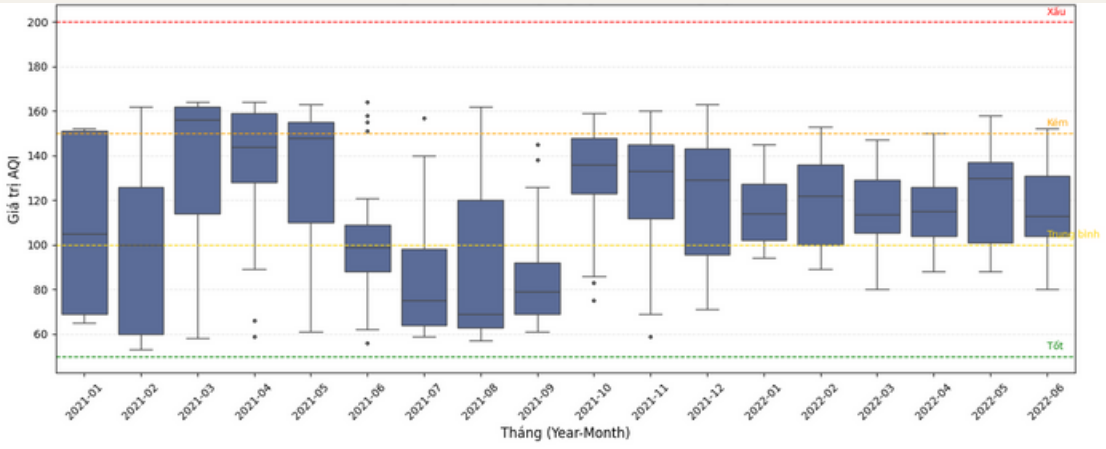
```
sns.boxplot(data=df,x='month',y='aqi_daily_total',
color='#5167a2', fliersize=2)
```

Trục hoành (x) biểu diễn các tháng, còn trục tung (y) thể hiện giá trị AQI trung bình hàng ngày. Mỗi hộp (box) thể hiện phân bố dữ liệu trong tháng gồm: giá trị trung vị (median), khoảng tứ phân vị (IQR) và các điểm bất thường (outliers).

Bên cạnh đường biểu diễn chính, biểu đồ còn được bổ sung các đường ngang tại các ngưỡng giá trị 50, 100, 150 và 200, tương ứng với các mức phân loại chất lượng không khí do Bộ Tài nguyên và Môi trường quy định. Cụ thể, vùng có giá trị AQI ≤ 50 được xem là tốt (màu xanh), từ 51 đến 100 là trung bình (màu vàng), từ 101 đến 150 là kém (màu cam), và trên 150 là xấu (màu đỏ). Việc thêm các đường tham chiếu này giúp người xem dễ dàng so sánh giá trị AQI trung bình từng tháng với các ngưỡng chuẩn, qua đó nhanh chóng nhận biết được mức độ ô nhiễm không khí tại từng thời điểm trong năm. Cách trình bày này làm tăng tính trực quan của biểu đồ và hỗ trợ quá trình phân tích, đánh giá tổng thể chất lượng môi trường đô thị.

4.1. Đánh giá tổng quan

4.1.1. Đánh giá chất lượng không khí tổng thể (theo giá trị AQI trung bình hàng ngày)



Biểu đồ 4.1. Phân bổ giá trị AQI hàng ngày theo tháng

Trong năm 2021, đặc biệt là các tháng đầu năm (từ tháng 1 đến tháng 5), giá trị AQI thường xuyên vượt ngưỡng 150 – mức được phân loại là "Kém", cho thấy chất lượng không khí giai đoạn này khá xấu. Tuy nhiên, từ giữa đến cuối năm 2021, giá trị AQI có xu hướng giảm dần và ổn định hơn, chủ yếu nằm trong vùng "Trung bình" (100–150), phản ánh sự cải thiện nhẹ về chất lượng không khí.

Bước sang năm 2022, chất lượng không khí có dấu hiệu cải thiện nhẹ so với cùng kỳ năm trước. Mặc dù trung vị AQI các tháng đầu năm vẫn nằm trong vùng "Trung bình", nhưng phạm vi dao động hẹp hơn và ít giá trị cực đoan hơn. Các tháng 4–6/2022 nhìn chung duy trì mức AQI ổn định quanh ngưỡng 100–130, tức là không khí ở mức chấp nhận được, tuy chưa thật sự tốt.

Tổng thể, biểu đồ cho thấy TP.HCM thường xuyên duy trì chất lượng không khí ở mức trung bình đến kém, đặc biệt vào các tháng mùa khô. Chỉ số AQI hiếm khi đạt mức "Tốt" và đôi khi tiến gần đến ngưỡng "Xấu", phản ánh thực trạng ô nhiễm không khí đáng lo ngại của thành phố trong giai đoạn này.

4.1. Đánh giá tổng quan

4.1.2. So sánh mức AQI trung bình 6 tháng đầu năm của TP.HCM giữa năm 2021 và 2022

Đặt vấn đề: Chỉ số chất lượng không khí (AQI) là một chỉ số báo cáo tổng hợp phản ánh mức độ ô nhiễm không khí và tác động của các chất ô nhiễm như bụi mịn ($PM_{2.5}$), khí NO_2, SO_2, CO, O_3. Trong phạm vi dữ liệu thu thập, nhóm chỉ có số liệu đến tháng 6 năm 2022, do đó tập trung so sánh 6 tháng đầu năm (tháng 1–6) của hai năm 2021 và 2022 để xác định sự thay đổi chất lượng không khí sau đại dịch COVID-19, khi các hoạt động sản xuất và giao thông được khôi phục. Câu hỏi đặt ra là:"*Mức AQI trung bình trong 6 tháng đầu năm 2022 có cao hơn so với cùng kỳ 2021 không?*"

Nhóm lựa chọn biểu đồ đường (line chart) đặt song song cho hai năm 2021 và 2022. Loại biểu đồ này được sử dụng vì có khả năng thể hiện rõ xu hướng biến động theo thời gian và so sánh trực quan giữa hai chuỗi dữ liệu có cùng cấu trúc (các tháng 1–6). Việc trình bày hai biểu đồ cạnh nhau giúp dễ dàng nhận thấy sự khác biệt về mức độ và độ dao động của AQI giữa hai năm. Quy trình xử lý và trực quan hóa dữ liệu gồm 2 bước chính: (1) Tạo 2 cột mới year và month_num, (2) Trực quan hóa dữ liệu bằng biểu đồ đường (Line chart).

4.1. Đánh giá tổng quan

4.1.2. So sánh mức AQI trung bình 6 tháng đầu năm của TP.HCM giữa năm 2021 và 2022

(1) Tạo 2 cột mới year và month_num

Dòng lệnh trên tạo thêm cột mới là year (năm) từ cột ngày gốc date. Việc tách riêng năm giúp dễ dàng lọc, nhóm, và trực quan hóa dữ liệu theo từng giai đoạn cụ thể, đặc biệt thuận tiện cho việc so sánh giữa hai năm 2021 và 2022.

```
df['year'] = df['date'].dt.year
```

Tiếp theo, nhóm tạo thêm một cột mới tên là month_num, chứa giá trị tháng (từ 1 đến 12) được tách ra từ cột date. Đồng thời lọc dữ liệu, chỉ giữ lại các bản ghi thuộc 6 tháng đầu năm và tính giá trị trung bình chỉ số AQI theo tháng và năm:

```
df['month_num'] = df['date'].dt.month
df = df[df['month_num'] <= 6]
df_monthly_year=df.groupby(['year','month_num'],as_index=False)
['aqi_daily_total'].mean()
```

Hàm groupby() nhóm các bản ghi theo hai yếu tố (year, month_num), sau đó hàm mean() tính trung bình chỉ số AQI trong từng nhóm. Điều này giúp đảm bảo tính công bằng khi so sánh giữa hai năm (cùng khoảng thời gian). Nếu giữ nguyên toàn bộ năm 2021, kết quả trung bình sẽ bị lệch do thiếu dữ liệu tương ứng ở 2022.

4.1. Đánh giá tổng quan

4.1.2. So sánh mức AQI trung bình 6 tháng đầu năm của TP.HCM giữa năm 2021 và 2022

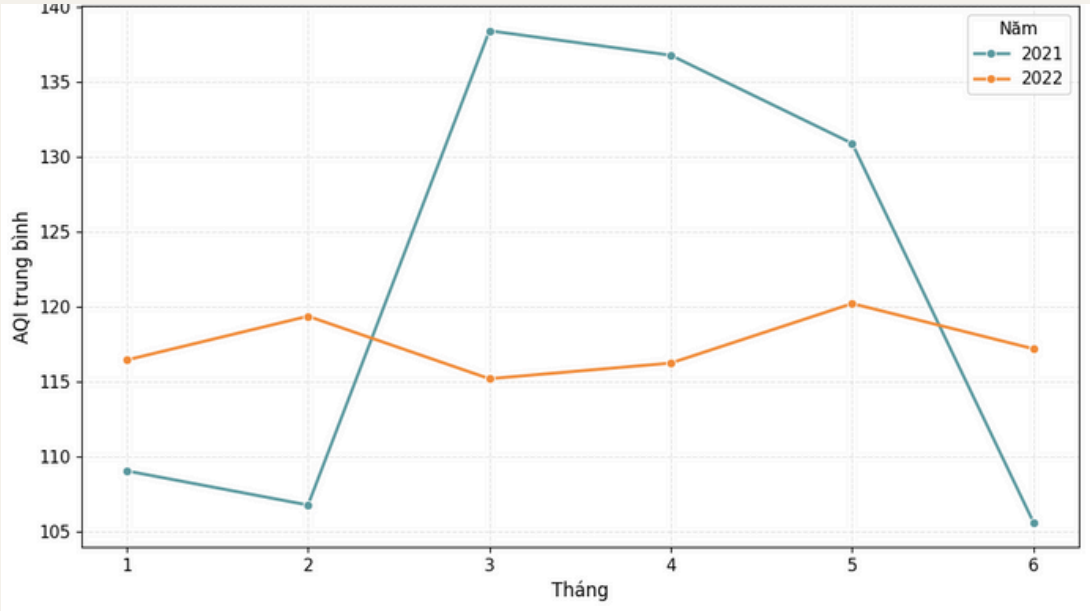
(2) Trực qua hóa dữ liệu bằng biểu đồ đường (Line chart)

Biểu đồ được tạo bằng hàm sns.lineplot(), trong đó biến month_num được đặt trên trục hoành, còn giá trị aqi_daily_total nằm trên trục tung. Tham số hue='year' được thêm vào để biểu diễn hai năm khác nhau trên cùng một biểu đồ, giúp người xem dễ dàng so sánh xu hướng giữa hai đường biểu diễn.

```
sns.lineplot(data=df_monthly_year,x='month_num',y='aqi_daily_total',
hue='year',palette=['#5b9aa0', '#f28c38'],marker='o',linewidth=1.8)
```

4.1. Đánh giá tổng quan

4.1.2. So sánh mức AQI trung bình 6 tháng đầu năm của TP.HCM giữa năm 2021 và 2022



Biểu đồ 4.2. Mức AQI trung bình 6 tháng đầu năm giữa năm 2021 và 2022

Năm 2021 ghi nhận biến động cực mạnh: AQI bắt đầu ở mức thấp (~108) vào tháng 1, sau đó tăng vọt lên đỉnh ~138 vào tháng 3 – mức Kém, duy trì cao trong tháng 4–5, rồi giảm mạnh về ~106 vào tháng 6. Xu hướng "tăng đột biến – giảm đột ngột" này cho thấy ô nhiễm nghiêm trọng đầu năm, có thể do khói bụi tích tụ từ hoạt động cuối năm cũ, giao thông tăng cao sau Tết và thời tiết khô hanh, trước khi được mưa đầu mùa rửa trôi vào tháng 6.

Ngược lại, năm 2022 thể hiện sự ổn định đáng kể: AQI dao động nhẹ trong khoảng 115–120, chủ yếu ở mức Trung bình đến Kém, không có bất kỳ đỉnh cao hay giảm sâu nào. Đường biểu diễn gần như phẳng, chỉ giảm nhẹ từ tháng 2 xuống tháng 3 rồi tăng nhẹ trở lại, phản ánh môi trường không khí đồng đều hơn, ít chịu tác động cực đoan từ thời tiết hay hoạt động xã hội.

4.1. Đánh giá tổng quan

4.1.2. So sánh mức AQI trung bình 6 tháng đầu năm của TP.HCM giữa năm 2021 và 2022

So sánh hai năm, 2021 có biến thiên lớn hơn hẳn (biên độ ~32 điểm), với đỉnh ô nhiễm cao hơn 18 điểm so với 2022 (~138 so với ~120). Trong khi 2021 trải qua giai đoạn ô nhiễm nặng kéo dài, thì 2022 không có ngày nào vượt quá mức Kém nghiêm trọng. Về mặt trung bình, AQI 6 tháng đầu 2022 thấp hơn và ổn định hơn, cho thấy chất lượng không khí được cải thiện rõ rệt so với cùng kỳ năm trước.

Tóm lại, biểu đồ minh chứng rằng 6 tháng đầu năm 2022 có chất lượng không khí tốt hơn đáng kể so với 2021, với ít ngày ô nhiễm nặng, biến động thấp và giá trị AQI trung bình an toàn hơn. Tuy nhiên, cả hai năm đều chưa đạt mức Tốt, và sự cải thiện của 2022 có thể chỉ là tạm thời nếu không duy trì các biện pháp kiểm soát phát thải lâu dài như giảm phương tiện cá nhân, kiểm soát bụi công trình và tăng cường cây xanh đô thị.

4.2. Đánh giá các thành phần gây ô nhiễm không khí

4.2.1. Cơ cấu các thành phần chính gây ô nhiễm (PM2.5, NO_2, SO_2, CO, O_3)

Đặt vấn đề: Chất lượng không khí bị ảnh hưởng bởi nhiều thành phần ô nhiễm khác nhau, trong đó phổ biến nhất gồm bụi mịn ($PM_{2.5}$), khí nitơ đioxit (NO_2), lưu huỳnh đioxit (SO_2), cacbon monoxit (CO) và ozon (O_3). Mỗi chất có đặc tính riêng, mức phát thải và tác động sức khỏe khác nhau. Việc xác định tỷ lệ đóng góp của từng chất giúp nhận diện nguồn gây ô nhiễm chính tại TP.HCM, từ đó hỗ trợ xây dựng chính sách kiểm soát phù hợp. Câu hỏi đặt ra là:"*Trong giai đoạn khảo sát, chất nào chiếm tỷ trọng lớn nhất trong tổng mức ô nhiễm không khí tại TP.HCM?*"

Để trực quan hóa cơ cấu đóng góp của các thành phần ô nhiễm, nhóm nghiên cứu sử dụng biểu đồ tròn. Biểu đồ tròn thể hiện tỷ lệ phần trăm (%) của từng chất trong tổng nồng độ các chất ô nhiễm, giúp dễ dàng hình dung cơ cấu tương đối và nhận biết thành phần chiếm ưu thế. Cụ thể, dữ liệu được xử lý bằng cách tính giá trị trung bình nồng độ của từng chất trong toàn bộ giai đoạn khảo sát, sau đó quy đổi thành tỷ lệ phần trăm so với tổng các giá trị trung bình này. Quy trình thực hiện gồm 3 bước chính: (1) Tính giá trị trung bình của từng chất ô nhiễm, (2) Tính tỷ lệ phần trăm đóng góp của từng chất và (3) Trực quan hóa dữ liệu bằng biểu đồ tròn (Pie chart).

4.2. Đánh giá các thành phần gây ô nhiễm không khí

4.2.1. Cơ cấu các thành phần chính gây ô nhiễm (PM2.5, NO_2, SO_2, CO, O_3)

(1) Tính giá trị trung bình của từng chất ô nhiễm

```
mean_values = df[['pm2.5', 'no2', 'so2', 'co', 'o3']].mean()
```

Đầu tiên, đoạn mã trích xuất các cột tương ứng với năm chất ô nhiễm chính gồm: PM2.5, NO_2, SO_2, CO và O_3 từ bộ dữ liệu df. Hàm .mean() được sử dụng để tính giá trị trung bình của từng chất trong toàn bộ khoảng thời gian khảo sát. Kết quả thu được là một chuỗi (Series) chứa nồng độ trung bình của từng chất ô nhiễm, phản ánh mức độ ảnh hưởng trung bình của mỗi chất đến tổng thể chất lượng không khí tại TP.HCM.

(2) Tính tỷ lệ phần trăm đóng góp của từng chất

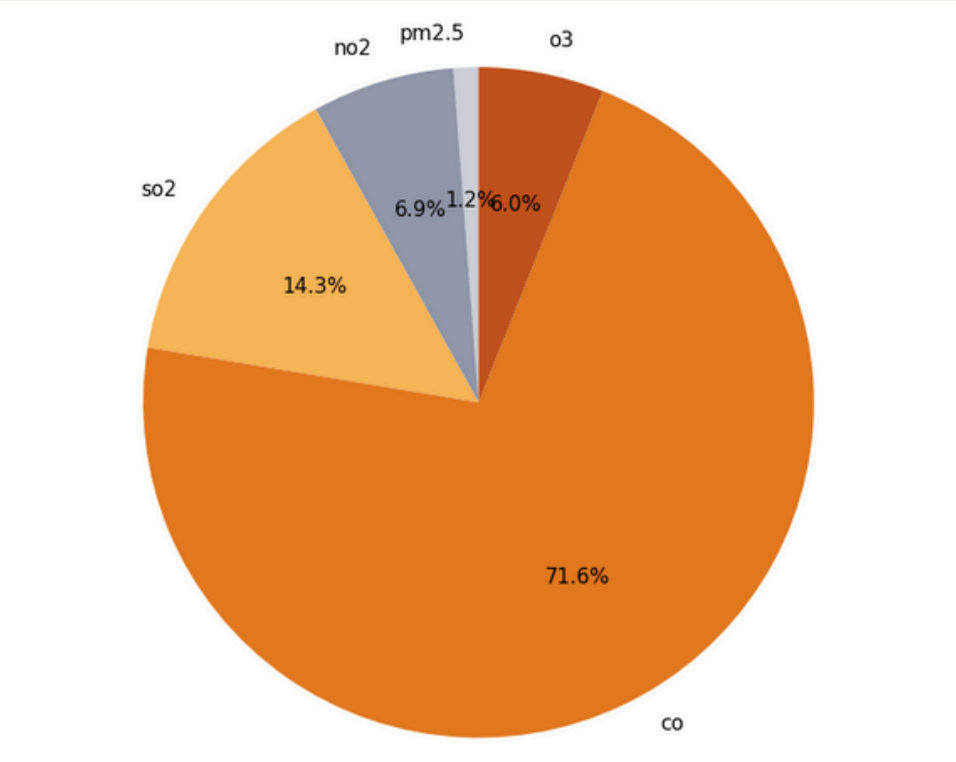
```
percent = (mean_values / mean_values.sum()) * 100
```

Sau khi có giá trị trung bình, đoạn mã tiếp theo tính tỷ lệ phần trăm đóng góp của từng chất so với tổng toàn bộ các chất ô nhiễm. Cụ thể, mỗi giá trị trung bình được chia cho tổng trung bình của tất cả 5 chất, rồi nhân với 100 để chuyển đổi sang phần trăm (%). Bước này giúp xác định cơ cấu tỷ lệ tương đối giữa các chất ô nhiễm, tạo tiền đề để biểu diễn trực quan bằng biểu đồ tròn.

```
plt.pie(mean_values,labels=mean_values.index,autopct='%1.1f%%',
startangle=90, colors=colors, textprops={'fontsize':11})
```

4.2. Đánh giá các thành phần gây ô nhiễm không khí

4.2.1. Cơ cấu các thành phần chính gây ô nhiễm (PM2.5, NO_2, SO_2, CO, O_3)



Biểu đồ 4.3. Tỷ lệ đóng góp trung bình của các chất ô nhiễm

4.2. Đánh giá các thành phần gây ô nhiêm không khí

4.2.1. Cơ cấu các thành phần chính gây ô nhiễm (PM2.5, NO_2, SO_2, CO, O_3)

Biểu đồ tròn trên thể hiện tỷ lệ đóng góp trung bình của năm thành phần ô nhiễm chính trong không khí tại TP.HCM, bao gồm CO, SO_2, NO_2, O_3 và PM2.5. Qua quan sát, có thể thấy khí CO chiếm tỷ lệ cao nhất, lên tới 71.6%, cho thấy đây là nguồn gây ô nhiễm chủ yếu trong khu vực. CO thường phát sinh từ hoạt động giao thông, đặc biệt là các phương tiện sử dụng xăng dầu, vốn phổ biến tại các đô thị lớn như TP.HCM.

Đứng thứ hai là SO_2 (14.3%), chủ yếu bắt nguồn từ hoạt động đốt nhiên liệu hóa thạch trong công nghiệp và giao thông. Tiếp theo, NO_2 (6.9%) cũng là một chất khí độc, thường xuất hiện do quá trình đốt cháy ở nhiệt độ cao, đặc biệt trong động cơ xe máy và ô tô. O_3 (6.0%) tuy không phát thải trực tiếp nhưng được hình thành từ phản ứng quang hóa giữa NOx và hợp chất hữu cơ dễ bay hơi (VOC) dưới ánh sáng mặt trời. Cuối cùng, PM2.5 (1.2%) chiếm tỷ lệ nhỏ nhất trong cơ cấu tổng thể, song vẫn là chất có tác động nghiêm trọng đến sức khỏe con người.

Tóm lại, biểu đồ cho thấy ô nhiễm không khí tại TP.HCM chủ yếu do khí CO chi phối, phản ánh rõ ảnh hưởng của hoạt động giao thông đô thị dày đặc. Các chất còn lại, dù chiếm tỷ trọng nhỏ hơn, vẫn góp phần đáng kể vào mức độ ô nhiễm tổng thể, đặc biệt trong những giai đoạn cao điểm hoặc khu vực có mật độ dân cư, phương tiện lớn. Kết quả này gợi ý rằng, kiểm soát khí thải từ phương tiện giao thông cần được xem là ưu tiên hàng đầu trong các chính sách cải thiện chất lượng không khí đô thị.

4.2. Đánh giá các thành phần gây ô nhiễm không khí

4.2.2. Xu hướng biến động theo mùa của các chất ô nhiễm tại TP.HCM

Đặt vấn đề: Một trong những yếu tố quan trọng ảnh hưởng đến chất lượng không khí là điều kiện khí hậu và mùa trong năm. Sự thay đổi về nhiệt độ, độ ẩm, lượng mưa và tốc độ gió có thể tác động đến khả năng phát tán hoặc tích tụ của các chất ô nhiễm. Vì vậy, việc phân tích sự thay đổi nồng độ trung bình của các chất ô nhiễm theo mùa giúp nhận diện rõ thời điểm ô nhiễm gia tăng, đồng thời hỗ trợ công tác giám sát và dự báo môi trường đô thị. Câu hỏi đặt ra là:"*Nồng độ của các chất ô nhiễm chính* (PM2.5, NO_2, SO_2, CO, O_3) *có xu hướng thay đổi như thế nào giữa các mùa trong năm tại TP.HCM?*"

Để trả lời câu hỏi nghiên cứu này, biểu đồ cột (barchart) được lựa chọn nhằm thể hiện trực quan sự khác biệt về nồng độ trung bình của các chất ô nhiễm giữa các mùa trong năm. Biểu đồ cột cho phép so sánh rõ ràng giá trị trung bình của từng thông số ($PM_{2.5}$, NO_2, SO_2, CO, O_3) giữa các nhóm mùa, qua đó dễ dàng nhận diện mùa nào có mức ô nhiễm cao hoặc thấp hơn. Ngoài ra, còn giúp minh họa xu hướng thay đổi của từng chất ô nhiễm một cách trực quan, hỗ trợ quan sát và đánh giá nhanh mối tương quan giữa yếu tố mùa vụ và chất lượng không khí. Quy trình thực hiện gồm 2 bước chính: (1) Tính trung bình theo mùa và (2) Trực quan hóa dữ liệu bằng biểu đồ cột (Bar chart).

4.2. Đánh giá các thành phần gây ô nhiễm không khí

4.2.2. Xu hướng biến động theo mùa của các chất ô nhiễm tại TP.HCM

(1) Tính trung bình theo mùa

```
seasonal_means = df.groupby('Season')[['pm2.5', 'no2', 'so2', 'co',
'o3']].mean().reset_index()
seasonal_long=pd.melt(seasonal_means,id_vars='Season',var_name='Pol
lutant', value_name='Mean_Value')
```

Dữ liệu được nhóm (groupby) theo cột Season, đại diện cho các mùa (mùa khô và mùa mưa), để tính giá trị trung bình của từng chất ô nhiễm (PM2.5, NO_2, SO_2, CO, O_3). Sau đó, dữ liệu được chuyển sang dạng long format bằng hàm pd.melt(). Cấu trúc này giúp thuận tiện cho việc vẽ biểu đồ cột nhóm trong Seaborn, vì mỗi hàng giờ tương ứng với một tổ hợp gồm: mùa, chất ô nhiễm, giá trị trung bình. Nhờ đó, ta có thể hiển thị dễ dàng sự khác biệt của các chất ô nhiễm giữa các mùa trong cùng một biểu đồ.

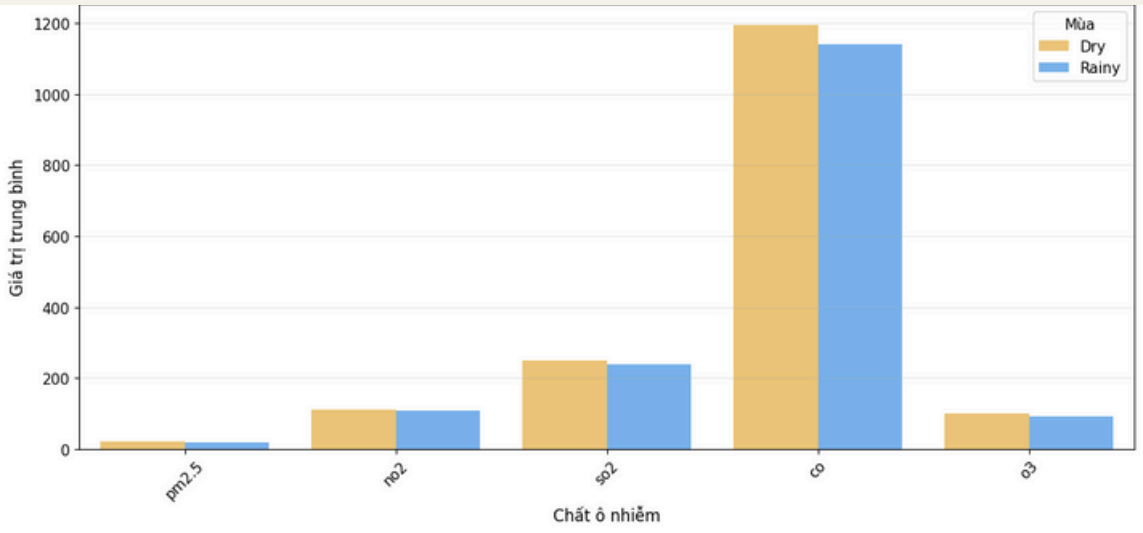
(2) Trực quan hóa dữ liệu bằng biểu đồ cột (Bar chart)

```
sns.barplot(data=seasonal_long,x='Pollutant',y='Mean_Value',
hue='Season', palette=colors)
```

Hàm sns.barplot() của thư viện Seaborn được dùng để vẽ biểu đồ cột. Biến x='Pollutant' đặt tên các chất ô nhiễm (PM2.5, NO_2, SO_2, CO, O_3) lên trục hoành, còn y='Mean_Value' thể hiện giá trị trung bình tương ứng trên trục tung. Tham số hue='Season' giúp chia cột thành hai nhóm màu đại diện cho hai mùa, cho phép so sánh trực tiếp giá trị trung bình giữa mùa khô và mùa mưa của từng chất ô nhiễm

4.2. Đánh giá các thành phần gây ô nhiêm không khí

4.2.2. Xu hướng biến động theo mùa của các chất ô nhiễm tại TP.HCM



Biểu đồ 4.4. Mức độ chất ô nhiễm theo mùa

Nhìn chung, mức độ ô nhiễm trong mùa khô cao hơn so với mùa mưa ở hầu hết các chất. Trong đó, khí CO chiếm giá trị vượt trội, cao nhất trong tất cả các chất, với sự chênh lệch đáng kể giữa hai mùa: mùa khô có nồng độ CO trung bình cao hơn rõ rệt. Điều này có thể do lượng phương tiện giao thông và hoạt động công nghiệp tăng mạnh trong mùa khô, khi điều kiện thời tiết khô nóng làm hạn chế quá trình khuếch tán chất ô nhiễm.

Các chất SO_2, NO_2 và O_3 cũng thể hiện xu hướng tương tự: nồng độ cao hơn vào mùa khô và giảm nhẹ vào mùa mưa. Trong khi đó, PM2.5 có mức thay đổi nhỏ nhất giữa hai mùa, cho thấy bụi mịn chịu ảnh hưởng ít hơn bởi yếu tố thời tiết so với các khí ô nhiễm khác.

Nhìn chung, biểu đồ phản ánh rằng mùa khô là giai đoạn TP.HCM có mức ô nhiễm không khí nghiêm trọng hơn, do điều kiện khí tượng bất lợi cho quá trình phát tán và rửa trôi các chất ô nhiễm.

4.3. Xu hướng AQI theo thời gian

4.3.1. Xu hướng AQI theo thời gian

Đặt vấn đề: Đánh giá tổng quan chất lượng không khí là bước đầu tiên trong việc phân tích dữ liệu AQI nhằm xác định mức độ ô nhiễm tại khu vực khảo sát. Chỉ số AQI trung bình hằng ngày phản ánh sự biến động tổng thể của chất lượng không khí trong ngắn hạn, qua đó giúp nhận biết các giai đoạn không khí trong lành hoặc ô nhiễm tăng cao. Câu hỏi đặt ra là: *Chỉ số AQI trung bình theo ngày tại TP.HCM trong giai đoạn khảo sát biến động như thế nào? Chất lượng không khí chủ yếu ở mức tốt, trung bình hay kém?*

Để mô tả xu hướng biến động chỉ số AQI trung bình qua các tháng trong năm, nhóm nghiên cứu sử dụng biểu đồ đường (line chart). Quy trình thực hiện bao gồm hai giai đoạn chính: (1) Tính toán giá trị trung bình theo tháng, (2) Trực quan hóa bằng biểu đồ.

(1) Tính toán giá trị trung bình theo tháng

```
df_aqi['month'] = df_aqi['date'].dt.to_period('M')
df_monthly = df_aqi.groupby('month', as_index=False)
['aqi_daily_total'].mean()
df_monthly['month'] = df_monthly['month'].astype(str)
```

Các giá trị AQI theo tháng được tạo ra bằng cách chuyển đổi dữ liệu thời gian từ dạng ngày sang tháng, sau đó nhóm toàn bộ các giá trị AQI trong cùng một tháng lại và tính trung bình. Cách tổng hợp này giúp tạo ra một chỉ số đại diện cho chất lượng không khí của từng tháng, giảm ảnh hưởng từ biến động ngắn hạn theo ngày và làm nổi bật xu hướng biến động AQI theo thời gian trong năm.

4.3. Xu hướng AQI theo thời gian

4.3.1. Xu hướng AQI theo thời gian

(2) Trực quan hóa bằng biểu đồ

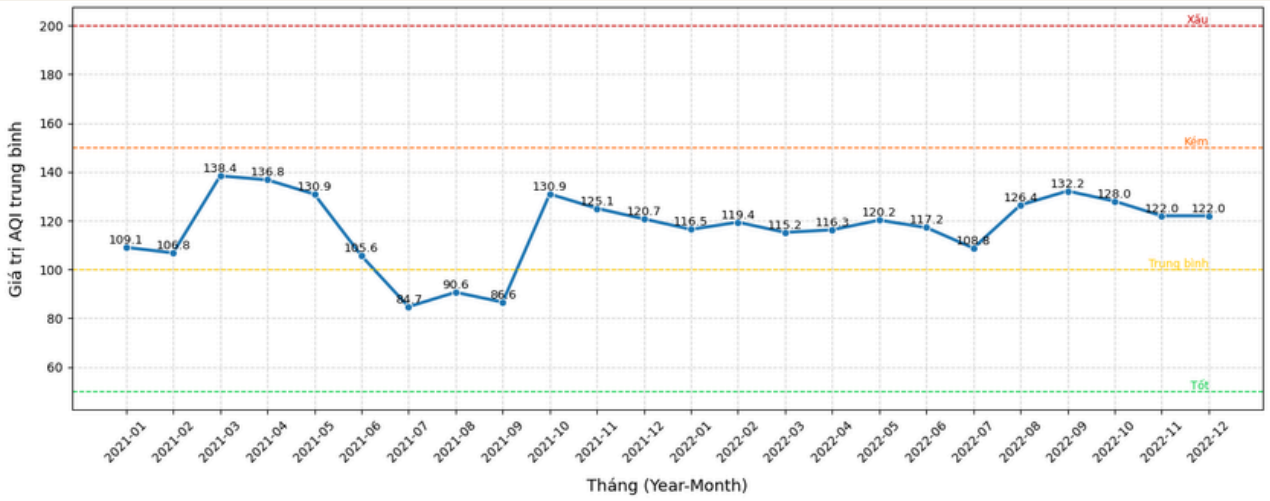
```
sns.lineplot(data=df_monthly,  x='month',  y='aqi_daily_total',
marker='o', linewidth=2.5)
```

Tiếp theo là dùng hàm sns.lineplot() của thư viện Seaborn được áp dụng để vẽ đường biểu diễn mối quan hệ giữa hai biến: trục X thể hiện tháng (month) và trục Y biểu thị giá trị AQI trung bình (aqi_daily_total). Biểu đồ minh họa được chính xác sự thay đổi của AQI qua từng tháng và đảm bảo tính thẩm mỹ và dễ đọc, phục vụ tốt cho mục tiêu phân tích và so sánh chất lượng không khí trong năm.

Bên cạnh đường biểu diễn chính, biểu đồ còn được bổ sung các đường ngang tại các ngưỡng giá trị 50, 100, 150 và 200, tương ứng với các mức phân loại chất lượng không khí do Bộ Tài nguyên và Môi trường quy định. Cụ thể, vùng có giá trị AQI ≤ 50 được xem là tốt (màu xanh), từ 51 đến 100 là trung bình (màu vàng), từ 101 đến 150 là kém (màu cam), và trên 150 là xấu (màu đỏ). Việc thêm các đường tham chiếu này giúp người xem dễ dàng so sánh giá trị AQI trung bình từng tháng với các ngưỡng chuẩn, qua đó nhanh chóng nhận biết được mức độ ô nhiễm không khí tại từng thời điểm trong năm. Cách trình bày này làm tăng tính trực quan của biểu đồ và hỗ trợ quá trình phân tích, đánh giá tổng thể chất lượng môi trường đô thị.

4.3. Xu hướng AQI theo thời gian

4.3.1. Xu hướng AQI theo thời gian



Biểu đồ 4.5. Xu hướng AQI trung bình theo tháng

Biểu đồ trên thể hiện xu hướng chỉ số chất lượng không khí (AQI) trung bình theo tháng tại Trạm 6 – Thành phố Hồ Chí Minh trong giai đoạn từ tháng 01/2021 đến tháng 12/2022. Qua quan sát, có thể thấy rằng giá trị AQI trung bình trong suốt hai năm dao động chủ yếu trong khoảng 84 – 138, tương ứng với mức "Trung bình" đến "Kém" theo thang đánh giá chất lượng không khí.

Trong năm 2021, giá trị AQI ở các tháng đầu năm dao động quanh mức 100 và có xu hướng tăng mạnh vào tháng 3 (138.4) và tháng 4 (136.8). Đây là thời điểm có chỉ số AQI cao nhất trong năm, phản ánh tình trạng chất lượng không khí kém, có thể ảnh hưởng đến nhóm người nhạy cảm như người già, trẻ em và người mắc bệnh hô hấp. Tuy nhiên, đến giai đoạn giữa năm (từ tháng 6 đến tháng 9/2021), chỉ số AQI giảm xuống còn khoảng 84.7 – 90.6, tương ứng mức trung bình, cho thấy chất lượng không khí được cải thiện đáng kể.

4.3. Xu hướng AQI theo thời gian

4.3.1. Xu hướng AQI theo thời gian

Từ tháng 10/2021 trở đi, chỉ số AQI tăng trở lại, dao động quanh 120 – 130, và duy trì xu hướng này sang năm 2022. Trong năm 2022, chất lượng không khí nhìn chung ổn định hơn, với chỉ số AQI trung bình hằng tháng nằm trong khoảng 115 – 130, đạt đỉnh vào tháng 9/2022 (132.2), vẫn thuộc mức kém. Mức AQI thấp nhất của năm là 108.8 vào tháng 7, cho thấy có sự cải thiện nhẹ trong giai đoạn mùa mưa.

Nhìn tổng thể, có thể kết luận rằng chất lượng không khí tại Trạm 6 trong hai năm 2021 – 2022 chủ yếu ở mức trung bình đến kém, chưa đạt tiêu chuẩn "tốt" cho sức khỏe cộng đồng. Việc AQI thường xuyên vượt ngưỡng 100 cho thấy khu vực này chịu ảnh hưởng bởi ô nhiễm không khí ở mức đáng lo ngại, đặc biệt trong các tháng khô, ít mưa.

4.3. Xu hướng AQI theo thời gian

4.3.2. So sánh AQI trung bình theo hai mùa mưa và khô

Đặt vấn đề: Sau khi đã phân tích xu hướng AQI theo thời gian, một khía cạnh khác cũng quan trọng không kém trong nghiên cứu chất lượng không khí là sự biến động giữa các mùa trong năm. Thành phố Hồ Chí Minh có hai mùa rõ rệt: mùa mưa (từ tháng 5 đến tháng 11) và mùa khô (từ tháng 12 đến tháng 4 năm sau). Câu hỏi đặt ra là: *Giá trị AQI trung bình giữa hai mùa mưa – khô có sự khác biệt đáng kể hay không? Và nếu có, mùa nào ghi nhận mức độ ô nhiễm cao hơn?*

Nhóm tiến hành xử lý dữ liệu và trực quan hóa bằng biểu đồ cột nhóm (Clustered Bar Chart) để so sánh các chỉ tiêu AQI theo mùa. Quy trình được chia thành hai bước chính: (1) nhóm dữ liệu theo mùa và các chỉ tiêu ô nhiễm, (2) Trực quan hóa thành biểu đồ Clustered Bar Chart

(1) Nhóm dữ liệu theo mùa và các chỉ tiêu ô nhiễm

```
cols_pollutants = ['aqi_pm2.5', 'aqi_so2', 'aqi_no2', 'aqi_co',
'aqi_o3']
df_season = df_aqi.groupby('season')
[cols_pollutants].mean().reset_index()
```

Sau khi đã xác định mùa, dữ liệu được nhóm lại dựa trên biến "season" cùng với các chỉ tiêu ô nhiễm gồm PM2.5, SO_2, NO_2, CO và O_3. Lệnh groupby('season') kết hợp với mean() giúp tính giá trị AQI trung bình của từng chất trong mỗi mùa. Nhờ đó, ta có thể xác định được loại chất ô nhiễm nào có xu hướng tăng hoặc giảm tùy theo mùa.

4.3. Xu hướng AQI theo thời gian

4.3.2. So sánh AQI trung bình theo hai mùa mưa và khô

(2) Trực quan hóa thành biểu đồ

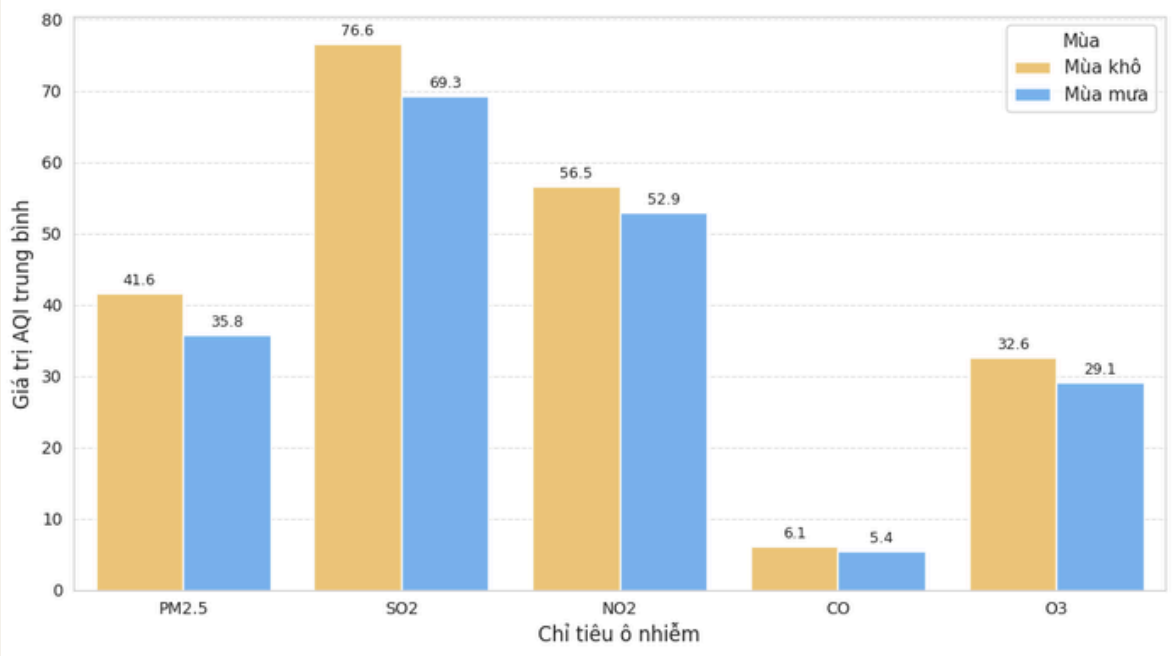
```
plt.figure(figsize=(10,6))
 sns.barplot(data=df_melted, x='Chỉ tiêu ô nhiễm', y='Giá trị
AQI trung bình',
 hue='season', palette=['#ffcc66', '#66b3ff'])
```

Sau khi xử lý dữ liệu, nhóm sử dụng hàm sns.barplot() để vẽ biểu đồ cột nhóm (Clustered Bar Chart). Trục hoành hiển thị các chỉ tiêu ô nhiễm, còn trục tung thể hiện giá trị AQI trung bình. Tham số hue='season' giúp chia nhóm theo hai mùa với hai màu sắc khác biệt: màu vàng nhạt cho mùa khô và màu xanh cho mùa mưa.

Biểu đồ được trang trí, chú giải thể hiện tên mùa, cùng các đường lưới ngang giúp người xem dễ đối chiếu giá trị, giúp người đọc dễ dàng nhận biết và so sánh trực tiếp mức độ ô nhiễm giữa hai mùa.

4.3. Xu hướng AQI theo thời gian

4.3.2. So sánh AQI trung bình theo hai mùa mưa và khô



Biểu đồ 4.6. So sánh mức độ ô nhiễm không khí giữa 2 mùa

Khi quan sát biểu đồ, có thể thấy rõ sự chênh lệch giữa mùa mưa và mùa khô. Ở hầu hết các chỉ tiêu, mùa khô có giá trị AQI trung bình cao hơn, phản ánh mức độ ô nhiễm nặng hơn. Ngược lại ở mùa mưa thì các giá trị AQI giảm đáng kể do mưa lớn và độ ẩm cao giúp rửa trôi và khuếch tán các chất ô nhiễm. Riêng chỉ tiêu O_3 có xu hướng ít biến động hơn giữa hai mùa, cho thấy sự ổn định tương đối của loại khí này trong không khí. Câu hỏi đặt ra là: *Phân bố giá trị AQI giữa hai mùa mưa – khô có sự khác biệt rõ rệt không? Và xu hướng biến động theo từng quý trong năm phản ánh điều gì về đặc điểm môi trường đô thị TP.HCM?*

4.3. Xu hướng AQI theo thời gian

4.3.2. So sánh AQI trung bình theo hai mùa mưa và khô

Quy trình xử lý dữ liệu được triển khai qua hai bước: (1) Tạo cột "Quý" để thể hiện chu kỳ ba tháng; (2) Trực quan hóa bằng biểu đồ Violin Plot

(1) Tạo cột "Quý" để thể hiện chu kỳ ba tháng

```
plt.figure(figsize=(10, 6))
sns.violinplot(
x='Quý',
y=aqi_col,
hue='Mùa',
data=df,
split=True,
```

Nhóm sử dụng hàm sns.violinplot() từ thư viện Seaborn để hiển thị phân bố dữ liệu AQI theo từng quý, đồng thời chia nhỏ theo hai mùa bằng tham số hue='Mùa'. Tham số split=True giúp hai phần phân bố (mùa mưa và mùa khô) được hiển thị đối xứng trên cùng một trục, tạo điều kiện dễ so sánh trực quan.

4.3. Xu hướng AQI theo thời gian

4.3.2. So sánh AQI trung bình theo hai mùa mưa và khô

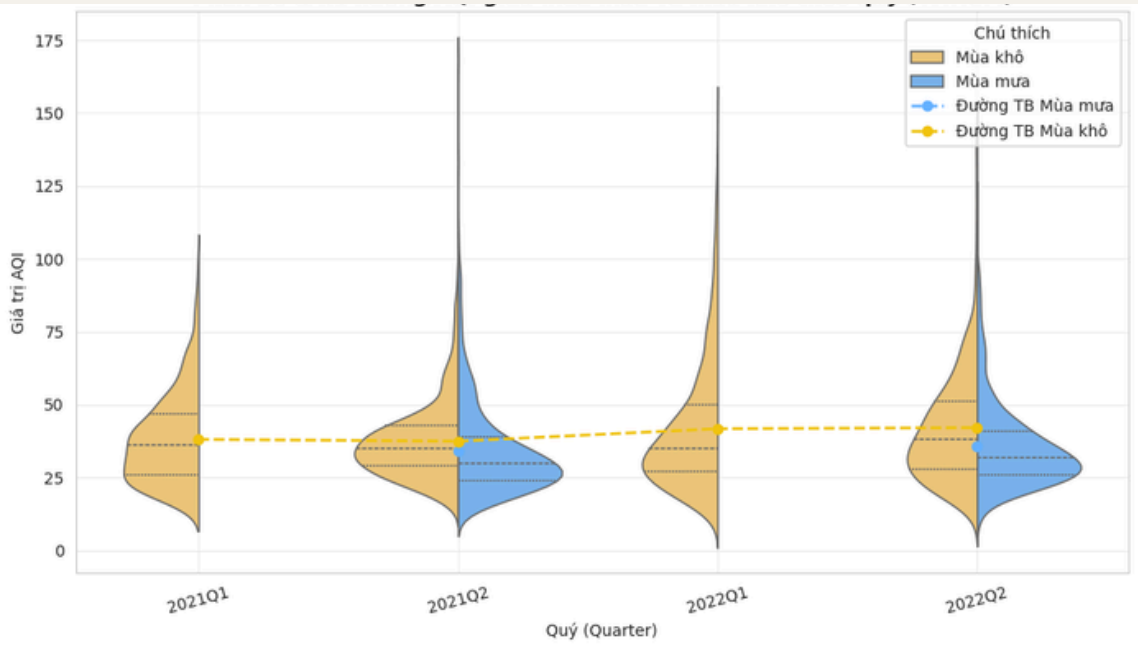
(2) Trực quan hóa bằng biểu đồ Violin Plot

```
mean_by_quarter = df.groupby(['Quý', 'Mùa'])[aqi_col].mean().reset_index()
```

Để biểu đồ có tính phân tích cao hơn, nhóm bổ sung thêm đường biểu diễn trung bình AQI theo mùa bằng lệnh plt.plot(). Mỗi đường được tạo từ kết quả nhóm groupby(['Quý', 'Mùa']) kết hợp với mean(), thể hiện xu hướng biến đổi AQI trung bình giữa các quý trong năm. Việc thêm đường trung bình vào biểu đồ giúp người xem dễ theo dõi xu hướng biến động theo thời gian. Cuối cùng, biểu đồ được trang trí, chú giải và lưới nhẹ giúp tăng tính trực quan.

4.3. Xu hướng AQI theo thời gian

4.3.2. So sánh AQI trung bình theo hai mùa mưa và khô



Biểu đồ 4.7. Phân bố & xu hướng AQI giữa 2 mùa theo quý

Quan sát biểu đồ cho thấy phân bố AQI giữa mùa khô và mùa mưa có sự khác biệt rõ rệt theo từng quý. Cụ thể, mùa khô có phân bố AQI rộng và lệch phải hơn, phản ánh mức độ ô nhiễm cao hơn và sự dao động lớn giữa các giá trị. Trong khi đó, mùa mưa có phân bố hẹp hơn, tập trung quanh giá trị trung bình thấp, cho thấy không khí trong lành và ổn định hơn.

Đường trung bình AQI cho thấy giá trị trung bình của mùa khô luôn cao hơn mùa mưa trong tất cả các quý. Đặc biệt, AQI giảm rõ trong quý II (2021Q2) và quý II (2022Q2) là hai giai đoạn cao điểm của mùa mưa, sau đó tăng trở lại vào các quý đầu năm, tương ứng mùa khô.

4.3. Xu hướng AQI theo thời gian

4.3.2. So sánh AQI trung bình theo hai mùa mưa và khô

⁕ Liên hệ và đối chiếu kết quả giữa hai biểu đồ

Khi đối chiếu kết quả từ biểu đồ 5.6 (So sánh mức độ ô nhiễm giữa 2 mùa) và biểu đồ 5.7 (Phân bố & xu hướng AQI theo quý), có thể thấy sự tương đồng rõ rệt trong xu hướng biến động chất lượng không khí giữa mùa khô và mùa mưa tại TP.HCM.
Biểu đồ 5.6 cho thấy giá trị AQI trung bình của hầu hết các chỉ tiêu (PM2.5, SO_2, NO_2, CO) đều cao hơn trong mùa khô, phản ánh tình trạng ô nhiễm tăng do điều kiện khô nóng, ít mưa và khả năng khuếch tán chất ô nhiễm kém. Ngược lại, mùa mưa có giá trị AQI thấp hơn, cho thấy hiệu quả rửa trôi bụi mịn và khí độc hại nhờ mưa và độ ẩm cao.

Biểu đồ 5.7 giúp bổ sung góc nhìn về phân bố và xu hướng AQI theo thời gian, thể hiện rằng mùa khô không chỉ có giá trị trung bình cao hơn mà còn có độ phân tán lớn hơn, tức là mức ô nhiễm vừa cao vừa biến động mạnh. Trong khi đó, mùa mưa có phân bố hẹp hơn, tập trung quanh giá trị thấp, phản ánh chất lượng không khí ổn định và trong lành hơn.

Đặc biệt, đường trung bình AQI theo quý cho thấy xu hướng giảm rõ rệt trong quý II của cả hai năm (2021Q2 và 2022Q2) là giai đoạn cao điểm mùa mưa, sau đó tăng trở lại vào quý I của năm sau, tương ứng với mùa khô. Xu hướng này phù hợp với đặc trưng khí hậu TP.HCM, khi mùa khô thường tích tụ bụi và khí thải giao thông, còn mùa mưa giúp khuếch tán và làm sạch không khí.

4.3. Xu hướng AQI theo thời gian

4.3.2. So sánh AQI trung bình theo hai mùa mưa và khô

Từ hai biểu đồ, có thể khẳng định rằng mùa khô là giai đoạn có nguy cơ ô nhiễm cao hơn, đặc biệt đối với PM2.5 và NO_2 là hai tác nhân chính ảnh hưởng đến sức khỏe hô hấp đô thị. Dù mùa mưa giúp cải thiện đáng kể chất lượng không khí, song một số chỉ tiêu như O_3 vẫn duy trì ở mức trung bình, cho thấy sự cần thiết của việc theo dõi AQI thường xuyên quanh năm.

Sự kết hợp giữa biểu đồ 5.6 và biểu đồ 5.7 đã giúp phân tích toàn diện hơn tác động của yếu tố mùa đến chất lượng không khí, đồng thời cung cấp cơ sở khoa học cho việc xây dựng chính sách kiểm soát ô nhiễm theo thời điểm tại TP.HCM.

4.4. Tính tương quan của AQI

4.4.1. Tương quan giữa AQI với biến nhiệt độ (Temperature)

Đặt vấn đề: Sau khi đã xem xét sự biến động của chỉ số AQI theo mùa và quý trong năm, nhóm tiếp tục đi sâu phân tích mối quan hệ giữa các yếu tố khí tượng và mức độ ô nhiễm không khí. Trong đó, nhiệt độ là một biến quan trọng, có khả năng ảnh hưởng đến khả năng khuếch tán hoặc tích tụ các chất ô nhiễm. Câu hỏi đặt ra là: *Liệu sự thay đổi của nhiệt độ có tương quan với biến động AQI hay không? Nói cách khác, khi nhiệt độ tăng hoặc giảm, chất lượng không khí có xu hướng cải thiện hay suy giảm?*

Để trả lời câu hỏi trên, nhóm nghiên cứu tiến hành ba bước xử lý chính: (1) Tự động nhận diện cột dữ liệu; (2) Tính hệ số tương quan Pearson; (3) Trực quan hóa bằng biểu đồ Hexbin Plot.

(1) Tự động nhận diện cột dữ liệu

Để mã có thể tái sử dụng linh hoạt cho nhiều bộ dữ liệu khác nhau, nhóm không chỉ định thủ công tên cột mà dùng vòng lặp list comprehension:

```
aqi_col = [c for c in df.columns if 'aqi' in c.lower()][0]
 temp_col = [c for c in df.columns if 'temp' in c.lower()][0]
```

Dòng lệnh này sẽ tự động tìm cột có chứa từ "aqi" trong tên và gán vào biến aqi_col; tương tự với biến temp_col cho cột nhiệt độ. Cách làm này giúp chương trình tự động thích nghi với dữ liệu có tên cột khác nhau, tăng tính tổng quát và giảm lỗi cú pháp.

4.4. Tính tương quan của AQI

4.4.1. Tương quan giữa AQI với biến nhiệt độ (Temperature)

(2) Tính hệ số tương quan Pearson

```
corr, _ = pearsonr(df[aqi_col], df[temp_col])
```

Trong bước tính hệ số tương quan Pearson, nhóm sử dụng hàm pearsonr() từ thư viện scipy.stats để đo lường mức độ liên hệ tuyến tính giữa hai biến: chỉ số AQI và nhiệt độ. Kết quả trả về là một giá trị nằm trong khoảng từ -1 đến +1. Nếu hệ số tương quan lớn hơn 0, điều đó cho thấy mối quan hệ thuận chiều – nghĩa là khi nhiệt độ tăng, chỉ số AQI cũng có xu hướng tăng theo. Ngược lại, nếu giá trị nhỏ hơn 0, mối tương quan là nghịch chiều, tức là nhiệt độ tăng sẽ đi kèm với việc giảm chỉ số AQI. Trường hợp hệ số gần bằng 0 cho thấy mối liên hệ yếu hoặc không rõ ràng giữa hai biến. Việc tính toán hệ số Pearson không chỉ giúp định lượng hóa mối quan hệ mà còn cung cấp cơ sở khoa học để lý giải các xu hướng thể hiện trên biểu đồ trực quan sau đó.

(3) Trực quan hóa bằng biểu đồ Hexbin Plot

Trong bước trực quan hóa mối quan hệ giữa nhiệt độ và chỉ số AQI, nhóm sử dụng biểu đồ Hexbin – một dạng biểu đồ mật độ hai chiều được thiết kế đặc biệt để hiển thị dữ liệu dày đặc một cách trực quan và dễ đọc hơn so với biểu đồ phân tán (scatter plot) truyền thống. Ý tưởng chính của biểu đồ này là chia mặt phẳng thành nhiều ô lục giác đều nhau, mỗi ô biểu thị số lượng điểm dữ liệu rơi vào khu vực đó. Các ô có màu càng đậm thể hiện mật độ quan sát càng cao, từ đó giúp người xem dễ dàng nhận biết các vùng dữ liệu tập trung hoặc thưa thớt.

4.4. Tính tương quan của AQI

4.4.1. Tương quan giữa AQI với biến nhiệt độ (Temperature)

```
plt.figure(figsize=(8,6))
 hb = plt.hexbin(
 df[temp_col], df[aqi_col],
 gridsize=30, cmap='Reds', linewidths=0.5, alpha=0.9
```

Đoạn mã trên thực hiện việc tạo biểu đồ. Trong đó, df[temp_col] đại diện cho trục hoành thể hiện nhiệt độ (°C), còn df[aqi_col] tương ứng với trục tung biểu thị chỉ số AQI. Tham số gridsize=30 quy định độ phân giải của biểu đồ. Tiếp đến, cmap='Reds' chỉ định bảng màu đỏ để hiển thị mức mật độ, trong đó những ô có màu đỏ đậm hơn cho thấy nơi tập trung nhiều điểm dữ liệu hơn.

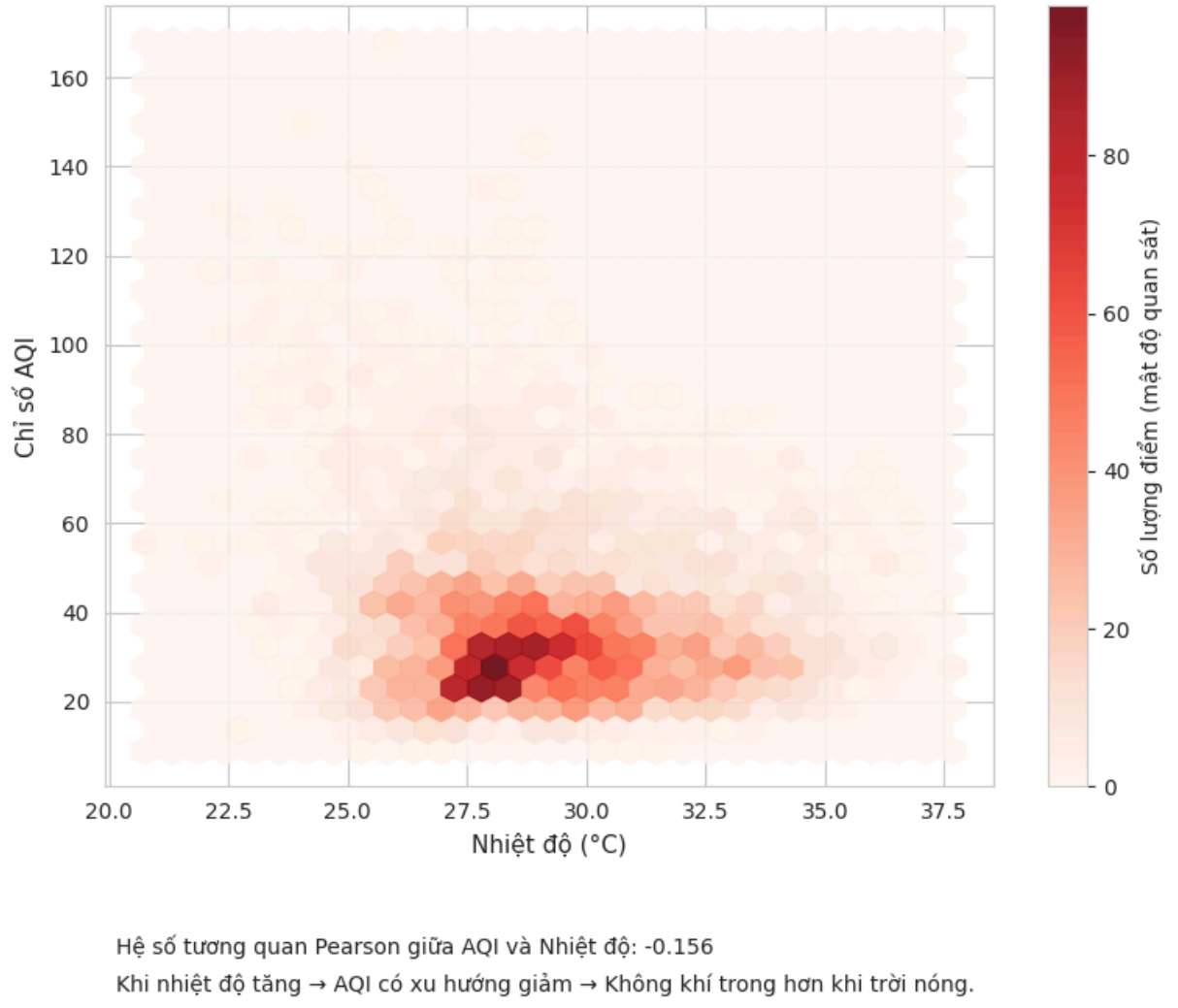
```
cb = plt.colorbar(hb)
 cb.set_label("Số lượng điểm (mật độ quan sát)")
```

Để biểu đồ thêm rõ ràng, nhóm bổ sung thanh màu chú thích Thanh màu này giúp người xem hiểu được ý nghĩa của từng sắc độ – từ nhạt đến đậm tương ứng với số lượng điểm dữ liệu tăng dần.

Nhờ sự kết hợp giữa mã lệnh và yếu tố thẩm mỹ, biểu đồ Hexbin không chỉ thể hiện được mối tương quan tổng thể giữa nhiệt độ và AQI, mà còn cho phép xác định nhanh các vùng có mật độ cao, qua đó giúp người đọc nhận thấy những khoảng giá trị mà hai biến thường xuyên cùng xuất hiện, góp phần làm sáng tỏ mối quan hệ giữa điều kiện thời tiết và chất lượng không khí.

4.4. Tính tương quan của AQI

4.4.1. Tương quan giữa AQI với biến nhiệt độ (Temperature)



Biểu đồ 4.8. Tương quan giữa nhiệt độ và AQI

Biểu đồ trên minh họa mối quan hệ giữa nhiệt độ (temperature) và chỉ số chất lượng không khí (AQI) tại TP.HCM. Mỗi ô lục giác đại diện cho một vùng giá trị của hai biến này, với sắc độ màu đỏ thể hiện mật độ quan sát – ô càng đậm chứng tỏ có nhiều điểm dữ liệu trùng lặp hoặc gần nhau.

4.4. Tính tương quan của AQI

4.4.1. Tương quan giữa AQI với biến nhiệt độ (Temperature)

Quan sát biểu đồ cho thấy phần lớn các điểm dữ liệu tập trung trong vùng nhiệt độ từ 26°C đến 30°C và chỉ số AQI dao động quanh mức 25–50. Điều này cho thấy phần lớn các ngày có chất lượng không khí ở mức "tốt" đến "trung bình" trong khoảng nhiệt độ này, tức là thời tiết ấm vừa phải thường đi kèm với bầu không khí trong lành hơn.

Đáng chú ý, hệ số tương quan Pearson giữa AQI và nhiệt độ được tính là -0.114, thể hiện mối tương quan âm yếu. Nói cách khác, khi nhiệt độ tăng, AQI có xu hướng giảm nhẹ, tức không khí trong lành hơn vào những ngày trời nóng. Tuy nhiên, mức tương quan nhỏ cho thấy mối quan hệ này không quá mạnh, nghĩa là nhiệt độ chỉ là một trong nhiều yếu tố ảnh hưởng đến chất lượng không khí, bên cạnh các yếu tố khác như lượng mưa, tốc độ gió hay mật độ giao thông.

Nhìn chung, biểu đồ này giúp làm rõ câu hỏi nghiên cứu đặt ra ở phần đầu. Kết quả cho thấy có mối liên hệ nhất định, nhưng không rõ rệt, gợi ý rằng nhiệt độ chỉ tác động gián tiếp đến mức độ ô nhiễm, cần kết hợp thêm các yếu tố thời tiết khác để mô hình hóa chính xác hơn sự biến động của AQI theo điều kiện khí hậu.

4.4. Tính tương quan của AQI

4.4.2. Tương quan giữa AQI với biến độ ẩm (Humidity)

Đặt vấn đề: Sau khi đã xem xét sự biến động của chỉ số AQI theo mùa và quý trong năm, nhóm tiếp tục đi sâu phân tích mối quan hệ giữa các yếu tố khí tượng và mức độ ô nhiễm không khí. Bên cạnh nhiệt độ, độ ẩm cũng là một biến quan trọng, có khả năng ảnh hưởng đến sự khuếch tán, lắng đọng và phản ứng hóa học của các hạt bụi trong không khí. Câu hỏi đặt ra là: *Liệu sự thay đổi của độ ẩm có tương quan với biến động của chỉ số AQI hay không? Nói cách khác, khi độ ẩm không khí tăng hay giảm, chất lượng không khí có xu hướng cải thiện hay suy giảm?*

Để trả lời câu hỏi trên, nhóm nghiên cứu tiến hành hai bước xử lý chính: (1) Tính hệ số tương quan Pearson; (2) Trực quan hóa bằng biểu đồ Hexbin Plot.

(1) Tính toán hệ số tương quan Pearson giữa hai biến bằng lệnh:

```
corr, _ = pearsonr(plot_df[humidity_col], plot_df[aqi_col])
```

Hệ số này cho biết mức độ và chiều hướng quan hệ tuyến tính giữa độ ẩm và chỉ số AQI. Nếu hệ số mang giá trị âm, điều đó cho thấy khi độ ẩm tăng thì chỉ số AQI có xu hướng giảm, tức là không khí trong lành hơn trong điều kiện ẩm ướt.

4.4. Tính tương quan của AQI

4.4.2. Tương quan giữa AQI với biến độ ẩm (Humidity)

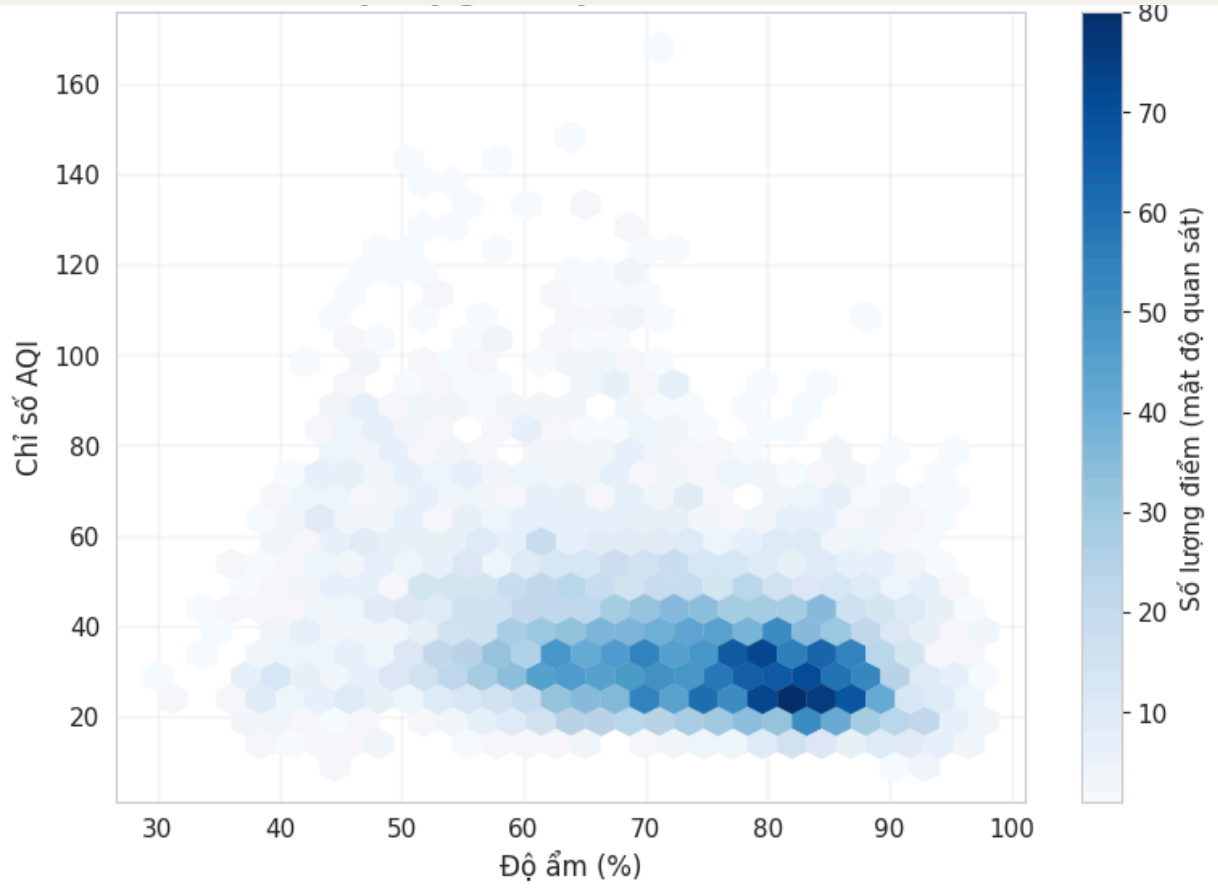
(2) Trực quan hóa bằng biểu đồ Hexbin Plot

```
plt.figure(figsize=(8, 6))
 hb = plt.hexbin(
 plot_df[humidity_col],
 plot_df[aqi_col],
```

Bước tiếp theo là phần trực quan hóa dữ liệu thông qua biểu đồ Hexbin. Thang màu được chọn là màu xanh lam, thể hiện cường độ mật độ quan sát — vùng nào có màu xanh đậm hơn thì ở đó tập trung nhiều điểm dữ liệu hơn. Sau khi biểu đồ được vẽ, thanh màu được thêm vào bên cạnh để người xem dễ dàng nhận biết mức độ mật độ điểm quan sát tương ứng với từng sắc độ màu.

4.4. Tính tương quan của AQI

4.4.2. Tương quan giữa AQI với biến độ ẩm (Humidity)



Biểu đồ 4.9. Tương quan độ ẩm và AQI

Biểu đồ trên minh họa mối quan hệ giữa độ ẩm không khí (humidity) và chỉ số chất lượng không khí (AQI) tại TP.HCM thông qua dạng biểu đồ mật độ Hexbin. Mỗi ô lục giác thể hiện số lượng điểm dữ liệu rơi vào vùng giá trị tương ứng của hai biến, với sắc độ xanh biểu thị mật độ quan sát: ô càng đậm thì lượng điểm dữ liệu càng nhiều. Nhìn tổng thể, phần lớn các điểm tập trung trong vùng có độ ẩm từ 60% đến 85% và chỉ số AQI trong khoảng 20–50. Điều này cho thấy đa số các ngày tại TP.HCM có độ ẩm cao đi kèm với chất lượng không khí ở mức tốt đến trung bình, phản ánh môi trường tương đối trong lành trong điều kiện khí hậu ẩm.

4.4. Tính tương quan của AQI

4.4.2. Tương quan giữa AQI với biến độ ẩm (Humidity)

Giá trị hệ số tương quan Pearson = -0.313 thể hiện mối tương quan âm trung bình giữa độ ẩm và AQI. Nói cách khác, khi độ ẩm tăng, chỉ số AQI có xu hướng giảm — tức không khí trở nên sạch hơn trong những ngày có độ ẩm cao. Mức tương quan này không quá mạnh, nhưng đủ cho thấy có một mối liên hệ đáng kể giữa hai yếu tố: độ ẩm dường như đóng vai trò hỗ trợ trong việc giảm nồng độ bụi mịn và chất ô nhiễm trong không khí, thông qua quá trình ngưng tụ và rửa trôi tự nhiên.

Quan sát chi tiết biểu đồ cho thấy vùng có mật độ cao nhất nằm ở khoảng độ ẩm 70–80% và AQI 30–40, chứng tỏ đây là điều kiện khí tượng phổ biến nhất trong khu vực khảo sát. Ở phía ngược lại, khi độ ẩm giảm xuống dưới 50%, AQI bắt đầu có xu hướng tăng, có thể do không khí khô làm hạn chế khả năng lắng đọng các hạt bụi, khiến nồng độ ô nhiễm tăng cao hơn. Tuy nhiên, cũng có một số điểm ngoại lệ nằm rải rác ở vùng độ ẩm cao nhưng AQI vẫn cao, điều này có thể xuất phát từ những yếu tố khác như nguồn thải giao thông, hoạt động công nghiệp, hoặc hiện tượng nghịch nhiệt làm hạn chế khuếch tán chất ô nhiễm.

Tóm lại, biểu đồ này giúp làm rõ mối quan hệ giữa điều kiện khí tượng và chất lượng không khí tại TP.HCM. Kết quả cho thấy độ ẩm có tác động gián tiếp và tích cực đến việc cải thiện chất lượng không khí, nhưng không phải là yếu tố quyết định duy nhất. Để đánh giá chính xác hơn, cần kết hợp thêm các biến khí tượng khác như nhiệt độ, tốc độ gió, lượng mưa hoặc áp suất khí quyển.

4.4. Tính tương quan của AQI

4.4.2. Tương quan giữa AQI với biến độ ẩm (Humidity)

⁕ Liên hệ và đối chiếu kết quả giữa hai biểu đồ

Kết quả phân tích cho thấy cả hai yếu tố khí tượng — nhiệt độ và độ ẩm — đều có mối tương quan nghịch với chỉ số chất lượng không khí (AQI), song mức độ ảnh hưởng không hoàn toàn giống nhau.

Cụ thể, nhiệt độ có tương quan âm yếu với AQI cho thấy khi thời tiết ấm hơn, mức ô nhiễm có xu hướng giảm nhẹ. Hiện tượng này có thể lý giải bởi khả năng khuếch tán và đối lưu của không khí tăng lên trong điều kiện nóng, giúp các hạt bụi và chất ô nhiễm được phân tán nhanh hơn. Tuy nhiên, vì mức độ tương quan thấp nên có thể khẳng định rằng nhiệt độ chỉ đóng vai trò hỗ trợ gián tiếp, chứ không phải là yếu tố quyết định đến chất lượng không khí. Ngược lại, độ ẩm có mối tương quan âm rõ rệt hơn với AQI, phản ánh ảnh hưởng đáng kể của yếu tố này. Khi độ ẩm tăng, hơi nước trong không khí giúp kết dính và lắng đọng các hạt bụi mịn, từ đó làm giảm nồng độ chất ô nhiễm. Nhờ vậy, không khí thường trong lành hơn vào những ngày có độ ẩm cao.

Từ hai mối quan hệ trên, có thể nhận định rằng độ ẩm đóng vai trò quan trọng hơn nhiệt độ trong việc điều hòa mức độ ô nhiễm không khí tại. Dẫu vậy, ảnh hưởng của khí tượng chỉ là một phần trong hệ thống các yếu tố tác động đến AQI. Chất lượng không khí thực tế còn chịu chi phối bởi các nguồn thải giao thông, công nghiệp, xây dựng và mật độ dân cư. Do đó, việc phân tích yếu tố khí tượng giúp hỗ trợ hiểu rõ xu hướng biến động tự nhiên của ô nhiễm, nhưng cần được kết hợp với biện pháp kiểm soát phát thải nhân tạo để hướng đến mục tiêu quản lý môi trường đô thị hiệu quả và bền vững hơn.

Chương 5:

5.1. Kết quả xử lý

5.2. Phát hiện chính

5.3. Giải pháp

5.1. Kết quả xử lý

Quy trình tiền xử lý dữ liệu được thực hiện chặt chẽ trên toàn bộ chuỗi quan trắc của trạm 6, bảo đảm đồng nhất tên cột và chuẩn hóa cột thời gian về một định dạng duy nhất. Dữ liệu gốc có chỗ gián đoạn nên nhóm chuyển các giá trị 0 và ô rỗng về NaN, sau đó điền thiếu theo thứ tự: ưu tiên lấy từ trạm 2 và trạm 4 → lấy trung bình theo ngày của trạm 6 → nội suy theo thời gian; nhờ vậy các biến còn thiếu chính của trạm 6 được phục hồi đầy đủ (số dòng thiếu = 0) và chuỗi thời gian được làm đầy ở mức sử dụng được cho phân tích. Các ngoại lệ ở các biến môi trường quan trọng (TSP, PM2.5, CO, SO_2, NO_2, O_3, nhiệt độ, độ ẩm) được phát hiện bằng IQR và Isolation Forest và không bị xoá mà được thay thế bằng giá trị cùng thời điểm, nên cấu trúc chuỗi vẫn được bảo toàn. Sau làm sạch, dữ liệu được chuẩn hóa thang đo khi đưa vào mô hình và được bổ sung các biến bối cảnh như Season và bộ chỉ số AQI theo quy định hiện hành, giúp phân tích theo mùa và đánh giá nhanh mức độ ô nhiễm. Kiểm tra đa cộng tuyến cho thấy hệ biến nhìn chung ổn định; chỉ cặp TSP–PM2.5 (và nhẹ hơn là CO–SO_2) có tín hiệu trùng lặp nên cần cân nhắc khi cùng đưa vào mô hình.

5.2. Phát hiện chính

Qua quá trình phân tích và trực quan hóa dữ liệu chất lượng không khí tại Thành phố Hồ Chí Minh trong giai đoạn khảo sát, có thể nhận thấy nhiều đặc điểm đáng chú ý phản ánh tình trạng ô nhiễm không khí đô thị hiện nay. Trước hết, TP.HCM thường xuyên duy trì chất lượng không khí ở mức trung bình đến kém. Điều này cho thấy chất lượng không khí tại khu vực TP.HCM chưa đạt mức an toàn cho sức khỏe cộng đồng, đặc biệt trong các tháng đầu và cuối năm khi nồng độ bụi mịn và khí độc có xu hướng tăng cao.

Về thành phần các chất ô nhiễm, kết quả cho thấy CO là chất chiếm tỷ lệ đóng góp lớn nhất trong tổng nồng độ trung bình các chất, tiếp theo là SO_2, NO_2 và O_3, trong khi $PM_{2.5}$ tuy chiếm tỷ lệ nhỏ hơn nhưng lại đặc biệt nguy hiểm đối với sức khỏe. Bụi mịn $PM_{2.5}$ có khả năng xâm nhập sâu vào phổi, gây ảnh hưởng đến hệ hô hấp và tim mạch, chủ yếu bắt nguồn từ hoạt động giao thông, công nghiệp và đốt nhiên liệu. Nồng độ NO_2 và CO cũng ở mức đáng lưu ý, phản ánh tác động rõ rệt của mật độ phương tiện giao thông dày đặc tại khu vực nội thành.

5.2. Phát hiện chính

Phân tích theo mùa cho thấy mùa khô có mức ô nhiễm cao hơn đáng kể so với mùa mưa. Nguyên nhân có thể do vào mùa khô, thời tiết khô nóng, ít mưa và gió yếu khiến các chất ô nhiễm khó khuếch tán, trong khi hoạt động giao thông và công nghiệp vẫn duy trì ở mức cao. Ngược lại, mùa mưa với lượng mưa lớn và độ ẩm cao giúp rửa trôi bụi mịn, làm giảm nồng độ các khí độc trong không khí.

Tổng quan lại, dữ liệu cho thấy ô nhiễm không khí tại TP.HCM mang tính chu kỳ theo mùa, với xu hướng nặng hơn vào các tháng khô, và nguồn phát thải chính đến từ bụi mịn và khí thải giao thông. Kết quả này phản ánh mối liên hệ chặt chẽ giữa điều kiện khí hậu, thời tiết, hoạt động đô thị và chất lượng không khí. Đây là cơ sở khoa học quan trọng để đề xuất các nhóm giải pháp quản lý, giảm phát thải và nâng cao chất lượng môi trường không khí được trình bày trong phần tiếp theo.

5.3. Giải pháp

Giải pháp quản lý & giám sát

TP.HCM cần tăng cường năng lực giám sát và quản lý chất lượng không khí thông qua việc mở rộng mạng lưới trạm quan trắc tự động đến các khu vực chưa được phủ sóng dữ liệu. Việc này giúp thành phố có bức tranh toàn diện hơn về tình trạng ô nhiễm theo thời gian và không gian. Cùng với đó, cần chuẩn hóa hệ thống dữ liệu môi trường giữa các sở, ngành và xây dựng cổng dữ liệu mở (open data), để các tổ chức nghiên cứu, doanh nghiệp và người dân cùng truy cập, giám sát và đề xuất sáng kiến cải thiện.

Ngoài ra, việc ứng dụng công nghệ AI và IoT trong dự báo, cảnh báo sớm các đợt ô nhiễm dựa trên dữ liệu AQI, thời tiết và lưu lượng giao thông sẽ giúp cơ quan quản lý chủ động hơn trong ứng phó. Thành phố cũng cần tăng cường kiểm định khí thải phương tiện giao thông và cơ sở công nghiệp, đặc biệt tại khu vực nội thành, nơi mật độ giao thông và dân cư cao. Những biện pháp này tạo nền tảng quản trị dữ liệu môi trường hiệu quả, hỗ trợ quá trình ra quyết định chính xác và kịp thời.

5.3. Giải pháp

Giải pháp giảm phát thải từ giao thông

Giao thông đường bộ là nguồn phát thải lớn nhất tại TP.HCM, đóng góp chủ yếu vào nồng độ $PM_{2.5}$, NO_2 và CO. Vì vậy, việc giảm ô nhiễm từ giao thông cần được xem là trọng tâm. Trước hết, thành phố nên phát triển mạnh hệ thống giao thông công cộng sạch như xe buýt điện, metro, tàu điện mặt đất và xe đạp công cộng. Cần ưu tiên làn đường riêng cho phương tiện công cộng nhằm giảm kẹt xe, một nguyên nhân khiến phát thải tăng đột biến vào giờ cao điểm.

Bên cạnh đó, cần khuyến khích người dân chuyển sang sử dụng phương tiện điện thông qua chính sách ưu đãi thuế, phí gửi xe hoặc hỗ trợ đổi xe cũ lấy xe điện. Mô hình "khu vực không khí sạch" hoặc thu phí vào trung tâm như các thành phố châu Âu có thể được xem xét áp dụng tại một số quận nội thành, nhằm hạn chế phương tiện chạy xăng dầu. Khi giao thông sạch và thông suốt hơn, không chỉ môi trường mà chất lượng sống, năng suất lao động và hình ảnh đô thị cũng được nâng cao.

5.3. Giải pháp

Giải pháp kiểm soát công nghiệp và xây dựng

Công nghiệp và xây dựng cũng là nguồn phát thải lớn gây ô nhiễm bụi mịn và khí độc (SO_2, NO_2). TP.HCM cần thực hiện kiểm soát nghiêm ngặt nguồn phát thải công nghiệp, yêu cầu các cơ sở sản xuất phải lắp đặt hệ thống lọc bụi, lọc khí độc đạt chuẩn quốc gia, đồng thời giám sát trực tuyến thông qua cảm biến kết nối trung tâm dữ liệu môi trường. Đối với lĩnh vực xây dựng, cần siết chặt quản lý bụi từ quá trình thi công, vận chuyển vật liệu, đồng thời xử phạt nghiêm các đơn vị không tuân thủ quy định che chắn, phun nước giảm bụi.

Về lâu dài, thành phố nên phát triển các khu công nghiệp xanh, áp dụng mô hình tuần hoàn tài nguyên và di dời nhà máy gây ô nhiễm khỏi khu dân cư, nhằm giảm thiểu tác động trực tiếp đến đời sống người dân.

5.3. Giải pháp

Giải pháp cộng đồng và nâng cao nhận thức

Bên cạnh chính sách và công nghệ, sự tham gia của cộng đồng đóng vai trò quyết định. Thành phố cần đẩy mạnh tuyên truyền thói quen sống xanh như sử dụng giao thông công cộng, hạn chế đốt rác, tiết kiệm điện và giảm sử dụng túi ni-lông. Việc xây dựng ứng dụng di động theo dõi AQI cá nhân, giúp người dân nắm bắt chất lượng không khí theo thời gian thực và nhận cảnh báo khi AQI vượt ngưỡng nguy hiểm, sẽ nâng cao nhận thức và khả năng tự bảo vệ sức khỏe.

Ngoài ra, có thể triển khai các chiến dịch xã hội như "Hồ Chí Minh xanh – không khí sạch", khuyến khích doanh nghiệp, trường học và tổ chức xã hội tham gia. Khi người dân được trang bị thông tin và có hành động cụ thể, quá trình cải thiện môi trường sẽ trở nên hiệu quả và bền vững hơn.

5.3. Giải pháp

Giải pháp quy hoạch đô thị và cây xanh

Quy hoạch đô thị có ảnh hưởng lớn đến chất lượng không khí. TP.HCM cần tăng mật độ cây xanh, hành lang gió và không gian mở, đặc biệt ở các khu vực có mật độ dân cư cao và dọc các trục giao thông chính. Cây xanh không chỉ hấp thụ bụi mịn, khí độc mà còn giúp giảm nhiệt độ đô thị, cải thiện vi khí hậu và tạo cảnh quan bền vững.

Song song, thành phố cần thiết kế quy hoạch đô thị thông minh, đảm bảo lưu thông không khí và hạn chế hiệu ứng đảo nhiệt. Có thể tích hợp vật liệu hấp thụ bụi mịn và khí độc như sơn sinh học, tường cây xanh, bê tông thấm trong hạ tầng đô thị. Việc phát triển "thành phố xanh và bền vững" không chỉ cải thiện chất lượng không khí mà còn góp phần xây dựng hình ảnh TP.HCM hiện đại, đáng sống.

5.3. Giải pháp

Giải pháp công nghệ và đổi mới sáng tạo

Việc ứng dụng công nghệ hiện đại là chìa khóa để nâng cao năng lực quản lý môi trường đô thị. TP.HCM nên triển khai các mô hình dự báo AQI bằng Machine Learning, sử dụng dữ liệu từ quan trắc, khí tượng và giao thông để dự đoán xu hướng ô nhiễm theo thời gian và mùa. Cùng với đó, phát triển bản đồ ô nhiễm thời gian thực giúp người dân và cơ quan chức năng theo dõi biến động môi trường tức thì.

Thành phố cũng nên tích hợp dữ liệu liên ngành, bao gồm khí tượng, giao thông, công nghiệp và dân cư, nhằm xác định nguồn ô nhiễm chính và đề xuất biện pháp kiểm soát phù hợp. Cuối cùng, việc xây dựng dashboard giám sát thông minh cho Sở Tài nguyên và Môi trường sẽ giúp cơ quan quản lý nhanh chóng phát hiện bất thường và đưa ra cảnh báo kịp thời.

5.3. Giải pháp

Kiến nghị chính sách

Về mặt thể chế, TP.HCM cần xây dựng bộ quy chuẩn khí thải riêng, phù hợp với đặc thù đô thị lớn, mật độ dân cư cao và phương tiện giao thông đông đúc. Cần tăng cường hợp tác quốc tế trong lĩnh vực công nghệ môi trường để học hỏi mô hình thành công từ các thành phố như Singapore, Seoul hay Tokyo. Đồng thời, nên khuyến khích đầu tư tư nhân vào các dự án năng lượng sạch, giao thông xanh và công nghệ xử lý khí thải thông minh. Chính sách tài chính xanh và ưu đãi cho doanh nghiệp thân thiện môi trường cũng là động lực thúc đẩy chuyển đổi xanh.

5.3. Giải pháp

Tổng kết

Kết quả phân tích dữ liệu chất lượng không khí tại TP.HCM cho thấy tình trạng ô nhiễm đã và đang trở thành một vấn đề môi trường nghiêm trọng. Các chỉ số AQI trong nhiều giai đoạn vượt ngưỡng an toàn, đặc biệt trong mùa khô, khi các yếu tố khí tượng bất lợi khiến nồng độ bụi mịn và khí độc khó khuếch tán. Trong các chất gây ô nhiễm, CO, SO_2, NO_2 và O_3 là những thành phần chiếm tỷ trọng cao nhất, trong khi $PM_{2.5}$ tuy chiếm tỷ lệ nhỏ hơn nhưng lại đặc biệt nguy hiểm về mặt sức khỏe, phản ánh rõ đặc trưng phát thải từ hoạt động giao thông, công nghiệp và xây dựng đô thị. Những dữ liệu này không chỉ thể hiện hiện trạng mà còn là cơ sở khoa học quan trọng để đề xuất các chính sách và biện pháp phù hợp, hướng đến cải thiện chất lượng không khí một cách bền vững.

5.3. Giải pháp

Tổng kết

Các giải pháp được trình bày ở trên cho thấy, để giải quyết vấn đề ô nhiễm không khí tại TP.HCM, cần sự kết hợp đồng bộ giữa công nghệ, quản lý và cộng đồng. Trong đó, việc xây dựng hệ thống giám sát và quản lý dữ liệu môi trường thông minh là nền tảng giúp chính quyền đưa ra quyết định nhanh chóng, chính xác. Cùng lúc, các biện pháp giảm phát thải giao thông, kiểm soát công nghiệp và tăng mảng xanh đô thị sẽ mang lại hiệu quả trực tiếp trong việc hạ nồng độ bụi mịn và khí độc. Ở tầm cộng đồng, việc nâng cao nhận thức và thay đổi hành vi người dân đóng vai trò then chốt giúp các chính sách được duy trì lâu dài và tạo chuyển biến thực sự.

Bên cạnh đó, TP.HCM cần tận dụng dữ liệu phân tích để phục vụ hoạch định chính sách và quy hoạch đô thị trong tương lai. Việc ứng dụng AI, IoT và Machine Learning trong dự báo ô nhiễm không chỉ giúp dự đoán xu hướng AQI theo thời gian và khu vực mà còn góp phần định hình mô hình quản lý môi trường chủ động. Sự phối hợp giữa các cơ quan quản lý, doanh nghiệp công nghệ và các tổ chức quốc tế sẽ là yếu tố quyết định trong việc tạo ra một hệ sinh thái "quản trị không khí thông minh", nơi dữ liệu được sử dụng hiệu quả để cải thiện chất lượng sống của người dân.

5.3. Giải pháp

Tổng kết

Tóm lại, các giải pháp nêu trong chương này không chỉ nhằm giảm thiểu ô nhiễm hiện tại mà còn hướng tới xây dựng năng lực quản lý môi trường hiện đại, hỗ trợ quá trình phát triển đô thị xanh và bền vững. Từ dữ liệu phân tích đến hành động chính sách, mỗi bước đi đều thể hiện sự gắn kết giữa khoa học, công nghệ và trách nhiệm xã hội. Nếu được triển khai đồng bộ, TP.HCM hoàn toàn có thể tiến tới mục tiêu trở thành "Thành phố xanh – không khí sạch – phát triển bền vững", nơi tăng trưởng kinh tế đi đôi với chất lượng cuộc sống và bảo vệ môi trường lâu dài.

Tài liệu tham khảo

1. World Health Organization, “Ambient (outdoor) air pollution: Key facts,” WHO Fact Sheets, 24 Oct. 2024. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/ambient-(outdoor)-air-quality-and-health.

2. Nhân Dân điện tử, “Bài 1: Báo động đỏ ô nhiễm không khí ở các đô thị lớn,” Nhân Dân, 22 Apr. 2024. [Online]. Available: https://nhandan.vn/bai-1-bao-dong-do-o-nhiem-khong-khi-o-cac-do-thi-lon-post805938.html.

3. Cơ quan Môi trường (CEM), “Kiến tạo mạng lưới quan trắc môi trường không khí hiện đại và đồng bộ,” CEM.gov.vn, [Online]. Available: https://cem.gov.vn/tin-tuc-moi-truong/kien-tao-mang-luoi-quan-trac-moi-truong-khong-khi-hien-dai-va-dong-bo.