TRƯỜNG ĐẠI HỌC THỦY LỢI



NHÓM 04

BÁO CÁO BÀI TẬP LỚN

HỌC PHẦN TIỀN XỬ LÝ DỮ LIỆU

DỰ ĐOÁN NỒNG ĐỘ Ô NHIỄM KHÔNG KHÍ

|  |  |
| --- | --- |
| NGƯỜI HƯỚNG DẪN | 1. TS. TẠ QUANG CHIỂU |
| Tên : Nguyễn Văn Quân  Lớp: 64TTNT1  MSV: 2251262627 |  |

Hà Nội, năm 2025

MỤC LỤC

[MỞ ĐẦU iii](#_Toc195131751)

[CHƯƠNG 1 GIỚI THIỆU BÀI TOÁN 1](#_Toc195131752)

[CHƯƠNG 2 TIỀN XỬ LÝ DỮ LIỆU 3](#_Toc195131753)

[1. Giới thiệu tập dữ liệu 3](#_Toc195131754)

[2. Tiền xử lý 4](#_Toc195131755)

[2.1 Xử lý dữ liệu thiếu 5](#_Toc195131756)

[2.1.1 Lý thuyết 5](#_Toc195131757)

[2.1.2 Triển khai 6](#_Toc195131758)

[2.1.3 Kết luận 10](#_Toc195131759)

[2.2 Chuẩn hóa 16](#_Toc195131760)

[2.3 Xử lý ngoại lai 17](#_Toc195131761)

[2.3.1 Kiểm tra ngoại lai 17](#_Toc195131762)

[2.3.2. Xử lý ngoại lai 21](#_Toc195131763)

[2.4 Giảm chiều dữ liệu 11](#_Toc195131764)

[2.4.1 Lý thuyết 11](#_Toc195131765)

[2.4.2 Áp dụng 13](#_Toc195131766)

[CHƯƠNG 3 XÂY DỰNG MÔ HÌNH 25](#_Toc195131767)

[3.1 Mô hình Random Forest kết hợp Rolling Forecast 25](#_Toc195131768)

[3.1.1 Lý thuyết 25](#_Toc195131769)

[3.1.2 Lý thuyết 25](#_Toc195131770)

[3.1.3 Triển khai 26](#_Toc195131771)

[3.2 Mô hình Hồi quy tuyến tính 30](#_Toc195131772)

[3.2.1 Lý thuyết 30](#_Toc195131773)

[3.2.2 Triền khai 31](#_Toc195131774)

[3.2.3 So sánh kết quả của mô hình với dữ liệu thực tế 33](#_Toc195131775)

[3.2.4 Kết luận 35](#_Toc195131780)

[CHƯƠNG 4 TỔNG KẾT 36](#_Toc195131781)

[TÀI LIỆU THAM KHẢO 37](#_Toc195131782)

MỞ ĐẦU

Trong những năm gần đây, vấn đề ô nhiễm không khí đang trở nên ngày càng nghiêm trọng, đặc biệt tại các khu vực đô thị và công nghiệp phát triển nhanh. Nồng độ các chất ô nhiễm như CO (Carbon Monoxide), NOx (Nitrogen Oxides), NO₂ (Nitrogen Dioxide) có thể ảnh hưởng trực tiếp đến sức khỏe con người, gây ra nhiều bệnh lý về hô hấp, tim mạch và làm suy giảm chất lượng sống. Vì vậy, việc dự đoán nồng độ các chất ô nhiễm trong không khí là cần thiết nhằm hỗ trợ công tác giám sát môi trường, đưa ra cảnh báo sớm và xây dựng các chính sách bảo vệ sức khỏe cộng đồng.

Tuy nhiên, trong thực tế, dữ liệu môi trường thường không đầy đủ, xuất hiện giá trị thiếu, nhiễu hoặc sai lệch do lỗi thiết bị, điều kiện thời tiết hoặc giới hạn kỹ thuật của hệ thống quan trắc. Điều này đặt ra yêu cầu quan trọng trong việc **tiền xử lý dữ liệu** trước khi tiến hành phân tích hoặc huấn luyện mô hình dự đoán. Việc xử lý các giá trị thiếu, chuẩn hóa dữ liệu, biến đổi đặc trưng và phát hiện outlier là những bước không thể thiếu để đảm bảo chất lượng và độ chính xác của mô hình.

Trong khuôn khổ môn học **Tiền xử lý dữ liệu**, em thực hiện đề tài **“Dự đoán nồng độ ô nhiễm không khí”** với mục tiêu chính là tìm hiểu và áp dụng các kỹ thuật xử lý dữ liệu trong thực tiễn, đồng thời kết hợp trực quan hóa để phân tích và đánh giá chất lượng dữ liệu sau tiền xử lý. Từ đó, em tiến hành xây dựng mô hình học máy nhằm dự đoán nồng độ của một số chất ô nhiễm trong tương lai, góp phần đưa ra cái nhìn cụ thể hơn về tầm quan trọng của tiền xử lý trong toàn bộ quy trình phân tích dữ liệu.

# GIỚI THIỆU BÀI TOÁN

1. **Giới thiệu bài toán**

Dữ liệu môi trường là một trong những dạng dữ liệu quan trọng nhưng thường gặp nhiều thách thức trong quá trình xử lý do đặc tính liên tục theo thời gian, dễ bị ảnh hưởng bởi yếu tố thời tiết, thiết bị đo đạc, hoặc điều kiện thu thập thực tế. Trong đó, dữ liệu liên quan đến nồng độ các chất ô nhiễm không khí như **CO (Carbon Monoxide)**, **NOx (Nitrogen Oxides)** và **NO₂ (Nitrogen Dioxide)** đóng vai trò quan trọng trong việc đánh giá chất lượng không khí và sức khỏe cộng đồng.

Bài toán đặt ra là làm thế nào để từ **dữ liệu đo đạc chưa hoàn chỉnh, chứa nhiều giá trị thiếu và biến động**, ta có thể xử lý, chuẩn hóa và khai thác một cách hiệu quả để xây dựng mô hình dự đoán chính xác nồng độ các chất ô nhiễm trong tương lai. Việc giải quyết bài toán không chỉ giúp tăng chất lượng dữ liệu đầu vào cho các mô hình học máy, mà còn giúp rút ra được các xu hướng và yếu tố ảnh hưởng đến mức độ ô nhiễm.

Trong đề tài này, em sử dụng một tập dữ liệu thực tế chứa thông tin về nồng độ ô nhiễm và thời gian đo đạc, trong đó có nhiều giá trị bị thiếu, bị nhiễu. Việc xử lý và biến đổi dữ liệu đúng cách sẽ đóng vai trò then chốt trong việc đảm bảo mô hình dự đoán hoạt động hiệu quả và phản ánh đúng xu hướng của hiện tượng ô nhiễm không khí.

1. **Mục tiêu**

Mục tiêu của đề tài là vận dụng các kiến thức về **tiền xử lý dữ liệu** để xử lý và chuẩn bị dữ liệu môi trường phục vụ cho việc xây dựng mô hình dự đoán nồng độ ô nhiễm không khí. Các mục tiêu cụ thể bao gồm:

1. **Tìm hiểu và phân tích dữ liệu đo đạc chất lượng không khí,** bao gồm các chỉ số ô nhiễm và thông tin thời gian đo đạc.
2. **Tiền xử lý dữ liệu đầu vào**, bao gồm:
   * Xử lý các giá trị thiếu (-200) bằng nhiều phương pháp như: trung bình, trung vị, nội suy tuyến tính, KNN, forward fill.
   * Chuẩn hóa dữ liệu bằng các kỹ thuật như StandardScaler.
   * Biến đổi đặc trưng từ thời gian.
   * Tìm kiếm ngoại lai bằng các phương pháp như IQR, z-score, Isolation Forest, thực hiện xử lý ngoại lai bằng nhiều phương pháp như điền trung vị, giới hạn IQR và xử lý bằng z-socre Liner.
   * Thực hiện giảm chiều dữ liệu bằng phương pháp Permutation Importance trong Random Forest.
3. **Thực hiện trực quan hóa dữ liệu** để đánh giá hiệu quả của các phương pháp xử lý giá trị thiếu và chuẩn hóa.
4. **Xây dựng mô hình học máy** để dự đoán nồng độ các chất ô nhiễm theo thời gian
5. **Đánh giá và so sánh hiệu suất mô hình** dựa trên dữ liệu đã được xử lý bằng các phương pháp khác nhau.

# TIỀN XỬ LÝ DỮ LIỆU

## Giới thiệu tập dữ liệu

Trong đề tài này, nhóm sử dụng tập dữ liệu **Air Quality** được công bố bởi **S. De Vito (2008)** và lưu trữ tại **UCI Machine Learning Repository** – một kho dữ liệu nổi tiếng dùng trong nghiên cứu và giảng dạy về học máy. Dữ liệu được thu thập tại một trạm quan trắc môi trường ở thành phố **Lucca, Ý**, trong khoảng thời gian từ tháng 3 đến tháng 4 năm 2004, với chu kỳ đo mỗi giờ.

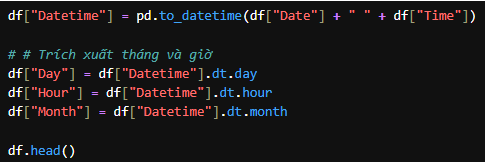
* 1. **Nguồn dữ liệu:**Vito, S. (2008). Air Quality [Dataset]. UCI Machine Learning Repository. [https://doi.org/10.24432/C59K5F](https://doi.org/10.24432/C59K5F" \t "_new).
  2. **Mô tả chung:**Tập dữ liệu gồm hơn **9358 bản ghi** (mỗi bản ghi là một giờ đo), với **15 thuộc tính**, bao gồm các chỉ số liên quan đến nồng độ các chất ô nhiễm như **CO (Carbon Monoxide)**, **NOx (Nitrogen Oxides)**, **NO₂ (Nitrogen Dioxide)**,... cùng với các thông số môi trường khác như nhiệt độ, độ ẩm, áp suất,…
  3. Các thuộc tính:

|  |  |
| --- | --- |
| Tên thuộc tính | Ý nghía |
| Date | Ngày/Tháng/Năm |
| Time | Giờ |
| CO(GT) | Nồng độ CO trung bình thực tế theo giờ tính bằng mg/m^3 |
| PT08.S1(CO) | Phản ứng cảm biến trung bình theo giờ (CO là mục tiêu) |
| NMHC(GT) | Nồng độ NMHC trung bình thực tế theo giờ tính bằng microg/m^3 |
| C6H6(GT) | Nồng độ Benzen trung bình thực tế theo giờ tính bằng microg/m^3 |
| PT08.S2 (NMHC) | Phản hồi cảm biến trung bình theo giờ (mục tiêu là NMHC) |
| NOx(GT) | Nồng độ NOx trung bình thực tế theo giờ tính bằng ppb |
| PT08.S3(NOx) | Phản ứng cảm biến trung bình theo giờ (NOx là mục tiêu) |
| NO2(GT) | Nồng độ NO2 trung bình thực tế theo giờ tính bằng microg/m^3 |
| PT08.S5(O3) | Phản ứng cảm biến trung bình theo giờ (O3 là mục tiêu) |
| T | Nhiệt độ |
| RH | Độ ẩm tương đổi |
| PT08.S4(NO2) | Phản ứng cảm biến trung bình theo giờ (NO2 là mục tiêu) |
| AG | Độ ẩm tuyệt đối |

Trong tập dữ liệu này, các giá trị bị thiếu không được ghi là NaN mà được mã hóa bằng **-200**. Do đó, cần phát hiện và xử lý chúng trong giai đoạn tiền xử lý dữ liệu.

## Tiền xử lý

Để thay thế cho dữ liệu Date và Time trong tập dữ liệu chúng em tách phần Date và Time thành các thuộc tính mới là Month, Day, Hour và xóa đi các cột Date và Time.



## Xử lý dữ liệu thiếu

Dữ liệu thiếu (missing data) là một hiện tượng phổ biến trong các tập dữ liệu thực tế, xảy ra khi một hoặc nhiều giá trị trong tập dữ liệu không được ghi nhận hoặc bị loại bỏ trong quá trình thu thập, lưu trữ hoặc xử lý. Nếu không được xử lý phù hợp, dữ liệu thiếu có thể gây sai lệch trong phân tích thống kê, giảm hiệu suất của các mô hình học máy, hoặc thậm chí khiến mô hình không thể huấn luyện được.

Các dạng thiếu dữ liệu phổ biến:

* **MCAR (Missing Completely at Random):** Thiếu ngẫu nhiên hoàn toàn, không liên quan đến bất kỳ biến nào trong dữ liệu.
* **MAR (Missing at Random):** Thiếu phụ thuộc vào các biến quan sát được, nhưng không phụ thuộc vào chính giá trị bị thiếu.
* **MNAR (Missing Not at Random):** Thiếu không ngẫu nhiên, phụ thuộc vào chính giá trị bị thiếu hoặc các giá trị không quan sát được.

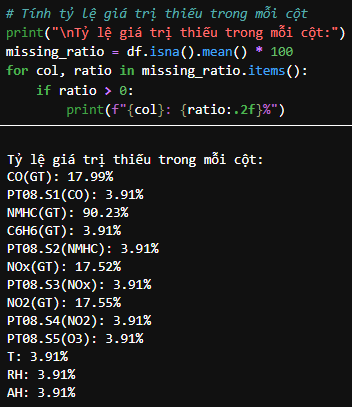
### Lý thuyết

Các phương pháp điền dữ liệu thiếu (imputation) hoạt động dựa trên giả định rằng có thể ước lượng các giá trị bị thiếu từ phần dữ liệu quan sát được. Mục tiêu là thay thế các ô trống (NaN) bằng các giá trị hợp lý để duy trì kích thước và tính toàn vẹn của tập dữ liệu. Có thể kể đến các phương pháp điền dữ liệu thiếu như:

* 1. **Trung bình** **(Mean Imputation):** Phương pháp này điền các giá trị thiếu bằng giá trị trung bình của cột tương ứng, tính trên các giá trị không bị thiếu.
  2. **Trung vị (Median Imputation):** Tương tự Mean, nhưng sử dụng giá trị trung vị – tức là giá trị ở giữa sau khi sắp xếp các giá trị hợp lệ.
  3. **Forward Fill (Điền theo giá trị trước đó):** Điền các giá trị thiếu bằng cách **sao chép giá trị gần nhất phía trước** trong cột đó. Phương pháp này phù hợp cho dữ liệu chuỗi thời gian.
  4. **Interpolation (Nội suy tuyến tính):** Dựa trên hai giá trị hợp lệ liền kề trước và sau vị trí bị thiếu để nội suy giá trị còn thiếu theo công thức tuyến tính, thường được sử dụng cho dữ liệu theo thời gian (time series) hoặc có thứ tự rõ ràng.
  5. **KNN Imputation:** Cũng gần giống như phương pháp Interpolation, nhưng thay vì lấy 2 giá trị liền kề thì KNN Imputaion tìm K hàng "gần giống nhất" (theo khoảng cách) và lấy trung bình từ đó để điền giá trị thiếu.

### Triển khai

* **Kiểm tra dữ liệu thiếu**

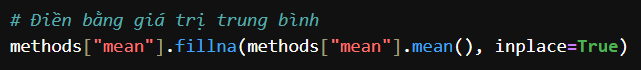


Nhận xét:

* **CO(GT)**, **NOx(GT)** và **NO2(GT)** đều có tỷ lệ thiếu quanh **17%**, khá đáng kể nhưng vẫn trong phạm vi xử lý được.
* **NMHC(GT)** có tỷ lệ thiếu cực cao (**96.23%**) → không đủ dữ liệu để nội suy hay học, nên chúng em **loại bỏ khỏi tập dữ liệu**.
* Các cột khác chỉ thiếu ~3.91% → xử lý đơn giản, không ảnh hưởng nhiều.

Trong phần điền dữ liệu thiếu này em áp dụng các phương pháp điền dữ liệu thiếu bằng giá trị Trung bình (Mean Imputation), Trung vị (Median Imputation), Forward Fill (Điền theo giá trị trước đó), Interpolation (Nội suy tuyến tính), KNN Imputation. Sau đó em thay vì sử dụng toàn bộ dữ liệu để so sánh phương pháp nào tốt hơn thì chúng em sử dụng đặc trưng CO(GT) để đánh giá độ hiệu quả của các phương pháp.

* **Trung bình (Mean Imputation)**



* **Trung vị (Median Imputauon)**



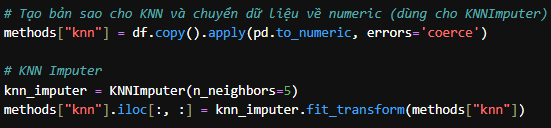
* **Forward Fill (Điền theo giá trị trước đó)**



* **Interpolation (Nội suy tuyến tính)**

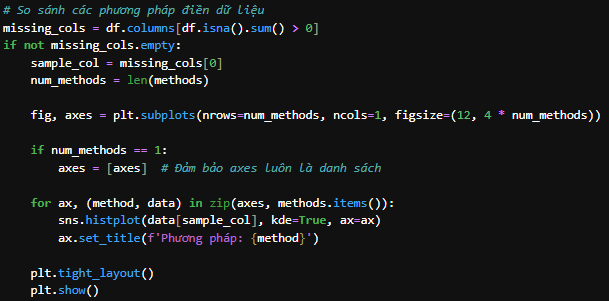


* **KNN Impulation**

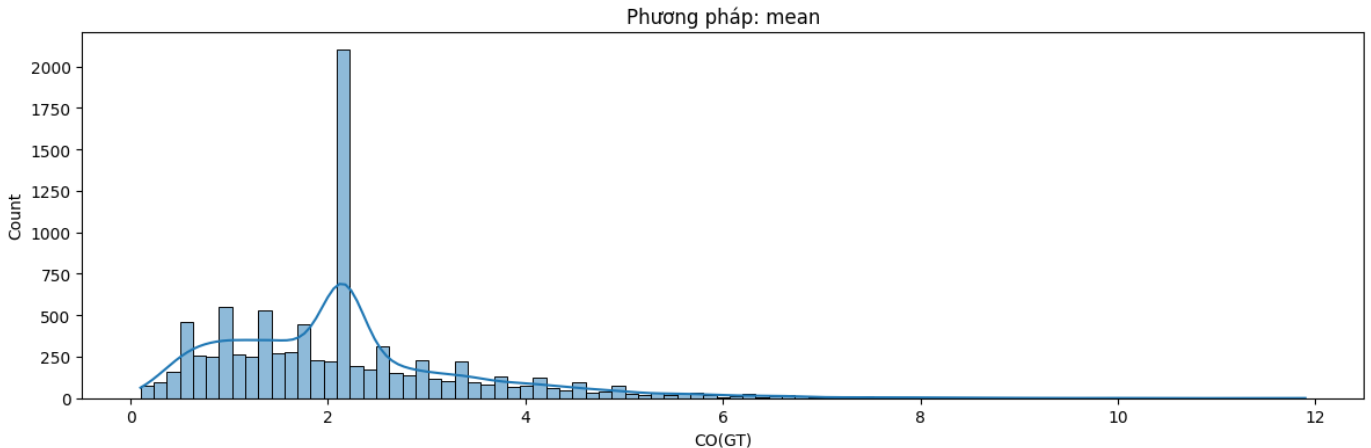


* So sánh các phương pháp

Để so sánh độ hiệu quả của các phương pháp em vẽ biểu đồ histogram để xem xét phân phối của dữ liệu từ đó chọn ra phương pháp tốt nhất.



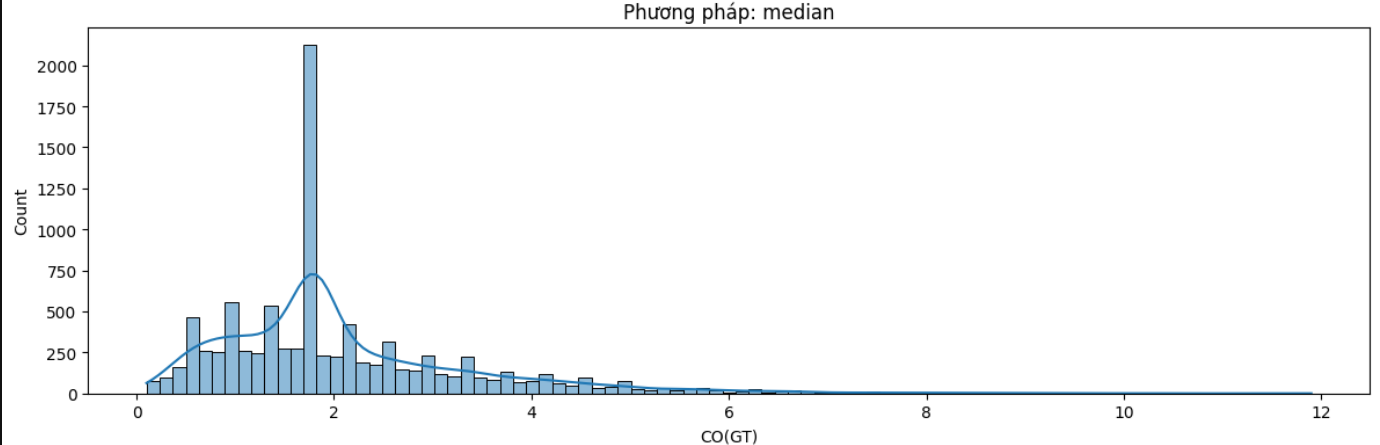
1. **Trung bình (Mean Imputation)**



**Nhận xét:** Có một **cột rất cao** tại giá trị khoảng **2.0**, biểu hiện rõ là hầu hết giá trị thiếu được thay thế bằng đúng giá trị trung bình, làm **biến dạng phân phối** gốc, gây lệch và mất tự nhiên, tạo ra “spike” không thật.

* Phương pháp đơn giản nhưng kém hiệu quả, không giữ được tính phân phối gốc.

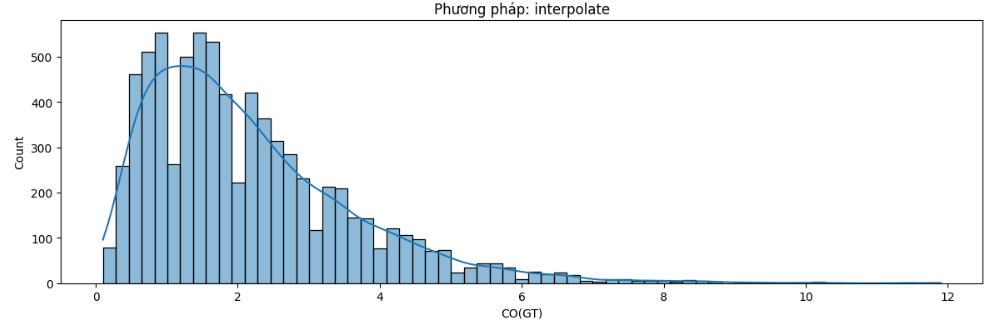
1. **Trung vị (Median Imputauon)**



**Nhận xét:** Tương tự phương pháp điền bằng giá trị trung bình, cũng xuất hiện một **đỉnh nhọn** gần giá trị trung vị (~2.0), làm mất tự nhiên. So với Mean: Ít bị ảnh hưởng bởi ngoại lệ hơn, nhưng vẫn gây **tụ điểm giá trị** nhân tạo.

* Vẫn gây lệch giá trị phân phối

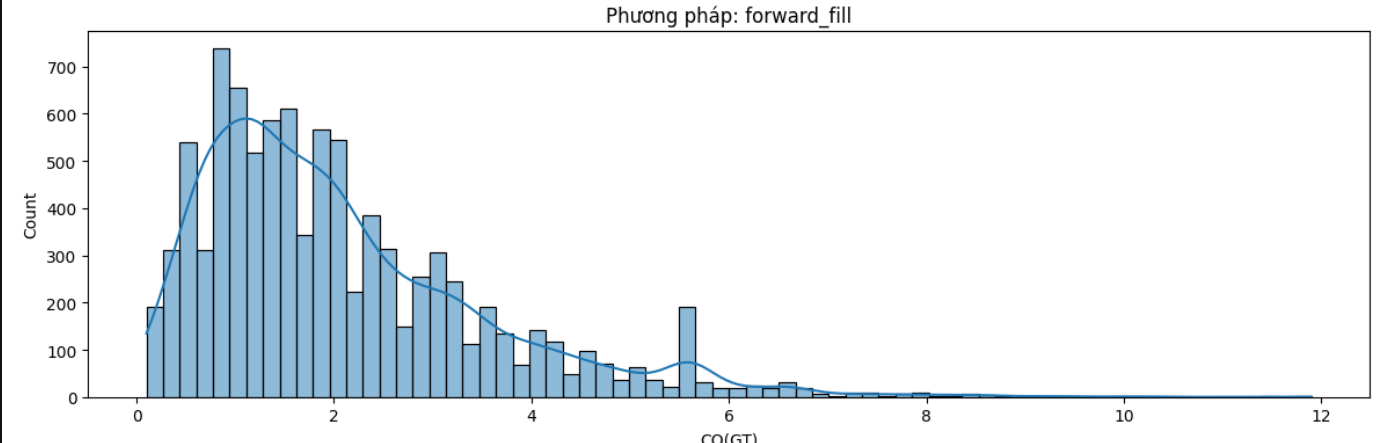
1. **Interpolation (Nội suy tuyến tính)**



**Nhận xét:** Phân phối lệch phải, với đỉnh chính quanh 1-2, và ít giá trị ngoại lai hơn so với Mean và Median. Các giá trị CO hiếm khi vượt quá 8.

* Phân phối lệch phải, với đỉnh chính quanh 1-2, và ít giá trị ngoại lai hơn so với Mean và Median. Các giá trị CO hiếm khi vượt quá 8.

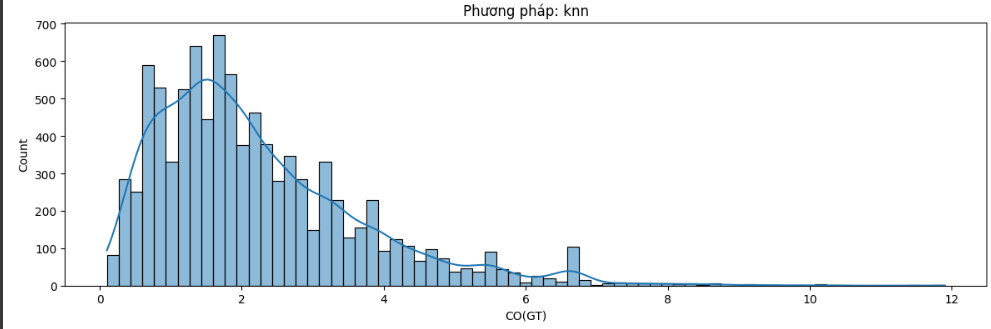
1. **Forward Fill (Điền theo giá trị trước đó)**



**Nhận xét**: Phân phối lệch phải, với đỉnh chính quanh 1-2, nhưng có một số cụm giá trị bất thường (ví dụ, quanh CO = 6 và CO > 8). Số lượng giá trị ngoại lai ít hơn Mean nhưng nhiều hơn Interpolate.

* Phương pháp Forward có vẻ không hiệu quả bằng Interpolate, vì vẫn có các cụm giá trị bất thường.

1. **KNN Impulation**



**Nhận xét:** Phân phối **mượt, đa dạng giá trị**, gần giống với interpolate, nhưng có thể thấy vẫn hơi **lốm đốm** do đặc điểm của KNN.

* Tốt hơn so với Mean, Median và Forward Fill nhưng vẫn còn một ít ngoại lai.

### Kết luận

Sau khi so sánh trực quan các phương pháp điền khuyết giá trị bao gồm: Mean, Median, Forward Fill, KNN và Interpolation trên cột dữ liệu CO(GT), có thể thấy rằng:

* Các phương pháp **Mean** và **Median** tạo ra sự tụ điểm giá trị nhân tạo, làm biến dạng phân phối gốc của dữ liệu.
* **Forward Fill** giữ được tính liên tục nhưng có thể làm mất đi xu hướng biến đổi thực tế của dữ liệu theo thời gian.
* **KNN imputation** cho kết quả phân phối tương đối tốt, tuy nhiên vẫn còn một vài ngoại lai.
* Trong khi đó, phương pháp **Interpolation** cho kết quả phân phối mượt mà, tự nhiên và gần nhất với phân phối ban đầu của dữ liệu.

Do đó, để đảm bảo giữ được đặc trưng gốc và tính liên tục tự nhiên của dữ liệu thời gian, tôi lựa chọn sử dụng phương pháp Interpolation cho quá trình xử lý dữ liệu thiếu trong bài toán này.

## Giảm chiều dữ liệu

Giảm chiều dữ liệu (dimensionality reduction) là một kỹ thuật trong học máy và xử lý dữ liệu nhằm giảm số lượng biến (hoặc chiều) trong tập dữ liệu, đồng thời cố gắng giữ lại phần lớn thông tin quan trọng. Thay vì làm việc với một tập dữ liệu có nhiều đặc trưng (features), giảm chiều giúp chuyển đổi dữ liệu sang một không gian có ít chiều hơn, giúp đơn giản hóa việc phân tích, giảm chi phí tính toán và đôi khi cải thiện hiệu suất của các mô hình học máy.

### Lý thuyết

1. Mutual Information (MI)

**Mutual Information (MI)** (Thông tin tương hỗ) là một phương pháp dùng để đo lường mức độ phụ thuộc lẫn nhau giữa hai biến ngẫu nhiên. Nó cho biết lượng thông tin mà một biến cung cấp về biến kia.

Nếu hai biến rời rạc Xvà Y**, Mutual Information** được tính bằng công thức:

***I(X;Y) =***

Trong đó:

* P(x,y): xác suất đồng thời của X=x, Y=y
* P(x): xác suất của X=x
* P(y): xác suất của Y=y

Nếu X và Y độc lập, thì P(x,y)=P(x).P(y), dẫn đến I(X;Y)=0. Ngược lại, nếu X và Y phụ thuộc mạnh vào nhau, I(X;Y) sẽ lớn hơn.

1. Permutation Importance (Tầm quan trọng hoán vị)

Áp dụng phương pháp **Permutation Importance(Tầm quan trọng hoán vị)** trong mô hình RandomForest để đánh giá tầm quan trọng của các đặc trưng từ đó phân tích, loại trừ các đặc trưng.

* Permutation Importance là một kỹ thuật kiểm tra mô hình, nó đo lường sự đóng góp của từng đặc trưng đầu vào(feature) đến hiệu suất của mô hình huấn luyện.
* **Nguyên lý hoạt động:**
  + Phương pháp này hoạt động bằng cách hoán vị ngẫu nhiên các giá trị trong một cột đặc trưng, tức là xáo trộn cột đó.
  + Sau đó, mô hình được dùng để dự đoán lại với cột đã bị xáo trộn, và so sánh sự thay đổi trong điểm số(score) so với lúc chưa xáo trộn.
  + Việc này làm phá vỡ mối quan hệ giữa đặc trưng và biến mục tiêu, từ đó giúp đo lường xem mô hình phụ thuộc bao nhiêu vào đặc trưng đó.
* **Các bước thực hiện:**
  + Bước1. Huấn luyện mô hình: Có thể sử dụng nhiều mô hình khác nhau, trong bài toán này chúng em sử dụng mô hình RandomForest để huấn luyện.
  + Bước 2. Đánh giá hiệu suất ban đầu: Tính toán hiệu suất của mô hình trên tập kiểm tra bằng một chỉ số cụ thể(ví dụ: độ chính xác- accuracy, MSE - mean squared error, R2\_score, v.v.). Đây là điểm chuấn (baseline).
  + Bước 3. Xáo trộn đặc trưng:
    - * Chọn một đặc trưng cần đánh giá(*​.*
      * Xáo trộn ngẫu nhiên các giá trị của *Fi* trong tập kiểm tra, trong khi giữ nguyên các đặc trưng khác.
      * Dự đoán lại trên tập kiểm tra với đặc trưng đã bị xáo trộn.
    - Bước 4. Đo lường sự thay đổi:
      * Tính lại hiệu suất của mô hình với tập dữ liệu đã bị xáo trộn.
      * Tầm quan trọng của *Fi* được tính bằng sự khác biệt giữa hiệu suất ban đầu và hiệu suất sau khi xáo trộn:

Importance( = -

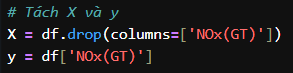
* Bước 5. Lặp lại: thực hiện bước 3 và 4 cho tất cả các đặc trưng trong tập dữ liệu.

### Áp dụng

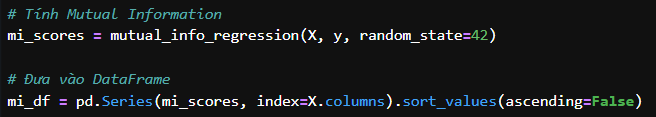
1. Mutual Information (MI)

Trong phần này chúng em sử dụng đặc trưng Nox(GT) để làm biến mục tiêu cho việc xây dựng mô hình.

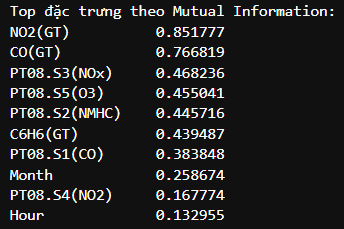
Thực hiện chia tập dữ liệu X và y:

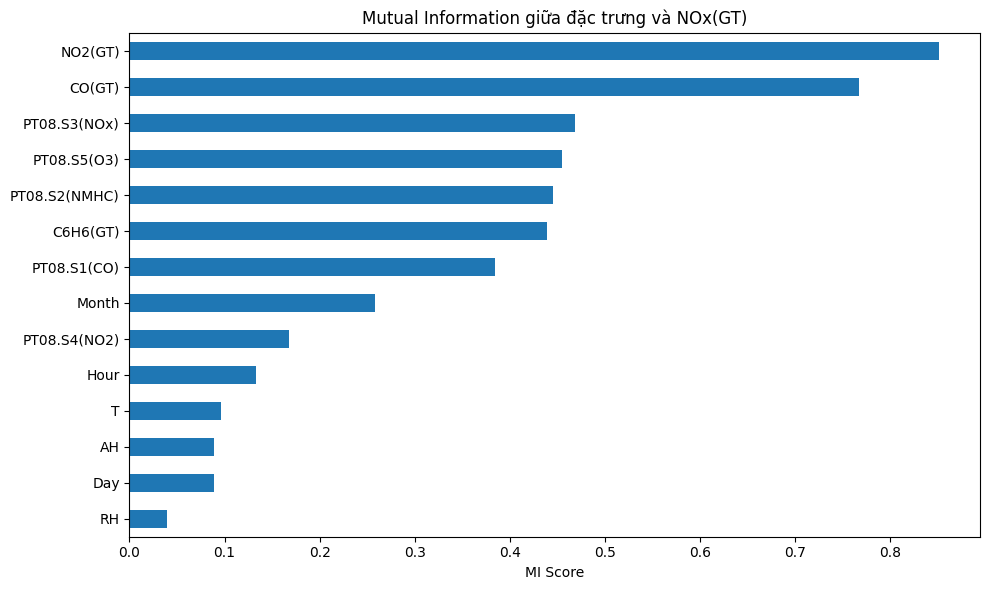


Tính MI giữa mỗi cột trong X với y, và đưa vào DataFrame



Kết quả





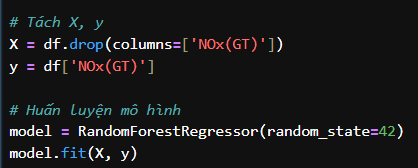
NO2(GT) và CO(GT) có MI rất cao, lần lượt là 0.85 và 0.77 -> Đây là đặc trưng có mối liên hệ mạnh nhất với NOx(GT)

Các đặc trưng PT08.S3(NOx), PT08.S5(O3), PT08.S2(NMHC), C6H6(GT), PT08.S1(CO) có MI trong khoảng 0.38 – 0.46 -> có mức liên hệ trung bình với NOx(GT)).

Sau khi đã phân tích bằng MI, em tiếp tục phân tích tiếp bằng phương pháp Permutation Importance để lựa chọn các đặc trưng cần được loại bỏ.

1. Permutaion Importance

Thực hiện chia tập dữ liệu và huấn luyện mô hình RandomForest:

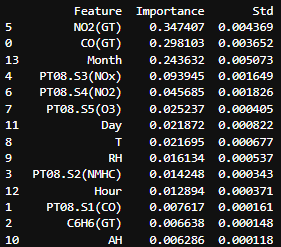


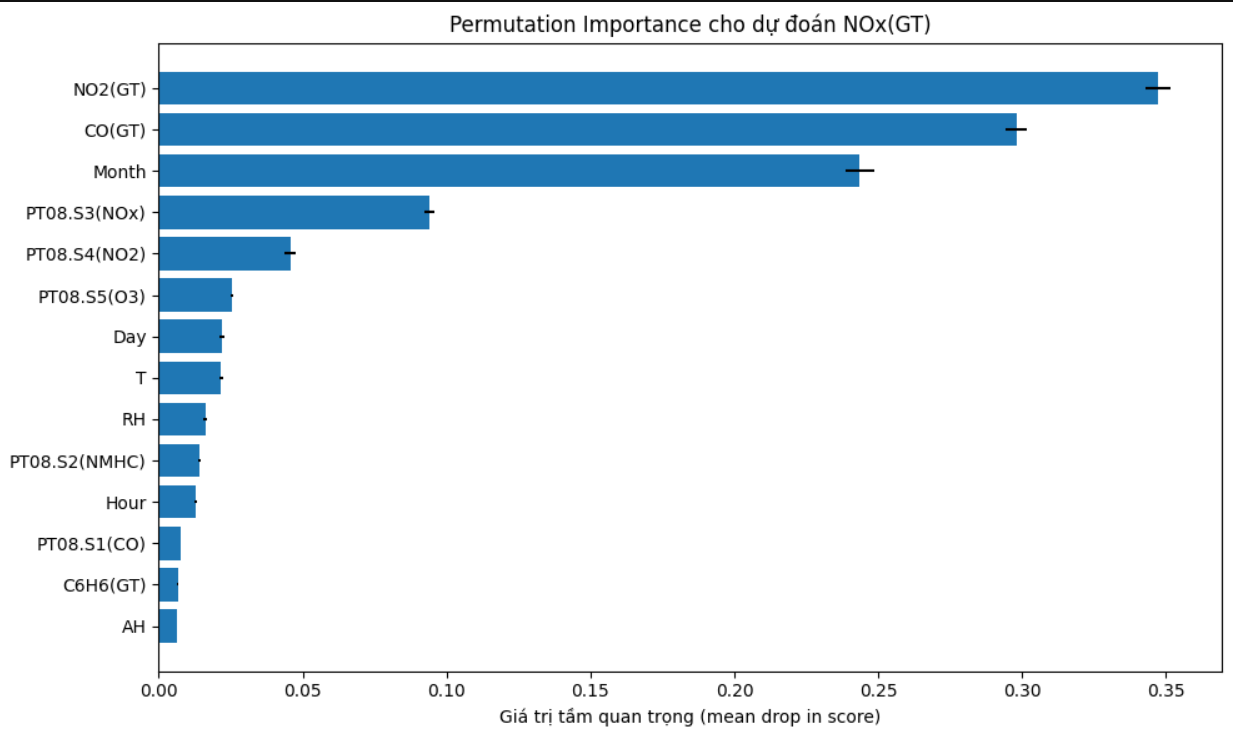
Tính Permutation:



* n\_repeats=10 : thực hiện hoán vị 10 lần với mỗi đặc trưng để tính trung bình và độ lệch chuẩn.
* n\_jobs=-1 : sử dụng tất cả CPU có sẵn để tính song song, giúp tăng tốc độ xử lý.
* result : Kết quả trả về là một đối tượng chứa *importances\_mean:* giá trị trung bình của tầm quan trọng cho mỗi đặc trưng, *importances\_std:* Độ lệch chuẩn của tầm quan trọng, thể hiện mức độ biến thiên qua các lần xáo trộn, *importances:* Ma trận chứa giá trị tầm quan trọng của từng đặc trưng qua từng lần lặp.

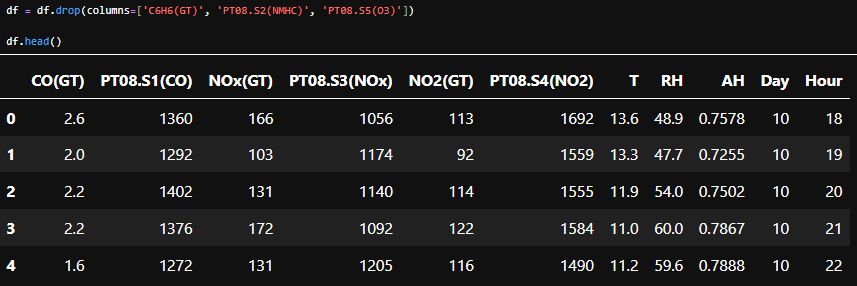
Kết quả:





* Dựa vào kết quả trên tôi thấy rằng các đặc trưng như NO2(GT), CO(GT), PT08.S3(Nox), PT08.S4(NO2) chiếm phần lớn importance; các đặc trưng PT08.S5(O3), T, PT08.S2(NMHC), RH, Hour, C6H6(GT) có ảnh hưởng trung bình; PT08.S1(CO), AH, Day có ảnh hương thấp, không đáng kể.
* Từ đó chúng tôi liệt kê các đặc trưng có ảnh hưởng trung bình và thấp vào nhóm có thể loại bỏ là PT08.S5(O3), T, PT08.S2(NMHC), RH, Hour, C6H6(GT), PT08.S1(CO), AH, Day. Nhưng các đặc trưng như T, RH, AH, PT08.S1(CO) chúng tôi thấy rằng nó có mối tương tác phi tuyến khác. Các đặc trưng như Hour, Day thì được tạo ra để có thể huấn luyện mô hình dự báo trong tương lai nên không cần loại bỏ.
* Loại bỏ các đặc trưng như C6H6(GT), PT08.S5(O3), PT08.S2(NMHC).

Dữ liệu sau khi giảm chiều

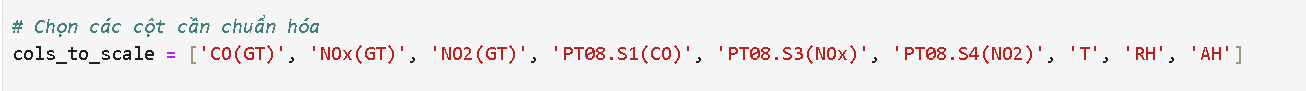


## Chuẩn hóa

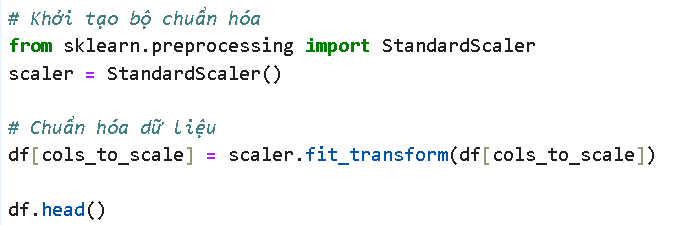
Trong quá trình tiền xử lý dữ liệu cho các thuật toán học máy, **chuẩn hóa dữ liệu** là một bước rất quan trọng nhằm đưa các đặc trưng đầu vào (features) về cùng một thang đo, giúp mô hình học hiệu quả và nhanh chóng hơn.

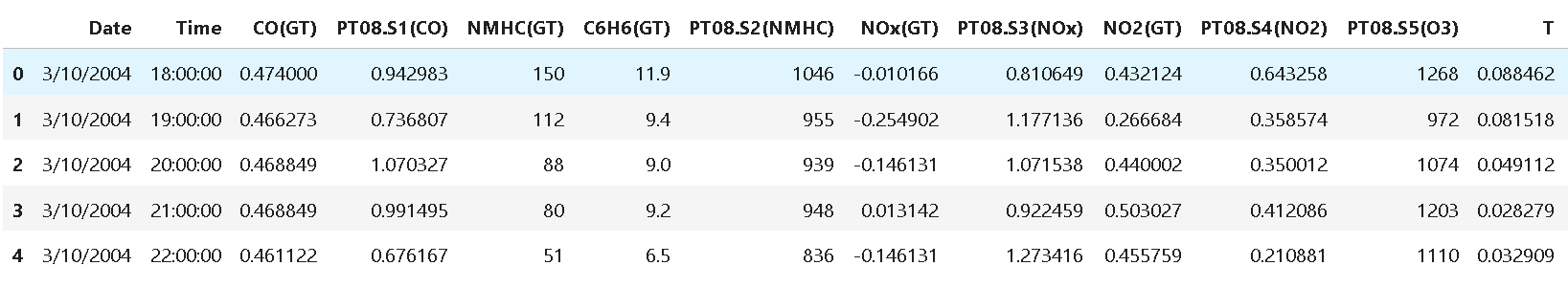
**Chuẩn hóa dữ liệu:**

* Lựa chọn các đặc trưng cần chuẩn hóa



* Chuẩn hóa bằng kỹ thuật Z-score (Standardizaion) thông qua thư viện sklearn được thực hiện như sau:





Phương pháp này chuyển dữ liệu về phân phối có trung bình bằng 0 và độ lệch chuẩn bằng 1:

Trong đó:

* μ: trung bình của đặc trưng
* σ: độ lệch chuẩn

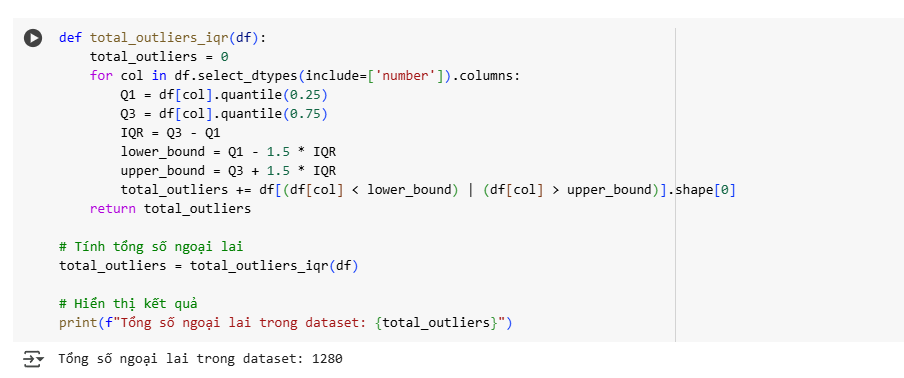
## Xử lý ngoại lai

Khái niệm : là những giá trị khác biệt lớn so với các giá trị còn lại trong tập dữ liệu. Chúng có thể gây ra sự sai lệch trong phân tích và ảnh hưởng đến độ chính xác của các mô hình dự đoán.

### Kiểm tra ngoại lai

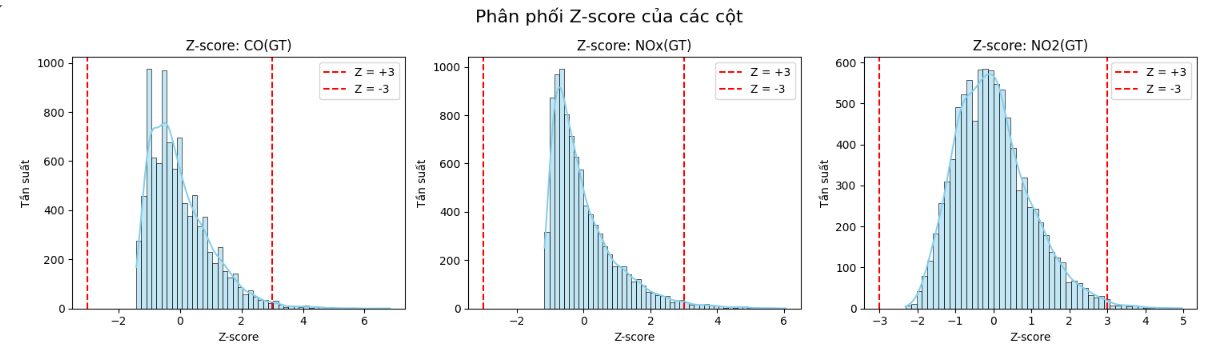
#### IQR

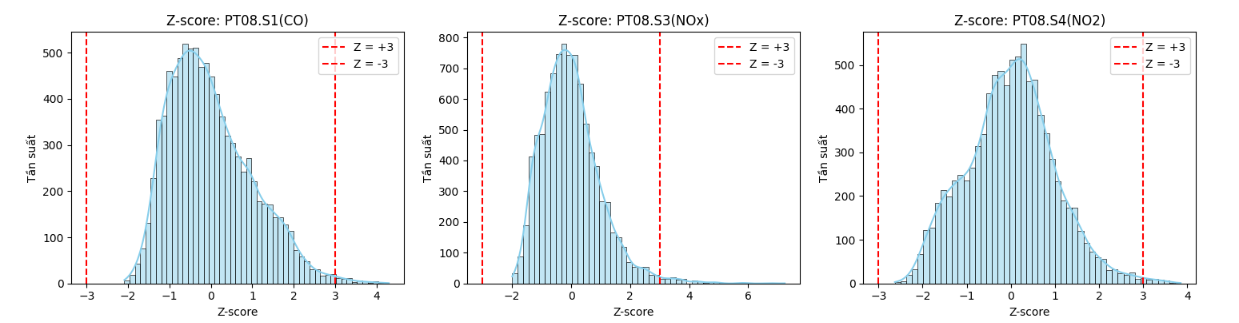
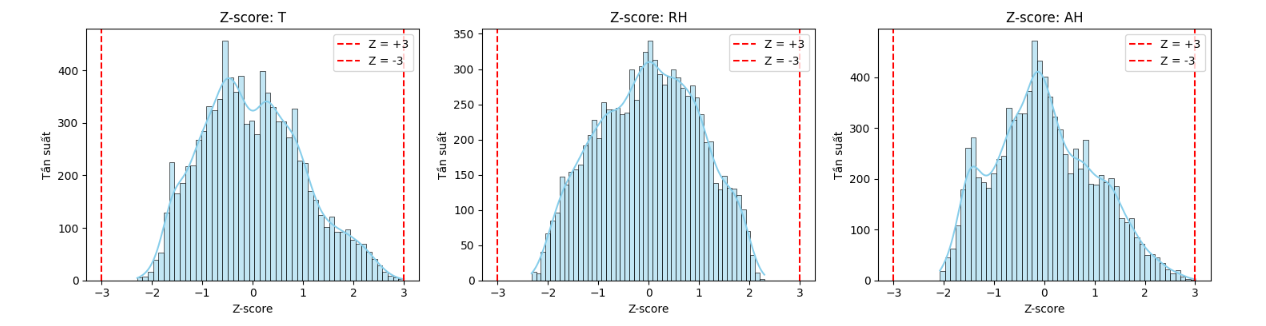
IQR (interquartile range) là khoảng tứ phân vị của tập dữ liệu dùng để xác định mức độ phân tán của dữ liệu, đồng thời cũng là một công cụ để **phát hiện ngoại lai (outliers)** trong tập dữ liệu

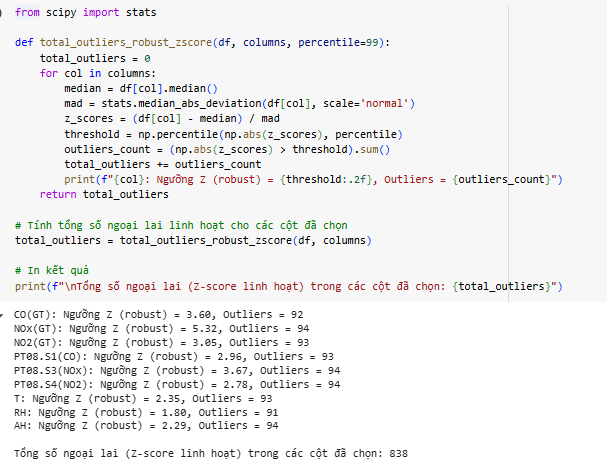


#### Z-Score linh hoạt

**Z-score linh hoạt** là một phương pháp mở rộng từ **z-score truyền thống**, dùng để xác định các điểm **ngoại lai (outliers)** trong tập dữ liệu có xu hướng thay đổi theo thời gian hoặc phân phối không ổn định.







#### Isolation Forest

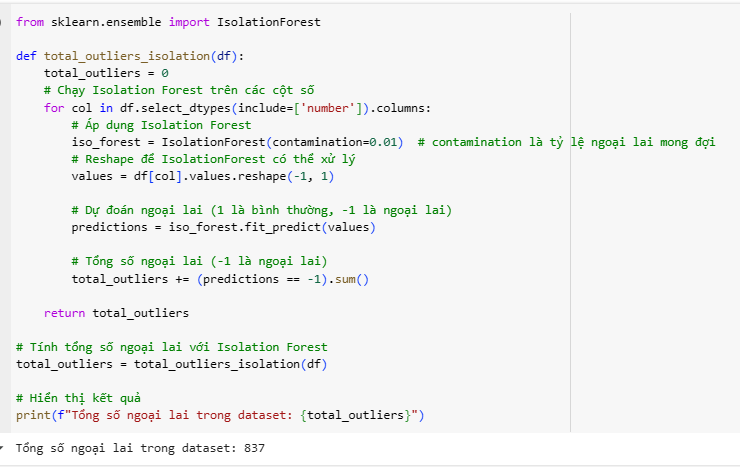
**Isolation Forest (rừng cô lập)** là một thuật toán học máy **không giám sát (unsupervised learning),** được thiết kế đặc biệt để **phát hiện điểm ngoại lai (outlier detection).**

Khác với các phương pháp truyền thống dựa vào mật độ hay khoảng cách (như kNN hay z-score), Isolation Forest hoạt động dựa trên **nguyên lý cách ly (isolation):**

**"Các điểm ngoại lai thường dễ bị cô lập hơn các điểm bình thường."**

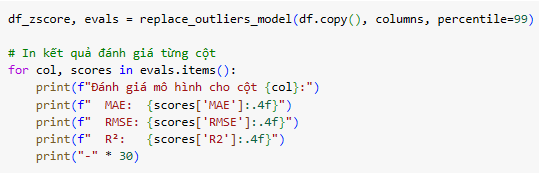
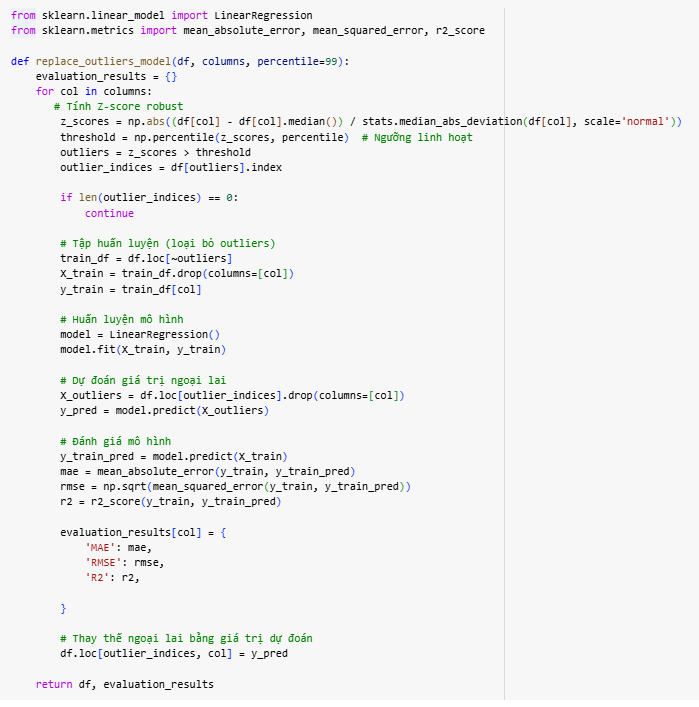
**Các bước tính:**

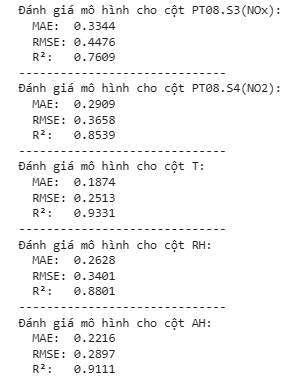
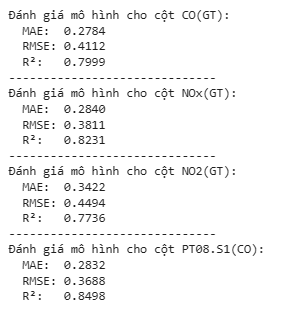
* 1. Chia ngẫu nhiên dữ liệu:
  + Chọn ngẫu nhiên một thuộc tính (feature).
  + Chọn ngẫu nhiên một giá trị chia (split value) trọng phạm vị của thuộc tính đó.
  1. Lặp lại việc chia:
* Dữ liệu được tiếp tục chia nhỏ cho đến khi:
  + Có một điểm dữ liệu duy nhất, hoặc
  + Đạt độ sâu tốt đa cho cây.
  1. Xây rừng (Forest)
* Xây nhiều cây ngẫu nhiên (Isolation Tree)
* Mỗi điểm sẽ có độ sâu trung bình qua các cây.
  1. Tính điểm ngoại lai
* Điểm bị cô lập sớm (ít chia) -> có độ sâu ngắn hạn -> khả năng là ngoại lai cao.
* Tính Outlier Score dựa trên độ sâu trung bình.
  + h(x): độ sâu trung bình để cô lập điểm x
  + c(n): giá trị chuẩn hóa phụ thuộc vào kích thước mẫu

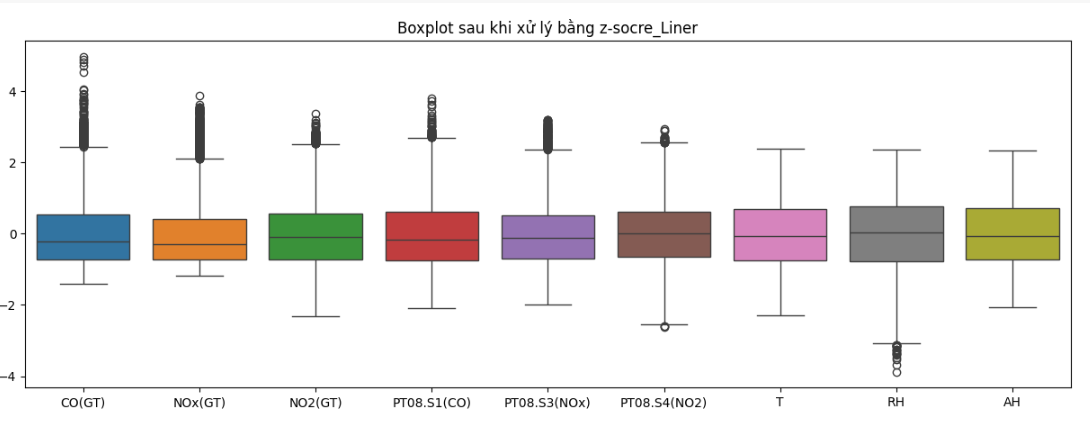


### 2.4.2. Xử lý ngoại lai

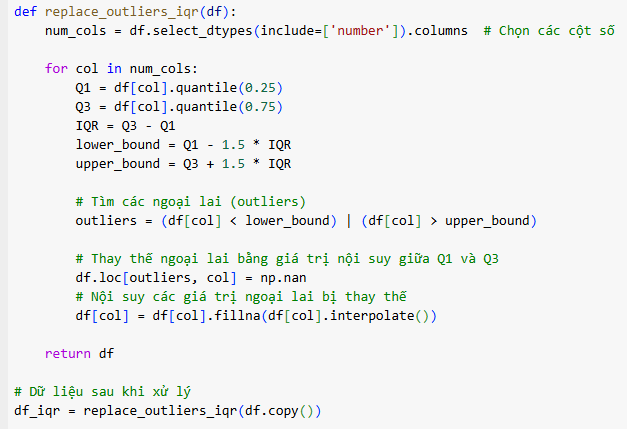
#### 2.4.2.1 Mô hình hồi quy

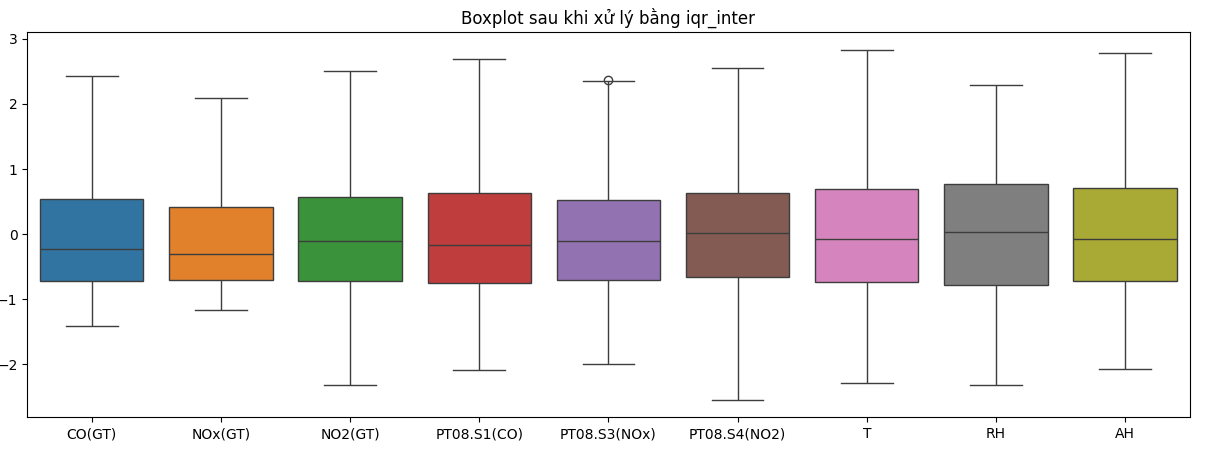




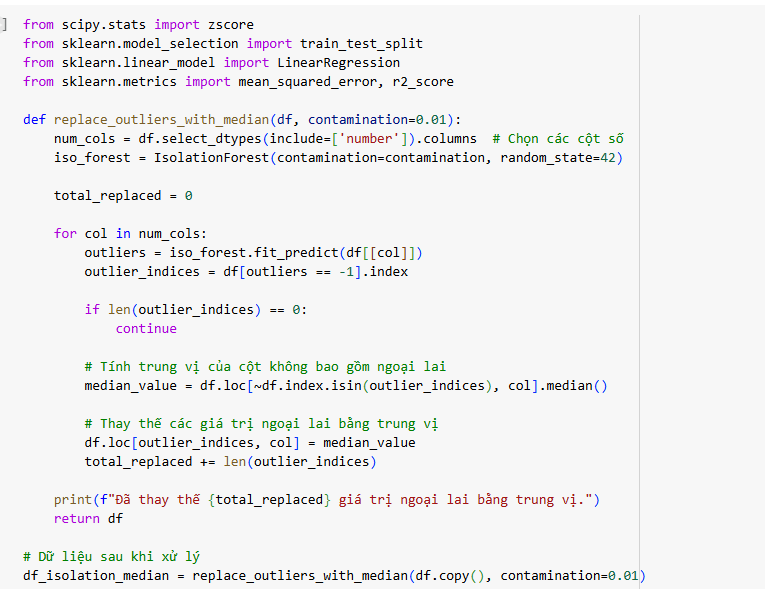


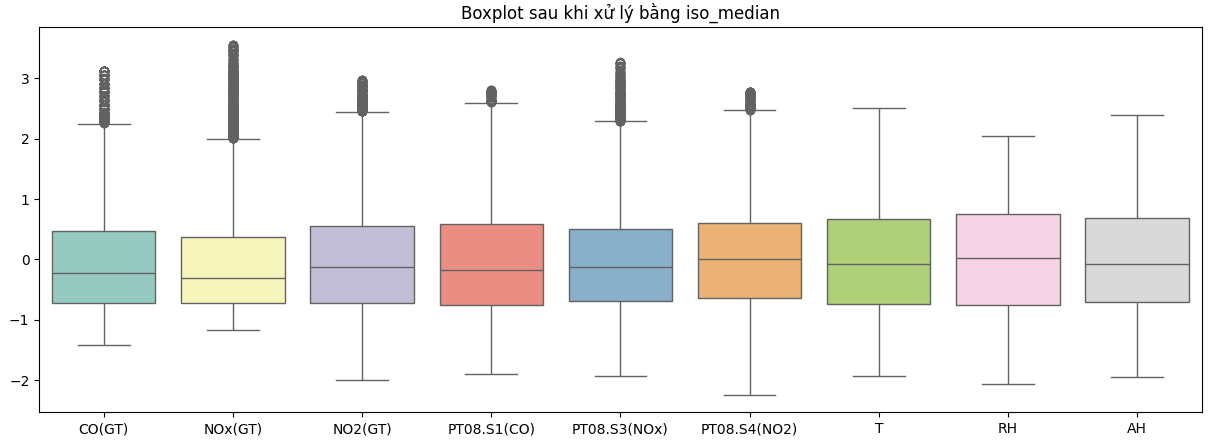
#### 2.4.2.2 Nội suy





#### 2.4.2.3 Trung vị





# XÂY DỰNG MÔ HÌNH

## Mô hình Random Forest kết hợp Rolling Forecast

### Lý thuyết

#### Giới thiệu

Dự báo chuỗi thời gian là một bài toán quan trọng trong nhiều lĩnh vực như kinh tế, khí tượng.... Trong bài tập lớn này, em áp dụng phương pháp Rolling Forecast kết hợp với mô hình Random Forest để dự báo nông độ các chất trong không khí, bao gồm CO(GT), NOx(GT), và NO2(GT), dựa trên dữ liệu thời gian đa biến. Phương pháp này tận dụng khả năng học phi tuyến của Random Forest và tính linh hoạt của Rolling Forecast để dự đoán từng bước trong tương lai, đồng thời cập nhật thông tin từ các giá trị dự báo trước đó.

### Lý thuyết

#### Random Forest

Random Forest là một thuật toán học máy được sử dụng cho bài toán phân loại và hồi quy trong lĩnh vực machine learning. Nó là sự kết hợp của nhiều cây quyết định (decision trees) để tạo ra một mô hình dự đoán mạnh hơn.

Cây quyết định trong Random Forest là các dạng cây phân chia dữ liệu dựa trên các thuộc tính và giá trị của chúng. Điều đặc biệt về Random Forest là việc sử dụng nhiều cây quyết định trong quá trình xây dụng mô hình, từ đó đưa ra các dự đoán chung dựa trên kết quả của từng cây quyết định thành viên. Kết quả cuối cùng của Random Forest được tính dựa trên việc lấy trung bình hoặc kết hợp kết quả từ tất cả các cây thành viên.

#### Rolling Forecast

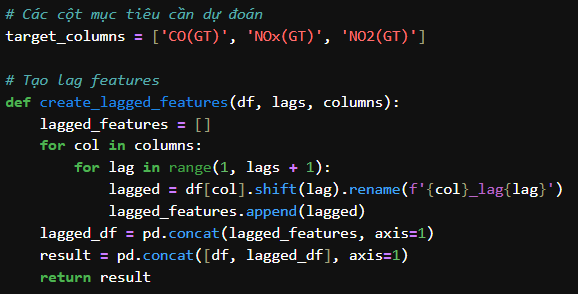
Phương pháp Rolling Forecast (dự báo cuộn) là một kỹ thuật dự báo thời gian phổ biến trong các bài toán chuỗi thời gian (time series forecasting). Phương pháp này mô phỏng cách mà mô hình dự báo được cập nhật liên tục khi có dữ liệu mới — rất sát với thực tế khi ta muốn sử dụng mô hình trong môi trường thực tế.

### Triển khai

Dữ liệu đầu vào là một tập hợp chuỗi thời gian đa biến, với các cột mục tiêu cần dự báo là CO(GT), NOx(GT), NO2(GT). Các bước chuẩn bị bao gồm:

#### Tạo đặc trưng trễ (Lagging Features)

Sử dụng hàm *create\_lagged\_features* để tạo 24 bước trễ (1 ngày) cho mỗi cột mục tiêu.

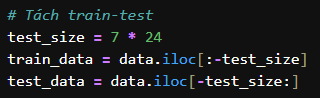


Ví dụ: Với CO(GT), tạo các cột CO(GT)\_lag1, CO(GT)\_lag2,..., CO(GT)\_lag24.

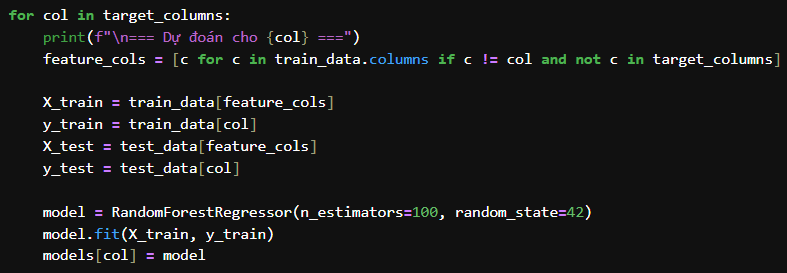


Tổng cộng, mỗi cột mục tiêu tạo ra 24 đặc trưng trễ, dẫn đến 72 cột đặc trưng mới.

Tiếp theo, chúng tôi tách dữ liệu thành tập train và test. Tập test gồm 168 dòng (7 ngày) cuối cùng của dữ liệu để vừa có thể dự đoán vừa có thể kiểm tra độ hiệu quả của mô hình. Phần còn lại là tập train.



#### Huấn luyện mô hình

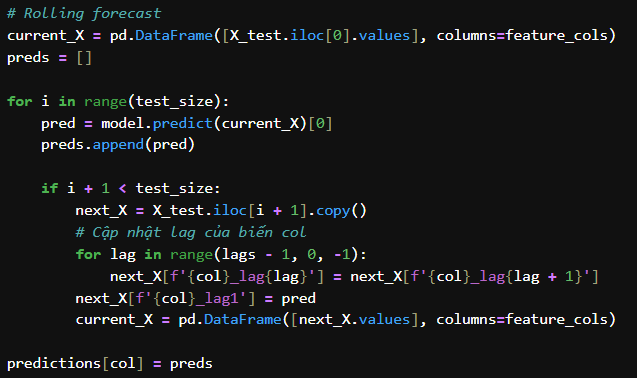


Mô hình Random Forest được huấn luyện riêng cho từng cột mục tiêu CO(GT), NOx(GT), NO2(GT).

Đặc trưng: Bao gồm tất cả các cột là của các biến khác và các cột không phải mục tiêu.

Tham số mô hình: Số lượng cây quyết định là 100, với số seed cố định *random\_state=42* để đảm bảo tính tái lập.

#### Rolling Forecast



Quy trình dự báo được thực hiện theo các bước:

1. Khởi tạo đặc trưng ban đầu bằng hàng đầu tiên của tập kiểm tra.
2. Dự báo giá trị tiếp theo bằng Random Forest.
3. Cập nhật các lag của cột mục tiêu bằng cách:

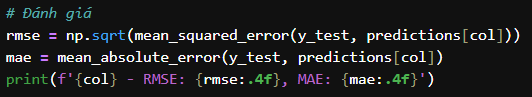
* Dịch chuyển các lag hiện tại (Ví dụ: lag2 thành lag1, lag3 thành lag2,...)
* Gán giá trị dự báo vào lag1.

1. Lặp lại cho toàn bộ 168 bước trong tập kiểm tra.

#### Đánh giá mô hình

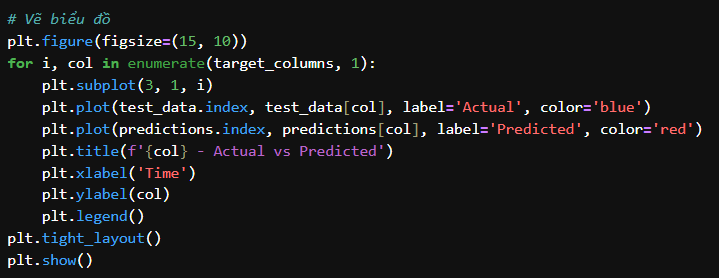
Hai chỉ số được sử dụng để đánh giá:

* RMSE (Root Mean Squared Error): Đo lường sai số bình phương trung bình.
* MAE (Mean Absolute Error): Đo lường sai số tuyệt đối trung bình.



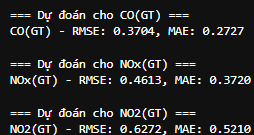
#### Trực quan hóa

Biểu đồ so sánh giá trị thực tế (màu xanh) và dự báo (màu đỏ) được vẽ cho từng cột mục tiêu, giúp đánh giá trực quan hiệu suất mô hình.

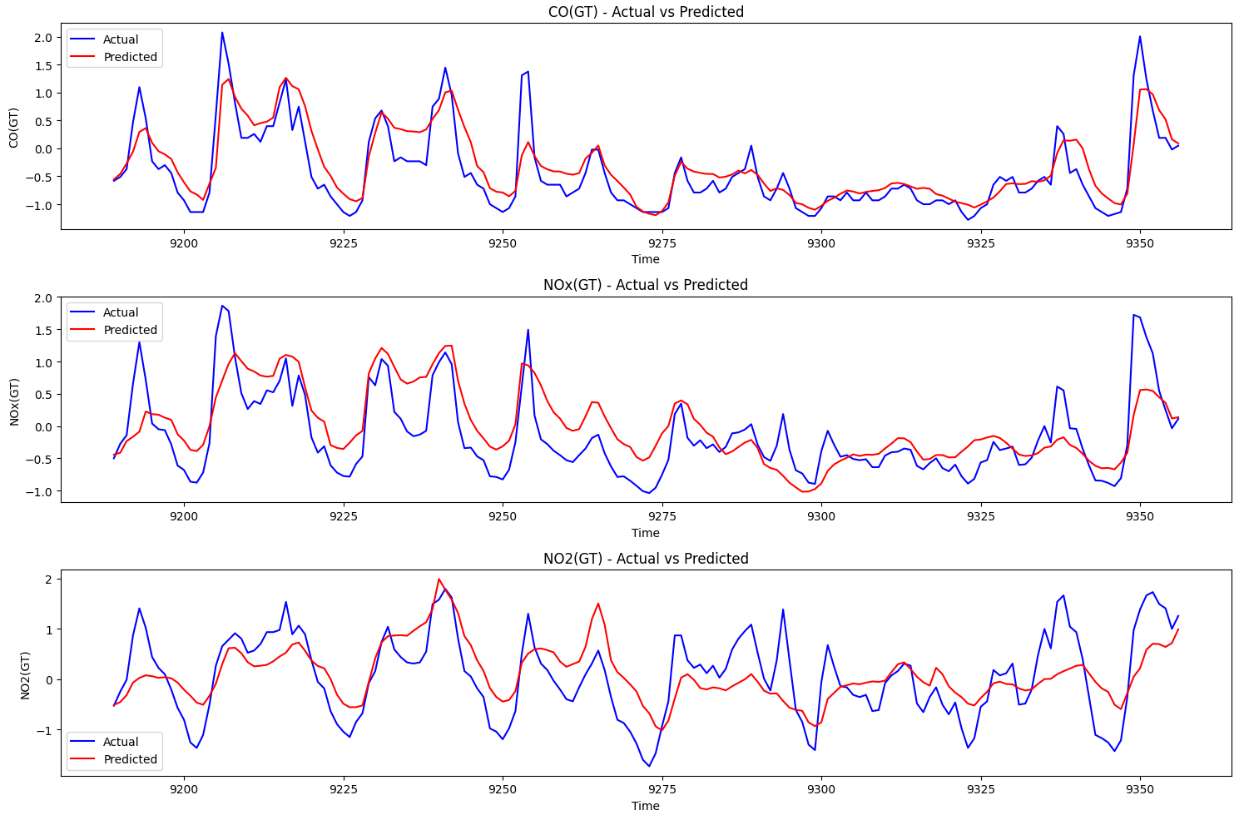


#### Kết quả và thảo luận

* Kết quả



* **CO(GT)**: Sai số chỉ chiếm <10% so với toàn bộ biên độ → rất chính xác.
* **NOx(GT)**: Sai số chiếm ~15-18% → vẫn tốt, nhưng độ chính xác giảm dần với độ biến động nhanh.
* **NO2(GT)**: Tốt nhưng mô hình có vẻ "làm mượt" quá, phản ứng hơi chậm với dao động mạnh.
* Trực quan



* **CO(GT):**
* Dự đoán bám khá sát thực tế.
* Đường dự đoán (đỏ) theo đúng xu hướng của dữ liệu thực (xanh), nhất là các đỉnh và đáy.
* Mô hình phản ứng tốt với các thay đổi ngắn hạn.
* Các đỉnh nhọn có thể hơi "mềm hóa", nhưng nhìn chung xu hướng dự báo khá chính xác.
* **NOx(GT):**
* Dự báo vẫn theo xu hướng tổng thể, nhưng bắt đầu có độ lệch rõ ở các đoạn dao động mạnh.
* Một số đỉnh/thung bị làm mượt quá mức → độ nhạy với thay đổi đột ngột thấp hơn.
* Dự đoán thấp hơn thực tế ở các pha tăng nhanh, và cao hơn thực tế ở các pha giảm sâu.
* Mô hình hơi "trễ" với các biến động mạnh.
* **NO2(GT)**
* Vẫn giữ được xu hướng chung nhưng không dự đoán được các đỉnh/thung lớn.
* Mô hình dự đoán chưa phản ánh được các dao động rõ rệt trong dữ liệu thực tế.
* Đường dự báo quá mượt, thiếu phản ứng với dao động mạnh, dẫn đến độ lệch lớn.

#### Kết luận

Nhìn chung, mô hình Random Forest kết hợp Rolling Forecast hoạt động hiệu quả trong việc dự đoán ngắn hạn cho dữ liệu thời gian ô nhiễm không khí, đặc biệt đối với CO(GT). Tuy nhiên, hiệu quả giảm dần với các chỉ số có biến động mạnh và phức tạp hơn như NO2(GT). Điều này cho thấy rằng mô hình có xu hướng làm mượt dữ liệu và không phản ứng tốt với các biến động nhanh, một điểm hạn chế đặc trưng của Random Forest khi áp dụng cho bài toán chuỗi thời gian.

## Mô hình Hồi quy tuyến tính

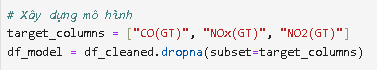
### Lý thuyết

Hồi quy tuyến tính là một thuật toán thuộc lĩnh vực học máy có giám sát (supervised learning), được sử dụng phổ biến trong các bài toán dự đoán. Mục tiêu của hồi quy tuyến tính là mô hình hóa mối quan hệ tuyến tính giữa một biến phụ thuộc (biến mục tiêu cần dự đoán) và một hoặc nhiều biến độc lập (các đặc trưng đầu vào).

### Triền khai

Dữ liệu đầu vào là một tập hợp chuỗi thời gian đa biến, với các cột mục tiêu cần dự báo là CO(GT), NOx(GT), NO2(GT).

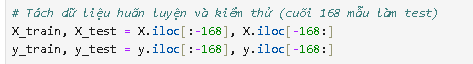
* Xây dựng mô hình:



* Chuẩn bị đặc trưng đầu vào và biến mục tiêu:

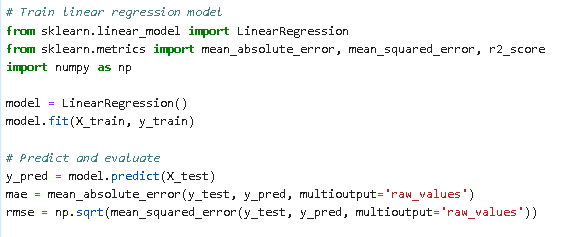


* Tách dữ liệu train và test



Tập dữ liệu được chia ra **168 mẫu cuối cùng** để kiểm tra mô hình (tương đương với 1 tuần dữ liệu đo mỗi giờ), phần còn lại dùng để huấn luyện.

* Huấn luyện và đánh giá mô hình



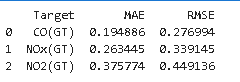
Mô hình hồi quy tuyến tính được huấn luyện riêng cho từng cột mục tiêu CO(GT), NOx(GT), NO2(GT) và đánh giá độ chính xác của mô hình trên tập kiểm tra.

Hai chỉ số được sử dụng để đánh giá:

· RMSE (Root Mean Squared Error): Đo lường sai số bình phương trung bình.

· MAE (Mean Absolute Error): Đo lường sai số tuyệt đối trung bình.

* Kết quả đánh giá



CO(GT): Mô hình dự đoán với sai số rất thấp.

* Sai số tuyệt đối trung bình (MAE) dưới 0.2 cho thấy mô hình dự đoán khá sát với thực tế.
* RMSE cũng thấp, nghĩa là ít có sai số lớn bất thường (outlier).
* → Mô hình hoạt động **rất hiệu quả** với chỉ số CO(GT).

NOx(GT): Sai số cao hơn một chút so với CO(GT), nhưng vẫn ở mức **chấp nhận được**.

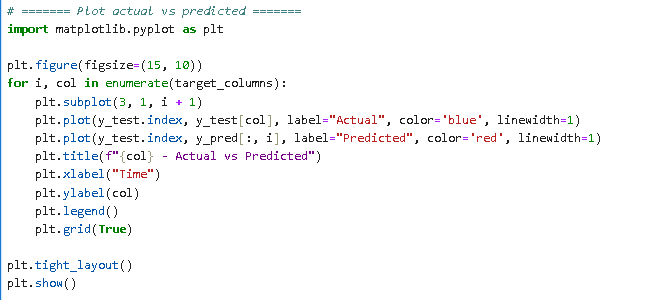
* Mô hình dự đoán được xu hướng nhưng có thể gặp khó khăn với các biến động lớn.
* RMSE cao hơn MAE → có thể tồn tại vài điểm sai số lớn hơn trung bình.

NO2(GT): Đây là chỉ số có **sai số cao nhất**.

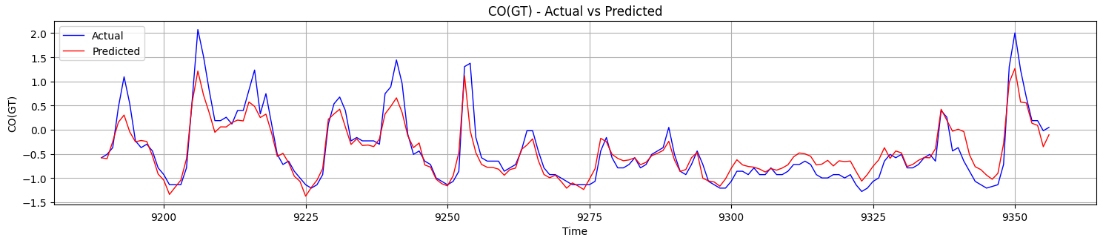
* Mô hình kém chính xác hơn trong việc dự đoán NO2(GT).
* Khả năng cao là dữ liệu NO2 có nhiều biến động, nhiễu hoặc ít tương quan với các đặc trưng đầu vào.
* RMSE lớn hơn MAE cho thấy mô hình **khó dự đoán được các biến động đột ngột**.

### So sánh kết quả của mô hình với dữ liệu thực tế

* Vẽ biểu đồ

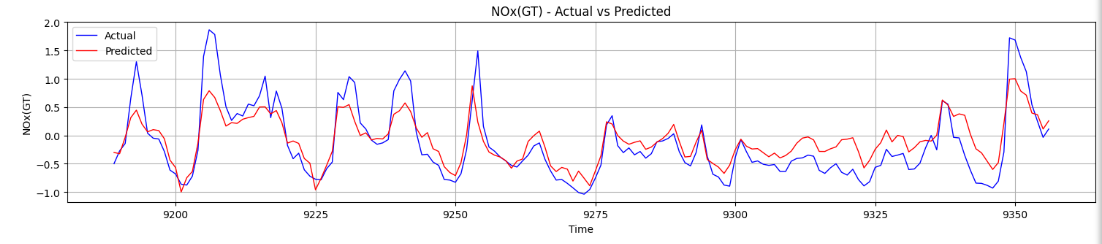


### 3.2.3.1 CO(GT)



* **Xu hướng tổng thể khớp tốt:** Dự đoán bám sát đường thực tế ở phần lớn các đoạn, đặc biệt trong những vùng có **biến động ổn định và đều**.
* **Các đỉnh và đáy chính được mô hình tái hiện tương đối tốt**, đặc biệt từ khoảng mốc 9200–9250 và sau 9330.
* **Một số đỉnh bị làm mượt**: Ở các điểm cực đại như gần mốc 9210 hoặc 9350, mô hình không lên được đúng độ cao như thực tế → cho thấy mô hình hồi quy tuyến tính **giới hạn trong việc nắm bắt những biến động mạnh**.
* **Sự chênh lệch nhỏ trong các dao động liên tiếp**: Trong vùng từ khoảng 9270–9320, mô hình dự đoán hơi **lệch pha** và **độ dao động nhỏ hơn thực tế**.

### 3.2.3.2 NOx(GT)



* **Bám theo xu hướng tổng thể** khá tốt, đặc biệt trong giai đoạn từ mốc ~9190 đến ~9240: các đỉnh – đáy chính được dự đoán đúng vị trí tương đối.
* Mô hình **dự đoán mượt và ổn định**, phù hợp với xu hướng trơn tru của nhiều đoạn trong dữ liệu.
* **Làm mượt quá mức tại các đỉnh nhọn**: Ví dụ, tại các mốc ~9195, ~9220, ~9250 và đặc biệt ~9350, mô hình **không lên tới đỉnh thực tế** → cho thấy mô hình chưa bắt kịp các biến động lớn, nhanh.
* **Độ trễ nhẹ (lag)**: Ở một số đoạn, như ~9300–9330, mô hình có vẻ **trễ pha so với thực tế**, khiến dự đoán bị lệch thời điểm.
* **Dao động nhỏ bị bỏ qua**: Một số dao động nhỏ trong đoạn 9260–9310 không được thể hiện rõ trong dự đoán → biểu hiện điển hình của mô hình tuyến tính khi xử lý tín hiệu nhiễu/biến động nhanh.

### 3.2.3.3 NO2(GT)

## 

* Mô hình **bắt được xu hướng chung** ở một vài giai đoạn, đặc biệt trong đoạn từ khoảng ~9190 đến ~9250 và từ ~9340 đến cuối.
* **Sự tương quan tổng thể vẫn có** ở các pha dao động lên/xuống theo chu kỳ.
* **Dự đoán thiếu độ nhọn và biên độ dao động nhỏ hơn thực tế**: Các đỉnh nhọn và đáy sâu trong đường thực tế (đường xanh) hoàn toàn không được mô hình tái hiện đúng. Biểu đồ đường đỏ bị **làm mượt quá mức**, khiến sai số lớn.
* **Không theo kịp biến động nhanh**: Các biến động lớn và đột ngột như tại mốc ~9270, ~9300 hoặc ~9345 hoàn toàn không được phản ánh đúng.
* **Sai lệch pha (phase shift)**: Có những đoạn dự đoán bị **trễ hoặc lệch pha** so với thực tế → mô hình không học tốt được cấu trúc tuần hoàn/phức tạp của dữ liệu.

### Kết luận

Nhìn chung, mô hình **Hồi quy tuyến tính** hoạt động tương đối hiệu quả trong việc dự đoán ngắn hạn đối với một số chỉ số ô nhiễm không khí, đặc biệt là **NOx(GT)** – nơi xu hướng biến động có thể được mô hình hóa khá tốt. Tuy nhiên, hiệu quả của mô hình giảm rõ rệt với các chỉ số có độ nhiễu cao và dao động phức tạp như **NO2(GT)** và **CO(GT)**.Mô hình có xu hướng **làm mượt dữ liệu**, phản ứng kém với những biến động mạnh hoặc bất thường – một điểm hạn chế phổ biến của hồi quy tuyến tính trong bài toán **dự báo chuỗi thời gian phi tuyến**.

# TỔNG KẾT

Trong khuôn khổ bài tập lớn môn học *Tiền xử lý dữ liệu*, em đã thực hiện đề tài *Dự đoán nồng độ ô nhiễm không khí* với mục tiêu xây dựng một quy trình xử lý và khai thác dữ liệu chất lượng không khí nhằm phục vụ cho bài toán dự báo nồng độ các chất ô nhiễm như CO, NOx và NO₂.

Thông qua các bước tiền xử lý dữ liệu như làm sạch dữ liệu, xử lý giá trị thiếu, chuẩn hóa, xử lý ngoại lai, giảm chiều dữ liệu, nhóm đã xây dựng được một pipeline hoàn chỉnh cho việc huấn luyện và đánh giá mô hình. Mô hình Random Forest kết hợp với Rolling Forecast cùng với mô hình Hồi quy tuyến tính được sử dụng cho bài toán hồi quy đã cho thấy khả năng dự đoán tương đối ổn định trong ngắn hạn (7 ngày kế tiếp), đồng thời phản ánh được xu hướng thay đổi của các chất ô nhiễm trong không khí.

Kết quả đạt được cho thấy vai trò quan trọng của bước tiền xử lý dữ liệu trong việc nâng cao hiệu suất và độ chính xác của mô hình học máy.

Tuy vẫn còn một số hạn chế như chưa khai thác toàn diện các yếu tố ngoại sinh (thời tiết, giao thông,...) hay thử nghiệm nhiều mô hình khác nhau, nhưng đề tài đã giúp nhóm hiểu rõ hơn về quy trình làm việc với dữ liệu thực tế và các kỹ thuật tiền xử lý dữ liệu quan trọng. Đây là nền tảng hữu ích cho các nghiên cứu và ứng dụng sâu hơn trong lĩnh vực phân tích dữ liệu và trí tuệ nhân tạo.

TÀI LIỆU THAM KHẢO

* 1. <https://machinelearningcoban.com/>
  2. <https://www.cubesoftware.com/blog/rolling-forecast>
  3. <https://scikit-learn.org/stable/index.html>