Chương 2.

Khám phá đặc tính dữ liệu chuỗi thời gian

PhD. Nguyễn Thị Khánh Tiên

email: tienntk@ut.edu.vn

Nội dung chính trong chương 2:

- Hàm tự tương quan (Autocorrelation / ACF)
- Khái niệm White Noise
- Phân rã chuỗi thời gian (Time-Series Decomposition)
- Vấn đề chất lượng dữ liệu trong Time Series
- Trực quan hóa dữ liệu

Mục tiêu thực hành chương 2:

- Hiểu và tính được hàm tự tương quan
- Nhận biết được chuỗi white noise
- Thực hiện phân rã chuỗi thời gian
- Xử lý các vấn đề chất lượng dữ liệu
- Thực hành trực quan hóa chuyên sâu cho Time Series

Hàm tự tương quan (Autocorrelation / ACF)

Hàm tự tương quan (ACF) là một trong những công cụ quan trọng nhất để hiểu cấu trúc của chuỗi thời gian trước khi làm mô hình dự báo. **ACF** cho biết mức độ "liên quan" giữa giá trị hiện tại và các giá trị trong quá khứ, giúp nhận biết chuỗi có xu hướng, mùa vụ hoặc quy luật lặp lại không.

Tự tương quan là mức độ tương quan giữa một chuỗi thời gian với chính nó tại các độ trễ khác nhau (lag).

- Lag = 1 → so sánh giá trị tại thời điểm *t* với *t−1*
- Lag = 2 → so sánh giá trị tại *t* với *t−2*
- Lag = k → so sánh giá trị tại *t* với *t−k*

Công thức tự tương quan tại độ trễ k:

$$\rho(k) = \frac{Cov(y_t, y_{t-k})}{Var(y_t)}$$

Giá trị của autocorrelation nằm trong [-1, 1]:

- +1 → tương quan dương mạnh (giá trị tăng → giá trị sau cũng tăng)
- -1 → tương quan âm mạnh (giá trị tăng → giá trị sau giảm)
- 0 → không có quan hệ

Ý nghĩa của các độ trễ (lags). Mỗi **lag** thể hiện mức độ ảnh hưởng của quá khứ lên hiện tại.

- **Lag nhỏ (1–3):** dùng để kiểm tra tính "nhớ" ngắn hạn của chuỗi → hữu ích cho mô hình AR (Autoregressive).
- **Lag mùa vụ** (ví dụ: lag=12 cho dữ liệu tháng): giúp nhận diện seasonality (mùa vụ lặp lại theo chu kỳ).
- **Lag lớn:** nếu autocorrelation vẫn cao → chuỗi có "trend mạnh" (tính dính).

Hiệp phương sai và phương sai

Phương sai (Variance). Phương sai đo mức độ một biến dao động quanh giá trị trung bình của chính nó.

- Nếu các giá trị nằm gần nhau → phương sai nhỏ.
- Nếu các giá trị phân tán rộng → phương sai lớn.

Công thức:

$$Var(Y) = \frac{1}{n}\sum_{t=1}^{n}(y_t - \bar{y})^2$$

Trong đó:

- y_t: giá trị tại thời điểm t
- $\bar{y}$: giá trị trung bình
- $(y_t - \bar{y})^2$: bình phương độ lệch so với trung bình

Ý nghĩa:

- Var lớn → dữ liệu dao động mạnh, không ổn định.
- Var nhỏ → dữ liệu ổn định quanh trung bình.

Ví dụ:

Dãy số: [10, 11, 10, 9, 10] → nằm rất sát nhau → phương sai nhỏ.
Dãy số: [10, 30, -5, 50, 2] → rất phân tán → phương sai lớn.

Hiệp phương sai (Covariance). Hiệp phương sai đo mức độ hai biến thay đổi cùng nhau.

- Cùng tăng cùng giảm → Cov > 0
- Một tăng, một giảm → Cov < 0
- Không liên hệ → Cov ≈ 0

Công thức:

$$Cov(X, Y) = \frac{1}{n}\sum_{t=1}^{n}(x_t - \bar{x})(y_t - \bar{y})$$

Trong đó:

- $(x_t - \bar{x})$: độ lệch của X so với trung bình
- $(y_t - \bar{y})$: độ lệch của Y so với trung bình
- Tích hai độ lệch cho biết hướng thay đổi của X và Y

Ý nghĩa dấu:

- Nếu hai số lệch cùng chiều → tích dương → Cov dương

Ví dụ: X tăng → Y tăng.

- Nếu lệch ngược chiều → tích âm → Cov âm

Ví dụ: X tăng → Y giảm.

- Nếu không có quy luật → Cov ≈ 0. Không có quan hệ tuyến tính.

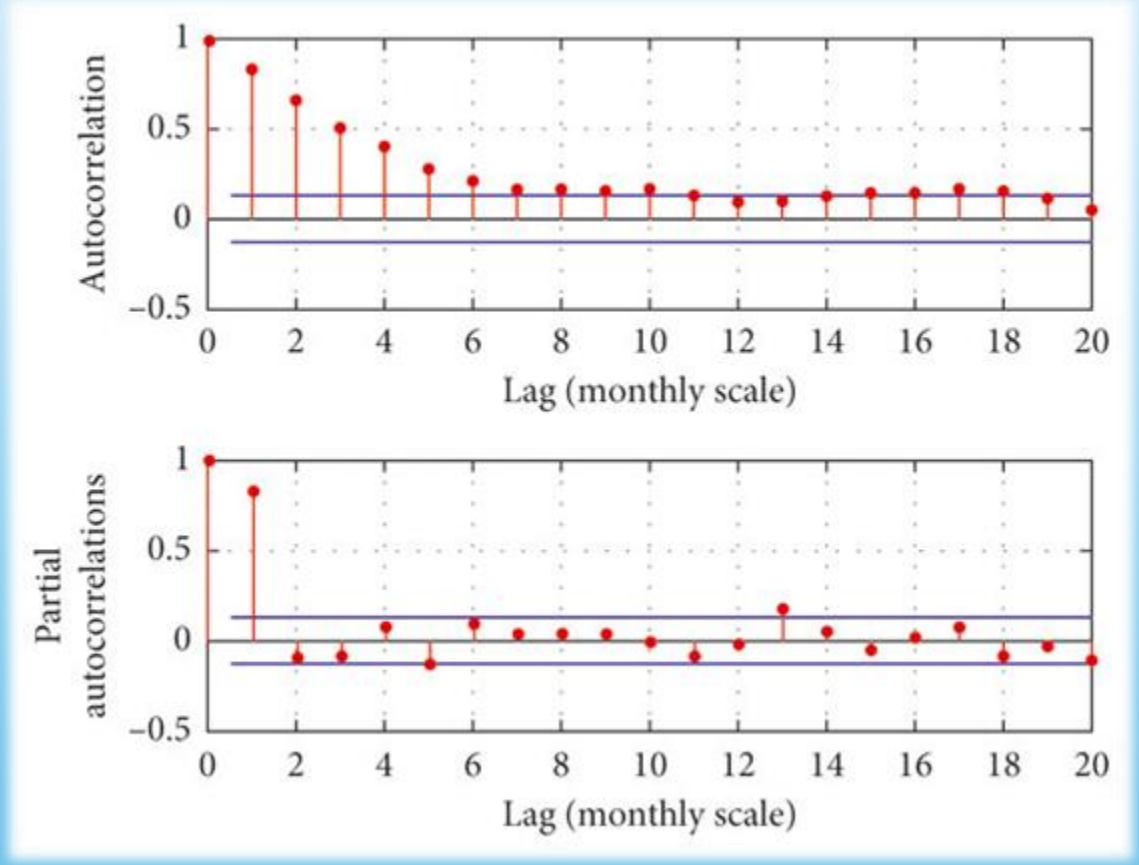
Cách đọc biểu đồ ACF và PACF

ACF (Autocorrelation Function)

- Đo tương quan giữa giá trị hiện tại và quá khứ bao gồm cả ảnh hưởng trung gian.
- Dùng để nhận dạng:
  - Trend (ACF giảm chậm)
  - Seasonality (ACF có đỉnh nhảy theo chu kỳ)
  - Thành phần MA (Moving Average)

PACF (Partial Autocorrelation Function)

- Đo tương quan trực tiếp giữa giá trị hiện tại và lag k, loại bỏ ảnh hưởng của các lag trung gian.
- Dùng để nhận dạng:
  - Thành phần AR (Autoregressive)
  - Xác định số lag cần thiết trong mô hình AR(p)



Ứng dụng trong việc xác định tính mùa vụ và chọn mô hình

- **Nhận diện mùa vụ (Seasonality).**
  - Nếu ACF có "đỉnh" lặp lại theo chu kỳ → chuỗi có mùa vụ.
  - Ví dụ:
    - Dữ liệu bán lẻ theo tháng → peak tại lag=12
    - Dữ liệu nhiệt độ ngày → peak tại lag=7 (chu kỳ tuần)
- **Chọn mô hình ARIMA**
  - Biểu đồ ACF và PACF hỗ trợ chọn tham số (p, d, q).
    - Nếu PACF cắt cụt → tăng p (AR)
    - Nếu ACF cắt cụt → tăng q (MA)
    - Nếu ACF giảm chậm → cần d (difference) để khử trend
  - Ví dụ:
    - ACF giảm chậm, PACF có đỉnh lớn tại lag=1 → ARIMA(1,1,0)
    - ACF có 2 lag cắt cụt → ARIMA(0,0,2)
- **Chọn mô hình Seasonal ARIMA (SARIMA)**
  - Nếu xuất hiện đỉnh tại lag=12 → ta thêm phần mùa vụ: (P, D, Q, 12)
- **Kiểm tra dữ liệu có cấu trúc hay chỉ là noise?**
  - Nếu ACF toàn nằm trong khoảng tin cậy → chuỗi gần như white noise → không dự báo được.

Khái niệm White Noise (Nhiễu trắng)

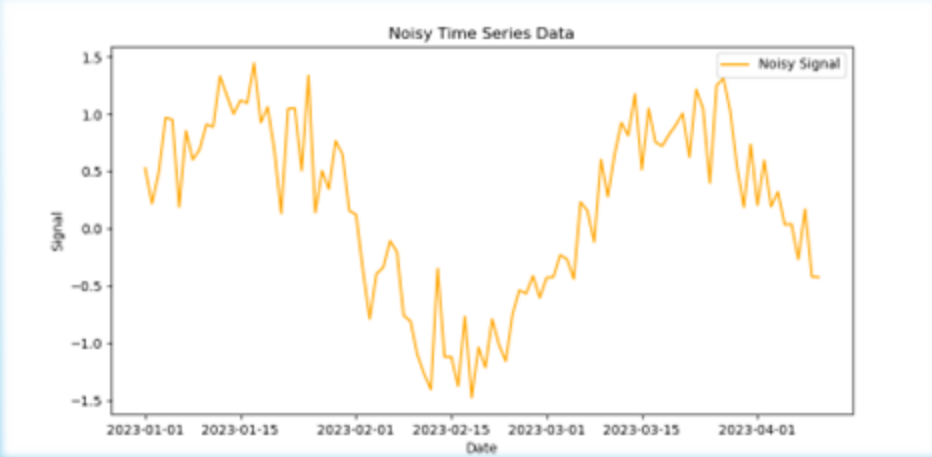
White Noise là một chuỗi thời gian mà các giá trị hoàn toàn ngẫu nhiên, không có bất kỳ cấu trúc hay quy luật nào. Nói cách khác, mỗi giá trị trong chuỗi là độc lập và không bị ảnh hưởng bởi các giá trị trước đó.

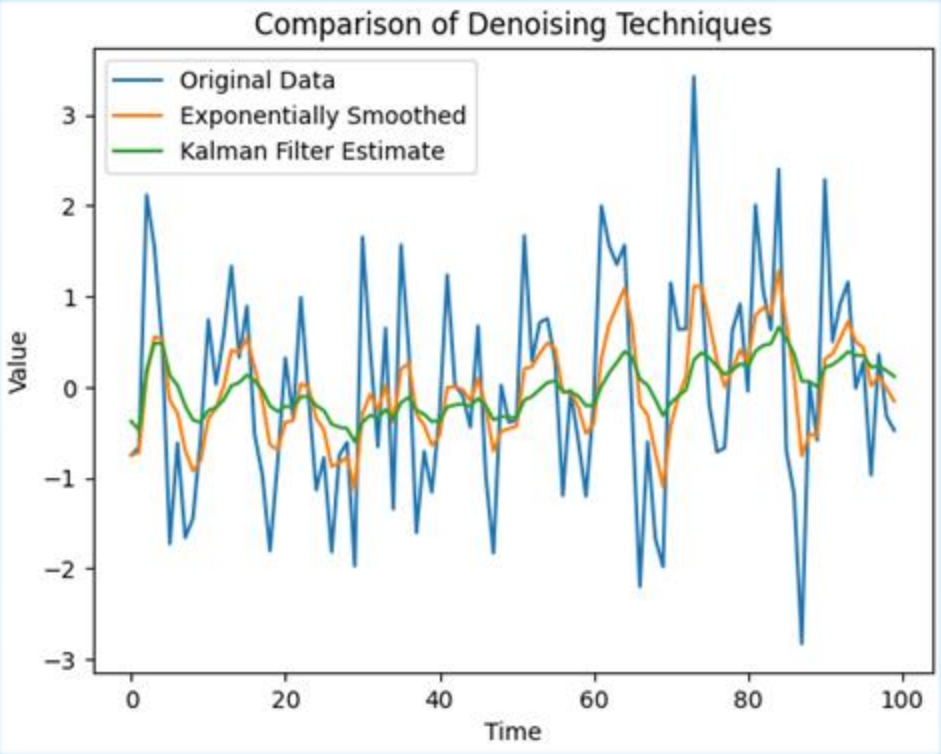
Mô hình toán học của White Noise:

$$y_t \sim N(0, \sigma^2)$$

→ White noise như **những dao động hoàn toàn vô nghĩa**, không có mẫu hình lặp lại, không có xu hướng, không có mùa vụ và không thể dự đoán (Mean = 0, variance = constant, No Autocorrelation, biểu đồ ACF sẽ nằm hoàn toàn trong dải tin cậy 95%)

→ Nhận diện white noise cực kỳ quan trọng để đánh giá chất lượng mô hình, tính dự báo của dữ liệu và tính hợp lệ của residual.





Phân rã chuỗi thời gian (Time-Series Decomposition)

Phân rã chuỗi thời gian

- là quá trình tách một chuỗi dữ liệu thành các thành phần cấu thành, giúp ta hiểu rõ hơn về xu hướng, mùa vụ và nhiễu.
- Đây là bước quan trọng trước khi mô hình hóa, dự báo hoặc phát hiện bất thường.

Mục tiêu thực tế của phân rã chuỗi thời gian:

- Hiểu cấu trúc của dữ liệu
- Hỗ trợ chọn mô hình dự báo
- Xử lý dữ liệu trước khi mô hình hóa
- Phát hiện bất thường (Anomaly Detection)
- Trực quan hóa dễ hiểu cho báo cáo

Các dạng phân rã (theo cách các thành phần kết hợp với nhau):

- Phân rã cộng (Additive Decomposition)
- Phân rã nhân (Multiplicative Decomposition)

Các thành phần chính của chuỗi thời gian

Xu hướng – Trend

- Là chiều hướng tổng quát của chuỗi trong thời gian dài: **tăng**, **giảm** hoặc **ổn định**.
- Xu hướng thường phản ánh tác động dài hạn như phát triển kinh tế, tăng trưởng dân số, mở rộng thị trường.

Mùa vụ – Seasonality

- Là thành phần lặp lại theo **chu kỳ cố định**.
- Chu kỳ có thể là:
  - 12 tháng → dữ liệu tháng
  - 7 ngày → dữ liệu theo ngày
  - 24 giờ → dữ liệu theo giờ
- Seasonality làm xuất hiện các "đỉnh lặp lại" trong biểu đồ ACF.

Nhiễu – Residual / Irregular

- Là phần còn lại sau khi loại bỏ trend và seasonality.
- Residual lý tưởng sẽ giống **white noise** → không có cấu trúc, khó dự đoán.
- Dùng để kiểm tra độ phù hợp của mô hình → residual mà có cấu trúc → mô hình thiếu thông tin.

Các dạng phân rã (theo cách các thành phần kết hợp với nhau):

Đặc điểm	Additive	Multiplicative
Biên độ mùa vụ	Biên độ mùa vụ **không đổi theo thời gian.**	Biên độ mùa vụ **phụ thuộc vào mức độ của chuỗi.**
Dạng dữ liệu	Dao động nhỏ và đều	Dao động mạnh, tăng theo thời gian
Công thức	$Y_t = T_t + S_t + R_t$	$Y_t = T_t \times S_t \times R_t$
Ví dụ	- Nhiệt độ, dân số. - Doanh thu mỗi tháng +/− tăng giảm khoảng 200 đơn vị → gần như không phụ thuộc mức doanh thu năm.	- Doanh thu, traffic website. - Tháng Tết lượng mua hàng tăng 30% so với mức nền → dao động phụ thuộc vào giá trị trend.

Vấn đề chất lượng dữ liệu trong Time Series

1. Missing values (Dữ liệu thiếu)

- Do lỗi ghi nhận, cảm biến hỏng, server downtime.
- Thiếu theo từng điểm hoặc theo cả đoạn dài (*gaps*).
- Thiếu theo pattern (ví dụ: luôn thiếu vào ban đêm do hệ thống không chạy).
- **Ảnh hưởng:** Làm sai lệch mô hình dự đoán, đặc biệt với ARIMA, LSTM.

2. Outliers / Anomalies (Ngoại lai / bất thường)

- Các giá trị spike đột ngột.
- Hệ thống đo sai trong vài phút/giờ.
- Các kỳ lễ, sự kiện đặc biệt tạo ra giá trị khác biệt lớn.
- **Ảnh hưởng:** Mô hình học sai xu hướng hoặc dự đoán sai lệch mạnh.

3. Noise / Measurement Errors (Nhiễu / lỗi đo đạc)

- Cảm biến bị sai số.
- Dao động ngẫu nhiên làm dữ liệu kém mượt.
- **Ảnh hưởng:** Làm giảm chất lượng tín hiệu → model khó học.

4. Non-stationarity (Không dừng)

- Time series thường: Có *trend* (xu hướng tăng/giảm), có *seasonality* (chu kỳ: ngày, tuần, tháng), có thay đổi cấu trúc theo thời gian (structural breaks).
- **Ảnh hưởng:** Mô hình ARIMA, Linear Regression khó học vì giả định tính dừng.

5. Irregular intervals (Khoảng thời gian không đều)

- Dữ liệu ghi không đúng tần suất
- Một số timestamp bị trùng, một số bị thiếu.
- **Ảnh hưởng:** Gây lỗi khi resample, khi tính rolling window.

6. Latency / Delayed recording (Trễ ghi nhận)

- Hệ thống gửi dữ liệu trễ vài phút/giờ.
- Thường gặp trong IoT hoặc hệ thống mạng không ổn định.
- **Ảnh hưởng:** Lệch pha so với thực tế → mô hình bị nhiễu.

7. Drift (Trôi)

- *Concept drift*: mối quan hệ giữa X và Y thay đổi theo thời gian.
- *Data drift*: phân phối dữ liệu đầu vào thay đổi.
- **Ví dụ:**Khách hàng năm 2020 mua sắm khác năm 2024.

8. Duplicate timestamps (Timestamp trùng lặp)

- Hệ thống gửi một điểm thời gian hai lần.
- Thường gặp trong log hoặc data streaming.
- **Ảnh hưởng:** Lỗi khi aggregate, phân tích theo thời gian.

9. Scaling issues (Thang đo không đồng nhất)

- Thay đổi thiết bị → giá trị đo bị đổi về thang đo khác.
- Ghép dữ liệu từ nhiều nguồn không cùng đơn vị.
- **Ví dụ:** Trước dùng °C, sau dùng °F.

10. Edge effects (Hiệu ứng biên)

- Khi tính rolling mean, ACF, trend decomposition → vài điểm đầu/cuối luôn thiếu hoặc bị nhiễu.

11. Timezone / daylight saving issues

- Không thống nhất múi giờ.
- Chuyển đổi DST → timestamp lặp hoặc nhảy.

Giải pháp xử lý từng vấn đề chất lượng dữ liệu trong Time Series

Missing Values — Dữ liệu thiếu

- **Forward fill (ffill):** Dùng giá trị gần nhất về trước.
- **Backward fill (bfill):** Dùng giá trị gần nhất về sau.
- **Interpolation (N ội suy):** Linear, Time-based, Polynomial, Spline.
- **Model-based imputation:** dùng ARIMA, Kalman Filter, Prophet, RandomForest để dự đoán giá trị bị thiếu.
- **Xoá đoạn dài bị missing:** Khi thiếu quá dài → dữ liệu mất ý nghĩa → nên remove.

Outliers / Anomalies — Ngoại lai

- **IQR / Z-score:** Loại hoặc thay thế điểm ngoại lai.
- **Rolling median / Hampel filter:**Tìm spike trong cửa sổ thời gian.
- **Model-based detection:** STL decomposition, ARIMA residuals, Isolation Forest / LOF, LSTM Autoencoder
- **Thay thế outlier**
  - nguyên (nếu là sự kiện đặc biệt thật)
  - median của cửa sổ
  - interpolate
  - hoặc giữ

Noise / Nhiễu đo đạc

- **Rolling mean / median smoothing:** Làm mượt tín hiệu.
- **Low-pass filter / Butterworth filter:** Loại bỏ tần số cao.
- **Exponential Moving Average (EMA):** Giảm nhiễu nhưng vẫn giữ được xu hướng.
- **Wavelet denoising:** Loại nhiễu nâng cao.

4. Non-stationarity — Không dừng

- **Differencing (D1, D2):** Loại trend.
- **Seasonal differencing:** Loại seasonality.
- **Log / sqrt / Box-Cox transform:** Ổn định phương sai.
- **Decomposition:** STL → trend + seasonality + residual.
- **Window transformation:** Sử dụng các feature: lag, rolling mean, rolling std.

5. Irregular Intervals — Khoảng thời gian không đều

- **Resampling**
  - resample('1H').mean()
  - resample('1D').sum()
- **Reindex toàn bộ timeline:** Tạo timeline chuẩn rồi fill dữ liệu.
- **c. Loại bỏ timestamp trùng:** group và aggregate (mean / sum / max tùy ngữ cảnh).

6. Latency & Delay — Trễ ghi nhận

- **Đồng bộ thời gian:** Đẩy dữ liệu về một chuẩn: UTC hoặc local timezone.
- **Đánh dấu điểm tới muộn:** Thêm cột is_delayed.
- **Sử dụng mô hình có khả năng xử lý delay:** Prophet, Kalman Filter.

7. Drift — Trôi dữ liệu

- **Drift detection:** ADWIN, Kolmogorov–Smirnov test, Population Stability Index (PSI)
- **Chia dữ liệu theo giai đoạn:** Train theo từng window (rolling training).
- **Online learning / Incremental training:** SGDClassifier, River ML.
- **Update model thường xuyên:** Theo tuần/tháng.

Giải pháp xử lý từng vấn đề chất lượng dữ liệu trong Time Series

8. Duplicate timestamps — Trùng timestamp

- **Gộp giá trị bị trùng:** mean, sum, max/min
  Tùy bài toán (dự báo, IoT, bán hàng).
- **Lấy giá trị mới nhất:** Khi sensor update liên tục.

9. Scaling Issues — Không đồng nhất thang đo

- **Chuẩn hóa đơn vị:** °C → °F, m → km, %
- **StandardScaler / MinMaxScaler:** Nếu ghép dữ liệu nhiều nguồn.
- **Lưu Meta / Docs:** Ghi lại thang đo theo từng giai đoạn để tracking.

10. Edge Effects — Hiệu ứng biên

- **Chấp nhận missing ở biên:** Khi rolling / smoothing → 2–10 giá trị đầu tiên/ cuối cùng thường bị ảnh hưởng.
- **Padding bằng giá trị gần nhất:** ffill / bfill.
- **Loại bỏ vài điểm đầu/cuối:** Nếu mô hình yêu cầu tính ổn định cao.

11. Timezone / DST Issues — Lỗi múi giờ

- **Chuẩn hóa về UTC:** Tránh DST và các vấn đề lặp giờ.
- **Sử dụng pandas timezone-aware:** tz_localize, tz_convert.
- **Làm sạch timestamp nhảy / lặp khi DST thay đổi:**
  - với timestamp trùng: chọn record thứ 1 hoặc thứ 2
  - với timestamp nhảy lên 1 giờ: nội suy

Trực quan hóa dữ liệu

- Vẽ biểu đồ đường theo thời gian.
- Biểu đồ phân phối, boxplot theo thời gian.
- ACF, PACF plot.
- Seasonal plot, heatmap seasonality.

Ref: https://machinelearningmastery.com/time-series-data-visualization-with-python/

XGBoost cho Time Series

Ref: https://machinelearningmastery.com/xgboost-for-time-series-forecasting/

Bài tập thực hành chương 2

Mục tiêu chung của bài tập

- Làm quen với ACF/PACF
- Hiểu white noise
- Phân rã chuỗi thời gian
- Kiểm tra chất lượng dữ liệu
- Trực quan hóa dữ liệu
- Viết báo cáo ngắn đánh giá dữ liệu

Dataset gợi ý: AirPassengers, Daily-Min-Temperatures.csv, hoặc bất kì chuỗi theo ngày.

Bài tập thực hành chương 2

Bài tập 1: Tính và phân tích ACF/PACF

1. Tải và xem trước dữ liệu (head, info, plot).
2. Tính ACF, PACF đến 40 lags.
3. Kết luận:

- Chuỗi có tự tương quan mạnh không?
- Có xu hướng/mùa vụ thể hiện qua ACF/PACF không?
- Lag nào quan trọng nhất?

Bài tập 2: Kiểm tra White Noise

1. Sinh dữ liệu white noise bằng NumPy (1000 điểm).
2. Vẽ ACF → kiểm tra xem các lag có nằm trong ngưỡng tin cậy 95% không.
3. So sánh ACF của white noise với ACF của chuỗi AirPassengers.
4. Viết 5 dòng mô tả sự khác nhau.

Bài tập 3: Phân rã chuỗi thời gian (Decomposition)

1. Chọn chuỗi dữ liệu theo tháng hoặc theo quý.
2. Dùng phương pháp STL hoặc classical decomposition.
3. Vẽ 4 thành phần: Observed, Trend, Seasonal, Residual
4. Mô tả:

- Xu hướng tăng/giảm?
- Mùa vụ mạnh hay yếu?
- Residual có giống white noise không?

Bài tập 4: Kiểm tra và xử lý chất lượng dữ liệu

1. Tạo dữ liệu thiếu (NaN) giả lập hoặc dùng dataset có missing.
2. Kiểm tra missing: isna().sum()
3. Xử lý missing bằng 2 cách: Forward-fill và Interpolation
4. Phát hiện outliers bằng: Rolling mean/variance; Z-score hoặc IQR
5. Vẽ trước và sau khi xử lý.

Bài tập 5: Trực quan hóa dữ liệu chuỗi thời gian

1. Vẽ line chart toàn bộ chuỗi.
2. Vẽ rolling mean (window=12) và rolling std.
3. Vẽ seasonal plot theo tháng (đối với dữ liệu theo tháng).
4. Vẽ heatmap seasonality (month × year).
5. Kết luận: Xu hướng chung, các mùa vụ nổi bật, bất thường (nếu có)

Bài tập 6: Viết **báo cáo** gồm:

1. Mô tả dataset
2. Kiểm tra ACF/PACF
3. Kiểm tra white noise
4. Phân rã chuỗi
5. Vấn đề chất lượng dữ liệu
6. Ý nghĩa trực quan hóa
7. Kết luận: dữ liệu có phù hợp để dự báo không?

CÂU HỎI ÔN TẬP CUỐI CHƯƠNG 2

Câu 1. Tự tương quan (autocorrelation) là gì? Nó giúp nhận diện điều gì trong chuỗi thời gian?

Câu 2. Sự khác nhau giữa ACF và PACF là gì? Lag quan trọng được hiểu như thế nào?

Câu 3. Thế nào là white noise? Làm sao biết một chuỗi có phải white noise?

Câu 4. Chuỗi white noise có dùng được để dự báo không? Vì sao?

Câu 5. Phân rã chuỗi thời gian gồm những thành phần nào? Mỗi thành phần mang ý nghĩa gì?

Câu 6. Phân biệt phân rã cộng và phân rã nhân (additive vs multiplicative).

Câu 7. Nêu 3 loại lỗi chất lượng dữ liệu phổ biến trong chuỗi thời gian.

Câu 8. Có những cách nào để xử lý missing values trong dữ liệu chuỗi thời gian?

Câu 9. Rolling mean được dùng để kiểm tra điều gì trong dữ liệu?

Câu 10. Biểu đồ ACF giúp ta nhận biết mùa vụ như thế nào?

Câu 11. Khi residual của decomposition giống white noise, điều đó có ý nghĩa gì?

Câu 12. Vì sao trực quan hóa dữ liệu là bước quan trọng trong phân tích chuỗi thời gian?