Data Science

Time Series Analysis

노기섭 교수

(kafa46@hongik.ac.kr)

Lecture Goals

- 시계열 데이터 이해
- 데이터 준비와 탐색
- 시계열 분해 (Time Series Decomposition)
- 정상성과 차분 (Stationarity & Differencing)
- ARIMA 계열 모델링
- 대안 모델과 최신 도구

시계열 (Time Series) 데이터 이해

시계열의 정의와 특징

시계열의 정의

- 일정한 시간 간격으로 기록된 관측값의 집합
- 시간 순서가 매우 중요 (순서가 바뀌면 패턴 소실)

주요 특징

- 자기상관 (Autocorrelation)인접 시점 간 유사성 존재
- 예: 오늘 전력 사용량 ↔ 어제 전력 사용량

비정상성 (Non-stationarity)

- 시간 경과에 따라 평균·분산이 변화
- 단순 통계로 전체 구조 설명 어려움

시계열 구성 요소 (Components of Time Series) - 1/2

Trend (추세)

- 장기적으로 상승 또는 하락하는 흐름
    - 예: 학기 중 유튜브 구독자 수 꾸준히 증가
- 분석 포인트: 최근 흐름을 더 강조해야 함

Seasonality (계절성)

- 일정한 간격으로 반복되는 패턴
    - 예: 시험 전 매출 급증, 방학 중 감소
- 분석 포인트: 주기 고려 차분, SARIMA 등 계절형 모델 필요

시계열 구성 요소 (Components of Time Series) - 2/2

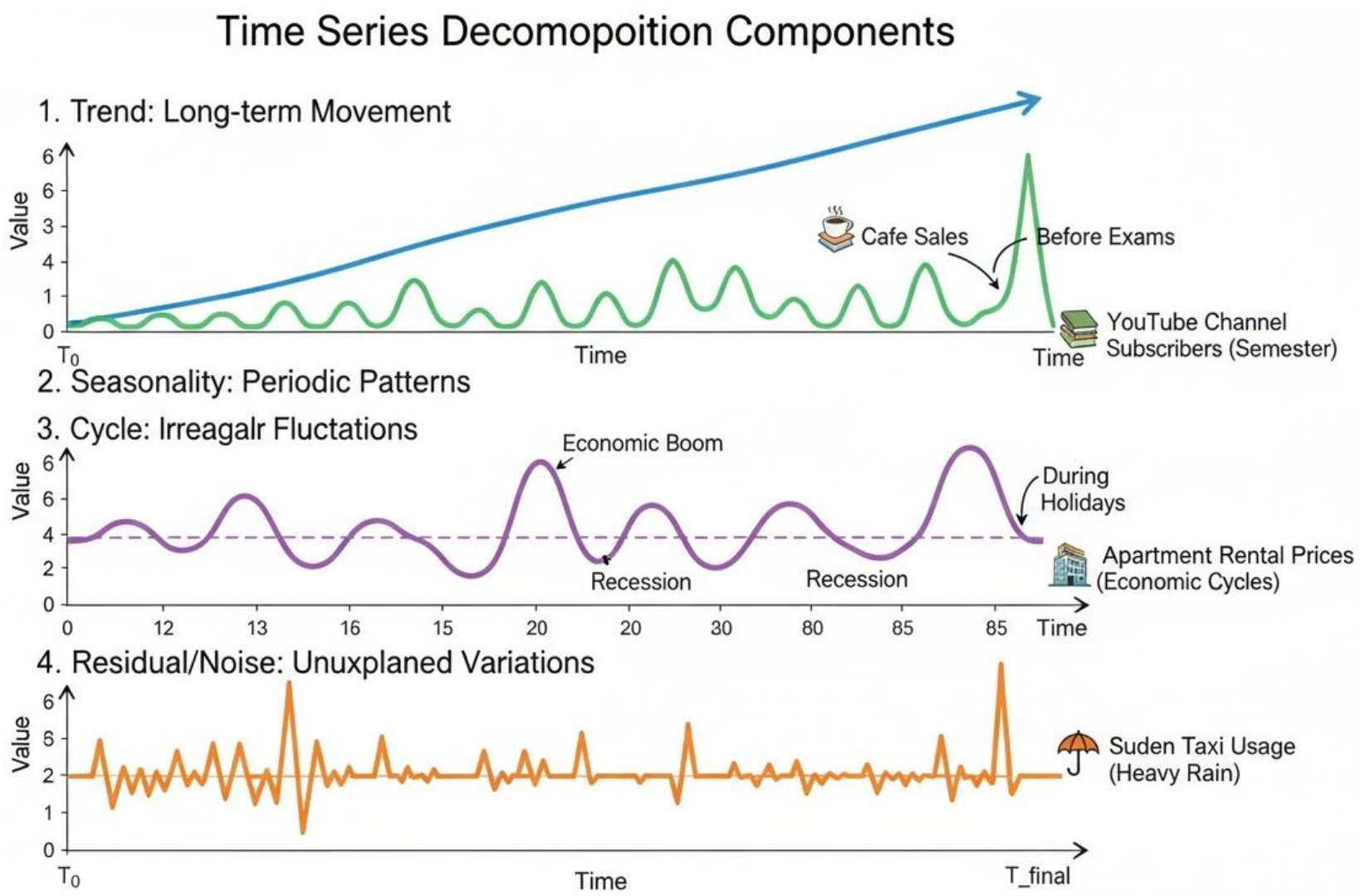
Cycle (주기성)

- 계절성보다 긴 불규칙한 변동
    - 예: 경기 변화로 인한 임대료 등락
- 분석 포인트: 거시경제 지표와 함께 해석

Residual / Noise (불규칙성)

- 추세·계절성·주기로 설명되지 않는 임시 변동
    - 예: 폭우로 하루 택시 이용 급증
- 분석 포인트: 스무딩 / 이상치 탐지로 영향 최소화

시계열을 네 가지 요소로 분해했을 때의 모습



데이터 준비와 탐색

AirPassengers 데이터

AirPassengers 데이터

- 1949년부터 1960년까지의 국제선 여객 수를 월별로 기록해 장기 추세와 뚜렷한 계절성을 동시에 관찰할 수 있는 대표적인 예제다.
- 시계열 전처리에서 자주 마주치는 과제(누락된 월, 문자열 형태의 날짜, 시간대 미지정 등)를 한 번에 연습할 수 있다

시계열(Time Series) 분석

- 단순히 날짜가 포함된 데이터를 다루는 것이 아니라, 시간의 흐름에 따라 규칙적으로 발생하는 데이터를 이해하고 예측하는 과정
- 컴퓨터(특히 pandas)가 "이건 시간에 따라 변화하는 데이터야"라고 인식하도록 형태를 변환해 주는 과정

Dataset 다운로드

https://www.deepshark.org/courses/data_science/w/07_time_series#airpassengers_dataset

시간 인덱스 처리

시계열 데이터를 다룰 때 가장 먼저 확인해야 하는 것

- "시간 정보가 규칙적인 간격으로 정렬돼 있는가?" 를 확인
- 날짜가 문자열로 남아 있거나, 몇 달이 빠져 있는 채로 분석하면 이후 모델링 단계에서 오류 발생

처리 절차

- 원시 데이터 구조 확인
  - `read_csv()` 후 `head()`로 열 이름과 값 범위를 즉시 확인
  - 시간 순서 컬럼을 잘못 이해하면 이후 변환에서 오류 발생
- 날짜 컬럼을 datetime으로 변환
  - "Month"는 문자열 상태로 들어오는데, 이를 변환하지 않으면 `asfreq`, 이동평균, 시각화 등 `Pandas`의 시계열 함수를 사용할 수 없음
- 시간 인덱스 설정과 간격 강제
  - `set_index("Month").asfreq("MS")`를 통해 월 첫 날짜(Month Start) 기준으로 인덱스

.asfreq() 메서드

■ **.asfreq(): 시계열 데이터 주기(frequency) 를 설정하거나 변경할 때 사용하는 메서드**

옵션	의미	설명	예시 인덱스
"A"	연도별 (Annual)	매년 12월 31일 기준으로 인덱스 설정	2020-12-31, 2021-12-31
"AS"	연초 기준 (Annual Start)	매년 1월 1일 기준으로 인덱스 설정	2020-01-01, 2021-01-01
"Q"	분기별 (Quarter End)	3월, 6월, 9월, 12월 말일	2020-03-31, 2020-06-30
"QS"	분기 시작 (Quarter Start)	분기 첫 달 1일	2020-01-01, 2020-04-01
"M"	월말 기준 (Month End)	매월 말일	2020-01-31, 2020-02-29
"MS"	월초 기준 (Month Start)	매월 1일	2020-01-01, 2020-02-01
"W"	주간 (Weekly)	매주 일요일 (기본값)	2020-01-05, 2020-01-12
"W-MON"	월요일 시작 주간	매주 월요일 기준	2020-01-06, 2020-01-13
"D"	일별 (Daily)	하루 단위 인덱스	2020-01-01, 2020-01-02
"B"	영업일 기준 (Business Day)	주말 제외한 평일 기준	2020-01-01, 2020-01-02
"H"	시간별 (Hourly)	1시간 단위	2020-01-01 00:00, 01:00, 02:00
"T" 또는 "min"	분 단위 (Minute)	1분 단위	2020-01-01 00:00, 00:01
"S"	초 단위 (Second)	1초 단위	2020-01-01 00:00:00, 00:00:01

타임존 명시 필요성

타임존 (Time Zone)

- 전 세계를 기준시(UTC, Coordinated Universal Time)로 나눈 표준 시간대
  - 한국: UTC+09:00
  - 미국 뉴욕: UTC-05:00
  - 동일한 시각이라도 위치에 따라 시각 표기 다름
- "AirPassengers 데이터"는 과거 로그에 시간대가 포함돼 있지 않음 → 무시해도 무방함.
- 그러나 서버 로그처럼 UTC 기준으로 떨어지는 데이터라면,
- 원본 Time Zome을 명시하고, 분석 대상 지역에 맞춰 변환하는 과정을 거쳐야 함

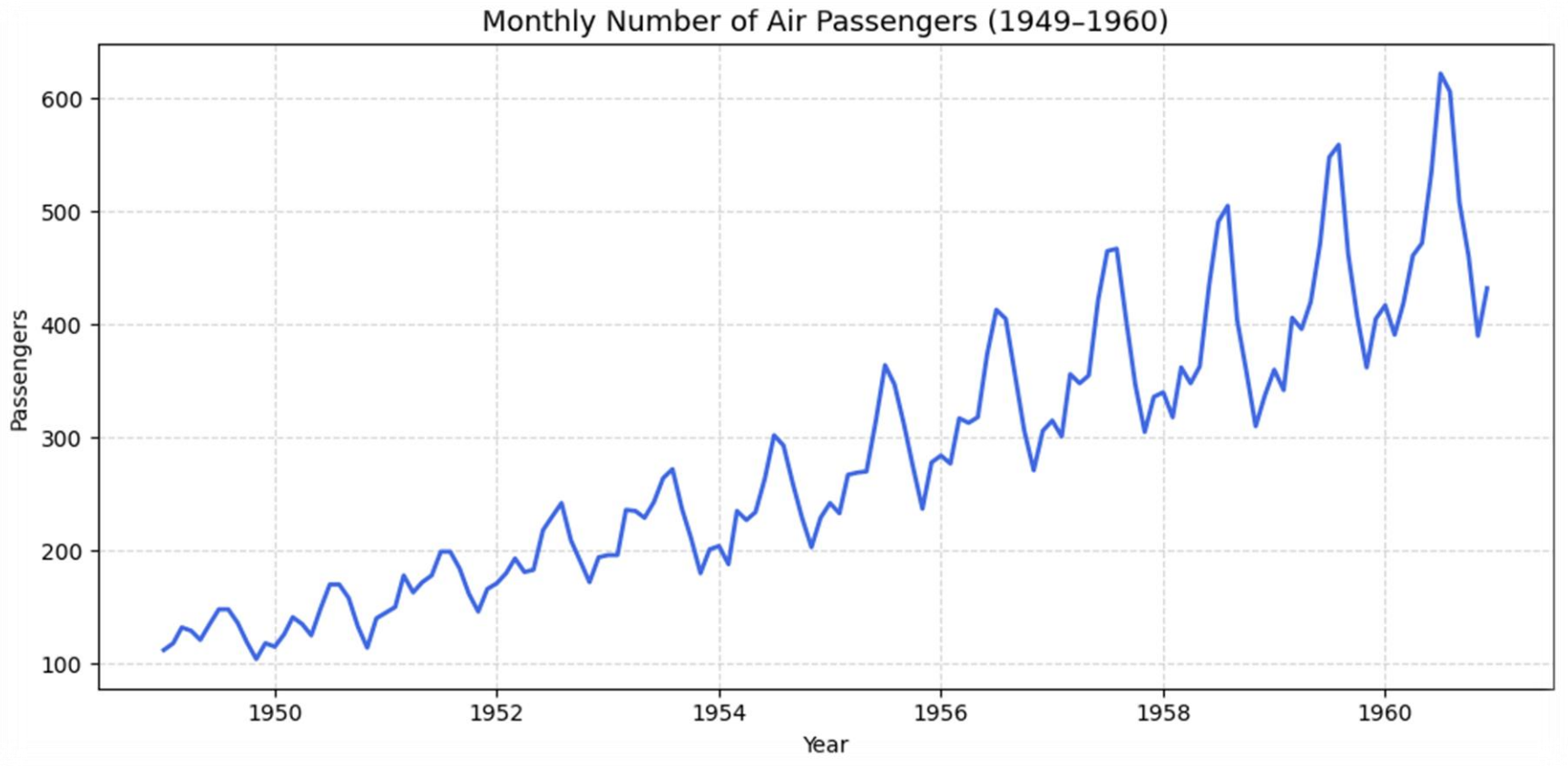
타임존 처리하지 않는 경우

- 시간대 처리를 소홀히 하면 일중 (daily) 패턴 분석에서 시간이 어긋난 결과를 얻을 수 있음

AirPassengers 데이터 실습

실습 코드

- https://www.deepshark.org/courses/data_science/w/07_time_series#time_series_practice



결측치 확인

결측치 (Missing Value)

- 데이터가 일부 누락된 상태
- 원인: 센서 오류, 수집 누락, 시스템 문제
- 영향: 평균·분산·상관계수 왜곡, 모델 학습 오류 발생

해결 방법

- 보간(Interpolation)
- 이전값 대체(Forward Fill)
- 도메인 지식 기반 보정

결측치 실습 - Dataset

Air Quality Dataset 및 변수(Features)

- Dataset: https://www.deepshark.org/courses/data_science/w/07_time_series#airquality_dataset
- 특징
  - 시간별 또는 일별로 측정된 대기 오염 물질(`PM2.5, PM10, NOx` 등) 데이터
  - 일부 시점에 결측치 또는 비정상값(`-200` 등)이 포함되어 있음.

실습 목표

- 시계열 인덱스 처리 및 기본사항 확인
- 결측치 탐지 및 보정 (보간, ffill/bfill 등)
- 이상치 탐지 및 보정
- 대기질 추세 분석

결측치 실습 - 구현 및 결과 확인

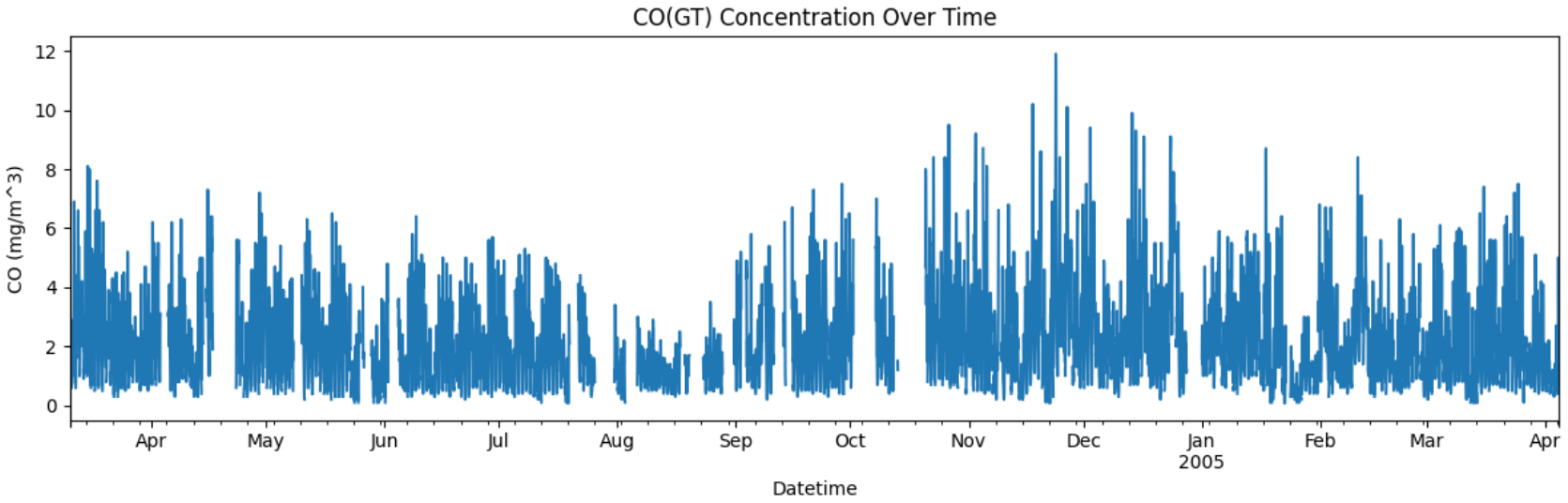
의존성 설치 (모듈 임포트 에러가 나는 경우)

ImportError: Missing optional dependency 'openpyxl'. Use pip or conda to install openpyxl.

```
pip install openpyxl
```

실습 코드

- https://www.deepshark.org/courses/data_science/w/07_time_series#airquality_codes



결측치 탐지 및 보정

결측치 보정

- 센서 오류, 통신 지연, 측정 중단 등으로 인해 결측치(missing value) 가 자주 발생
- 결측치는 분석이나 모델 학습에 직접적인 영향을 주므로, 반드시 탐지하고 적절히 보정

결측치를 처리하는 대표적인 방법

방법	설명	특징
interpolate()	이전·다음 값의 선형 관계를 기반으로 보간(보간법)	연속적인 시계열에 적합
ffill() (forward fill)	직전의 관측값으로 결측치 채움	센서 일시 오류에 유용
bfill() (backward fill)	이후의 관측값으로 결측치 채움	데이터 앞부분의 공백에 사용

선형보간법(Linear Interpolation)

선형보간법

- 데이터 사이에 빠진 값을 이전 값과 다음 값을 직선(선형) 연결해서 그 중간 값을 채워 넣는 방법
- "결측된 구간에서도 값이 일정한 속도로 변한다고 가정하고, 그 사이의 값을 계산"하는 접근

시간	온도(℃)
10시	20
11시	**NaN (측정 누락)**
12시	24

- 10시엔 20℃, 12시엔 24℃니까

"11시는 두 값의 중간쯤인 22℃ 정도일 거야"

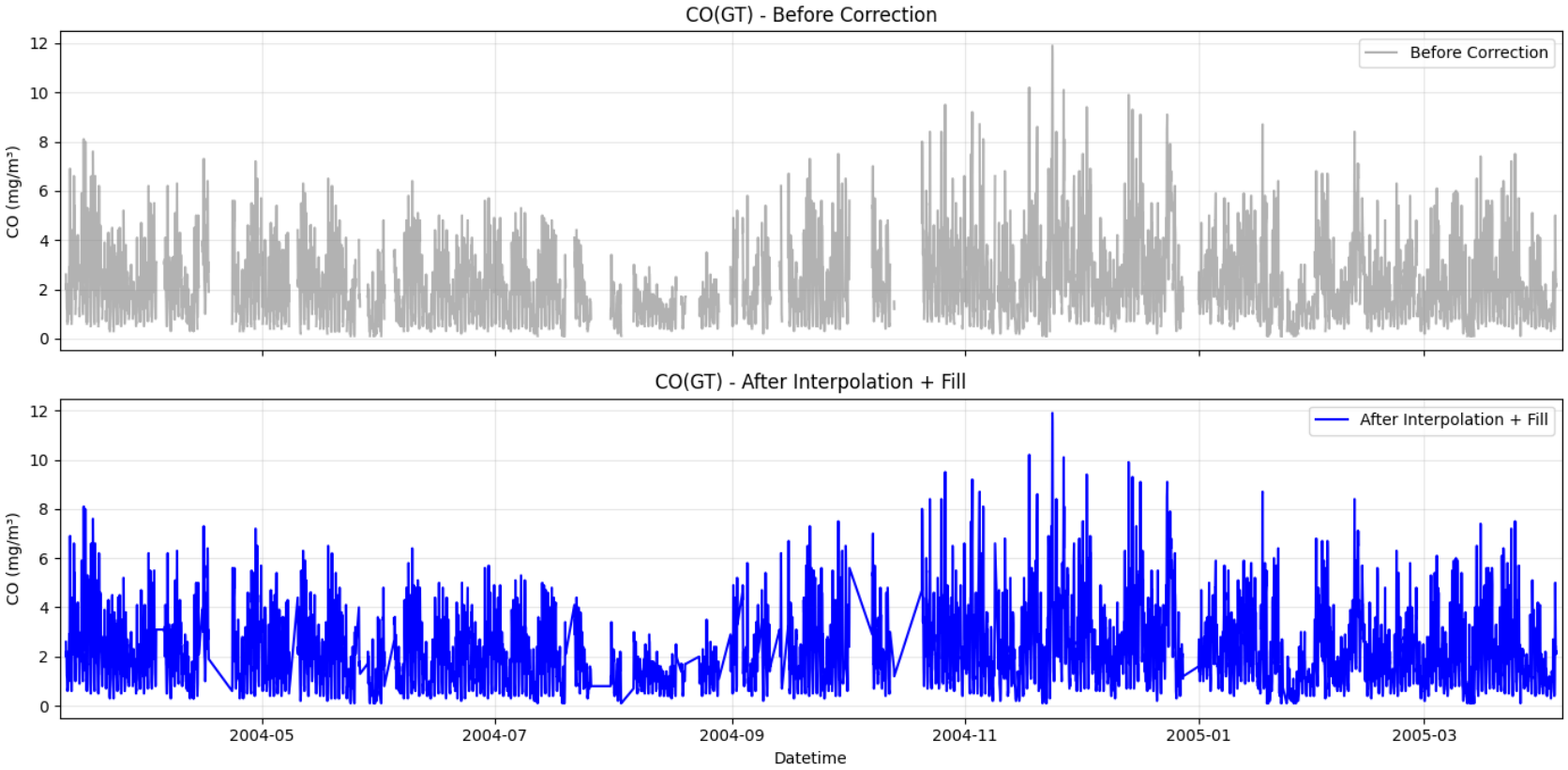
라고 선형적으로 추정

$$y(x) = y_1 + \frac{y_2 - y_1}{x_2 - x_1}(x - x_1)$$

결측치 보정 실습

소스코드

- https://www.deepshark.org/courses/data_science/w/07_time_series#correction_before_after



이상치 (Outlier) 처리

이상치 (Outlier)

- 정상 패턴에서 벗어난 비정상적 값
- 원인: 센서 오작동, 외부 환경 요인
- 영향: 평균 왜곡, 회귀 기울기 비정상 변화

탐지 및 처리

- IQR(사분위 범위) 기반 탐지
- 이동평균 편차 확인
- 도메인 기준 임계값 설정 → 필요 시 보정 또는 제거

$$IQR = Q_3 - Q_1$$

- $Limit_{low} = Q_1 - 1.5 \times IQR$,
- $Limit_{upper} = Q_3 + 1.5 \times IQR$

$$|x_t - \bar{x}_t| > k \times \sigma_t$$

- $\bar{x}_t$: 최근 일정 구간의 평균
- σ_t: 최근 구간의 표준편차
- k: 임계값 (보통 2~3)

이상치 보정 방법

방법	설명	예시
삭제(drop)	명백한 오류값 제거	데이터 손실 가능
평균/중앙값 대체	주변 통계값으로 치환	극단값 완화
보간(interpolation)	시간 흐름 고려 선형 대체	연속성 유지
윈도우 스무딩	급격한 변화 완화	장기 추세 반영에 유리

이동평균과 스무딩

■ 이동평균(Moving Average)

- 일정한 창(window)에 포함된 관측값의 평균으로, 단기 변동을 완화하고 장기 추세를 강조
- 창 크기 ↑ ➔ 노이즈 ↓ ➔ 하지만 반응 속도가 느려짐 (분석 목적에 맞게 적정 값 선택)
- 가중 이동평균(Weighted Moving Average)이나 지수평활(Exponential Smoothing)처럼 최근 관측치에 더 큰 가중치를 주는 변형도 실무에서 자주 사용

■ 윈도우 스무딩 (Window Smoothing)

- 시계열 데이터의 단기적인 요동(잡음)을 완화
- 전체적인 장기 추세(trend) 를 부드럽게 보기 위해 사용하는 기법

$$\tilde{y}_t = \frac{1}{n} \sum_{i=t-n+1}^{t} y_i$$

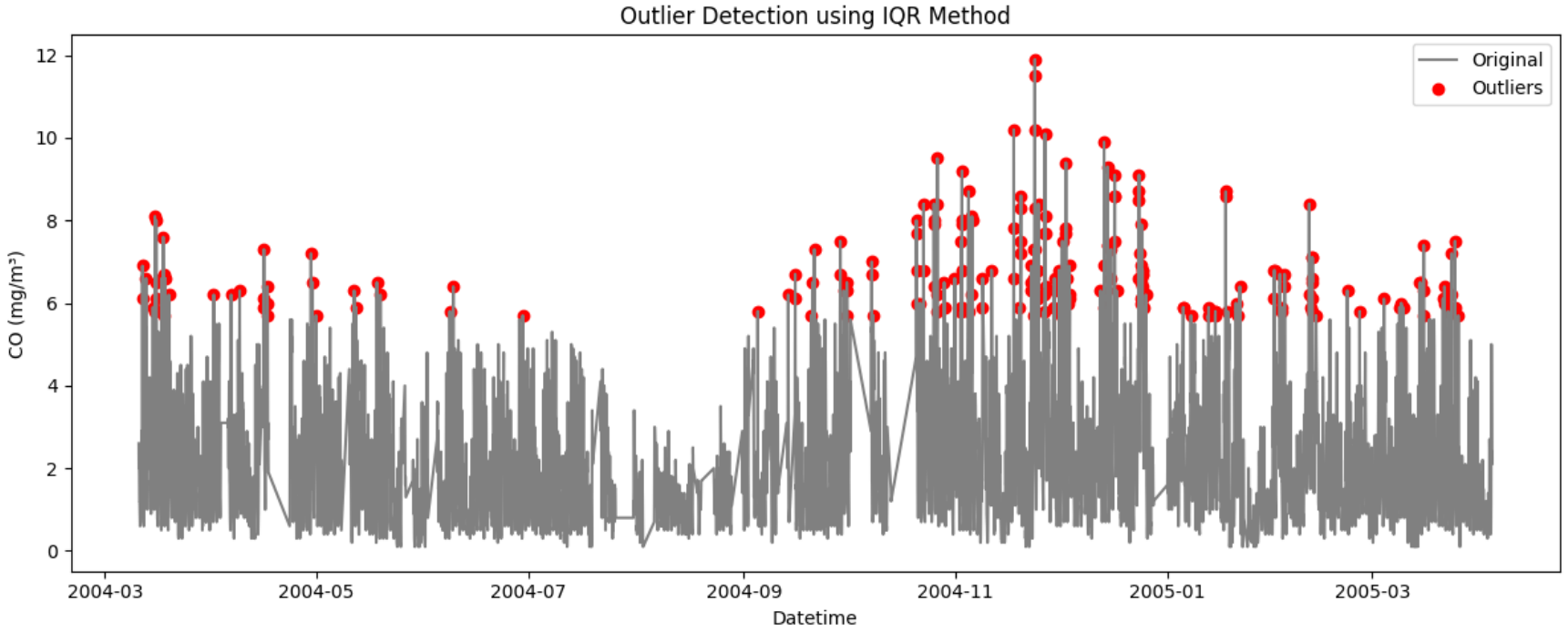
- $\tilde{y}_t$: 스므딩(평활화) 된 값
- n : 윈도우 크기 (예: 3시간, 7일 등)
- y_i: 시점 i의 실제 관측값

이상치 탐지 및 시각화

소스코드

- 이상치 탐지 및 시각화

  https://www.deepshark.org/courses/data_science/w/07_time_series#outlier_detection_visualization

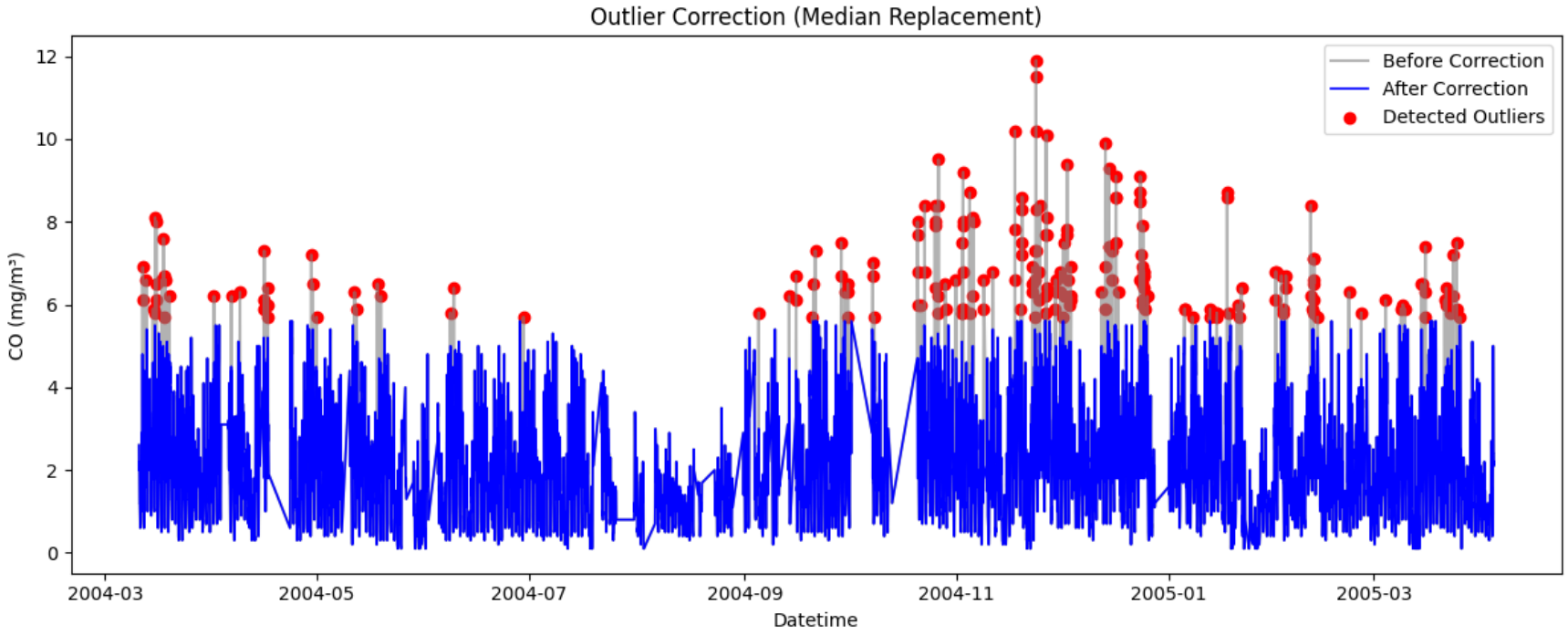


이상치 탐지 및 중앙값 보정 실습

이상치 탐지 후 중앙값으로 보정하는 실습

- 실습 코드

- https://www.deepshark.org/courses/data_science/w/07_time_series#outlier_correction_median



종합 실습 - 대기질 추세 분석

대기질 추세 분석

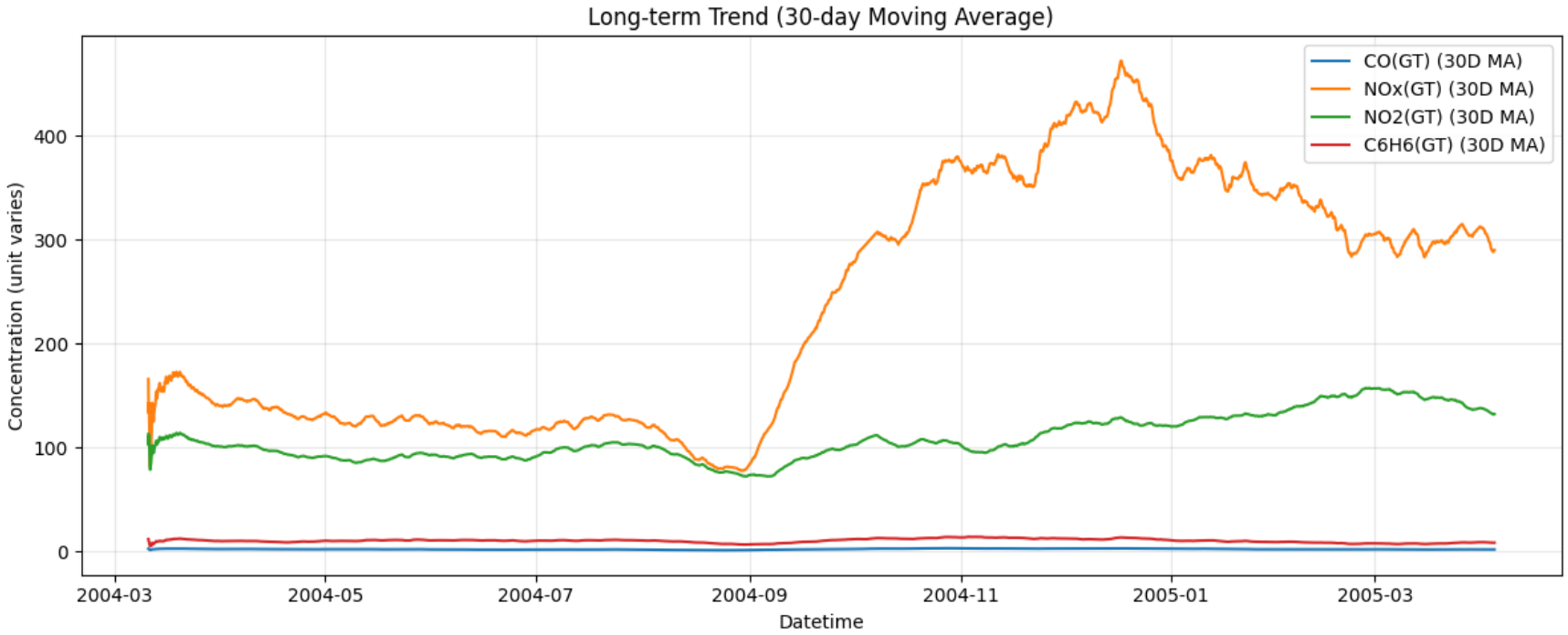
분석 관점	설명
장기 추세 (Long-term Trend)	1년 단위로 CO, NOx, Benzene 등의 평균 농도 변화를 분석 (예: 겨울철 오염 증가)
계절별 변화 (Seasonality)	월별 또는 분기별 평균값 비교를 통해 봄·여름·가을·겨울 간 대기질 차이 탐색
일중 패턴 (Daily Pattern)	하루 중 오전/오후/야간의 오염물질 변화 관찰 (예: 출퇴근 시간대 CO 증가)
센서 간 상관관계	여러 센서(PT08.S1~S5)와 오염물질 간의 관계 파악

실습 코드

- https://www.deepshark.org/courses/data_science/w/07_time_series#airquality_overall_practice

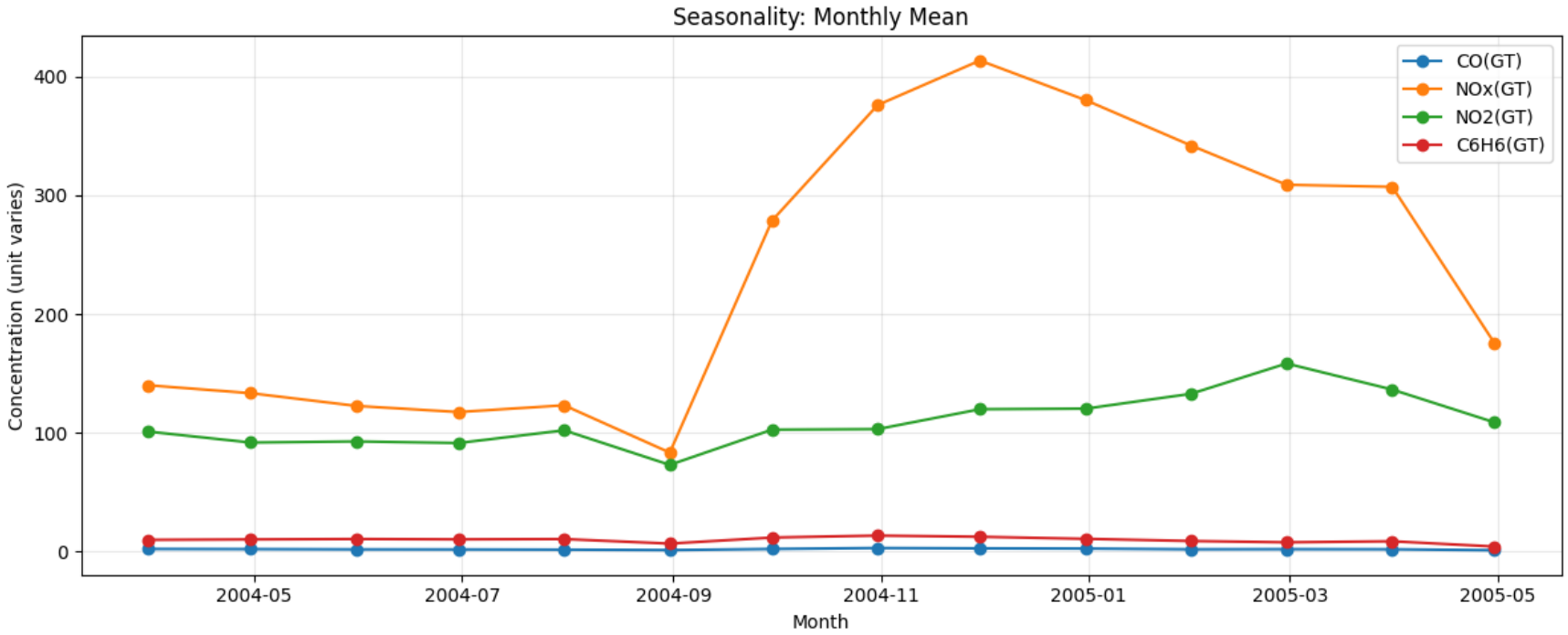
종합 실습 - 대기질 추세 분석 – 실행 결과

장기 추세(Long-term Trend)



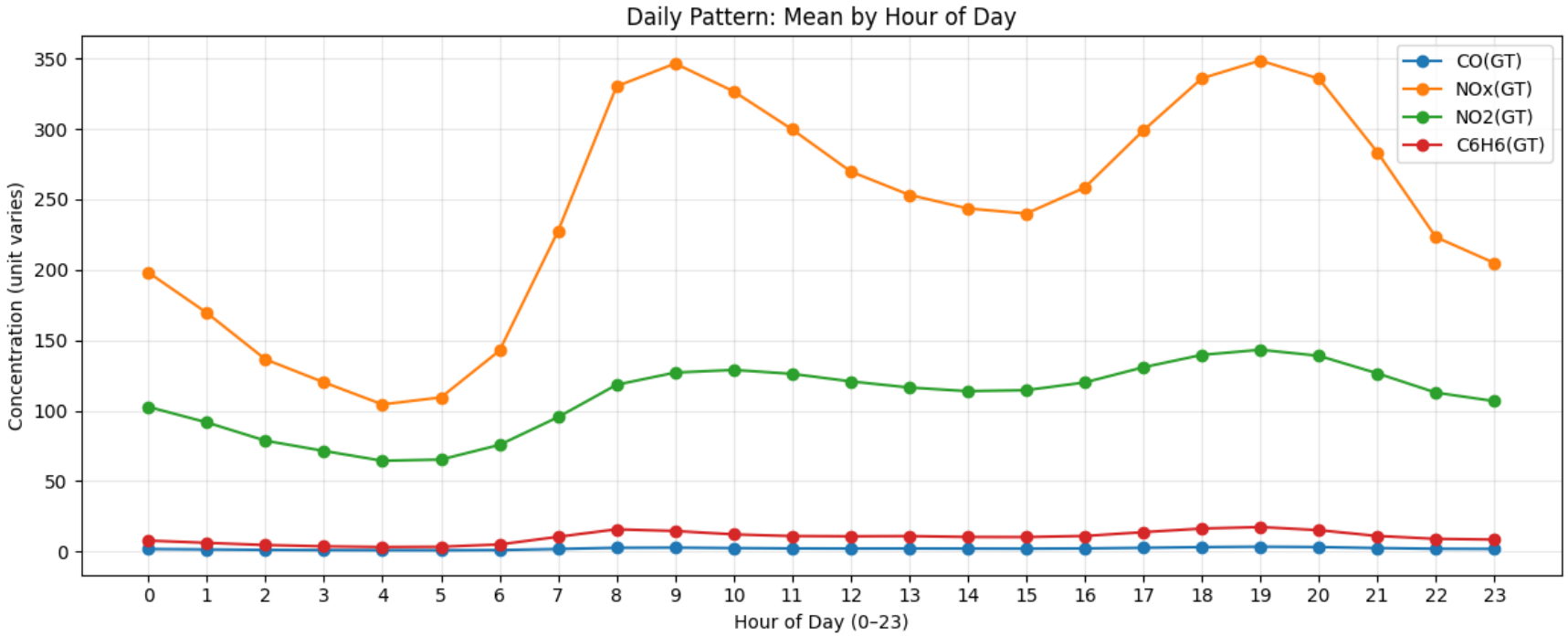
종합 실습 - 대기질 추세 분석 – 실행 결과

계절별 변화(Seasonality)



종합 실습 - 대기질 추세 분석 – 실행 결과

일중 패턴(Daily Pattern)



종합 실습 - 대기질 추세 분석 – 실행 결과

센서 간 상관관계

시계열 분해
(Time Series Decomposition)

시계열 분해

시계열 데이터

- 단순히 시간에 따른 값의 나열이 아니라, 다양한 패턴의 합성으로 이루어진다.

시계열 분해(Time Series Decomposition)

- 다양한 합성 패턴을 분해하는 과정

시계열 분해를 하는 이유

목적	설명	예시
구조 이해	추세·계절·노이즈를 분리하여 패턴 파악	온도 상승 + 계절 변동
예측 개선	비정상성 제거로 모델 성능 향상	ARIMA 학습 안정화
이상 탐지	잔차를 통해 예외적 사건 탐색	폭염으로 인한 전력 급등
해석력 향상	각 요인의 기여도를 설명	매출 상승 원인 분리

시계열의 구성 요소 (Components of a Time Series)

■ **시계열의 구성 요소: 4가지 성분의 합으로 표현**

$$Y_t = T_t + S_t + C_t + I_t$$

구성 요소	설명	예시
추세 (Trend, T_t)	데이터가 장기간에 걸쳐 증가하거나 감소하는 전반적 경향	평균기온 상승, 판매량 증가
계절성 (Seasonality, S_t)	일정 주기마다 반복되는 주기적 패턴	월별 전기 사용량, 요일별 교통량
순환성 (Cyclic, C_t)	경기 변동과 같은 비정기적 장기 요동	경기 침체, 호황 주기
불규칙성 (Irregular, I_t)	예측 불가능한 우연적 요인	천재지변, 이벤트 효과

※ 대부분의 분석에서는 C_t 와 I_t 를 합쳐서 단순히 잔차(residual)로 다루기도 한다.

분해 방식 (Types of Decomposition)

가법모형 (Additive Model)

$$Y_t = T_t + S_t + I_t$$

- 각 성분이 더해진 상태로 결합
- 성분 진폭이 일정한 경우에 해당 (예: 월별 온도 변화)

승법모형 (Multiplicative Model)

$$Y_t = T_t \cdot S_t \cdot I_t$$

- 각 성분이 곱으로 결합된 형태
- 데이터의 변동 폭이 시간에 따라 달라질 때 (예: 매출액이 점점 커지며 변동 폭도 커질 때)

시계열 분해 실습 - 구현

의존성 설치

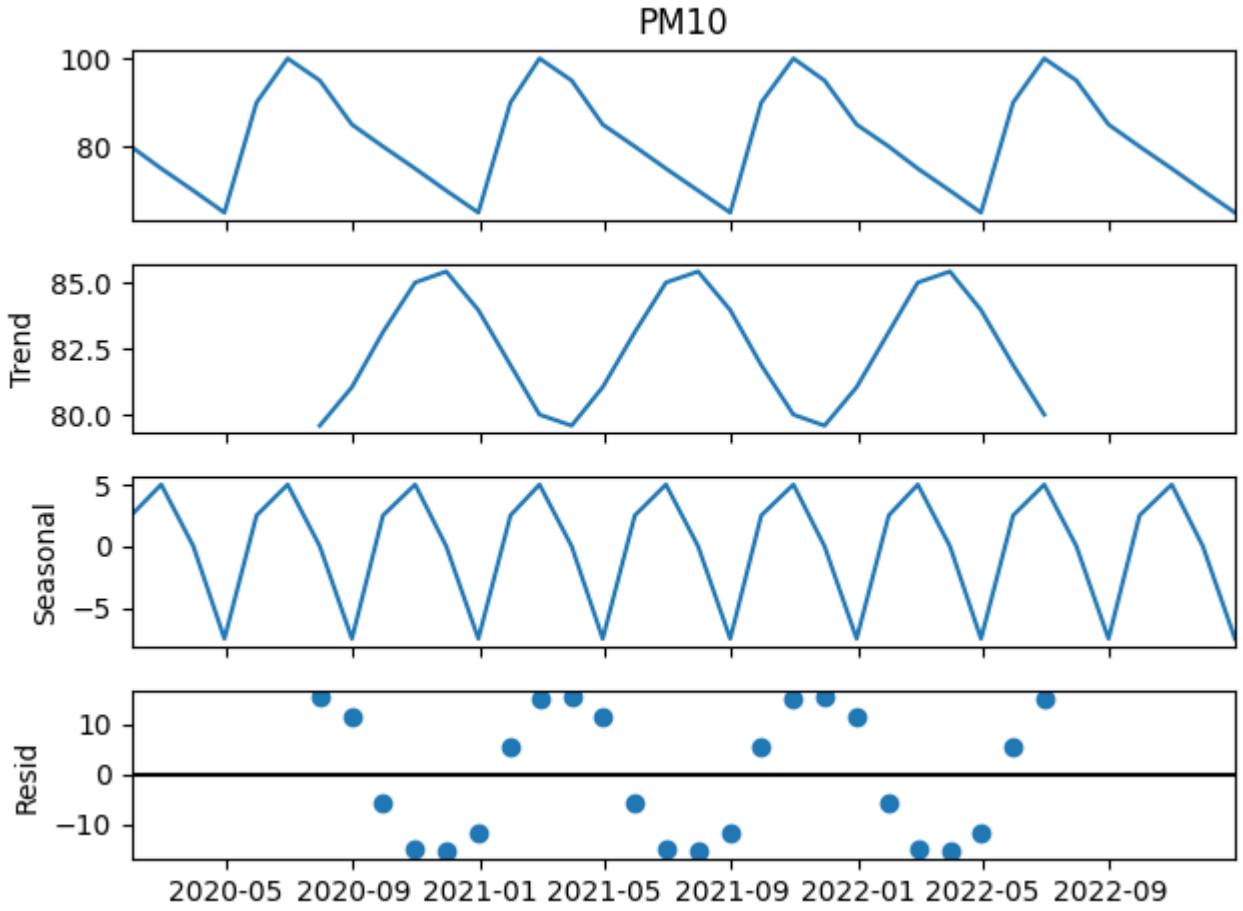
```
# 통계 패키지 설치
pip install statsmodels
```

실습 코드

- https://www.deepshark.org/courses/data_science/w/07_time_series#decompose_practice

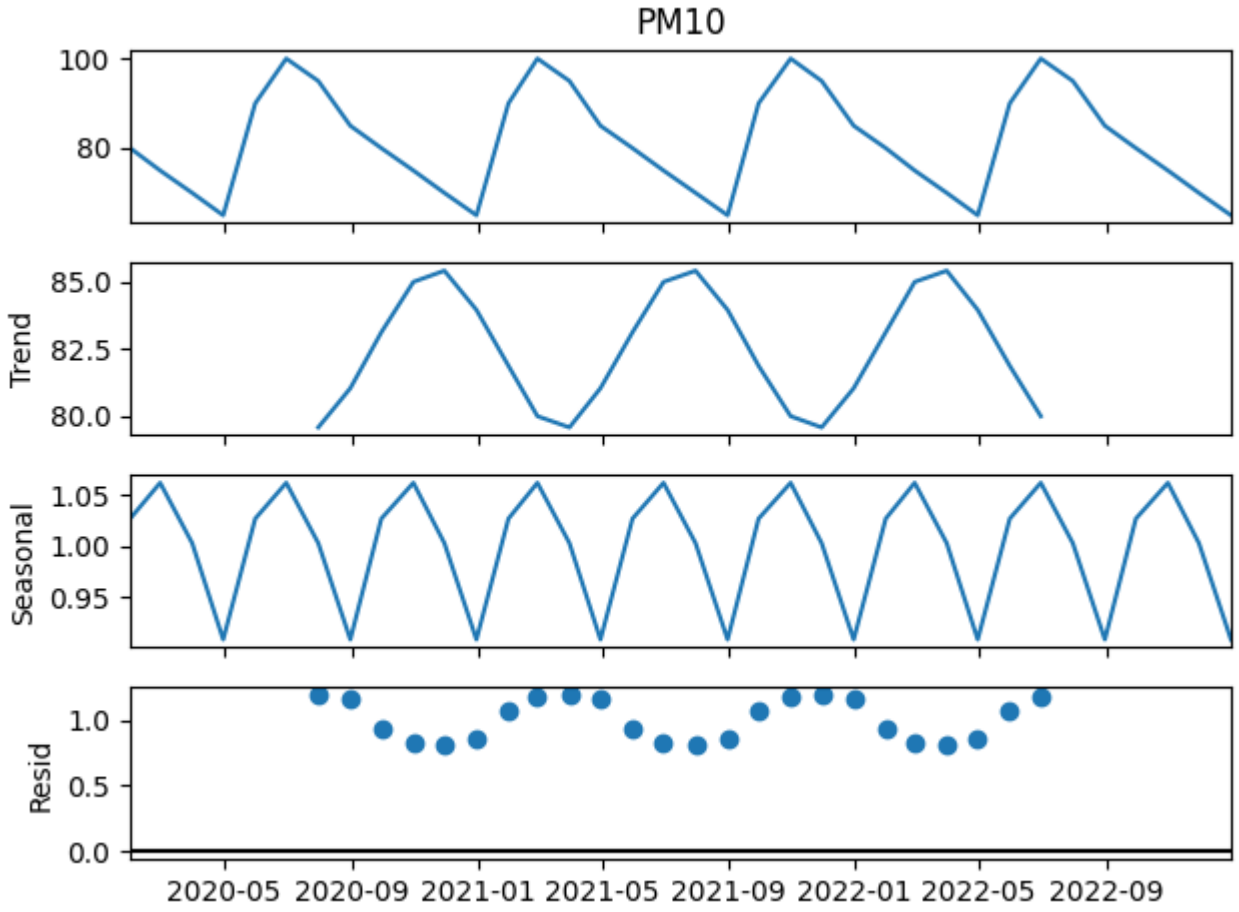
시계열 분해 실습 - 시각화

■ 시계열 분해 - 가법 (Additive) 모형 적용 결과



시계열 분해 실습 - 시각화

■ 시계열 분해 - 승법 (Multiplicative) 모형 적용 결과



정상성과 차분
(Stationarity & Differencing)

정상성(Stationarity)

정상성의 개념

- 시간이 지나도 데이터의 통계적 성질이 변하지 않는 성질

항목	내용
정의	시간에 따라 평균, 분산, 공분산이 일정한 시계열
의의	모델이 안정적으로 동작하고 미래 예측의 타당성을 확보
비정상 예시	꾸준히 증가하는 매출, 계절적 반복이 있는 데이터
정상 예시	평균을 중심으로 진동하는 노이즈형 데이터

정상성의 개념과 필요성

- 전통적 시계열 모델은 데이터가 정상적(stationary) 일 때만 통계적으로 타당한 예측이 가능

정상성(Stationarity) 판별

■ 단위근(Unit Root)?

- 시계열 모델에서 단위근(Unit Root)?
  - "시계열이 자기 자신의 과거 값에 지나치게 의존하는 상태" 를 의미
  - 즉, 이전 시점의 영향이 시간이 지나도 사라지지 않고 누적되는 경우

$$y_t = \phi y_{t-1} + \epsilon_t$$

- ϕ: 자기 회귀 계수
- ϵ_t: 백색 잡음

만약 $\phi = 1$ 이라면, 누적 오차가 쌓여서

시간이 지날수록 평균과 분산이 계속 변함

➔ 비정상 시계열 (Non-stationary)

현상	의미
평균이 일정하지 않음	시간에 따라 평균이 변함
분산이 증가	시간이 지날수록 변동성 커짐
충격이 누적	일시적 변동이 장기적으로 영향 미침

예: 주가, 환율: 한 번의 급등이 이후에도 누적되어 남음

ADF Test

Augmented Dickey-Fuller Test (단위근 검정, ADF Test)

- "시계열에 단위근이 존재하는가?"를 검정하는 통계적 방법

항목	설명
귀무가설 (H_0)	단위근 존재 → 비정상 시계열
대립가설 (H_1)	단위근 없음 → 정상 시계열
판단 기준	p-value < 0.05 → 귀무가설 기각 ➔ 정상성 확보

결과 해석	의미
p-value < 0.05	정상 시계열 (차분 불필요)
p-value ≥ 0.05	비정상 시계열 (차분 필요)

```
from statsmodels.tsa.stattools import adfuller


adf_result = adfuller(df["Passengers"])
print(f"ADF Statistic: {adf_result[0]:.3f}")
print(f"p-value: {adf_result[1]:.4f}")
```

차분(Differencing)

차분(Differencing)

- 이전 시점의 값을 현재 시점에서 빼서,
- 변화량을 기반으로 한 새로운 시계열을 만드는 과정
- 비정상 시계열은 차분(differencing) 을 통해 정상 시계열로 변환 가능

차분의 종류

차분 종류	수식	설명
1차 차분	$Y_t' = Y_t - Y_{t-1}$	추세(Trend) 제거
계절 차분	$Y_t'' = Y_t - Y_{t-s}$	계절성(Seasonality) 제거
복합 차분	1차 차분 + 계절 차분 순차 적용	추세와 계절성이 모두 존재할 때 사용

차분(Differencing) 수행 예시

```
# 1차 차분 (Trend 제거)

df["diff1"] = df["Passengers"].diff(periods=1)


# 계절차분 (Seasonal Difference): 계절성 제거

#   12는 월별 데이터에서 1년 주기(12개월) 를 의미

df["diff_seasonal"] = df["Passengers"].diff(periods=12)


# 복합 차분: 추세 + 계절성 모두 제거

df["diff_both"] = df["Passengers"].diff(periods=1).diff(periods=12)
```

변동성(분산) 안정화

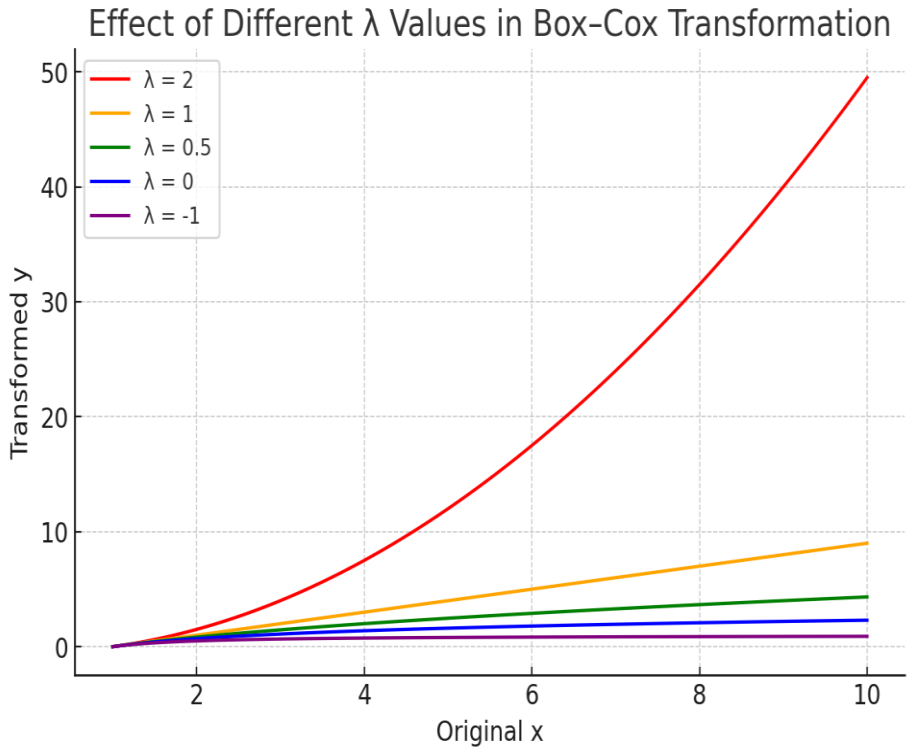
■ 시계열의 변동 폭이 일정하지 않은 경우,

- 로그 변환(Log Transform) 또는 Box-Cox 변환을 사용하여 분산을 안정화할 수 있음

방법	수식	설명
로그 변환	$Y_t' = \log(Y_t)$	값이 커질수록 변동 폭이 커지는 경우 사용
Box-Cox 변환	$Y_t' = \frac{Y_t^{\lambda} - 1}{\lambda}$	로그보다 일반화된 형태로, 통계적 안정화에 효과적

λ: 변환 파라미터

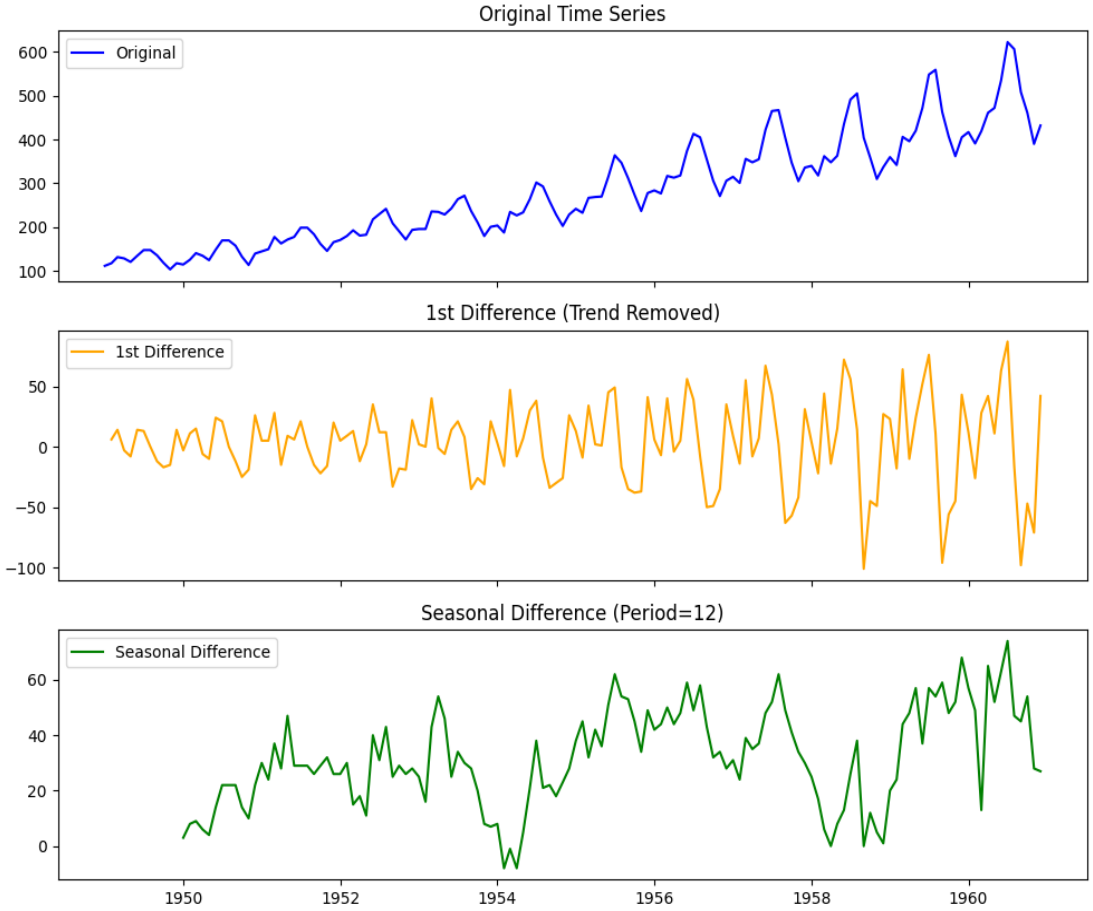
- λ 값에 따라 데이터의 형태가 달라짐
- λ 는 보통 MLE(최대우도추정)으로 자동 추정됨



차분(Differencing) 실습

실습 코드

- https://www.deepshark.org/courses/data_science/w/07_time_series#stationary_differencing_practice



1949년부터 1960년까지의 국제 항공 여객 수를 나타낸 원본 시계열이다. 전반적으로 시간이 지남에 따라 여객 수가 꾸준히 증가하고 있으며, 연도마다 일정한 주기적 변동(계절성)이 반복된다. 이러한 패턴은 비정상 시계열(Non-stationary) 의 전형적인 특징이다.

1차 차분을 적용한 결과로, 전 시점과의 차이를 계산하여 추세(Trend) 성분이 제거되었다. 데이터의 평균이 일정한 중심을 기준으로 진동하는 형태로 바뀌며, 장기적인 증가 경향이 사라졌다. 그러나 여전히 주기적인 변동(계절성)은 남아 있다.

12개월 주기의 계절 차분을 적용한 결과로, 계절적 반복 패턴이 제거되었다. 이제 데이터는 특정 주기에 따른 진폭 변화가 감소하고, 평균이 일정하게 유지되는 정상 시계열(Stationary Series) 형태에 가까워졌다.

ARIMA 계열 모델링

ARIMA 계열 모델링

ARIMA?

- ARIMA(**A**uto **R**egressive **I**ntegrated **M**oving **A**verage)는 비정상 시계열 데이터를 차분을 통해 정상화한 뒤,
- AR(자기회귀)과 MA(이동평균) 요소를 결합하여 예측하는 모델

ARIMA 구성 요소

구성요소	의미	설명
AR(p)	AutoRegressive	과거 p개의 값이 현재 값에 영향을 미침
I(d)	Integrated	d차 차분을 통해 비정상성을 제거
MA(q)	Moving Average	과거 q개의 오차항이 현재 값에 영향을 미침

ARIMA 파라미터 및 확장 모델

ARIMA 파라미터

모델	구성	설명
$ARIMA(p,d,q)$	p: 자기회귀(AR) 차수 d: 차분 횟수(정상성 확보 수준) q: 이동평균(MA) 차수	비계절형 시계열 모델
$SARIMA(p,d,q)(P,D,Q)_s$	(P, D, Q):계절 성분 차수 s: 주기(예: 12개월)	계절성이 있는 시계열 모델

확장된 모델

모델	형태	설명
$ARIMA(p,d,q)$	기본형	추세 중심
$SARIMA(p,d,q)(P,D,Q)_s$	계절형	계절성 주기 s 포함
$ARIMAX$	ARIMA + 외생변수	외부 요인(x변수)을 함께 고려
$SARIMAX$	SARIMA + 외생변수	계절성 + 외생변수 모델

모델 성능지표 소개

번호	지표	의미	해석 기준 / 특징
1	AIC (Akaike Information Criterion)	모델의 적합도 + 복잡도(파라미터 수)를 동시에 고려	값이 작을수록 적합 / 과적합 위험 낮음 여러 모델 비교 시, AIC가 작은 모델 선택
2	BIC (Bayesian Information Criterion)	AIC와 유사하나 복잡도에 더 강한 패널티 부여	AIC와 유사하지만, 복잡한 모델에 더 강한 패널티 부여 값이 작을수록 좋은 모델
3	coef (Coefficient)	AR, MA, Trend 등 각 항의추정 계수	부호(+, −)로 변수의 영향 방향 해석
4	std err (Standard Error)	계수 추정의 불확실성 정도	추정치가 얼마나 불확실한지 나타냄 작을수록 신뢰도 높음(변동성 낮음)
5	Z (z-statistic)	계수의 유의성 검정 통계량 (coef / std err)	|z| 값이 크면, 계수가 통계적으로 유의할 가능성이 높음
6	P > |z| (p-value)	유의확률: 귀무가설(H0: 계수=0)을 기각할 확률	
7	Ljung–Box (Q)	잔차의 자기상관(독립성) 검정	p > 0.05 → 잔차 독립적, 모델 적합 양호
8	Jarque–Bera (JB)	잔차의 정규성(평균0, 대칭성) 검정	p > 0.05 → 정규분포 가정 만족
9	Heteroskedasticity (H)	잔차의 등분산성 여부 판단	값이 1에 가까울수록등분산(좋음)
10	Durbin–Watson	잔차의 1차자기상관 검정	2에 가까울수록 자기상관 없음 (1↓: 양의 상관, 3↑: 음의 상관)

모델 출력 값 확인

SARIMAX Results

Dep. Variable:	Passengers	No. Observations:	144
Model:	SARIMAX(1, 1, 1)x(1, 1, 1, 12)	Log Likelihood	-456.103
Date:	Fri, 17 Oct 2025	AIC	922.205
Time:	12:24:02	BIC	936.016
Sample:	01-01-1949 - 12-01-1960	HQIC	927.812
Covariance Type:	opg		

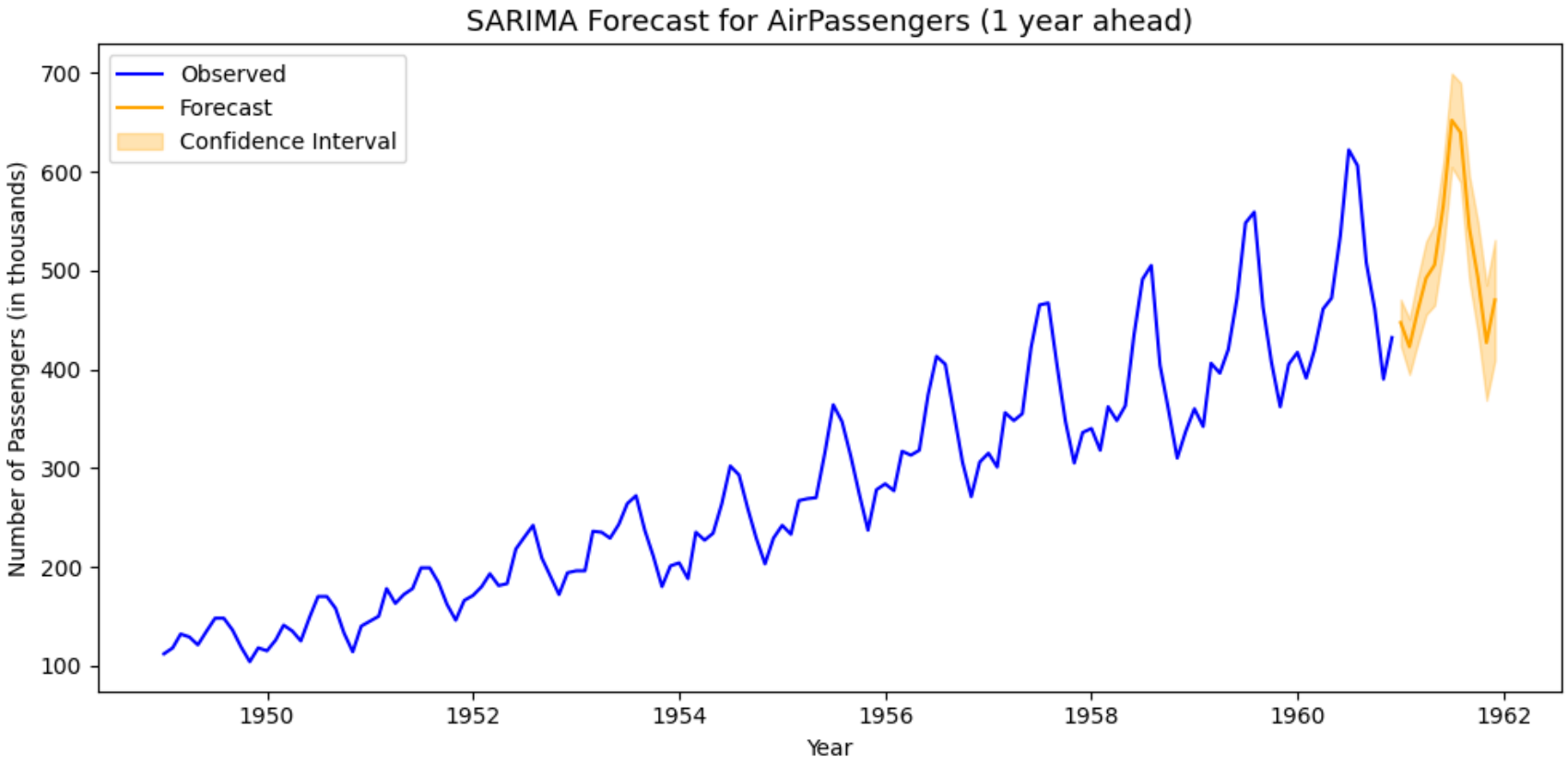
	Coef	std err	z	P>|z|	[0.025	0.975]
ar.	-0.2298	0.401	-0.573	0.567	-1.016	0.557
ma.L1	-0.0987	0.374	-0.264	0.792	-0.832	0.634
ar.S.L12	-0.5460	0.299	-1.825	0.068	-1.133	0.041
ma.S.L12	0.3959	0.352	1.125	0.261	-0.294	1.086
Sigma2	140.2945	17.997	7.795	0.000	105.020	175.569

Ljung-Box (L1) (Q):	0.00	Jarque-Bera (JB):	5.42
Prob(Q):	0.95	Prob(JB):	0.07
Heteroskedasticity (H):	2.51	Skew:	0.12
Prob(H) (two-sided):	0.01	Kurtosis:	4.03

SARIMAX 실습

실습 코드

- https://www.deepshark.org/courses/data_science/w/07_time_series#sarimax_practice



대안 모델과 최신 도구

통계 기반 시계열 모델

Prophet (by Facebook / Meta)

- 추세(Trend), 계절성(Seasonality), 이벤트(Holiday) 독립적 모델링
- 선형 회귀 기반 구조로 해석이 쉬움
- 결측치·이상치에 강건(Robust)
- 공휴일, 정책 변경 등 도메인 이벤트 반영 용이
- Python / R 모두 지원

TBATS

- Trigonometric + Box–Cox + ARMA + Trend + Seasonal
- 다중 계절성(Multiple Seasonality) 처리 가능
    - 예: 주기 7일(주간) + 365일(연간) 동시 반영

⚠ 계산량 많고, 대규모 데이터에서는 학습 느림

딥러닝 기반 시계열 모델

LSTM (Long Short-Term Memory)

- 순환신경망(RNN)의 확장형
- 장기 의존성(Long-term dependency) 학습
- 비선형·복잡한 패턴 학습 가능 (주가, 센서, 로그 등)다변량 시계열 + 외부 요인 입력 가능

⚠ GPU 자원 요구 많고, 해석 어려움

TCN (Temporal Convolutional Network)

- 1D CNN 기반 시계열 모델
- 시점 순서를 유지한 채 병렬 학습 가능
- RNN보다 빠르고 안정적

⚠ 짧거나 불규칙한 시계열에는 성능 저하

Transformer 기반 모델

Transformer 기반 모델

- Attention 메커니즘 으로 시점 간 중요도 동적 학습
- 대표 모델: Informer, TFT, FEDformer
- 다변량 시계열 + 외부 변수 입력 가능
- 고정 주기 한계를 넘어 복잡 패턴 학습

⚠ 대규모 데이터와 GPU 필요, 해석 어려움

시계열 예측의 최신 트렌드

- 딥러닝 + Attention 기반 고정 주기 한계를 극복

수고하셨습니다 ..^^..