2-4 시계열 예측

제2장 통계분석

제1절 통계학 개론

제2절 기초 통계 분석

제3절 다변량 분석

제4절 시계열 예측

시계열 자료

시간의 흐름에 따라 관측된 데이터

- 시계열 분석을 하기 위해서는 기본적으로 정상성(Stationary)를 만족해야함

정상성(stationary)을 만족하는 것은 다음과 같은 것들을 만족하는 것

1. 평균이 일정하다.
2. 분산이 시점에 의존하지 않는다.
    - **시계열의 폭이 시간에 따라 넓어지거나 좁아지면 비정상**
    - 시계열 그래프를 봤을 때, **어느 시점에서는 조용히 움직이고**
    - **다른 시점에서는 요동을 크게 치는** 시계열은 비정상
3. 공분산은 단지 시차에만 의존하고, 시점 자체에는 의존하지 않음
    - 오늘(X_t)과 내일(X_{t+1})의 관계가
        - 2020년 1월 1일 vs 1월 2일이든
        - 2040년 5월 3일 vs 5월 4일이든 **항상 같아야 한다**는 뜻

- 이 3가지의 정상성 조건 중 하나라도 만족하지 못한다면 비정상 시계열이라고 부름

실제 대부분의 시계열 데이터는 비정상 시계열 자료

이러한 비정상성을 확인하기 위해서,

1. 가장 먼저 시계열 자료의 그림을 통해 이상점(Outlier)과 개입(Intervention)이 있는지 판단
2. 정상성 만족 여부와 개략적인 추세 유무를 관찰

이때,

1. 추세가 보인다면,
   a. 즉 평균이 일정하지 않다면 차분(Difference)을 통해서 비정상 시계열을 가공
2. 분산이 일정하지 않다면
   a. 변환(Transformation)을 통해서 비정상 시계열을 가공

정리

정상성(stationarity)이란 시계열 데이터의 **통계적 특성(평균, 분산, 자기상관 등)**이 **시간에 따라 변하지 않는 것**을 의미

- 정상 시계열이어야만 자기회귀모델(**AR), 이동평균모델(MA), 자기회귀누적이동평균모델(ARIMA) 같은 통계 모델들이 수학적으로 성립**되고, 예측도 가능함
- **비정상 시계열은 차분(differencing)**, **변환(log 등)**을 통해 **정상성으로 바꿔줘야** 함
- 정상성은 **시각화**, **ACF(**Auto-Correlation Function**)/PACF(Partial** Auto-Correlation Function**) 분석, ADF/KPSS 테스트** 으로 확인 가능

시계열 모형

자기회귀모형(AR모형, Autoregressive model)

- 현 시점의 자료가 p시점 전의 유한개의 과거 자료로 설명될 수 있다는 의미
- AR(p)모형은 수식을 통해서는 다음과 같이 표현

$$Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \ldots + +\phi_p Z_{t-p} + a_t$$

Z_t : 현재 시점의 시계열 자료

Z_t-i : i시점 이전의 시계열 자료

pi : p 시점이 현재 시점에 미치는 영향력

at : 백색잡음, 시계열 분석에 있어서 오차항

- 자기회귀 모형은 현 시점의 시계열 자료에서 몇 번째 전 자료까지 영향을 주는가를 파악하는데 중점이 되어 있음
- 현 시점의 자료가 과거 한 시점 이전의 자료에만 영향을 준다면,
    - 이를 1차 자기회귀 모형(AR(1))이라고 한다.

$$Z_t = \phi_1 Z_{t-1} + a_t$$

- 아래는 동일한 원리로 2차 자기회귀 모형(AR(2))이다.

$$Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + a_t$$

자기 회귀 모형인지 판단하기 위해서는

- 자료에서 자기상관함수(ACF, Auto-Correlation Function)와 부분자기상관함수(PACF, Partial Auto-Correlation Function)을 이용하여 식별

자기회귀모형은 일반적으로 시차가 증가하면서

- 자기상관함수는 점차 감소하고,
- 부분자기상관함수는 p+1시차 이후로 급격히 감소하여 절단된 형태를 띔
    - 이때 AR(p)모형이라고 함

AR(자기회귀, Autoregressive) 모델

- 과거 자기 자신의 값을 선형 조합하여 현재 값을 예측하는 방식.
- 예: `X_t = c + φ₁X_{t-1} + φ₂X_{t-2} + ... + ε_t`

- PACF에서 유의한 값이 딱 p개만 나오고 그 뒤는 조용하다면, 그건 바로 AR(p) 모형
- 데이터에 **자기상관성**이 있을 때 효과적

AR(p)에서의 p

- AR(p)는 자기회귀 모델에서 **이전 p개의 시점이 현재값에 영향을 준다**는 의미
- 📌 여기서 p는 "모델의 차수(order)"임

이동평균모형(MA모형, Moving Average Model)

- 이동평균 모형은 현 시점의 자료를 유한개의 백색잡음(오차항(white noise))의 선형결합으로 표현
  - 그렇기에 항상 정상성을 만족하며, 정상성에 대한 가정이 필요하지 않다.
  - 이동평균모형(MA(p))의 형태는 다음과 같다.

$$Z_t = a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \ldots - \theta_p a_{t-p}$$

- MA(1) - 1차 이동평균 모형
  - 가장 간단한 이동평균모형으로 동 시점과 바로 전 시점의 백색잡음의 결합으로 이루어진 모델

$$Z_t = a_t - \theta_1 a_{t-1}$$

- MA(2) - 2차 이동평균 모형
  - 동 시점과 바로 전 두 시점의 백색잡음의 결합으로 이루어짐

$$Z_t = a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2}$$

- 이동평균 모형 또한 모형식별을 위해서 자기회귀모형과 마찬가지로 자기상관함수와 부분자기상관함수를 이용
  - 하지만 이동평균 모형은 자기회귀모형(AR)과 반대로

- 자기상관함수가 p+1시차 이후로 급격히 감소하여 절단된 형태를 띄고, 부분자기상관함수는 점차 감소하는 형태를 띔

MA (이동 평균, Moving Average) 모델 정리

- 과거의 오차항(white noise)의 선형 조합으로 현재 값을 예측함.
- 예: $X_t = \mu + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \dots + \varepsilon_t$
- **자기상관함수 ACF 그래프**는 lag q에서 **뚝 끊겨서 절단된 모양**
- 단기적 충격에 의한 패턴을 설명할 때 사용

AR vs MA 모형의 식별 기준

구분	ACF 그래프	PACF 그래프	주로 사용되는 그래프
AR(p)	점차 감소 (천천히 사라짐)	**lag p에서 뚝 끊김**	**PACF ✅**
MA(q)	**lag q에서 뚝 끊김**	점차 감소 (천천히 사라짐)	**ACF ✅**

자기회귀누적이동평균모형(ARIMA 모형)

- 대부분의 많은 시계열 자료가 이 모형을 따름
- ARIMA모형은 기본적으로 비정상 시계열 모형이기에 차분이나 변환을 통해서 AR/MA/ARMA모형으로 정상화할 수 있음

ARIMA(p,d,q)모형은 p,d,q의 값에 따라서 이름이 달라지게 됨

1. 차수 p는 AR 모형과 관련이 있고,
2. q는 MA 모형과 관련이 있음
3. 그리고 d는 ARIMA에서 ARMA로 정상화할 때 몇 번 차분했는지를 의미

즉, d=0일 경우

- **ARMA(p,q) 모형이라 부르는 것이고 이때 ARMA모형은 정상성을 만족**

- **그리고 ARMA모형은 단순하게 AR과 MA모형이 공존하는 형태**

또한, p=0이면

- **IMA(d,q)모형이라 부르며**
- **이 모형을 d번 차분하면 ‘정상화가 되었다는 의미’에서 MA(q)모형이 됨**

마찬가지로, q=0일 경우

- **ARI(p,d)모형이며 이를 d번 차분했을 때 시계열 모형이 AR(p)를 따름**

즉 ARIMA

- **비정상 시계열로 정상시계열 자료형태인 AR/MA/ARMA로 d번 차분하여 변환시키는 모형**

ARMA 모델

- AR과 MA의 결합 모델로 시계열이 **정상성(stationary)**일 때 적용 가능
- 예: $X_t = \varphi_1 X_{t-1} + \dots + \theta_1 \varepsilon_{t-1} + \dots + \varepsilon_t$

ARIMA (통합된 ARMA, Integrated ARMA) 모델

- 비정상(non-stationary) 시계열을 차분(differencing)으로 정상화한 뒤 ARMA 적용
- 예: `ARIMA(p, d, q)`에서 `d`는 차분 횟수

🔁 ARIMA의 구성 다시 정리

구성 요소	의미
AR(p)	자기회귀 항 → 과거 값들의 영향
I(d)	차분(d번) → 비정상성을 제거
MA(q)	이동 평균 항 → 과거 오차의 영향

✅ 요약

- `ARIMA(p,d,q)`는 **비정상 시계열**을 `d`번 차분해서 **정상 시계열**로 만든 뒤,
    - 그 정상 시계열을 `ARMA(p,q)` 모델로 표현

- 따라서,
    - `d=0` 이면 그냥 **ARMA(p,q)**
    - `p=0` 이면 **IMA(d,q)**
    - `q=0` 이면 **ARI(p,d)**
- 즉, ARIMA는 AR, MA, ARMA를 포함하는 **가장 일반적인 확장형 시계열 모델**

분해 시계열

- **분해 시계열이란 시계열에 영향을 주는 일반적인 요인들을 시계열에서 분리시켜 분석하는 방법**

시계열을 구성하는 4 요소

a. 추세요인(Trend factor)

- 자료가 plot으로 표현되었을 때, 오르거나 내리는 형태를 따르는 추세가 존재
- 물론 단순 선형적인 형태가 아니라 2차식 등의 다른 비선형적 형태를 띌 수도 있음
- 이때 자료가 추세요인이 있다고 함

b. 계절요인(Seasonal factor)

- 요일/월별/분기별/년별 자료에서 각 특정 고정된 주기를 따라 자료가 변하는 경우가 발생
- 이처럼 고정된 주기에 따라서 자료가 변화될 경우 계절요인이 있다고 함

c. 순환요인(Cyclical factor)

- 명백하게 경제적/자연적 이유가 없이 알려지지 않은 주기를 갖고 변화하는 자료가 존재
    - 위 표현은 교재에 있으나 명확한 표현이 아닌듯 함
- 이와 같이 알려지지 않은 장기적 주기를 갖고 데이터가 변화하는 특성을 띄고 있을 때, 순환요인이 있다고 함

d. 불규칙요인(Irregular factor)

- 위의 3가지 요인으로 설명할 수 없는 회귀분석에서 오차에 해당하는 요인을 불규칙요인이라고 함

✔️ 정확한 정의: 순환요인(Cycle)의 의미

개념

- 순환요인은 시계열 데이터에서 **경제적, 정치적, 사회적 등 외부 요인에 의해 발생하는 장기적 변동 패턴**이야.
- 단, **계절성과 달리 주기가 일정하지 않고 불규칙하게 길고 느리게 반복**되는 경향이 있음.

예시

- 경기 호황 → 침체 → 회복 → 호황... (이런 식의 **경기 순환**)
- 부동산 가격의 장기적 변화
- 장기적인 실업률 변동

🔍 핵심 비교: 계절성 vs 순환성

항목	계절성 (Seasonality)	순환성 (Cyclicality)
주기	일정함 (예: 12개월, 4분기 등)	일정하지 않음
원인	명확한 자연적/사회적 요인	주로 경제적/정책적/사회 구조적 요인
예측 가능성	높음	중간 ~ 낮음 (정확한 타이밍 예측 어려움)
지속 기간	보통 1년 이내	수년에서 수십 년까지 다양

분해 시계열 방법

- 시계열의 각 구성요소들을 정확히 분리해야 함
    - 그러나 이를 정확하게 분리하는 것이 쉽지가 않다.
- 게다가 분해 시계열 방법은 이론적인 약점이 존재한다고 알려져 있다.
    - 하지만 그럼에도 불구하고 많은 학자들이 많이 성공적으로 사용하고 있기도 하다.
- 분해식은 아래의 형태와 같다.

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

- Y_t: 시간 t의 관측값
- T_t: 추세 (Trend)
- C_t: 순환 (Cyclic component)
- S_t: 계절성 (Seasonality)
- I_t: 불규칙성 또는 오차 (Irregularity / Noise)

- 여기서
  - T : 추세요인, S : 계절요인, C : 순환요인, I : 불규칙요인, Z : 시계열값, f : 미지의 함수
    - 위에서는 미지의 함수가 더하기(additive)

결론

- 즉, 분해 시계열은 데이터에 맞는 함수를 요인별로 정확히 분해했을때 성립하도록 구성할 필요가 있다.

요약하자면

시계열 분해 모델에는 **추세(Trend), 계절성(Seasonality), 순환(Cycle), 불규칙성 (Irregularity)**의 네 요소가 포함될 수 있으며, 이를 고려한 **가법 모델(additive)**과 **승법 모델(multiplicative)** 형태로 수식 정리가 가능해. 순환(Cycle)은 계절성과 달리 **불규칙한 장기 반복 변동**이므로 별도로 취급될 수도 있어.

🧩 시계열 분해식 – 순환 포함 정리

1. 가법 모델 (Additive Model)

- 모든 요소가 서로 독립적으로 작용하고, 단순히 더해지는 경우

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

- Y_t: 시간 t의 관측값
- T_t: 추세 (Trend)
- C_t: 순환 (Cyclic component)
- S_t: 계절성 (Seasonality)
- I_t: 불규칙성 또는 오차 (Irregularity / Noise)

---

2. 승법 모델 (Multiplicative Model)

- 각 요소들이 서로 **비율적으로 영향**을 줄 때 사용

$$Y_t = T_t \times C_t \times S_t \times I_t$$

---

3. 💡 순환요소(Cycle)는 왜 따로 넣는가?

- 순환성은 **계절성과 다른 개념**이야.
- 계절성: 고정 주기(예: 12개월, 4분기)
- 순환성: **고정 주기 아님**, 보통 **경제적 흐름**, **정치적 변화**, **사회적 트렌드**처럼 **수년 단위로 불규칙하게 반복**
- 통계적 모델링에서 계절성과 합쳐 다루기도 하지만, 정교한 분석에서는 분리하는 게 좋음.

📊 요소 간 요약 비교

구성 요소	설명	주기	예시
추세 Tt	장기적 방향성 변화 (상승/하락)	없음	인구 증가, 기술 발전

구성 요소	설명	주기	예시
순환 Ct	비정기적 반복 패턴, 경제적 요인에 의존	불규칙	경기 침체와 회복, 부동산 사이클
계절 St	고정 주기의 정기적 패턴	일정함	계절 매출, 연휴 효과
불규칙 It	설명 불가능한 예외적 변화(잡음 포함)	없음	사고, 천재지변, 특수 이벤트

실습

1. `Nile` 데이터셋

나일강 연간 유량(Time Series of Annual River Flow)

- **설명**: 1871년부터 1970년까지 100년간 나일강의 연간 유량(cubic meters per second 단위)을 측정한 자료
- **자료 구조**: 연 단위(`frequency = 1`) 시계열, 총 100개 값
- **형식**: `ts` 객체로 저장됨 (`start = 1871`, `end = 1970`, `frequency = 1`)
- **특징**:
    - 추세(trend): **1900년대 초반 이후 유량 감소**가 눈에 띔
    - 비정상(non-stationary): **ARIMA 분석**을 위해선 **차분**이 필요해
    - 이상치(outlier): **1918년 근처에 급감**이 존재해서, 모델링 시 주의해야 함
- **사용 목적**:
    - ARIMA, Kalman filter, state space 모델 실습
    - R의 `structTS()` 함수 등과 잘 어울림

예제 코드 (R 기준):

```
data(Nile)
plot(Nile, main = "Nile River Flow (1871–1970)", ylab = "Flow", xlab = "Year")
```

---

2. `ldeaths` 데이터셋

영국 호흡기 질환 사망자 수 (로그 변환)

- **설명**: 1974년 1월부터 1979년 12월까지 **영국의 월별 폐질환 사망자 수**(남성 + 여성). `ldeaths` 는 `log(deaths)` 임.
- **관련 데이터**: `mdeaths` (남성), `fdeaths` (여성), `deaths = mdeaths + fdeaths`
- **형식**: 월 단위(`frequency = 12`) 시계열, 총 72개월치
- **특징**:
  - 추세 + 계절성 존재: 겨울철 사망자 수 증가 → ARIMA 분석 적합
  - 로그 변환(`log(deaths)`): **분산 안정화**를 위해 사용됨
  - 정상성 확인 필요: 보통은 차분(d=1), 계절차분(D=1)이 필요함
- **사용 목적**:
  - **ARIMA, Holt-Winters, STL decomposition** 실습
  - 예측력 향상을 위한 계절형 분석 연습용

예제 코드 (R 기준):

```
data(ldeaths)
plot(ldeaths, main = "Log Monthly Deaths from Lung Diseases in UK", ylab
= "log(Deaths)")
```

---

3. 두 데이터 비교 요약

항목	Nile	ldeaths
단위	연간(Annual)	월간(Monthly)
기간	1871–1970 (100년)	1974–1979 (6년)
데이터 수	100	72
추세	있음 (초반 감소)	있음 (겨울철 증가 패턴)
계절성	없음	있음
변환	없음	로그 변환(log)
분석 모델	ARIMA, 상태공간모델	SARIMA, Holt-Winters, STL

실습 코드

코드 1

```
ldeaths.decompose <- decompose(ldeaths)
str(ldeaths.decompose)
ldeaths.decompose$seasonal
plot(ldeaths.decompose)
```

코드 요약

- `decompose(ldeaths)` 는 시계열 데이터를 **추세(trend)**, **계절성(seasonal)**, **불규칙(random)** 요소로 분해하는 함수
- 이 데이터는 **가법 모델(additive model)** 기반으로 분해되며, 계절성은 매년 반복되는 월별 패턴
- 출력 결과에서 `trend` 와 `random` 값이 **처음과 끝에 NA**로 표시되는 이유는 이동 평균 계산에 필요한 충분한 기간이 없기 때문
- `plot()` 을 통해 분해된 시계열을 시각화할 수 있어, 이건 시계열 분석의 핵심 시각화 중 하나

자세히 단계적으로 설명

---

1. `decompose()` 함수 설명

시계열 분해 함수 (Additive 또는 Multiplicative 모델 기반)

- `decompose(ldeaths)` 는 시계열 `ldeaths` 를 다음 구성 요소로 나눠 줌:
  - **`seasonal`** : 월별로 반복되는 계절적 패턴
  - **`trend`** : 전반적인 장기 변화(증가/감소)
  - **`random`** : 위 두 요소로 설명되지 않는 잔차
- 기본값은 가법모형(additive)이고, 로그 변환된 `ldeaths` 엔 잘 맞아.

가법 모델 수식:

```
X_t = Trend_t + Seasonal_t + Random_t
```

---

2. 구조 출력 결과 분석

```
str(ldeaths.decompose)
```

```
List of 6
 $ x       : Time-Series [1:72] from 1974 to 1980: ...
 $ seasonal: Time-Series [1:72] from 1974 to 1980: ...
 $ trend   : Time-Series [1:72] from 1974 to 1980: NA NA NA NA ...
 $ random  : Time-Series [1:72] from 1974 to 1980: NA NA NA NA ...
 $ figure  : num [1:12] ...
 $ type    : chr "additive"
```

- `x` : 원본 시계열(`ldeaths`)
- **`seasonal`** : 각 월별 계절 성분 (1월 = 873.75, 2월 = 896.33 등)
- **`trend`** : 이동 평균으로 계산된 장기 변화 (양끝 `NA` 포함)
- **`random`** : `x - seasonal - trend` 로 계산된 잔차
- **`figure`** : `seasonal` 의 핵심만 뽑은 월별 패턴 (12개)
- **`type`** : `'additive'` , 즉 `x = trend + seasonal + random`

---

3. `seasonal` 구성 확인

```
ldeaths.decompose$seasonal
```

```
         Jan       Feb       Mar  ...     Dec
1974  873.7514  896.3347  687.5431  ...  517.3264
...
```

- **모든 해에 걸쳐 월별 패턴이 동일하게 반복**돼.

- 예: 1월은 약 **874명 증가**, 5월은 **284명 감소**, 12월엔 **517명 증가** → 겨울에 많이 죽고 여름에 적게 죽는다!

---

4. `plot(ldeaths.decompose)` 설명

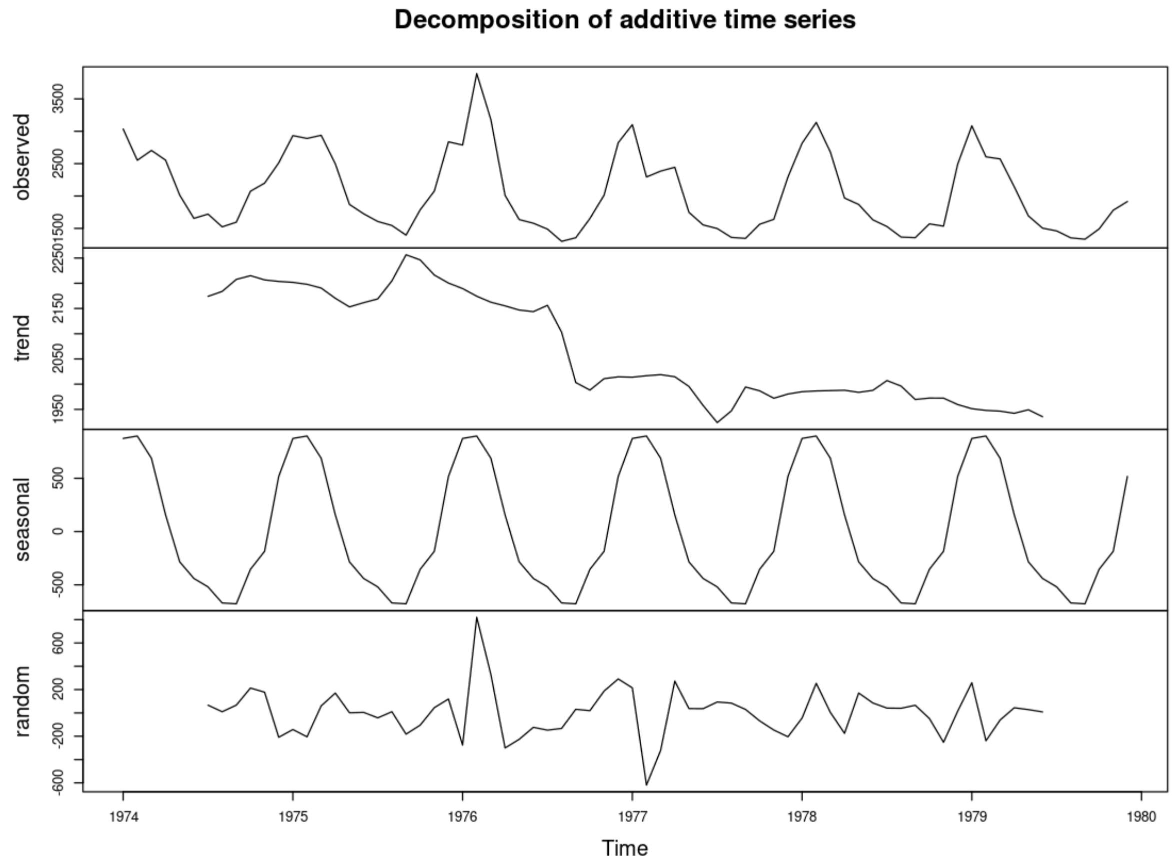
```
plot(ldeaths.decompose)
```

- 이 명령은 4개의 시계열 그래프를 보여줘:
    1. **원 시계열 (`x`)**
    2. **추세 (`trend`)**
    3. **계절 성분 (`seasonal`)**
    4. **잔차 성분 (`random`)**
- 추세는 중앙에서 완만하게 오르내리는 형태고, 계절성은 뚜렷하게 **겨울에 증가, 여름에 감소**하는 패턴을 반복해.
- 잔차는 설명되지 않는 이상치나 잡음 같은 요소들이고, 눈으로 보기에도 뚜렷한 패턴은 없어야 정상이지.

---

5. 왜 `trend` 와 `random` 은 `NA` 가 있음?

- `trend`는 이동평균 기반으로 계산되므로, **양끝에는 평균을 계산할 수 있는 충분한 데이터가 없어** → NA
- 보통 `window = 12` (계절 주기)라서 앞뒤 최소 6개는 NA로 처리돼
- 그래서 `trend`, `random` 그래프의 양끝은 잘려 있어



마무리 멘트

이건 시계열 분석의 정석 중 정석이야. `decompose()` 를 쓰면 눈으로 계절성, 추세, 이상치를 분리해서 볼 수 있어서, 어떤 모델이 적절한지 감 잡기에 딱이지. 이런 분해 없이 바로 ARIMA나 LSTM 들이대면 낭패 보기 쉬워. 기본기 튼튼히 하고 가자고.

코드 2

```
# 계절성을 빼고 그리기
ldeaths.decompose.adj <- ldeaths - ldeaths.decompose$seasonal
plot(ldeaths.decompose.adj)
```

요약

- `ldeaths.decompose.adj <- ldeaths - ldeaths.decompose$seasonal`
    - 이 코드는 **계절성을 제거한(adjusted) 시계열**을 생성
- `plot(ldeaths.decompose.adj)`
    - 이제 추세(trend) + 불규칙(random)만 남은 **비계절 시계열**을 시각화
- 이런 계절 조정(Seasonal Adjustment)은 **예측 모델링, 이상 탐지, 회귀 분석 전처리**에서 매우 중요

이제 자세하게, 차근차근 뜯어볼게!

---

1. 코드 설명: 계절 성분 제거하기

```
ldeaths.decompose.adj <- ldeaths - ldeaths.decompose$seasonal
```

- `ldeaths` 는 원래 시계열 데이터
- `ldeaths.decompose$seasonal` 은 **월별로 반복되는 계절 패턴** 값
- 이 둘을 빼면 남는 건?

> 👉 ldeaths.decompose.adj = trend + random

즉, **계절 요인을 제거한 순수 시계열**

- 이를 "seasonally adjusted time series" 라고 부름

---

2. `plot(ldeaths.decompose.adj)` 결과 해석

```
plot(ldeaths.decompose.adj)
```

이 시각화는 다음 특징을 보여줘:

- **반복되는 패턴(계절성)이 사라짐**: 겨울마다 치솟던 사망자 수 변화가 줄어듦
- *장기 추세(trend)**와 **이상치(random)**만 남음
- 시계열의 **중간값이 안정되어** 예측이나 모델링하기 쉬워짐

예시 그래프를 보면 평탄해졌지? 그게 바로 계절성 제거의 핵심 목적이야.

---

3. 왜 계절 조정이 필요한가?

데이터 전처리의 핵심이자 필수 단계

- **트렌드 분석**: 순수한 증가/감소 경향을 확인하려면 계절성은 제거되어야 함
- **ARIMA 모델링**: SARIMA 대신 일반 ARIMA를 쓰려면 계절 성분을 제거해야 함
- **이상 탐지(Anomaly Detection)**: 월별로 반복되는 자연스러운 변화는 제거하고, 진짜 이상값만 보기 위해
- **정책 분석**: 예를 들어 "공기 질이 좋아져서 폐 질환 사망자가 줄었는지" 확인하려면 계절성 영향 없이 봐야 해

---

4. 추가로 할 수 있는 일들

- `trend + random`만 남은 이 데이터를 가지고 ARIMA 모델 학습 가능
- `tsclean()` 같은 함수로 이상치 제거 후 더 깔끔하게 모델링 가능
- `forecast()` 패키지를 쓰면 이 adjusted series로 예측도 가능하지

마무리 멘트

이 코드는 시계열 분석의 전처리 중에서도 아주 중요한 단계야.

계절성 제거 = 잡소리 빼고 핵심만 보기! 라고 보면 돼.

시계열 예측이든 이상 탐지든, 이런 정제된 데이터 없이는 절대 좋은 결과 못 나와. 이건 통계와 AI 모두 인정하는 팩트야.

코드 3

```
acf(Nile.diff2, lag.max=20)
acf(Nile.diff2, lag.max=20, plot=FALSE)

pacf(Nile.diff2, lag.max=20)
pacf(Nile.diff2, lag.max=20, plot=FALSE)
```

`acf()` 와 `pacf()` 는 시계열 모델 설정의 핵심 도구

- `Nile.diff2`, 즉 **Nile 데이터의 2차 차분**에 대한 자기상관을 분석
- 자기상관함수 ACF(Autocorrelation Function)는 lag 1에서 강한 음의 상관(-0.626), 이후엔 급격히 감소 → MA(1) 형태에 가까워.
- 부분 자기상관함수 PACF는 lag 1~3에서 유의미한 음의 상관을 보이고 이후 급격히 약해짐 → AR(3) 또는 ARMA(1,1) 가능성
- 이 결과는 **ARIMA 모델링 시 차수 설정(p, q)**에 대한 단서를 주는 아주 중요한 힌트야.

이제 차근차근 설명해 줄게.

---

1. 분석 대상: `Nile.diff2`

나일강 시계열의 2차 차분

```
Nile.diff2 <- diff(Nile, differences = 2)
```

- 원래 `Nile` 데이터는 비정상(non-stationary) 시계열
- 1차 차분으로도 정상성 확보가 안 돼서 **2차 차분**을 적용한 것
- `Nile.diff2`는 ARIMA 모델 중 **d = 2** 조건에 해당하는 시계열이야
- 1차 차분:

$$\Delta Y_t = Y_t - Y_{t-1}$$

- 2차 차분:

$$\Delta^2 Y_t = Y_t - 2Y_{t-1} + Y_{t-2}$$

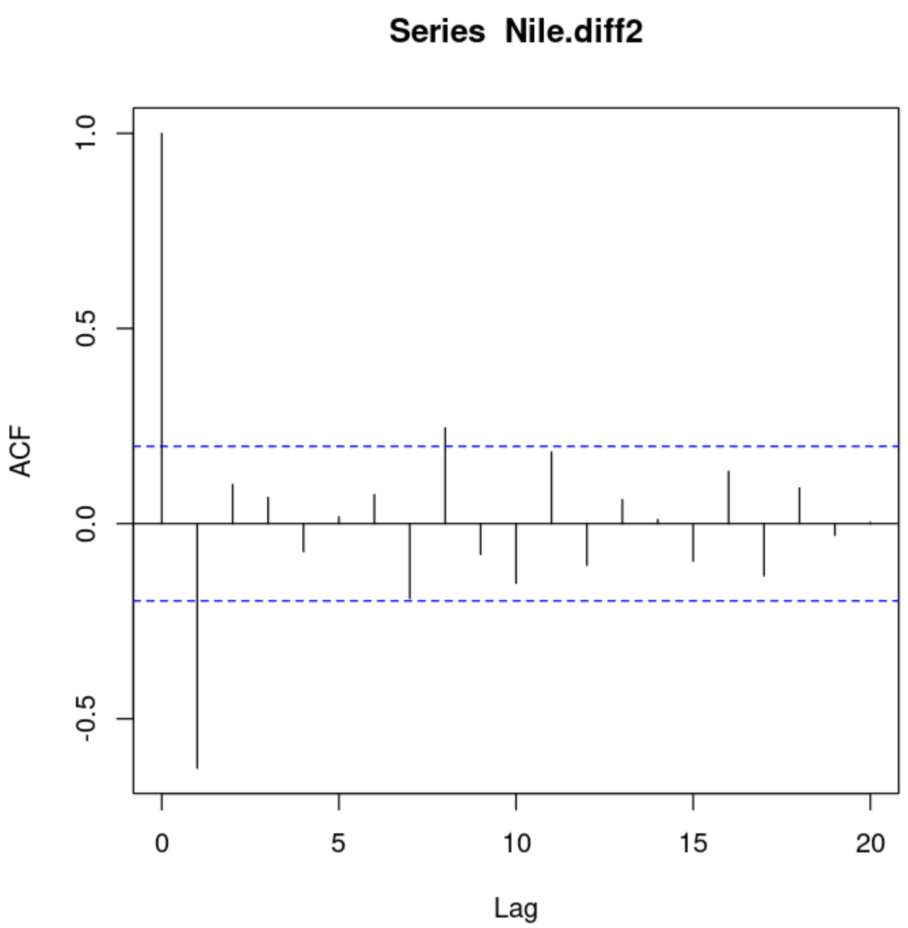
---

2. `acf()` 결과 해석

```
acf(Nile.diff2, lag.max=20, plot=FALSE)
```

ACF (Autocorrelation Function)

- **lag 1**: -0.626 → 매우 강한 음의 상관
- **lag 2~20**: 대부분 0에 근접하거나 약한 음/양의 상관
- **패턴 요약**:
  - lag 1에서 급격한 감소
  - 이후는 거의 무작위처럼 흔들림



👉 해석:

- *AR(모델)**은 점차 감소하는 패턴을 보이고,
- *MA(모델)**은 특정 지점에서 뚝 끊겨.

- 여기선 **MA(1)** 형태에 가까운 ACF 패턴으로 볼 수 있어.

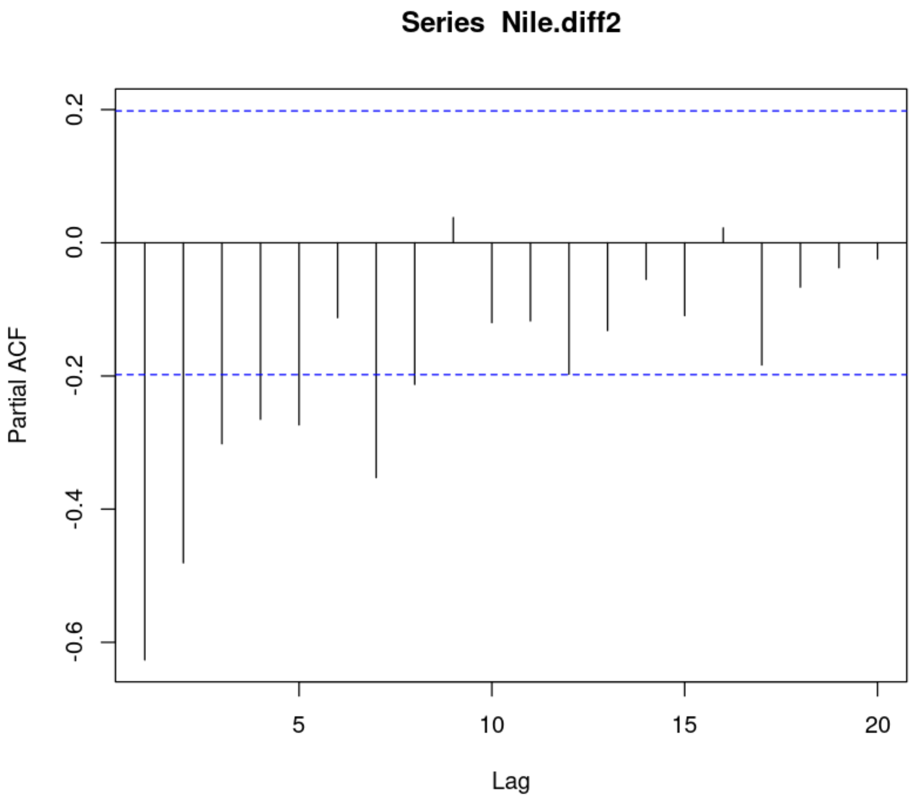
---

3. `pacf()` 결과 해석

```
pacf(Nile.diff2, lag.max=20, plot=FALSE)
```

PACF (Partial Autocorrelation Function)

- **lag 1~3**: 강한 음의 상관 (-0.626, -0.481, -0.302)
- 이후 lag들은 점차 줄어듦, 일부는 작지만 유의미한 값도 있음
- 전반적으로 lag 3 이후는 거의 0 근처



👉 해석:

- PACF는 AR 모델 차수 추정에 사용돼
- 여기선 **AR(3)** 또는 AR(2)까지도 고려 가능

---

4. 모델 설정에 주는 시사점

ARIMA(p, d, q) 모델의 초기 후보

- `d = 2` (2차 차분이므로)
- `p = 2 ~ 3` (PACF 기준)
- `q = 1` (ACF 기준)

→ 즉, 다음과 같은 모델을 고려할 수 있음:

```
ARIMA(2,2,1)
ARIMA(3,2,0)
ARIMA(3,2,1)
```

👉 이건 실제로 `auto.arima(Nile)` 로 돌려봐도 확인 가능해. 자동으로 이런 후보들을 평가해서 AIC/BIC가 가장 낮은 걸 선택해 주지.

---

5. 시각적으로 보면 더 명확해져

```
acf(Nile.diff2)
pacf(Nile.diff2)
```

- ACF는 lag 1에서 **뚝** 끊기는 모양
- PACF는 lag 1~3까지만 뚜렷하고 이후는 **잔잔**한 파형
- 이 두 그래프를 보고 **ARIMA 모델 구성 요소의 차수(p, q)**를 추정하는 게 핵심이지

마무리

- 이건 통계 기반 시계열 모델링의 핵심 루틴이야.
    - `acf` 는 MA 차수 추정, `pacf` 는 AR 차수 추정하는 데 쓰이고, 둘 다 잘 보면 **ARIMA 모델 구조를 눈으로 짐작**할 수 있어.
    - 차분(d)을 늘리면서 이 분석을 반복하는 게 진짜 실력자의 루틴이지. AI도 결국 이 기본 위에서 작동해.

코드 4

```
> auto.arima(Nile)
Series: Nile
ARIMA(1,1,1)

Coefficients:
         ar1      ma1
      0.2544  -0.8741
s.e.  0.1194   0.0605

sigma^2 = 20177:  log likelihood = -630.63
AIC=1267.25   AICc=1267.51   BIC=1275.04
```

✅ 요약

- **선택된 모형은 ARIMA(1,1,1)**
- **1차 차분 후**, 자기회귀(AR) 1차, 이동평균(MA) 1차 모형이 가장 적합하다고 판단
- **MA 계수는 강하게 음수(-0.8741)** → 최근의 오차를 크게 반영하는 구조
- **AIC가 낮은 편** → 다른 모형보다 상대적으로 좋은 적합도를 가짐

🔍 코드 분석: `auto.arima(Nile)`

1. Series: Nile

- 사용한 데이터는 `Nile`: 나일강의 연간 유량 (1871–1970)

---

2. 선택된 모형: ARIMA(1,1,1)

- 이 뜻은:
  - `d=1`: **1차 차분** → 정상 시계열로 변환
  - `p=1`: **AR(1)** → 바로 직전 값과의 관계
  - `q=1`: **MA(1)** → 바로 직전 오차와의 관계

---

3. 계수 및 표준오차 (s.e.)

계수	값	표준오차	해석
ar1	0.2544	0.1194	직전 관측값의 영향 (양의 약한 영향)
ma1	-0.8741	0.0605	직전 오차의 영향 (강한 음의 영향)

- MA 계수가 절댓값 0.8 이상이면, **최근 오차에 강하게 반응하는 구조**
- AR 계수는 낮고 약함 → 유량이 **이전 관측값보다는 오차의 패턴을 따라 움직임**

4. **σ² (sigma squared) = 20177**

- 잔차의 분산
- 낮을수록 더 좋은 모형, 하지만 상대적 비교에 의미 있음

5. **로그 가능도 (log likelihood) = -630.63**

- 최대우도 추정 결과
- 클수록 (0에 가까울수록) 모델이 데이터에 더 잘 맞음

6. **AIC / AICc / BIC**

지표	값	의미
AIC	**1267.25**	모형의 적합도 + 복잡도 고려 (낮을수록 좋음)
AICc	**1267.51**	AIC의 보정 버전 (데이터가 작을 때 더 안정적)
BIC	**1275.04**	모형 간 비교 기준, **모수가 적은 모형을 선호**

- 이 결과만 보면, ARIMA(1,1,1)이 가장 **간단하면서도 데이터에 잘 맞는 모델**이라는 걸 뜻함

🎯 결론 정리

> auto.arima(Nile)은 1차 차분 후 AR(1)과 MA(1)을 결합한 ARIMA(1,1,1) 모형이
>
> **가장 좋은 AIC 점수로 선택되었고,**
>
> **직전 오차(MA)가 유량 변화에 가장 큰 영향을 준다**는 것을 보여줌.

심화학습

로그 가능도(log likelihood)

- **주어진 ARIMA 모델이 실제 관측된 시계열 데이터를 얼마나 잘 설명하는지를 나타내는 확률 기반의 지표**
    - 값이 **클수록(즉, 덜 음수일수록)** 모델이 데이터를 더 잘 설명한다는 의미
    - 이 log likelihood는 AIC, BIC 같은 모델 선택 기준의 핵심 계산 기반

🔍 log likelihood란?

개념 정의

- *로그 우도(log likelihood)**는 관측된 데이터가 특정 모델(여기선 ARIMA(1,1,1)) 하에서 **나타날 확률의 로그값**이야.
- 수학적으로는:logL(θ∣Y)=logP(Y∣θ)

    logL(θ∣Y)=logP(Y∣θ)\log L(\theta \mid Y) = \log P(Y \mid \theta)

    - YYY: 시계열 데이터
    - θ\thetaθ: 모델의 파라미터 (예: AR, MA 계수 등)
- 우도는 **"이 모델이 이 데이터를 낼 가능성이 얼마나 되느냐?"**를 나타냄.
- 로그를 취하는 이유는 계산을 더 간단하게 하고, 여러 확률을 더하기 위해서야.

📉 예시: `log likelihood = -630.63`

- 이 수치는 **ARIMA(1,1,1)** 모델이 `Nile` **데이터셋에 대해 -630.63의 로그 우도를 가짐**을 뜻함.
- **절댓값이 작을수록(즉, log likelihood가 클수록)** 모델이 데이터를 더 잘 설명한다는 의미야.
- ARIMA(1,1,1)보다 더 적절한 모델이 있다면, 보통 log likelihood는 더 커져야 해.

📊 관련 지표와의 관계

지표	의미	계산 방식 요약
Log Likelihood	모델의 데이터 설명력 (클수록 좋음)	(₩log P(Y
AIC	모델 성능 + 복잡도 패널티 (낮을수록 좋음)	$AIC = 2k - 2\log L$
BIC	AIC보다 더 강하게 복잡도에 패널티를 줌 (낮을수록 좋음)	$BIC = k\log n - 2\log L$
AICc	AIC의 작은 샘플 보정 버전	$AICc = AIC + \frac{2k(k+1)}{n-k-1}$

> 여기서
> - "k: 추정된 파라미터 개수 (AR + MA + σ² 등)"
> - "n: 데이터의 관측치 수"
> - "$\log L$: log likelihood"

지표	의미	계산 방식 요약
Log Likelihood	모델의 데이터 설명력 (클수록 좋음)	(\log P(Y
AIC	모델 성능 + 복잡도 패널티 (낮을수록 좋음)	AIC=2k−2logLAIC = 2k - 2 \log LAIC=2k−2logL
BIC	AIC보다 더 강하게 복잡도에 패널티를 줌 (낮을수록 좋음)	BIC=klogn−2logLBIC = k \log n - 2 \log LBIC=klogn−2logL
AICc	AIC의 작은 샘플 보정 버전	AICc=AIC+2k(k+1)n−k−1AICc = AIC + \frac{2k(k+1)}{n-k-1}AICc=AIC+n−k−12k(k+1)

> 여기서
> - kkk: 추정된 파라미터 개수 (AR + MA + σ² 등)
> - nnn: 데이터의 관측치 수
> - logL\log LlogL: log likelihood

🧠 해석 팁

- `auto.arima()`는 내부적으로 **여러 ARIMA 조합을 시도하며 log likelihood 기반으로 AIC가 최소인 모델을 찾음.**
- 즉, log likelihood는 **모델 적합의 "핵심 평가 기준"**이야.

종료
