1-1 R 기초

📘 핵심 01 R 기초

R은 통계 계산과 시각화 등을 위해 개발된 오픈 소스 프로그램으로 1993년 뉴질랜드 오클랜드 대학교의 로스 이하카와 로버트 젠틀맨에 의해 개발되었다.

■ R의 특징

항목	설명
최적화된 통계 분석 도구	R은 다양한 통계 분석에 특화된 프로그래밍 언어
강력한 데이터 시각화	다양한 시각화 패키지를 제공하며, 이를 통해 데이터를 직관적으로 이해할 수 있게 도와줌
방대한 패키지 제공	CRAN(Comprehensive R Archive Network)을 통해 통계, 시각화, 데이터 마이닝, 금융 등 다양한 분야에 사용할 수 있는 패키지를 제공
여러 운영체제와 호환	Windows, MacOS, Linux 등 다양한 운영체제에서 R을 사용할 수 있음
객체지향 및 함수형 프로그래밍 지원	R은 객체지향 언어이며, 동시에 함수형 프로그래밍과 패러다임을 지원

📘 핵심 02 R 패키지

- 특정 기능 또는 작업을 수행하기 위해 미리 작성된 코드의 집합으로 함수, 객체, 데이터, 도움말 등이 포함되며 R에서는 CRAN이라는 서버에 패키지를 저장하고 사용자에게 제공한다.
- 패키지를 설치하려면 `install.packages("패키지명")` 명령문을 사용한다.
- 패키지를 활성화하려면 `library(패키지명)` 명령문을 사용한다.

3. 행렬(matrix)

- 같은 데이터 형태로 구성된 **2차원 데이터 구조**이다.
- **행(row)** 과 **열(column)** 로 구성된다.
- 행렬(matrix)의 생성 방법:

```
matrix(data, nrow, ncol, byrow)
```

매개변수	설명
`data`	행렬에 넣을 데이터
`nrow`	행렬의 행 수를 지정
`ncol`	행렬의 열 수를 지정
`byrow`	`TRUE`이면 데이터를 **행 우선 채움** (기본값은 `FALSE`, 즉 열 우선)

📌 관련 함수

- `cbind()` 함수 : 입력된 데이터를 **열 방향**으로 결합
- `rbind()` 함수 : 입력된 데이터를 **행 방향**으로 결합

---

4. 데이터 프레임(dataframe)

- 데이터베이스에서 테이블과 유사한 데이터 객체를 의미한다.
- 행렬과 유사한 **2차원 목록 구조**이지만, **각 열이 서로 다른 형태의 객체**를 가질 수 있다.
- 데이터 프레임의 생성 : `data.frame()`

---

5. 리스트(list)

- **서로 다른 유형의 데이터 구조를 결합한 것**이다.
- 리스트의 생성 : `list()`

📘 핵심 05. 결측값 처리

1. 결측값(missing value)

- 결측값: 데이터에 **값이 없는 항목**
  - 예) 통계조사에서 응답자가 문항에 응답하지 않은 경우
- 결측값이 포함된 데이터는 분석 시 문제 발생 가능
- 가능하면 제거하는 것이 좋으나, 결측 자체가 의미 있는 경우도 있음
  - 예) 특정 정보를 의도적으로 입력하지 않은 경우

- **R에서는 결측값을 `NA` (Not Available)로 표시**

---

(1) R에서 결측값 처리

① 결측값 확인

- `is.na( )` : 결측값이면 `TRUE`, 아니면 `FALSE`

② 결측값 처리 함수

- `na.rm = T` : 결측값 제거
- `na.omit( )` : NA 포함 행 전체 삭제
- `complete.cases( )` : 결측값이 없는 완전한 행 추출

---

(2) 결측값 처리 방법

① 완전 분석법 (Complete Case Analysis)

- 결측값이 있는 레코드 **전체 삭제**
- 결측이 많으면 데이터 손실과 통계적 문제 발생

② 평균 대치법 (Mean Imputation)

- 관측 또는 실험을 통해 얻은 데이터의 평균으로 대체
  - 비조건부 평균 대치: 전체 평균
  - 조건부 평균 대치: 회귀분석 등을 이용해 예측값으로 대체

③ 단순 확률 대치법 (Single Stochastic Imputation)

- 평균 대치법에 표준오차를 더해 **추정의 오차**를 보완하는 방식

④ 다중 대치법 (Multiple Imputation)

- 여러 차례 결측값을 대치해 여러 개의 완전한 데이터셋 생성
- **대치 → 분석 → 결합** 순으로 처리
- 오차 최소화, 분석 정확도 향상

📘 핵심 06. 이상값 처리

1. 이상값(Outlier)

- **정상적 분포 범위 바깥에 위치하는 값**
- 예) 입력 실수, 고의적 제외 대상 등

---

2. 이상값 처리 방법

방법	설명
제거(Deleting)	이상값을 제거 (→ 데이터 손실 가능성 존재)
대치(Imputation)	평균, 중앙값, 최빈값으로 대체 (극단값 완화)
변환(Transformation)	

- 오른쪽 꼬리 길면: 로그, 제곱근 변환
- 왼쪽 꼬리 길면: 지수, 제곱 변환
