R 정리

■ 주요 수학함수

절댓값	abs(-3)	3
제곱근	sqrt(16)	4
원주율	pi	3.14159
부호	sign(-3)	-1
반올림	round(2.345, digits=2)	2.35
올림	ceiling(2.3)	3
내림	floor(2.7)	2
지수	exp(10)	22026.47
자연로스	log(10)	2.302585
상용로그	log10(10)	1
로그(2)	log2(10)	3.321928
계승	factorial(4)	24
조합	choose(4,2)	6
곱	prod(1:4)	24

■ 데이터 유형 검정 함수, 변환 함수

is.numeric()	as.numeric()	수치형
is.integer()	as.integer()	정수형
is.double()	as.double()	실수형
is.character()	as.character()	문자형
is.logical()	as.logical()	논리형
is.complex()	as.complex()	복소수형
is.null()	x	NULL 여부
is.na()	x	NA 여부
is.finite()	x	유한 수치 여부
is.infinite()	x	무한 수치 여부

■ 벡터 생성 방법

① c()

- 규칙이 없는 데이터로 이루어진 벡터를 생성할 때
- 벡터들을 하나로 합쳐서 하나의 새로운 벡터를 생성할 때
- vec <- c(, , ,)

② :

- 콜론은 수치형에만 적용
- 1씩 증가되거나 1씩 감소되는 규칙이 있는 값으로 이루어진 벡터를 생성할 때
- vec <- start : end

③ seq()

- sequence 약자
- 증감이 되는 규칙 있는 수치형 벡터를 생성할 때
- vec <- seq(from = , to = , by =)

④ sequence()

- 1과 지정한 숫자 사이의 정수로 이루어진 수치형 벡터를 생성할 때
- 지정한 숫자가 0 이면 벡터를 초기화할 때 사용할 수 있음
- 지정한 숫자가 음수면 오류가 발생함
- vec <- sequence(숫자)

⑤ rep()

- replicate의 약자로 복사, 반복의 의미
- 수치형, 문자형, 논리형, 복수형으로 된 벡터를 생성할 때
- 지정된 데이터를 복사해주는 기능
- vec <- rep(스칼라 또는 벡터, times = 반복횟수, each = 원소 각각의 반복횟수, length.out = 벡터의 길이)

⑥ paste()

- 함수에 지정된 데이터를 합쳐서 하나의 문자형 데이터를 생성할 때
- 데이터가 합쳐질 때 구분자를 지정해줌 (데이터 사이 연결)
- 주로 변수명, 관찰치명을 작성할 때 유용하게 사용
- vec <- paste(여러 스칼라, sep = “구분자”)

■ 벡터 속성

속성	R 코드	반환값
데이터 유형	mode(vec)	벡터의 요소 유형
	is.numeric(vec)	TRUE / FALSE
	is.character(vec)	TRUE / FALSE
	is.logical(vec)	TRUE / FALSE
	is.complex(vec)	TRUE / FALSE
데이터 개수	length(vec)	벡터의 원소 개수
데이터 이름	names(vec)	벡터의 이름
	names(vec) <- c(원소 이름)	벡터의 원소에 이름 붙이기
데이터 추출	vec[index]	벡터 원소 뽑아내기 (인덱스 1부터 시작) 인덱스에 c(), :, seq(), sequence 등도 올 수 있음

■ 벡터 연산

① 벡터의 길이가 동일한 경우

- 벡터들간의 연산 (+, -, *, /, **). 결과는 벡터가 됨.
- 각 벡터에 있는 동일한 위치의 값들끼리 연산이 됨.

② 벡터의 길이가 동일하지 않은 경우

- 데이터의 개수가 적은 쪽의 벡터는 데이터 개수가 많은 쪽의 벡터와 동일하게 데이터 개수를 맞춤.
- 데이터 개수가 차이나는 만큼 임시적으로 데이터가 생성됨. recycling

■ 행렬 생성

① rbind()

- 2개 이상의 벡터를 행을 기준으로 합쳐서 하나의 행렬 생성
- 벡터의 개수 = 행의 개수, 벡터 중에서 데이터의 개수가 큰 벡터의 길이 = 열의 개수
- matrix <- rbind(vec, vec)

② cbind()

- 2개 이상의 벡터를 열을 기준으로 합쳐서 하나의 행렬 생성
- 벡터의 개수 = 열의 개수, 벡터 중에서 데이터의 개수가 큰 벡터의 길이 = 행의 개수
- matrix <- cbind(vec, vec)

③ matrix()

- data : 행렬의 원소에 채워질 데이터로 벡터 지정
- nrow = 행의 개수 지정
- ncol = 열의 개수 지정
- byrow = TRUE / FALSE(default)
- matrix(data, nrow = , ncol = , byrow =)

■ 행렬 속성

속성	R 코드	반환값
데이터 유형	mode(matrix)	행렬의 데이터 유형
데이터 개수	length(matrix)	행렬이 갖고있는 원소 개수
행 / 열의 개수	nrow(matrix) / ncol(matrix)	행 / 열의 개수
행 / 열의 이름	rownames(matrix) / colnames(matrix)	행/ 열의 이름
차원	dim(matrix)	행의 개수, 열의 개수
차원의 이름	dimnames(matrix)	행의 이름, 열의 이름
데이터 추출	matrix[row index, col index]	하나의 행 또는 열만 추출하면 벡터가 됨. 행렬의 형태를 유지하고 싶으면 drop = FALSE 입력

■ 행렬 연산

행렬 원소들간의 연산 (+, -, *, /)

행렬 곱하기 %*%

역행렬 solve(matrix)

전치행렬 t(matrix)

■ 데이터 프레임

- 데이터 프레임의 형태는 행렬과 같지만, 데이터 프레임은 열끼리 데이터 유형이 달라도 아무런 영향이 없음.
- 데이터 프레임에서 length() 함수는 열의 개수를 알려줌
- 데이터 프레임 생성방법 : data.frame() 함수에 벡터나 행렬을 넣음.

■ 요인

- 데이터를 질적 자료(범주형)로 변환해주는 기능
- 질적 자료로 변경되면 집단별로 통계분석 가능
- factor(x, levels, labels, ordered)

    x = 벡터 지정

    levels = 그룹으로 지정할 문자형 벡터 지정. default : 오름차순으로 구분하여 자체적으로 그룹 지정

    labels = levels에 대한 문자형 벡터 지정

    ordered = levels에 대해 특정한 순서를 정하고 싶으면 TRUE 지정

■ 리스트

- 통계분석의 결과를 저장할 때 사용
- list(x) 함수

    x = 스칼라, 벡터, 행렬, 배열, 데이터프레임, 요인, 리스트

- 리스트의 원소를 추출할 때 : [index] = 결과가 리스트 , [[index]] = 결과가 해당 원소의 데이터 형태

■ 텍스트 데이터 불러오기

- read.table(file, header, sep, stringsAsFactors, na.strings)

    file = R에서 읽어올 외부 데이터가 있는 파일의 위치(경로)와 파일명(확장자 포함)을 문자형으로 지정함

    header = TRUE - 외부 데이터에 있는 변수명을 R 데이터에서도 동일하게 사용

        FALSE - 외부 데이터의 변수명을 사용하지 않고 R에서 자체적으로 변수명 지정 (V1, V2)

    sep = 구분자로 문자형 형태로 지정

    stringsAsFactors = 문자형 데이터를 요인으로 자동으로 변경할지 설정. FALSE를 지정하면 문자형으로 읽어들임.

    na.strings = 데이터에 결측치가 있거나 결측치로 지정할 내용을 문자형으로 지정함

■ csv 데이터 불러오기

- csv 데이터는 엑셀 데이터에서 콤마로 구분된 형태의 데이터
- read.csv(file, header)

    file = R에서 읽어올 외부 데이터가 있는 파일의 위치(경로)와 파일명(확장자 포함)을 문자형으로 지정함

    header = csv 데이터에 있는 변수명을 사용하려면 TRUE

R 데이터 핸들링 정리

■ 데이터 프레임 속성

속성	R 코드	반환값
행의 개수	nrow(data)	행의 개수
열의 개수	ncol(data)	열의 개수
행의 이름	row.names(data) row.names(data) <- c() or paste()	행의 이름 (문자형, 1부터 시작하는 일련번호)
열의 이름	colnames(data) colnames(data) <- c() or paste()	열의 이름 (문자형)
차원	dim(data)	행의 개수, 열의 개수
차원의 이름	dimnames(data)	리스트 형태 행의 이름, 열의 이름
데이터의 구조	str(data)	데이터 프레임의 구조

■ 데이터 추출

① 데이터 일부 보기

- head(data, n = 정수) : 상위 n개의 행 (default = 6개)
- tail(data, n = 정수) : 하위 n개의 행 (default = 6개)

② 행 또는 열 추출하기

- data[행 index, 열 index] : 행 인덱스를 비워두면 해당하는 열만 추출.
- drop = FALSE : 하나의 열 또는 행만 추출하면 결과가 벡터 형태가 되므로 데이터프레임 결과를 유지하고 싶을 때.

- 인덱스에 c() : 규칙 없이 2개 이상의 행/열을 추출할 때
- 인덱스에 콜론 : 연달아 있는 행/열을 여러개 추출할 때
- 인덱스에 seq() : 일정한 간격으로 떨어져 있는 행/열을 여러개 추출할 때

- 인덱스에 c(변수명) : 추출하고 싶은 행/열의 위치를 모르고 변수명을 아는 경우
- 인덱스에 grep() : 변수명 중에서 특정 문자로 시작하거나 끝나거나 포함하고 있는 것을 추출할 때
    * grep(pattern, x, value)
        pattern = 문자형 ("^a" : a로 시작, "a$" : a로 끝, "a" : a 포함)
        x = 문자형 벡터 지정 (보통 rownames() 또는 colnames())
        value = TRUE - pattern을 만족하는 벡터의 값 반환 / FALSE - pattern을 만족하는 벡터의 위치 반환

- 인덱스에 substr() == “특정문자” : 변수명에서 일부 문자 추출하고 특정한 문자인지 비교하여 특정 행/열 추출

    * substr(x, start, stop)

        x = 문자형 벡터 지정 (보통 rownames() 또는 colnames())

        start = 추출할 문자의 첫 번째 위치를 정수형으로 지정함

        stop = 추출한 문자의 마지막 위치를 정수형으로 지정함

- 인덱스에 (data$변수명 == “특정 관측값”) : 데이터의 변수안에서 특정 값에 해당하는 행/열 추출
- 인덱스에 (data$변수명 >= 숫자) : 데이터의 변수안에서 특정 값 이상인 행/열 추출
- 인덱스에 (data$변수명1 == “특정 관측값”) & (data$변수명2 >= 숫자) : 두 조건을 모두 만족하는 행/열 추출
- 인덱스에 (data$변수명1 == “특정 관측값”) | (data$변수명2 >= 숫자) : 두 조건 중 적어도 하나를 만족하는 행/열 추출

■ 데이터 정렬

- sort(x, decreasing) : 벡터 정렬

    x = 정렬하고 싶은 벡터 지정

    decreasing = TRUE / FALSE(default)

- order(x, decreasing) : 데이터프레임 정렬 => 결과는 정렬했을 때의 위치를 정수형으로 알려줌.

    x = 데이터 프레임에서 정렬하고 싶은 변수(열) 지정

    decreasing = TRUE / FLASE(default)

- order의 결과값(인덱스)을 데이터 프레임의 행의 위치에 넣어주면 데이터 프레임이 정렬됨.
- order(data$변수명1, data$변수명2) : 2개 이상의 변수를 기준으로 정렬하고 싶을 때. 우선순위는 변수명1.
- order(data$변수명1, -data$변수명2) : 변수명1로 오름차순 정렬하고 변수명2로 내림차순 정렬. 수치형 변수일 때.

■ 데이터 합치기

① 데이터를 위아래로 합치기

- rbind(df1, df2)

② 데이터를 옆으로 합치기

- merge(x, y, by, all, all.x, all.y)

    x = 합쳐질 첫 번째 데이터

    y = 합쳐질 두 번째 데이터

    by = 두 데이터를 합칠 때 사용할 변수명. 데이터에 공통적으로 있는 변수. 변수는 각 데이터를 식별할 수 있어야함.

    all = TRUE - full join 방식

    all.x = TRUE - left join 방식

    all.y = TRUE - right join 방식

- inner join : 두 데이터에 공통으로 기준이 되는 변수가 가지는 값의 교집합, 즉 동일한 것만 합쳐짐. (ex. id)
- full join : 두 데이터에 공통으로 기준이 되는 변수가 가지는 값의 합집합. 변수의 값이 없는 부분은 결측값이 된다.
- left join : 두 데이터에 공통으로 기준이 되는 변수가 가지는 값 중에서 첫 번째로 들어가는 데이터의 값에 대하여 두 번째 데이터와 일치하는 경우에는 값을 가져오며, 일치하지 않을 경우에는 결측값.
- right join : 두 데이터에 공통으로 기준이 되는 변수가 가지는 값 중에서 두 번째로 들어가는 데이터의 값에 대하여 첫 번째 데이터와 일치하는 경우에는 값을 가져오며, 일치하지 않을 경우에는 결측값.

R 일변량 질적 자료의 분석 정리

■ 빈도와 백분율

표를 만들 때 데이터의 특징을 잘 알 수 있도록 내림차순으로 정렬.

표에 대한 해석을 기술할 때는 빈도(백분율) 또는 백분율(빈도) 형태로 함.

① 빈도

- 자료가 가지는 각각의 값이 몇 개가 있는지 구한 수치
- table(x)

    x = 분석하고 싶은 질적 자료. data$변수명, data[, "변수명"], data[, 변수위치] 등으로 표현

- sort(x, decreasing) : 빈도의 결과를 정렬할 때

    x = table() 결과

    decreasing = TRUE / FALSE(default)

② 백분율

- 자료가 가지는 각각의 값이 전체를 100으로 봤을 때 얼마나 차지하고 있는지 알려주는 수치
- prop.table(x) * 100

    x = 질적 자료에 대한 빈도 결과. table(data$변수명), table(data[, "변수명"]), table(data[, 변수위치])

- round(x, digits) : 반올림

    x = 반올림시키고 싶은 숫자가 들어 있는 벡터. 백분율의 결과

    digits = 표현되고 싶은 소수점의 자리수.

■ 막대그래프와 원그래프

① 막대그래프

- 빈도 또는 백분율을 이용하며 일반적으로 빈도를 더 많이 사용함.
- 축의 제목이 있어야함.
- 최솟값은 0이고 최댓값이 포함되도록 눈금을 설정함.
- 정렬을 하여 표현하는 것이 질적 자료의 특징을 파악하는데 좋음.
- barplot(height, col, main, xlab, ylab, xlim, ylim, horiz)

    height = 빈도의 정보. table() 결과

    col = 막대 색상. 기본은 grau. c()함수로 여러 색상 설정 가능.

    main = 차트 제목

    xlab = x축 제목

    ylab = y축 제목

    xim = x축 눈금. c() 함수로 최소값과 최대값 설정.

    ylim = y축 눈금. c() 함수로 최소값과 최대값 설정.

    horiz = 세로 막대그래프/ 가로 막대그래프 설정. TRUE - 세로, FALSE - 가로.

② 원그래프

- 질적 자료가 가지는 항목이 5개 이하인 경우에 적당함.
- 각 항목이 전체 중에서 얼마나 차지하고 있는지 표현하는데 유용함.
- 강조하고 싶은 항목이 있다면 해당 조각을 분리해서 표현하는 것도 좋음.
- pie(x, radius, init.angle)

    x = 빈도의 정보. table() 결과

    radius = 원의 반지름. -1 ~1 사이의 값. 절댓값이 1에 가까울수록 원의 반지름이 최대.

    init.angle = 원조각이 처음으로 시작하는 각도. 양의 값을 가지면 왼쪽으로 회전, 음의 값을 가지면 오른쪽으로 회전.

③ 그래프 저장

- 파일형태 저장 R코드

  그래프 코드

  dev.off()
- png 방식 : png(file="주소.png")
- jpeg 방식 : jpeg(file="주소.jpg")
- bmp 방식 : bmp(file="주소.bmp")
- pdf 방식 : pdf(file="주소.pdf")
- pdfscript 방식 : pdfscript(file="주소.ps")

R 일변량 양적 자료의 분석 정리

■ 표

- cut(data, breaks) : 구간의 정보를 가지는 새로운 변수 생성

    data = 분석할 일변량 양적 자료. 데이터의 형태는 벡터.

    breaks = 구간의 정보로 c(), :, seq(), quantile() 함수를 적절히 사용

- table(x) : 구간의 빈도 구하기

    x = 구간의 정보를 가지는 변수

- prop.table(x) * 100

    x = 빈도 결과. table()

■ 그래프

① 히스토그램

- 빈도나 백분율 이용
- x축은 양적 자료의 구간, u축은 각 구간의 빈도 또는 백분율
- 막대의 가로는 각 구간의 너비
- 알 수 있는 특징 : 구간의 현황, 빈도가 가장 많은 구간, 최댓값을 포함하는 구간, 대칭여부, 이상치 유무, 봉우리 개수
- hist(x, breaks)

    x = 분석할 일변량 자료. 데이터 형태는 벡터.

    breaks = 구간의 정보로 c(), :, seq(), quantile() 함수 또는 구간의 개수를 적절히 사용

② 상자 그림

- 이상치가 있는지 파악 가능한 그래프
- 최솟값, 제 1 사분위수, 중위수, 제 3 사분위수, 최댓값, 사분위 범위 등의 정보를 얻을 수 있음.
- boxplot(formular, x, range)

    formular = 집단별로 상자그림을 그릴 때 사용. 양적 자료(y) ~ 질적자료(group) 형태

    x = 분석할 일변량 자료. 데이터 형태는 벡터.

    range = 이상치를 판단하는 기준으로 default = 1.5
- boxplot(data$x ~ data$group) 와 boxplot(x~group, data=data)는 같음.

■ 기술통계량 (요약통계량)

- 자료의 대표값, 퍼짐, 분포의 모양을 알려줌.
- summary(x) 함수를 사용하면 6개의 기술통계량 출력 (최솟값, Q1, 중위수, 평균, Q3, 최댓값)

① 중심 (대표값)

- 평균 : mean(x, na.rm)
- 중위수 : median(x)
- 최빈값 : index <- which.max(table(x))

    table(x)[index]

② 산포 (퍼짐)

- 범위 : diff(range(x)
- 분산 : var(x)
- 표준편차 : sd(x)

③ 분포의 모양

- 왜도
- 첨도

* 집단별로 기술통계량 구하기

- 집단별 평균 : by(y, group, mean)
- 집단별 표준편차 : by(y, group, sd)
- 집단별 6개 기술통계량 : by(y, group, summary)