샘플데이터 활용하기

열 추출하기

- 열 추출하기
  - 특정한 열(또는 변수) 추출 시 열이 있는 위치로 추출하거나 열의 이름을 가지고 추출할 수 있음
- 열의 위치를 아는 경우
  - 열의 위치를 아는 경우, 열의 위치를 입력함
  - 하나의 열만 추출할 수도 있고 여러 개의 열을 추출할 수도 있음
    - 하나인 경우 - index 자리에 해당 열이 있는 위치를 수치로 입력
    - 여러 개의 경우 => c(), :, seq() 함수를 사용
      - c()는 추출할 열들의 위치가 규칙이 없을 때 사용
      - :은 연달아 있는 경우에 사용
      - seq() 함수는 일정한 간격으로 떨어진 열을 추출할 때 사용

열 추출하기

- 열의 위치를 아는 경우
  - 데이터명[, index]
  - 데이터명[, 2]
  - 데이터명[, 2, drop=FALSE]
  - 데이터명[, c(2, 3, 7)]
  - 데이터명[, 7:10]
  - 데이터명[, seq(from=2, to=10, by=2)]
  -
  - diamonds[, 2]
    - diamonds에서 두 번째 열을 추출 => 최종 결과는 벡터(vector)
  - diamonds[, 2, drop=FALSE]
    - diamonds에서 두 번째 열을 추출 => 최종 결과는 데이터 프레임의 유지
  - diamonds[, c(2, 3, 7)]
    - 2, 3, 7번째 열을 추출, 총 3개의 열을 추출 => 최종 결과는 데이터 프레임
    - 두 개 이상의 열을 추출하면 그 결과는 데이터 프레임이
  - diamonds[, 7:10]
    - 7~10번째 열을 추출, 총 4개의 열이 추출됨
    - :은 연달아 있는 열을 추출할 때 사용
  - diamonds[, seq(from=2, to=10, by=2)]
    - 짝수 번째 있는 모든 열을 추출
    - 두 번째 열부터 시작해서 10번째 열을 넘지 않을 때까지 추출할 열의 위치를 2씩 증가시켜서 추출

열 추출하기

- 열의 위치를 아는 경우
  - diamonds[, index]
  -
  - diamonds[, “cut”]
    - diamonds의 cut이라는 변수명을 가지는 열을 추출 => 최종 결과는 벡터(vector)
  - diamonds[, “cut”, drop=FALSE]
    - cut 이라는 변수명을 가지는 열을 추출 최종 결과는 데이터 프레임
  - diamonds[, c(“cut”, “price”)]]
    - cut과 price 변수명을 가지는 두 개의 열을 추출 => 최종결과는 데이터 프레임
  - diamonds[, grep(“^c”, colnames(diamonds))]
    - 변수명 중에서 c라는 문자로 시작하는 열들을 추출
  - diamonds[, grep(“c”, colnames(diamonds))]
    - 변수명 중에서 c라는 문자를 포함하고 있는 열들을 추출
  - diamonds[, grep(“c$”, colnames(diamonds))]
    - 변수명 중에서 c라는 문자로 끝나는 열들을 추출

행 추출하기

- 특정한 행을 추출할 때에도 행의 위치를 알거나 행의 이름을 알면 열을 추출하는 방법과 동일
- 데이터명[index,] 형태로 조건을 줌
- 행에서만 할 수 있는 방법으로 비교 연산자나 논리 연산자를 이용하여 특정 조건을 만족하는 행을 추출하는 방법을 학습함

행 추출하기

- diamonds[diamonds$cut == “Fair”,]
  - cut이라는 변수에 저장된 값들 중에서 “Fair”인 데이터만 추출
- diamonds[diamonds$price >= 18000,]
  - price가 18000 달러 이상인 데이터를 추출
- diamonds[(diamonds$cut == “Fair”) & (diamonds$price >= 18000),]
  - cut이 “Fair”이고 price가 18,000 달러 이상인 데이터를 추출
  - &는 주어진 조건을 동시에 만족할 때만 TRUE가 되는 논리 연산자임
- diamonds[(diamonds$cut == “Fair”) | (diamonds$price >= 18000),]
  - cut이 “Fair”이거나 또는 price가 18,000 달러 이상인 데이터를 추출
  - |은 주어진 조건 중에서 하나만을 만족하면 TRUE가 되는 논리연산자임

새로운 변수 만들기

- 데이터를 분석하는 과정에서 원자료를 이용하여 새로운 변수를 만드는 일을 자주 하게 됨
- 여러 개의 변수에 대한 평균이나 합계를 이용하거나 다른 수식 등을 이용하여 새로운 변수를 만듦
  - [x, y, z라는 세 개의 변수가 가지는 값의 평균, 합계를 이용하여 새로운 변수 생성]
  - 데이터명$새로운변수명 = 함수 또는 수식
  -
  - diamonds$xyz.sum = (diamons$x + diamonds$y + diamonds$y)
    - xyz.sum을 새롭게 생성되는 변수
    - x, y, z의 합계 (diamonds$x + diamonds$y + diamonds$z)의 수식 이용
    - 구해진 합계는 xzy.sum 이라는 변수에 저장
  - diamonds$xyz.sum = rowSums(diamonds[, c(“x”, “y”, “z”)]
    - rowSums() 함수는 수식을 함수로 구현된 것으로 수식보다는 함수가 더 편리함
  - diamonds$xyz.mean = (diamons$x + diamonds$y + diamonds$y) / 3
    - xyz.mean도 새롭게 생성되는 변수이며, x, y, z의 평균 값을 구함
  - diamonds$xyz.mean = rowMeans(diamonds[, c(“x”, “y”, “z”)]

데이터 수정하기

- 데이터 중에서 조건에 맞는 데이터의 값을 변경할 수 있음
  - 예 : 입력오류의 경우, 이상치가 있는 경우에 해당 값을 다른 값으로 수정
  - 데이터명[행의 조건, 열] = 변경할 값
  - diamonds[diamonds$price >= 18000, “price”] = 18000
    - diamonds의 price가 18,000 달러 이상이면 price를 18,000 달러로 수정
  - diamonds[(diamonds$cut == “Fair”)&(diamonds$price >= 18000), “x”] = NA
    - cut이 “Fair”이고 price가 18000 달러 이상이면 x값을 NA로 수정

    - NA은 Not Available의 약자로 결측치 또는 결측값(missing value)을 의미

데이터 삭제하기

- 행 삭제하기
  - 데이터 중에서 특정한 행을 삭제하는 방법

  - diamonds[-c(10, 20, 30),]
    - 10번째, 20번째, 30번째의 행을 삭제
  - diamonds[-(100:200),]
    - 100~200번째의 행을 삭제
  - diamonds[-seq(from=1, to=length(diamonds), by=10),]
    - 첫 번째 행부터 시작해서 10행씩 떨어져 있는 행들을 삭제

데이터 삭제하기

- 행 삭제하기
  - diamonds$table = NULL
    - diamonds의 table 변수를 삭제
  - subset(diamonds, select=-c(2, 3))
  - subset(diamonds, select=-c(cut, color))
  - within(diamonds, rm=c(cut, color))
    - cut, color라는 두 개의 변수를 삭제

데이터 정렬하기

- 데이터 프레임 형태의 데이터를 정렬할 때는 order() 함수를 사용
- order() 함수는 기본으로 오름차순으로 정렬
- 내림차순으로 정렬하고 싶으면 order() 함수의 argument에 decreasing=TRUE를 지정

  - diamonds[order(diamonds$price),]
    - price를 기준으로 오름차순으로 정렬
  - diamonds[order(diamonds$price, decreasing=TRUE),]
    - price를 기준으로 내림차순으로 정렬
  - diamonds[order(diamonds$cut, diamonds$color),]
    - 1) cut으로 먼저 오름차순으로 정렬하고,
    - 2) color로 오름차순으로 정렬
  - diamonds[order(diamonds$cut, diamonds$color, decreasing=TRUE),]
    - 1) cut으로 내림차순으로 정렬
    - 2) color로 내림차순으로 정렬