통계학 실습

8. 데이터프레임 및 그래프출력

R 데이터프레임

- R에는 데이터를 다루기 위한 ‘데이터프레임’이라는 자료구조가 존재한다.

- 다른 프로그래밍 언어의 ‘사전’ 개념과 비슷하며, R에서 사용되는 행렬 자료 구조와 같은 형태를 띄고 있다.
  - 행렬과 같이 2차원 배열 구조를 가지고 있다.
  - 각 열에는 이름이 주어지며, 데이터에 대한 해당 속성을 가지고 있는 형태이다.
  - 동일한 길이의 벡터로 이루어진 리스트가 하나의 요소가 된다.

- 여러 가지 함수와 함께 쓰이면서 다양한 통계 자료를 보여줄 수 있다.

데이터프레임 사용

```
> name <- c("James", "Alice", "Tim")
> score <- c(75, 85, 90)
> class <- data.frame(name, score)
> class
   name score
1 James    75
2 Alice    85
3   Tim    90
> class[[1]]
[1] James Alice Tim
Levels: Alice James Tim
> class[1]
   name
1 James
2 Alice
3   Tim
> class[1,]
   name score
1 James    75
> class[,1]
[1] James Alice Tim
Levels: Alice James Tim
> 
```

- 데이터프레임을 생성하기 위해 자료를 가진 벡터를 생성한다.
- 같은 길이를 가진 벡터들을 묶어 데이터프레임으로 생성한다.
- Levels : 벡터 값 중에 겹치지 않는 unique한 값을 뜻한다.
- 다양한 접근법이 가능하다.
  - [[1]] : 구성하는 벡터 자체에 접근
  - [1,] : 행렬에서 행에 접근
  - [,1] : 행렬에서 열에 접근

데이터프레임 사용 (2)

```
> class$name
[1] James Alice Tim
Levels: Alice James Tim
> class$score
[1] 75 85 90
> str(class)
'data.frame':   3 obs. of  2 variables:
 $ name : Factor w/ 3 levels "Alice","James",..: 2 1 3
 $ score: num  75 85 90
> summary(class)
    name        score
 Alice:1   Min.   :75.00
 James:1   1st Qu.:80.00
 Tim  :1   Median :85.00
           Mean   :83.33
           3rd Qu.:87.50
           Max.   :90.00
>
```

- ‘데이터프레임$열이름’ 으로 열에 주어진 이름으로도 접근 가능하다.

- str()이나 summary() 함수를 이용하여 통계적 정보를 미리 파악할 수 있다.
  - str() 함수는 해당 자료의 직접적인 정보를 분석하여 준다. ex) 변수 자료형
  - summary() 함수는 해당 자료로 얻을 수 있는 다른 정보를 보여준다. ex) 최대값

데이터프레임 사용 (3)

```
> class$gender <- c("M", "F", "M")
> class
   name score gender
1 James    75      M
2 Alice    85      F
3   Tim    90      M
> newRow
   name score gender
1 Sarah    90      F
> class <- rbind(class, newRow)
> class
   name score gender
1 James    75      M
2 Alice    85      F
3   Tim    90      M
4 Sarah    90      F
> passes
   name pass
1 James    P
2   Tim    P
3 Alice    P
4 Sarah    P
> class <- merge(class, passes, key="name")
> class
   name score gender pass
1 Alice    85      F    P
2 James    75      M    P
3 Sarah    90      F    P
4   Tim    90      M    P
```

- 변수이름으로 접근하여 하나의 열을 추가할 수도 있다.
- 행렬과 같이 결합 함수를 이용하여 데이터프레임끼리 결합이 가능하다.
- merge() 함수를 이용하여 데이터프레임을 결합시킬 수 있다.
  - 행의 길이가 다를 경우 merge 결합이 되지 않지만, key를 이용하여 유연하게 결합할 수 있다.

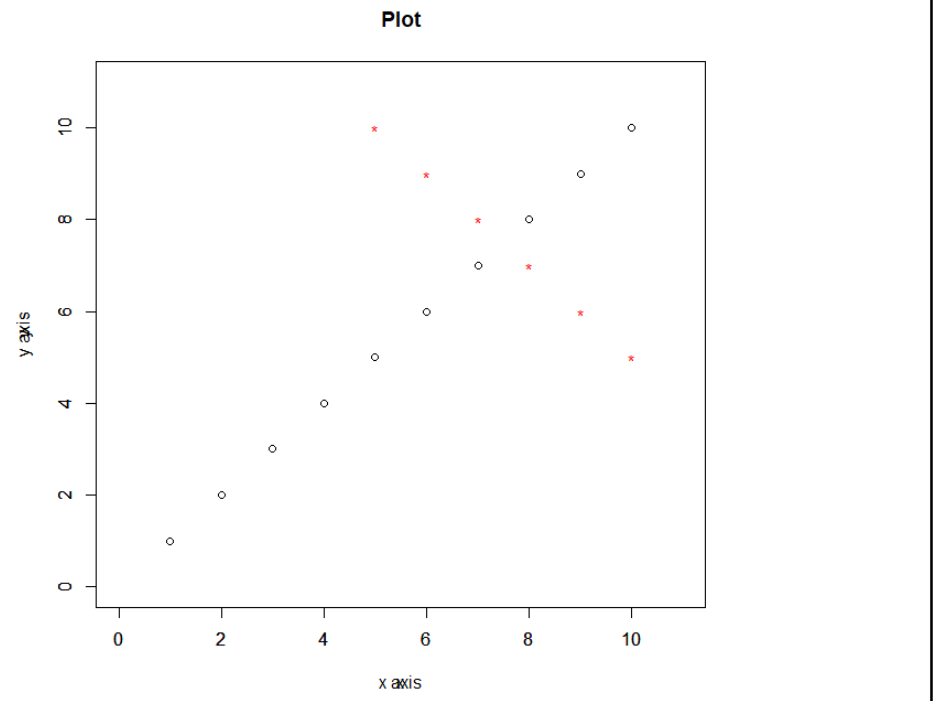
그래프 출력

그래프 옵션	
옵션	**설명**
main	제목
xlab="문자열" ylab="문자열"	축 이름
xlim=벡터, ylim=벡터	축의 한계 범위
그래프 타입	
타입	**설명**
type="p"	점 모양 (default)
type="l"	선 모양 (꺾은선)
type="b"	점+선 모양
type="c"	"b"에서 점 제거

그래프 모양	
옵션	**설명**
lty="blank"	투명선
lty="dashed"	대쉬선
lty="dotted"	점선
col="색깔"	기호의 색깔
pch="문자"	점의 모양
bg="색깔"	배경색
lwd="숫자"	선의 굵기
cex="숫자"	점의 크기

그래프 출력 – 선형 그래프 plot()

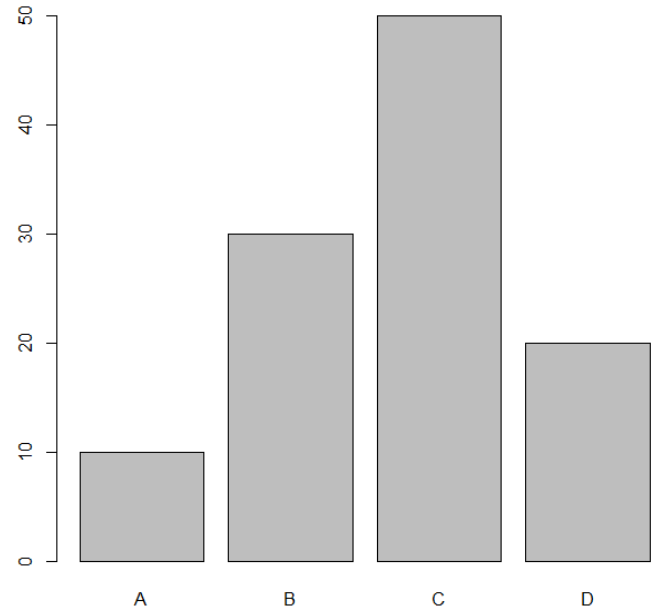
```
> x <- 1:10
> y <- 1:10
> plot(x, y, xlim=c(0,11), ylim=c(0,11), xlab="x axis", ylab="y axis", main="Plot")
>
> par(new=T)
> x <- 5:10
> y <- 10:5
> plot(x, y, xlim=c(0,11), ylim=c(0,11), col="red", pch="*")
>
```



- 선형 그래프 출력
  1. 먼저 출력할 데이터를 마련한다.
  2. plot() 함수에서 옵션을 고른다.
     - 그래프 제목과 축 제목은 넣는 것이 좋다.
     - 축의 범위를 정해주어 안전하게 그려준다.
  3. 여러 그래프를 동시에 출력하려면 par(new=T)를 그래프 출력 전에 사용한다.
     - 여러 그래프를 그리는 경우 겹치지 않도록 색깔이나 기호를 달리하여 그려주는 것이 좋다.

그래프 출력 – 막대 그래프 barplot()

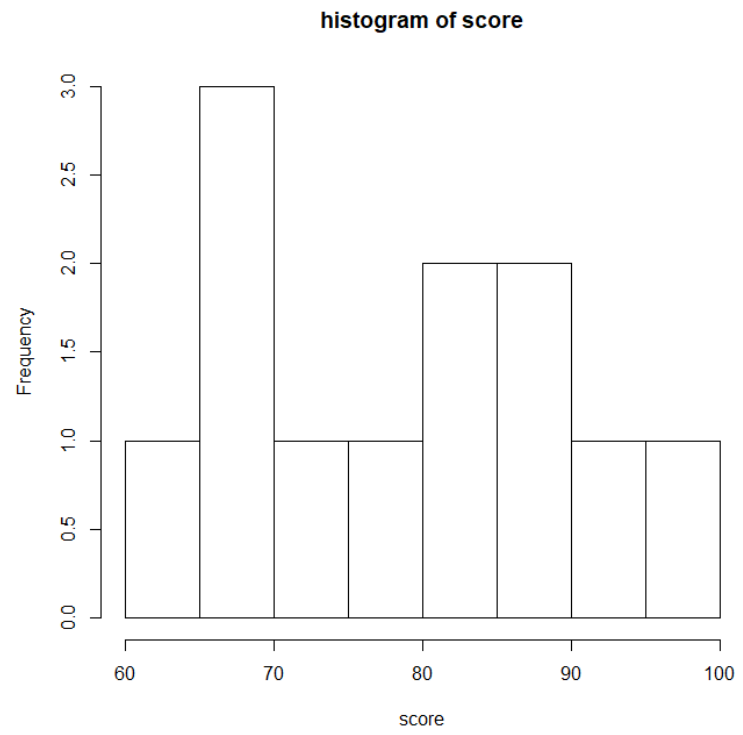
```
> y <- c(10, 30, 50, 20)
> barplot(y, names=c("A","B","C","D"))
>
```



- 막대그래프 출력
  1. 출력할 데이터를 준비한다.
     - 막대그래프는 데이터 간의 연속적인 속성이 없기 때문에 각자의 이름을 옵션으로 설정한다.
  2. 막대그래프 함수의 옵션을 설정한다.
     - names : 각 막대의 라벨을 설정하는 벡터
     - horiz=T : T이면, 막대를 옆으로 눕힌다.
     - beside=T : T이면, 그룹으로 묶어서 하나의 그래프 출력에 여러 막대를 그릴 수 있다. (여러 데이터를 막대로 그리므로 행렬로 가져야 한다.)

그래프 출력 – 히스토그램 hist()

```
> score <- c(62, 66, 68, 70, 75, 80, 82, 85, 86, 90, 95, 99)
> hist(score, main="histogram of score", breaks=8)
>
```

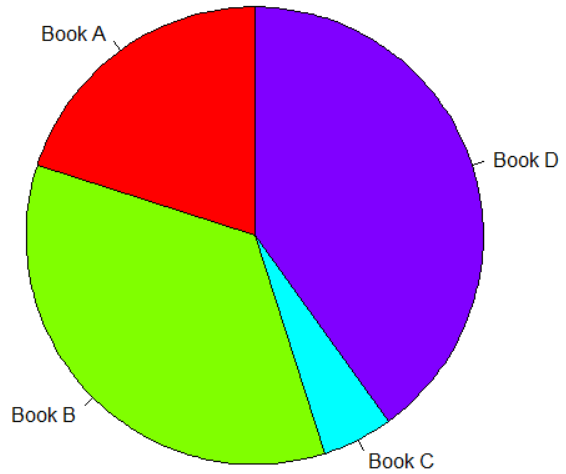


- 히스토그램은 범위 내의 빈도수에 대한 그래프를 나타낸다.

- 히스토그램 출력
  1. 데이터를 준비한다.
  2. 데이터의 최대값과 최소값을 구한다.
  3. 최대값과 최소값을 적절한 구간으로 나눈다.
     - breaks : 표시될 막대의 개수를 나타내므로, 숫자가 커질수록 막대가 가늘어진다.

그래프 출력 – 파이 차트 pie()

```
> lab <- c("Book A", "Book B", "Book C", "Book D")
> data <- c(20, 35, 5, 40)
> pie(data, init.angle=90, label=lab, col=rainbow(length(data)))
>
```



- 파이 차트는 전체 합을 100%로 만들어 전체에서의 비율을 한 눈에 확인할 수 있게 한다.

- 파이 차트 출력
  1. 데이터를 준비한다.
  2. 다양한 옵션을 준비한다.
     - label : 데이터의 이름
     - init.angle : 그래프 출력 시작 각도
     - clockwise : 시계/반시계 방향 (기본:반시계)
     - 각 분포의 %를 같이 출력하려면 label에 아래와 같이 추가하여 label을 써준다.

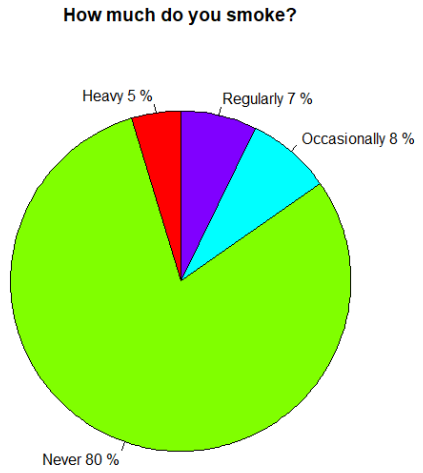
```
pct <– round(data / sum(data) * 100)
lab <– paste(lab, pct)
lab <– paste(lab, “%” , sep= “ ” )
```

실습

- 철수는 설문조사 자료를 보고 사람들의 흡연 여부 분포를 확인하려고 한다.
  - 설문조사 데이터셋 : MASS 패키지의 'survey'
  - survey$Smoke : 흡연 여부를 조사한 내용
  - summary() 함수를 이용하여 각 여부에 대한 사람들의 수를 확인한다.
    1. Heavy : 엄청 많이 피는 사람
    2. Never : 전혀 안 피는 사람
    3. Occas : 가끔 피는 사람
    4. Regul : 주기적으로 피는 사람

    ※ table() 함수를 이용하면, 각 항목에 대한 개수를 직접 벡터로 따로 저장할 수 있다.

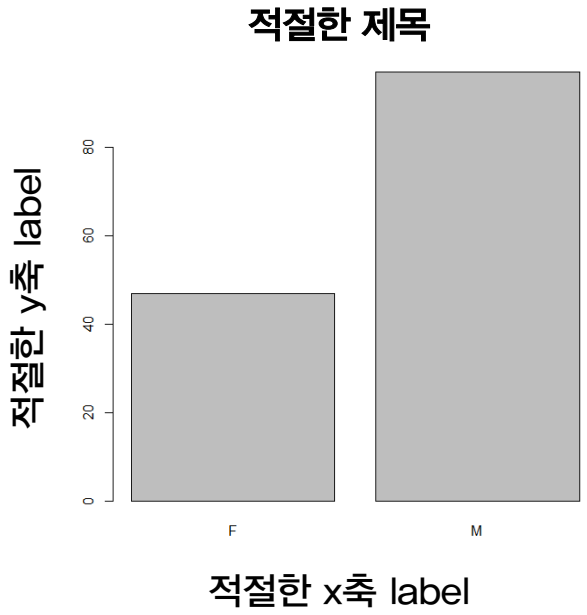
  - 파이 차트를 이용하여 흡연 여부 분포를 한 눈에 확인하자.



숙제

- 1. MASS 패키지를 이용하여 cats 데이터를 가져온다.

- 2. 고양이의 성별에 따른 개체 수를 그리기 위해 막대 그래프로 출력해본다.
  - summary() 함수를 이용하여 성별에 따른 개체의 수를 파악한다.
  - 개체의 수를 막대그래프로 나타내고, 제목, x축과 y축의 label을 적절히 붙인다.
  - 〈예시〉

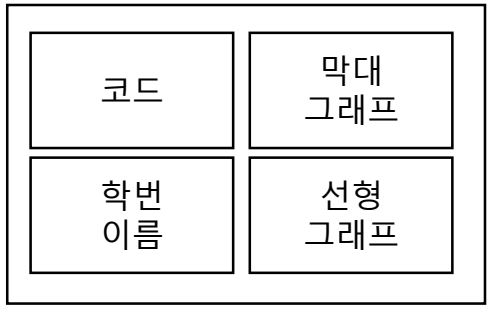


숙제 (2)

- 3. cats 데이터의 Bwt와 Hwt의 관계를 plot하여 그래프로 출력한다.
  - 아래의 옵션을 지켜서 코드를 작성하여 출력해본다.
    1. X축 : Bwt (Body Weight), Y축 : Hwt (Heart Weight)
    2. X축 Label : Body Weight(kg), Y축 Label : Heart Weight(g)
    3. 각 축의 범위는 최소값보다 크지 않은 정수, 최대값보다 작지 않은 정수로 한다.
       - 위의 정보를 알려면 summary() 함수가 필요하다.
    4. 제목 : Heart Weight(g) by Body Weight(kg) of cats
    5. 기호 : ♡, 색상 : red

제출

- 1. 아래 내용이 한 화면에 들어오게 해서 스크린샷 찍기
  - 프로그램 내에 작성한 코드 (R 작업창 또는 텍스트로 옮겨서)
  - 막대 그래프 – 숙제 2번 스크린샷
  - 선형 그래프 – 숙제 3번 스크린샷
  - 메모장에 학번, 이름



- 2. 휴대폰 촬영 말고 캡쳐 도구 등을 이용해서 캡쳐할 것.

- 제출 서식
  - 스크린샷 파일 이름과 이메일 제목을 모두 아래 서식으로 동일하게 할 것
  - [8주차][학번][이름]통계학실습

- 제출 이메일
  - gtsk623@gmail.com