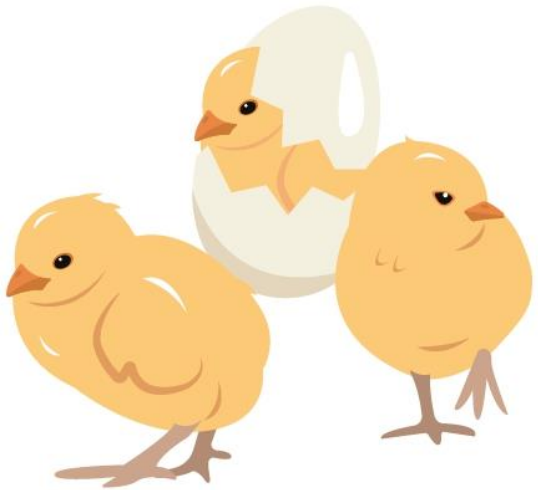
모두를 위한

R 데이터 분석 입문

2판



Chapter 05
단일변수 자료의 탐색

목차

Section 01

자료의 종류

1. 자료의 특성에 따른 분류

그림 5-1 자료의 특성에 따른 분류

1.1 범주형 자료

- 범주형 자료(categorical data)는 질적 자료(qualitative data)라고도 부르며, 성별과 같이 범주 또는 그룹으로 구분할 수 있는 값으로 구성된 자료

1. 자료의 종류

- 범주형 자료의 값들은 기본적으로 숫자로 표현할 수 없고, 대소(大小) 비교나 산술 연산이 적용되지 않음

표 5-1 범주형 자료의 예

범주형 자료	범주형 자료의 표현
성별	M, F, F, M, M, M, F
혈액형	A, B, O, AB, B, A, O
선호하는 색	빨강, 파랑, 노랑, 빨강, 초록, 검정
찬성 여부	YES, NO, NO, YES, NO

- 아래와 같이 범주형 자료를 숫자로 표기했다고 해서 계산 가능한 연속형 자료가 되는 것은 아님

- 성별 : 0, 1
- 혈액형 : 1, 2, 3, 4

1.2 연속형 자료

- 연속형 자료(numerical data)는 양적자료(quantitative data)라고도 부르며, 크기가 있는 숫자들로 구성된 자료
- 연속형 자료의 값들은 대소 비교가 가능하고, 평균, 최댓값, 최솟값과 같은 산술 연산이 가능

표 5-2 연속형 자료의 예

연속형 자료	연속형 자료의 표현
몸무게	57.4, 64.1, 71.0, 65.1, 90.1
키	162, 180, 174, 171, 181, 167
일평균 온도	19.1, 20.5, 20.5, 21.1, 22.0
자녀의 수	0, 2, 1, 3, 0, 1, 2

2. 변수의 개수에 따른 분류

- 통계학에서 말하는 변수는 우리가 R에서 배운 변수와는 의미상 다소 차이가 있음
- 통계학에서의 변수는 우리가 '연구, 조사, 관찰하고 싶은 대상의 특성'을 말하며, 키, 몸무게, 혈액형, 매출액, 습도, 미세먼지 농도 등

그림 5-2 변수의 개수에 따른 분류

- 단일변수 자료(univariate data): 하나의 변수로만 구성된 자료, '일변량 자료'라고도 부름
- 다중변수 자료(multivariate data): 두 개 이상의 변수로 구성된 자료, 다변량 자료라고 부름. 특별히 두 개의 변수로 구성된 자료를 이변량 자료(bivariate data)라고 함

1. 자료의 종류

몸무게
62.4
65.3
59.8
46.5
49.8
58.7

(a) 단일변수 자료

키	몸무게	성별
168.4	62.4	M
169.5	65.3	F
172.1	59.8	F
185.2	46.5	M
173.7	49.8	M
175.2	58.7	F

(b) 다중변수 자료

그림 5-3 단일변수 자료와 다중변수 자료

- R에서는 단일변수 자료는 벡터에, 다중변수 자료는 매트릭스나 데이터프레임에 저장하여 분석
- 매트릭스 또는 데이터프레임 형태의 자료에서 하나의 열(column)이 하나의 변수를 나타냄
- 열(column)의 개수 = 변수의 개수

1. 자료의 종류

그림 5-4 변수의 개수와 자료의 특성에 따른 분류

- 변수의 개수와 자료의 특성에 따라 세분화된 분류가 가능
- 세분화된 분류에 따라 각각 서로 다른 분석 방법들이 존재

Section 02

단일변수 범주형 자료의 탐색

2. 단일변수 범주형 자료의 탐색

- **단일변수 범주형 자료(또는 일변량 질적 자료):** 특성이 하나이면서 자료의 특성이 범주형인 자료
- 범주형 자료에 대해서 할 수 있는 기본적인 작업은 자료에 포함된 관측값들의 종류별로 개수를 세는 것
- 개수를 세면 종류별 비율을 알 수 있음
- 막대그래프나 원그래프의 작성이 가능
- 단일변수 범주형 자료의 예: 학생들이 선호하는 계절

WINTER	SUMMER	SPRING	SUMMER	SUMMER
FALL	FALL	SUMMER	SPRING	SPRING

1. 도수분포표의 작성

코드 5-1

```
favorite <- c('WINTER', 'SUMMER', 'SPRING', 'SUMMER', 'SUMMER',
              'FALL', 'FALL', 'SUMMER', 'SPRING', 'SPRING')
favorite                                    # favorite의 내용 출력
table(favorite)                             # 도수분포표 계산
table(favorite)/length(favorite)            # 비율 출력
```

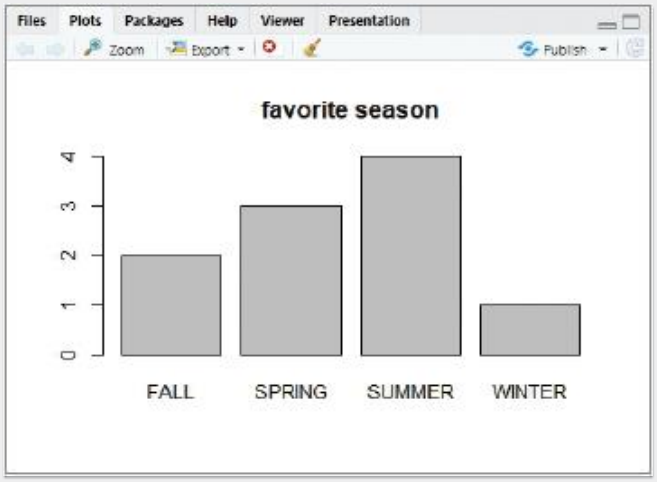
```
> favorite <- c('WINTER', 'SUMMER', 'SPRING', 'SUMMER', 'SUMMER',
+               'FALL', 'FALL', 'SUMMER', 'SPRING', 'SPRING')
> favorite
 [1] "WINTER" "SUMMER" "SPRING" "SUMMER" "SUMMER" "FALL"   "FALL"
 [8] "SUMMER" "SPRING" "SPRING"
> table(favorite)                       # 도수분포표 계산
favorite
  FALL SPRING SUMMER WINTER
     2      3      4      1
> table(favorite)/length(favorite)  # 비율 출력
favorite
  FALL SPRING SUMMER WINTER
   0.2    0.3    0.4    0.1
```

2. 막대그래프의 작성

코드 5-2

```
ds <- table(favorite)
ds
barplot(ds, main='favorite season')
```

```
> ds <- table(favorite)
> ds
favorite
  FALL SPRING SUMMER WINTER
     2      3      4      1
> barplot(ds, main='favorite season')
```



3. 원그래프의 작성

코드 5-3

```
ds <- table(favorite)
ds
pie(ds, main='favorite season')
```

```
> ds <- table(favorite)
> ds
favorite
  FALL SPRING SUMMER WINTER
     2      3      4      1
> pie(ds, main='favorite season')
```

4. 숫자로 표현된 범주형 자료

- 숫자 형태의 범주형 자료도 문자 형태의 범주형 자료와 마찬가지로 도수분포를 계산한 후 막대그래프와 원그래프를 그려서 자료의 내용을 확인
- 학생 15명이 선호하는 색깔을 조사한 자료

2, 3, 2, 1, 1, 2, 2, 1, 3, 2, 1, 3, 2, 1, 2

(1=초록, 2=빨강, 3=파랑)

코드 5-4

```
favorite.color <- c(2, 3, 2, 1, 1, 2, 2, 1, 3, 2, 1, 3, 2, 1, 2)
ds <- table(favorite.color)
ds
barplot(ds, main='favorite color')
colors <- c('green', 'red', 'blue')
names(ds) <- colors            #자료값 1,2,3을 green, red, blue로 변경
ds
barplot(ds, main='favorite color', col=colors)      # 색 지정 막대그래프
pie(ds, main='favorite color', col=colors)          # 색 지정 원그래프
```

2. 단일변수 범주형 자료의 탐색

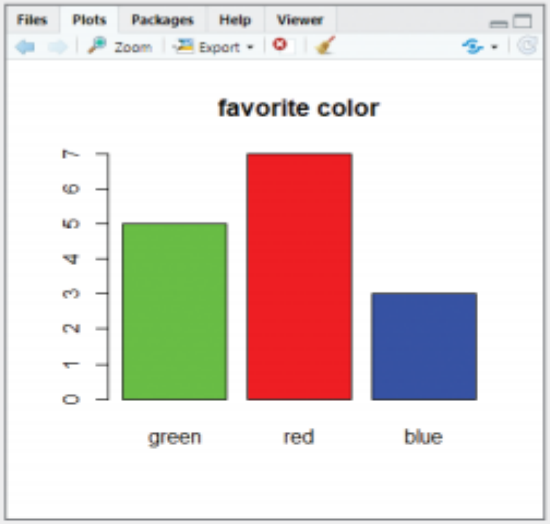
```
> favorite.color <- c(2, 3, 2, 1, 1, 2, 2, 1, 3, 2, 1, 3, 2, 1, 2)
> ds <- table(favorite.color)
> ds
favorite.color
1 2 3
5 7 3
> barplot(ds, main='favorite color')
```


```
> colors <- c('green', 'red', 'blue')
> names(ds) <- colors        # 자료값 1,2,3을 green, red, blue로 변경
> ds
```

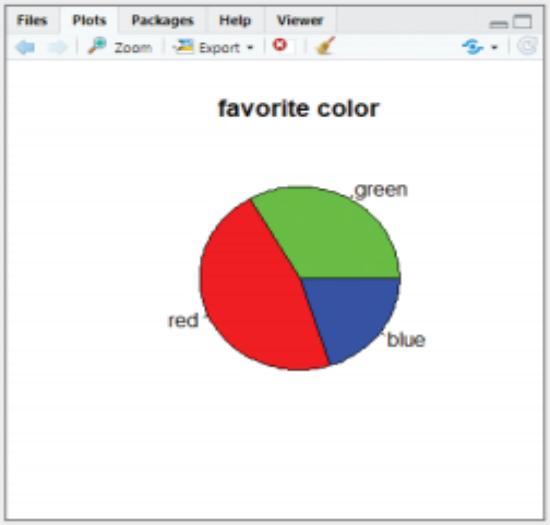
2. 단일변수 범주형 자료의 탐색

```
green   red  blue
    5     7     3
```

```
> barplot(ds, main='favorite color', col=colors)      # 색 지정 막대그래프
```




```
> pie(ds, main='favorite season', col=colors)         # 색 지정 원그래프
```



여기서 잠깐! 플롯 창의 Zoom 아이콘과 Export 아이콘

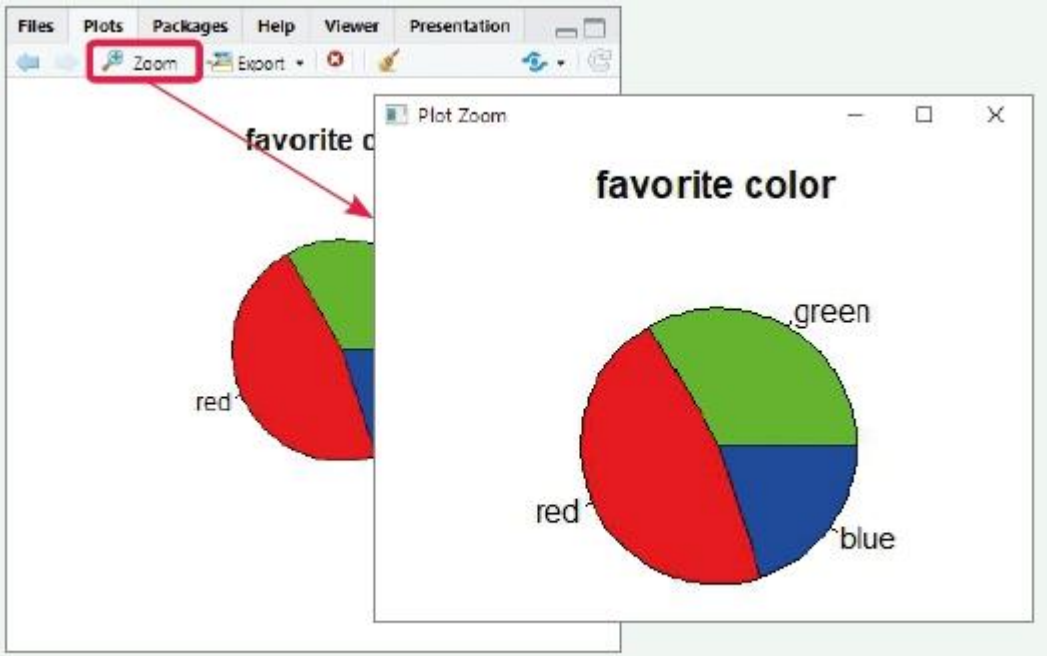


그림 5-5 Zoom 아이콘의 활용

플롯창의 크기가 그래프를 표시할 수 있을 만큼 충분히 커야 함.
그렇지 않은 경우 에러 메시지가 콘솔창에 표시됨

Section 03

단일변수 연속형 자료의 탐색

1. 평균과 중앙값

- 연속형 자료는 관측값들이 크기를 가지기 때문에 범주형 자료에 비해 다양한 분석 방법이 존재
- 평균, 중앙값 : 전체 데이터를 대표할 수 있는 값
- 평균

$$\bar{x} = \frac{1}{n} \cdot \sum_{i=1}^{n} x_i$$

- 중앙값(median) : 자료의 값들을 크기순으로 일렬로 줄 세웠을 때, 가장 중앙에 위치하는 값

그림 5-6 평균과 중앙값

- 절사평균(trimmed mean)은 자료의 관측값들 중에서 작은 값들의 하위 n%와 큰 값들의 상위 n%를 제외하고 중간에 있는 나머지 값들만 가지고 평균을 계산

코드 5-5

```
weight <- c(60, 62, 64, 65, 68, 69)
weight.heavy <- c(weight, 120)
weight
weight.heavy

mean(weight)                          # 평균
mean(weight.heavy)                    # 평균

median(weight)                        # 중앙값
median(weight.heavy)                  # 중앙값

mean(weight, trim=0.2)                # 절사평균(상하위 20% 제외)
mean(weight.heavy,trim=0.2)           # 절사평균(상하위 20% 제외)
```

3. 단일변수 연속형 자료의 탐색

```
> weight <- c(60, 62, 64, 65, 68, 69)
> weight.heavy <- c(weight, 120)
> weight
[1] 60 62 64 65 68 69
> weight.heavy
[1]  60  62  64  65  68  69 120
> mean(weight)                          # 평균
[1] 64.66667
> mean(weight.heavy)                    # 평균
[1] 72.57143
> median(weight)                        # 중앙값
[1] 64.5
> median(weight.heavy)                  # 중앙값
[1] 65
> mean(weight, trim=0.2)                # 절사평균(상하위 20% 제외)
[1] 64.75
> mean(weight.heavy,trim=0.2)           # 절사평균(상하위 20% 제외)
[1] 65.6
```

2. 사분위수

- 사분위수(quatile)란 주어진 자료에 있는 값들을 크기순으로 나열했을 때 이것을 4등분하는 지점에 있는 값들을 의미
- 자료에 있는 값들을 4등분하면 등분점이 3개 생기는데, 앞에서부터 '제1사분위수(Q1)', '제2사분위수(Q2)', '제3사분위수(Q3)'라고 부르며, 제2사분위수(Q2)는 중앙값과 동일
- 전체 자료를 4개로 나누었기 때문에 4개의 구간에는 각각 25%의 자료가 존재

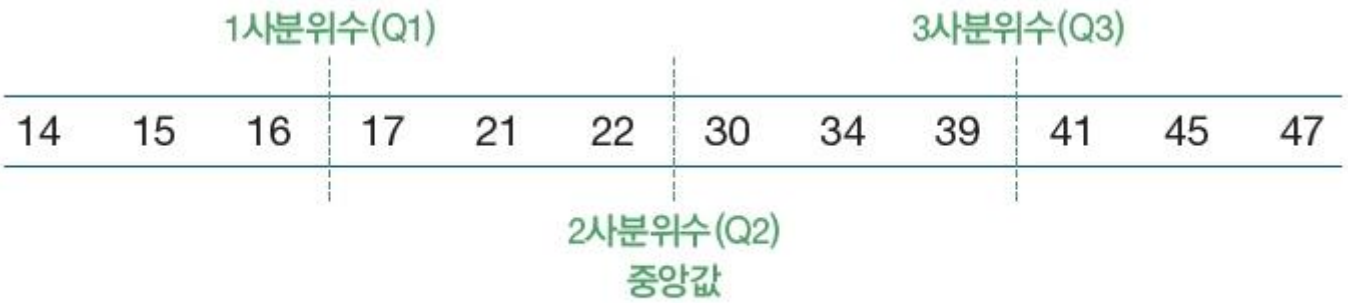


그림 5-7 사분위수의 예

사분위수 계산법

기본 type=7

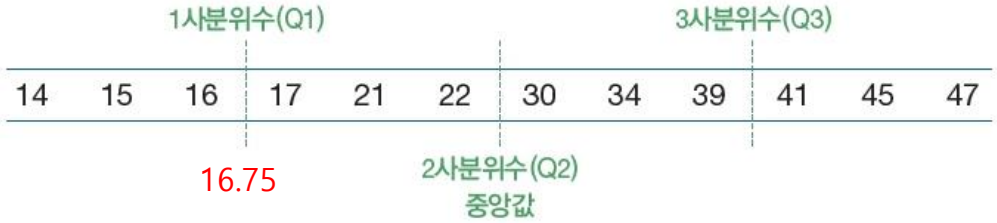


그림 5-7 사분위수의 예

R의 `quantile()` 함수의 기본 작동 방식

R의 `quantile()` 함수는 `type` 인수에 따라 여러 가지 방법으로 분위수를 계산할 수 있으며, 기본적으로 `type = 7`을 사용합니다. `type = 7`은 다음과 같은 방법으로 분위수를 계산합니다:

$P = (n - 1) \times p + 1$

여기서 n은 데이터의 개수, p는 분위수 (0.25, 0.5, 0.75 등)를 나타냅니다.

예제: `mydata`의 25% 분위수 구하기

주어진 데이터셋 `mydata`는 다음과 같습니다:

$[14, 15, 16, 17, 21, 22, 30, 34, 39, 41, 45, 47]$

데이터셋의 개수 n은 12입니다. 25% 분위수를 계산하려면:

$P = (12 - 1) \times 0.25 + 1 = 11 \times 0.25 + 1 = 2.75$

위치 계산

- 2.75번째 위치를 찾습니다. 이는 2번째 값(15)과 3번째 값(16) 사이에 있습니다.
- 보간을 통해 값을 계산합니다:
  $Q1 = 15 + 0.75 \times (17 - 16) = 15 + 0.75 \times 1 = 15 + 0.75 = 16.75$

따라서, 1사분위수(Q1)는 16.75로 계산됩니다.

중요

- 100명의 학생을 대상으로 영어시험을 본 결과에 대해 사분위수를 구하였더니 Q1=60, Q2=80, Q3=90이라고 가정하면 →

  25명의 학생은 성적이 60점 미만이다.

  25명의 학생은 성적이 60점~80점 사이이다.

  25명의 학생은 성적이 80점~90점 사이이다.

  25명의 학생은 성적이 90점 이상이다.

  90점 이상인 학생이 25명이나 되기 때문에 이번 영어시험은 매우 쉬웠다.

  전체 50%의 학생이 80점 이상의 성적을 받았다.

3. 단일변수 연속형 자료의 탐색

코드 5-6

```
mydata <- c(60, 62, 64, 65, 68, 69, 120)
quantile(mydata)
quantile(mydata, (0:10)/10)                # 10% 단위로 구간을 나누어 계산
summary(mydata)
```

```
> mydata <- c(60, 62, 64, 65, 68, 69, 120)
> quantile(mydata)
   0%   25%   50%   75%  100%
 60.0  63.0  65.0  68.5 120.0
> quantile(mydata, (0:10)/10)          # 10% 단위로 구간을 나누어 계산
   0%   10%   20%   30%   40%   50%   60%   70%   80%   90%  100%
 60.0  61.2  62.4  63.6  64.4  65.0  66.8  68.2  68.8  89.4 120.0
> summary(mydata)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  60.00   63.00   65.00   72.57   68.50  120.00
```

3. 산포

- 산포(distribution)란 주어진 자료에 있는 값들이 퍼져 있는 정도(흩어져 있는 정도)
- 산포는 수학시간에 배운 분산과 표준편차를 가지고 파악
- 분산 (variance)

$$S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2$$

- 표준편차 (standard deviation)

$$s = \sqrt{s^2}$$

- 자료의 분산과 표준편차가 작다는 의미는 자료의 관측값들이 평균값 부근에 모여 있다는 뜻

3. 단일변수 연속형 자료의 탐색

코드 5-7

```
mydata <- c(60, 62, 64, 65, 68, 69, 120)
var(mydata)                          # 분산
sd(mydata)                           # 표준편차
range(mydata)                        # 값의 범위
diff(range(mydata))                  # 최댓값, 최솟값의 차이
```

```
> var(mydata)                        # 분산
[1] 447.2857
> sd(mydata)                         # 표준편차
[1] 21.14913
> range(mydata)                      # 값의 범위
[1] 60 120
> diff(range(mydata))                # 최댓값, 최솟값의 차이
[1] 60
```

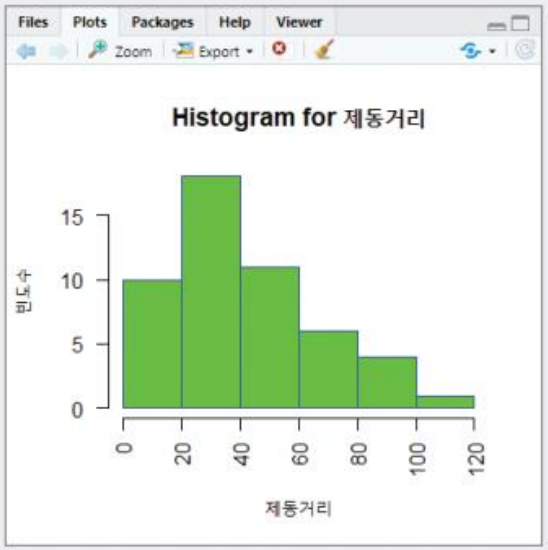
4. 히스토그램

- 히스토그램(histogram)은 외관상 막대그래프와 비슷한 그래프로, 연속형 자료의 분포를 시각화할 때 사용
- 막대그래프를 그리려면 값의 종류별로 개수를 셀 수 이어야 하는데, 키와 몸무게 등의 자료는 값의 종류라는 개념이 없어서 종류별로 개수를 셀 수 없음
- 대신에 연속형 자료에서는 구간을 나누고 구간에 속하는 값들의 개수를 세는 방법을 사용

코드 5-8

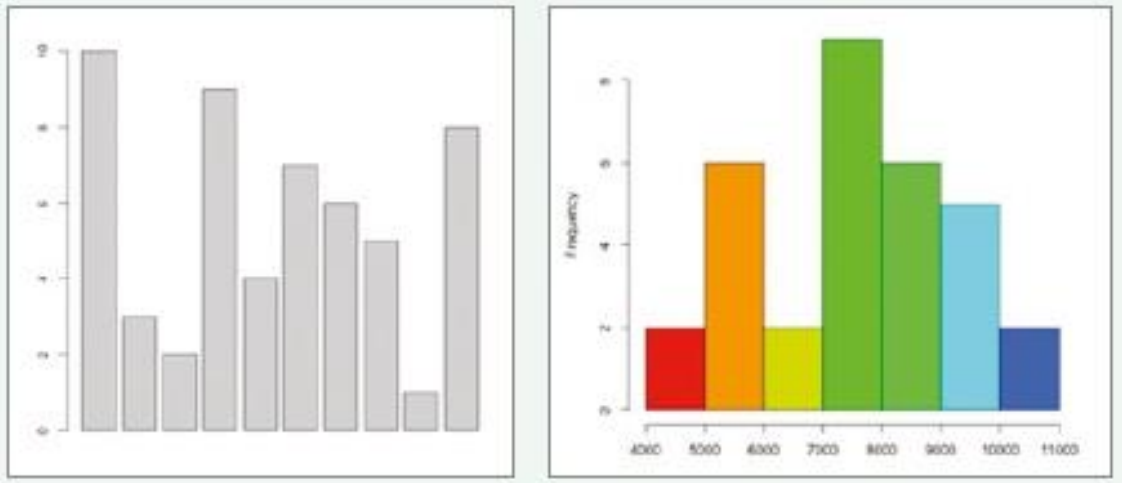
```
dist <- cars[,2]                                  # 자동차 제동거리
hist(dist,                                        # 자료(data)
        main="Histogram for 제동거리",            # 제목
        xlab ="제동거리",                         # x축 레이블
        ylab="빈도수",                            # y축 레이블
        border="blue",                            # 막대 테두리색
        col="green",                              # 막대 색
        las=2,                                    # x축 글씨 방향(0~3)
        breaks=5)                                 # 막대 개수 조절
```

3. 단일변수 연속형 자료의 탐색



여기서 잠깐! 막대그래프와 히스토그램 비교

- 히스토그램은 외관상 막대그래프와 유사
- 일반적으로 막대 사이에 간격 있으면 막대그래프, 간격 없이 막대들이 붙어 있으면 히스토그램
- 막대그래프에서는 막대의 면적이 의미가 없지만 히스토그램에서는 막대의 면적도 의미가 있음



(a) 막대그래프 (b) 히스토그램

그림 5-8 막대그래프와 히스토그램

5. 상자그림

- 상자그림(box plot)은 상자 수염 그림(box and whisker plot)으로도 부르며, 사분위수를 시각화하여 그래프 형태로 나타낸 것
- 하나의 그래프로 데이터의 분포 형태를 포함한 다양한 정보를 전달하기 때문에 단일변수 수치형 자료를 파악하는 데 자주 사용

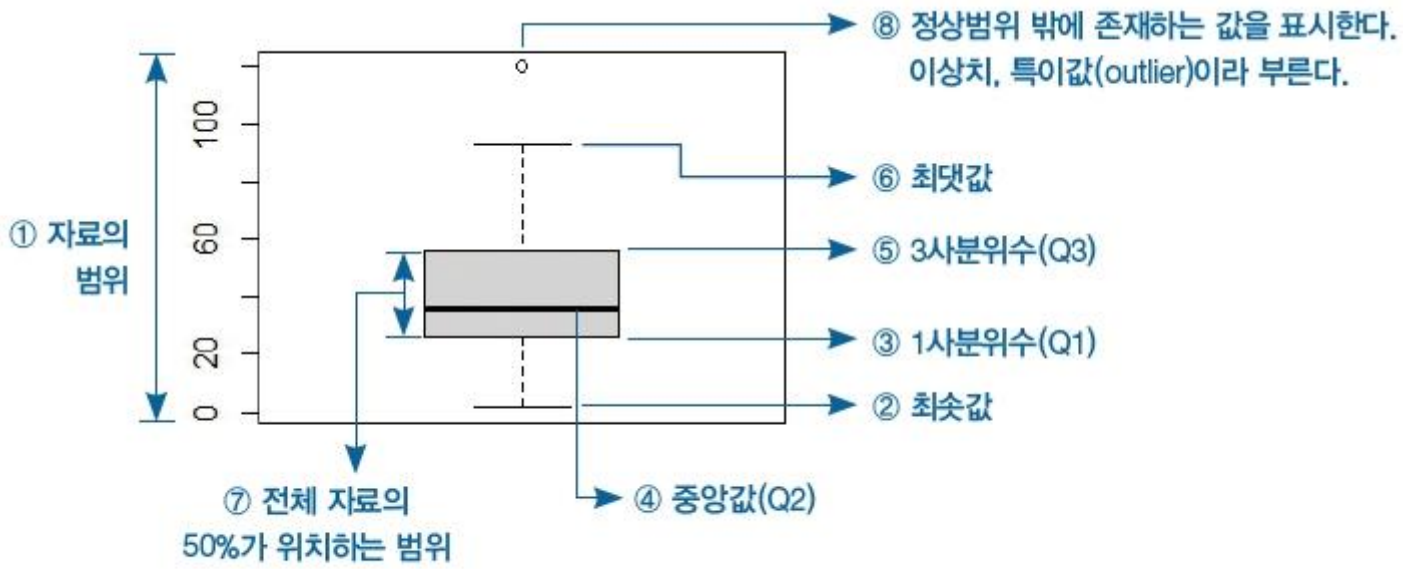
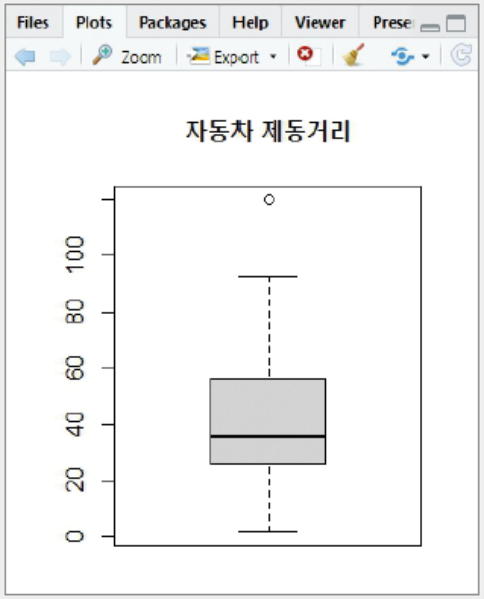


그림 5-9 상자그림의 구성 요소

코드 5-9

```
dist <- cars[,2]                                    # 자동차 제동거리(단위: 피트)
boxplot(dist, main="자동차 제동거리")
```

```
> dist <- cars[,2]                    # 자동차 제동거리(단위: 피트)
> boxplot(dist, main="자동차 제동거리")
```



코드 5-10 상자그림에 사용된 통계값 확인

```
boxplot.stats(dist)
```

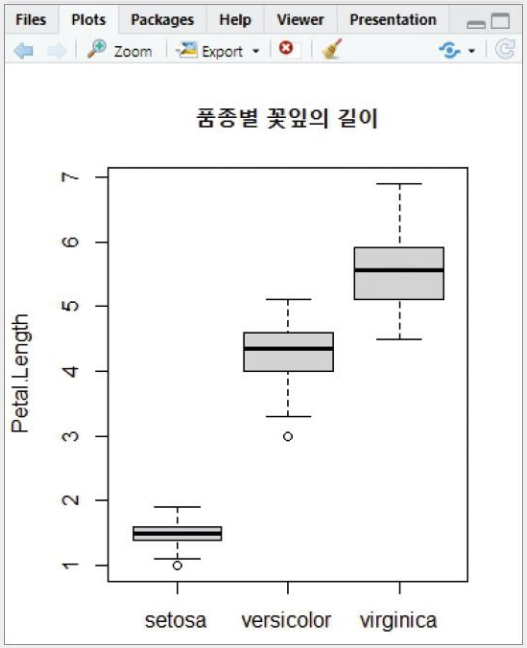
```
> boxplot.stats(dist)
$stats
[1]  2 26 36 56 93

$n
[1] 50

$conf
[1] 29.29663 42.70337

$out
[1] 120
```

6. 그룹이 있는 자료의 상자그림

코드 5-11

```
boxplot(Petal.Length~Species, data=iris, main="품종별 꽃잎의 길이")
```

```
> boxplot(Petal.Length~Species, data=iris, main="품종별 꽃잎의 길이")
```



여기서 잠깐! 한 화면에 그래프 여러 개 출력하기

```
par(mfrow=c(1,3))              # 1x3 가상화면 분할
barplot(table(mtcars$carb),
        main="Barplot of Carburetors",
        xlab="#of carburetors",
        ylab="frequency",
        col="blue")
barplot(table(mtcars$cyl),
        main="Barplot of Cylender",
        xlab="#of cylender",
        ylab="frequency",
        col="red")
barplot(table(mtcars$gear),

        main="Barplot of Grar",
        xlab="#of gears",
        ylab="frequency",
        col="green")
par(mfrow=c(1,1))              # 가상화면 분할 해제
```

여기서 잠깐! 한 화면에 그래프 여러 개 출력하기

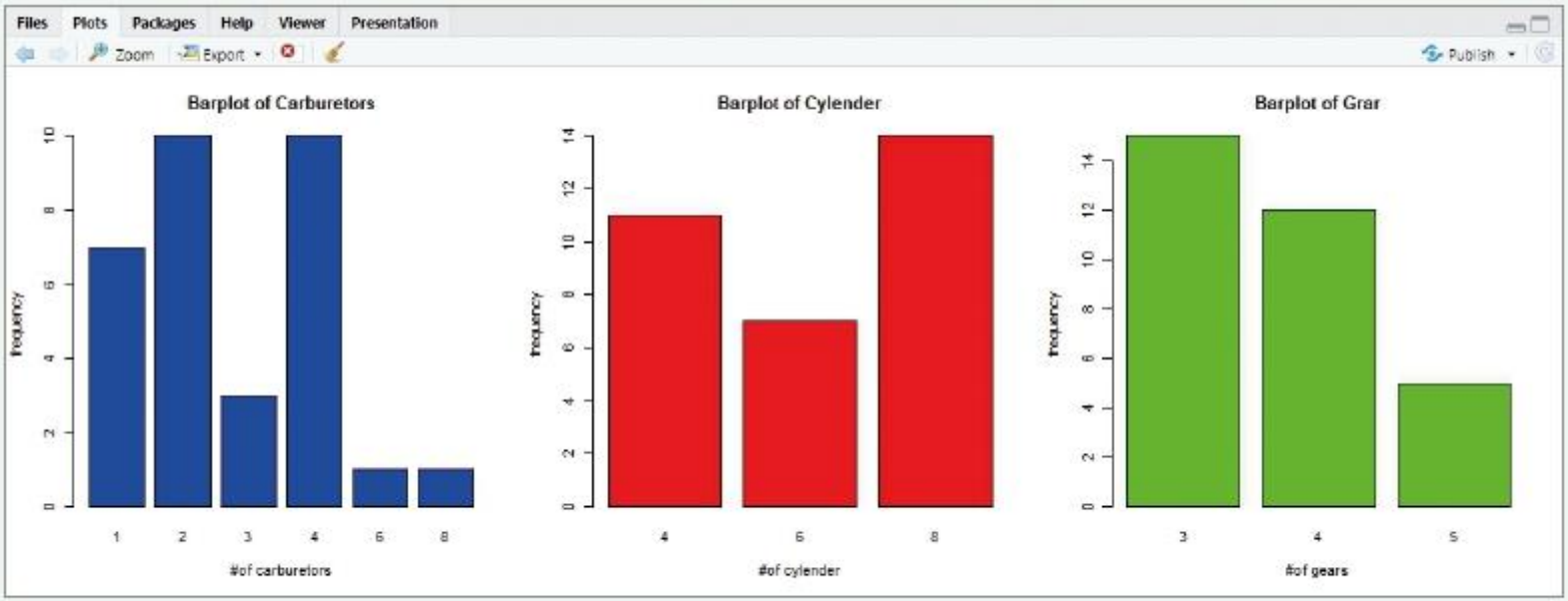


그림 5-11 한 화면에 여러 개의 그래프 출력

Thank you!

한빛아카데미 Hanbit Academy, Inc.