

강의교안 이용 안내

- 본 강의교안의 저작권은 이윤환과 한빛아카데미(주)에 있습니다.
- 이 자료를 무단으로 전제하거나 배포할 경우 저작권법 136조에 의거하여 벌금에 처할 수 있고 이를 병과(併科)할 수도 있습니다.





제대로 알고 쓰는
R 통계분석

CHAPTER 02

기술통계학

Contents

2.1

그래프

- 기술통계학의 개요
- 그래프의 개요
- 산점도
- 막대그래프와 히스토그램
- 원 도표

2.2

자료의 특성 : 모수와 통계량

- 최댓값과 최솟값
- 최빈값
- 평균과 중앙값
- 표준편차와 사분위수 범위

3장을 위한 준비



01. 그래프

: 자료의 모양을 그림으로 표현하기

1. 통계학의 두 분야인 기술통계학과 추측통계학의 개념을 이해한다.
2. 기술통계학에서 자료를 시각적으로 표현하는 방법인 그래프에 대해 학습한다.
3. 그래프의 종류와 표현법을 살펴본다.

기술통계학의 개요

• [표 1-1]의 자료

- ▣ 468,284 명으로 부터 5개 (ID 제외)의 변수 측정
- ▣ 자료의 의미를 어떻게 파악할 것인지?

[표 1-1] 2010년 인구주택총조사 중 일부

ID	성별	나이	가구주와의 관계	학력	출생아 수
1	여자	68	가구주의 배우자	초등학교	3
2	여자	29	가구주의 배우자	초등학교	0
3	여자	7	자녀	초등학교	결측
4	여자	3	자녀	안 받았음	결측
5	남자	26	자녀	중학교	결측
6	여자	52	가구주의 배우자	초등학교	2
7	여자	62	가구주의 배우자	고등학교	1
8	여자	10	자녀	초등학교	결측
9	남자	58	가구주	중학교	결측

기술통계학의 개요

- 기술통계학

- 수집한 자료들을 정리하고 요약하여 자료가 어떤 특성을 갖고 있는지 해석하는 통계학의 한 분야
- 자료 요약 방법
 - 시각적인 방법(그래프)을 통한 자료의 요약
 - 각종 통계 숫자를 이용한 자료의 요약

- 추측통계학

- 모집단의 특성을 알고자 과학적인 방법으로 표본의 특성을 통해 모집단의 특성을 추측하는 통계학의 한 분야

그래프의 개요

• 개요

- 자료의 크기가 크더라도 전체 자료의 모양을 한 눈에 파악하기 쉽습니다.
- 컴퓨팅 환경의 발달로 그래프 작성이 비교적 쉬워졌습니다.
- 자료의 유형과 밝히고자 하는 내용에 따라 다양한 그래프가 있습니다.

• 예시 : 남녀의 성비

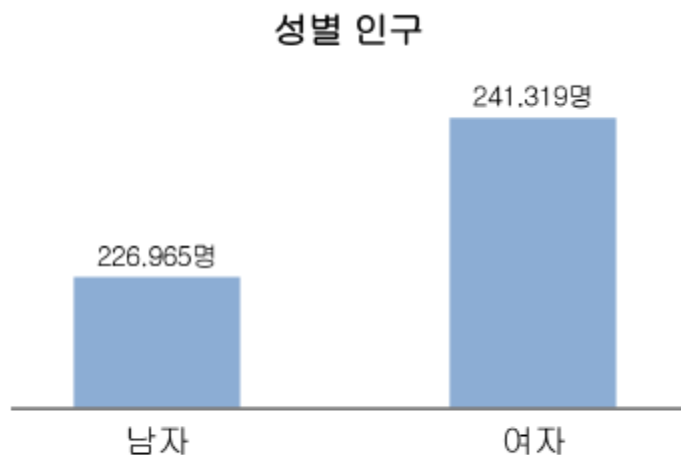
- 통계청 ‘마이크로데이터 통합서비스포털’에서 제공받은 [표1-1]의 자료 중 남자와 여자의 인구수와 이에 따른 비율을 다음과 같이 요약하였습니다.

[표 2-1] 성별 현황

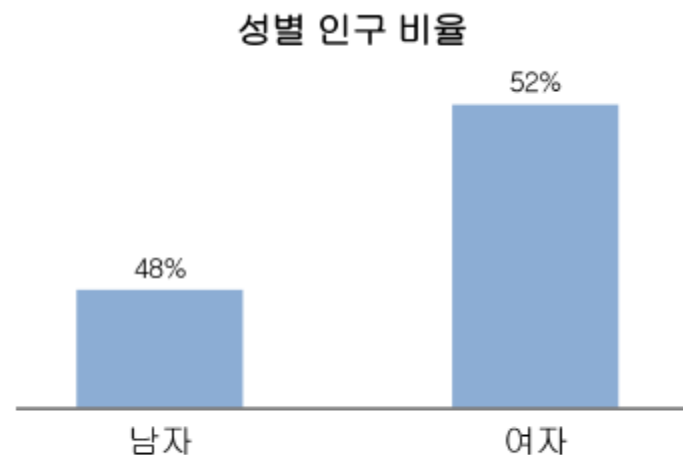
구분	조사자 수	비율
남자	226,965명	48%
여자	241,319명	52%

그래프의 개요

- 다음과 같이 ‘막대그래프’로 남녀 인구수에 대한 그래프와, 남녀 비율에 대한 그래프를 작성해 보았습니다.



[그림 2-1] 남녀 인구수에 대한 막대그래프

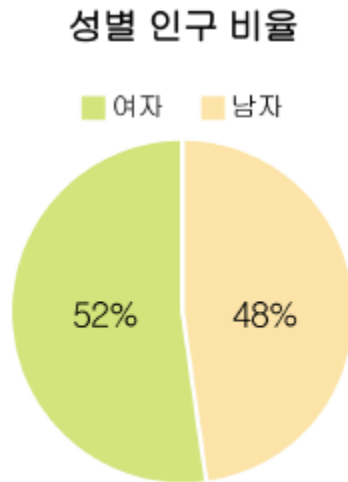


[그림 2-2] 남녀 비율에 대한 막대그래프

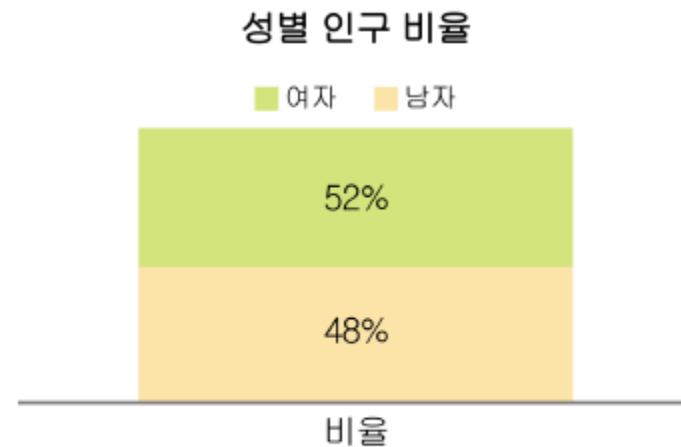
- 두 그래프 모두 여자의 인구수가 많고 마찬가지로 여자의 비율이 전체 인구중에 더 많음을 명확히 보여주고 있습니다.
- 두 그래프는 인구수와 비율의 차이를 강조하고 있습니다.

그래프의 개요

- ▣ 비율을 나타내는 그래프의 경우 전체에서 차지하는 비중을 나타내려면 다음과 같은 그래프 들을 작성해 볼 수 있습니다.



[그림 2-3] 원 도표로 작성한 성별 인구 비율



[그림 2-4] 누적 막대그래프로 작성한 성별 인구 비율

• 그래프의 사용

- ▣ 나타내고자 하는 목적과 자료의 특성에 맞는 그래프를 사용합시다.
- ▣ R을 이용하여 목적과 자료의 특성에 맞는 몇 가지 그래프를 작성해 봅시다.

산점도

- 산점도 : 가장 기본이 되는 그래프

- x축과 y축으로 구성된 좌표계 위에 이차원(양적변수 두 개) 자료를 점으로 표현하여 두 변수 간의 관계를 나타내는 데 사용하는 그래프입니다.
 - 기본 목적외에 다양하게 응용하여 사용할 수 있습니다.
- R에서의 함수 : `plot(x, y, ...)`
 - `x, y` : x축과 y축에 그릴 자료(벡터변수)
 - `main` : 제목 (문자열)
 - `xlim, ylim` : 각 축별 표시 영역 (벡터로 된 수치자료)
 - `xlab, ylab` : x축과 y축의 제목 (문자열)
 - `type` : 산점도 표시 유형 (기본값은 'p'이며, 'l', 'b', 'b', 'o', 'h', 's', 'S', 'n' 등이 있음)
 - `pch` : 표시할 점의 유형 (0부터 25까지의 숫자와 키보드 상의 글자 하나)

산점도

예제 2-1 두 변수 간의 관계를 나타내는 산점도

준비파일 | 01.plot.R

- 자동차와 속도에 대한 산점도 작성

- 1920년대에 수집한 cars는 50대의 차량으로부터 speed와 dist 두 변수를 측정한 자료로 speed 변수는 차량의 속도(mph)를, dist 변수는 제동거리(ft)를 나타냅니다.
 - 각 벡터에 접근하는 방법은 cars\$speed, cars\$dist 입니다.

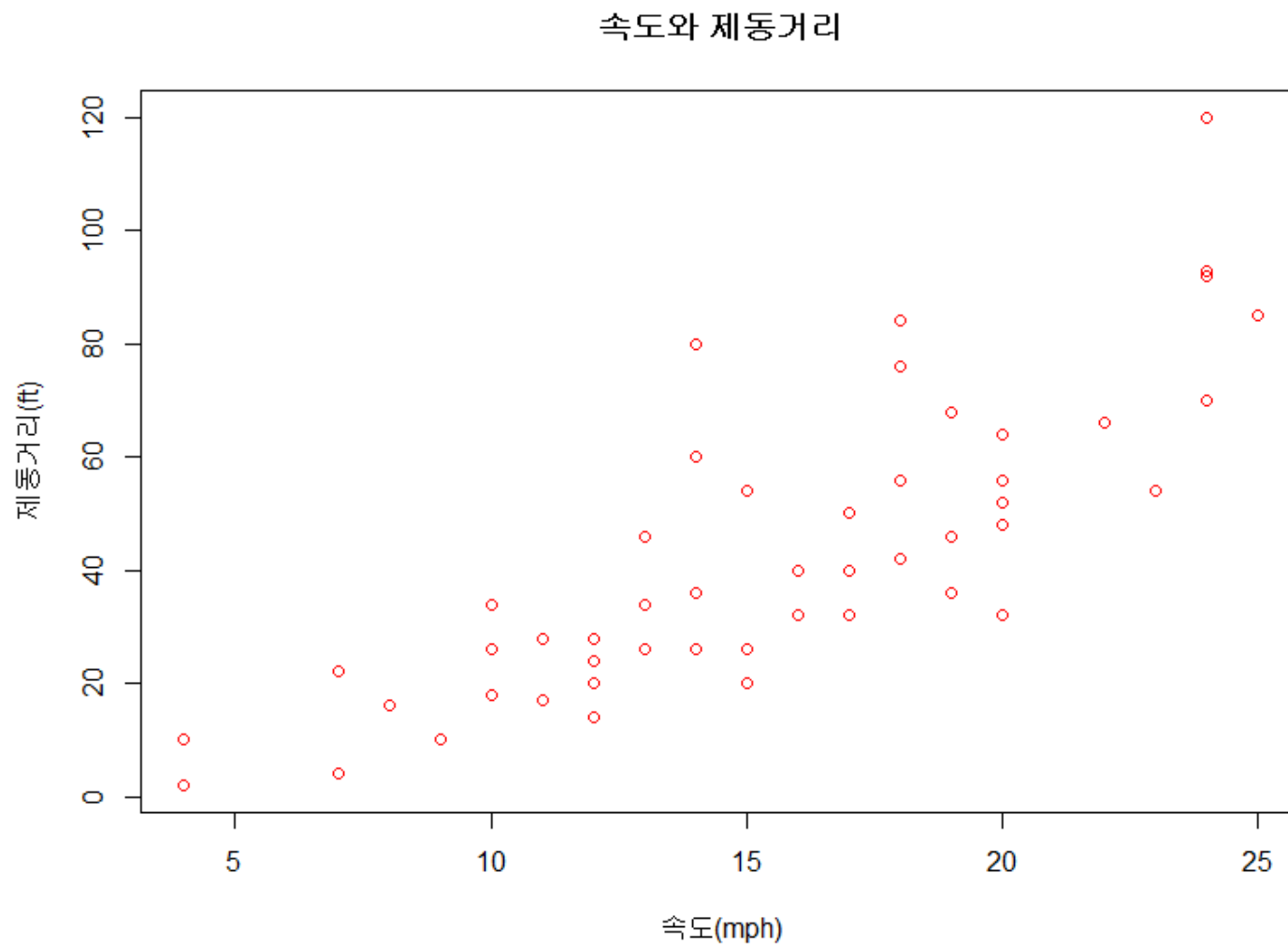
```
2: plot( cars$speed, cars$dist,  
        main="속도와 제동거리",  
        xlab="속도(mph)", ylab="제동거리(ft)",  
        pch=1, col="red" )
```

산점도

- 코드 설명

- ▣ `cars$speed, cars$dist,`
 - x축 좌표값으로 사용할 변수는 `cars$speed`, y축 좌표값으로 사용할 변수는 `cars$dist` 입니다.
- ▣ `main="속도와 제동거리",`
 - 산점도의 제목을 “속도와 제동거리”로 합니다.
- ▣ `xlab="속도(mph)", ylab="제동거리(ft)",`
 - x축의 제목으로 “속도(mph)”, y축의 제목으로 “제동거리(ft)”로 합니다.
- ▣ `pch=1, col="red"`
 - ○ (`pch=1`)이고, 점의 색상은 붉은색이 되도록 합니다

산점도



산점도

- **plot() 을 이용하여 시간의 흐름에 따라 값이 변하는 시계열 자료의 그래프 작성**
 - R의 내장자료인 Nile 은 1871년부터 1970년까지 연도별 나일강의 유량을 기록하고 있는 시계열 자료입니다.
 - 시계열 자료는 시간의 변화에 따라 값을 측정한 자료를 기록한 자료로 주가, 기온 등이 대표적이며, 보통 각 점들을 연결한 그래프를 작성합니다.
 - 시계열 자료를 그래프로 작성시 기본적으로 x축은 관찰 시점의 시간이, y축은 관찰값을 배치합니다.

```
6: plot( Nile, main="Nile강의 연도별 유량 변화",  
        xlab="연도", ylab="유량")  
7: plot( Nile, type="p", main="Nile강의 연도별 유량 변화",  
        xlab="연도", ylab="유량")
```

산점도

- 코드 설명

- Nile

- 하나의 시계열 자료를 plot() 함수에 전달하면, 시계열 자료로부터 시간정보와 관찰값을 읽어와 시간을 x축으로 그 시점에 관찰된 유량값을 y축으로 하는 산점도를 그립니다.

- main="Nile강의 연도별 유량 변화",

- 산점도의 제목을 “Nile강의 연도별 유량 변화”로 합니다.

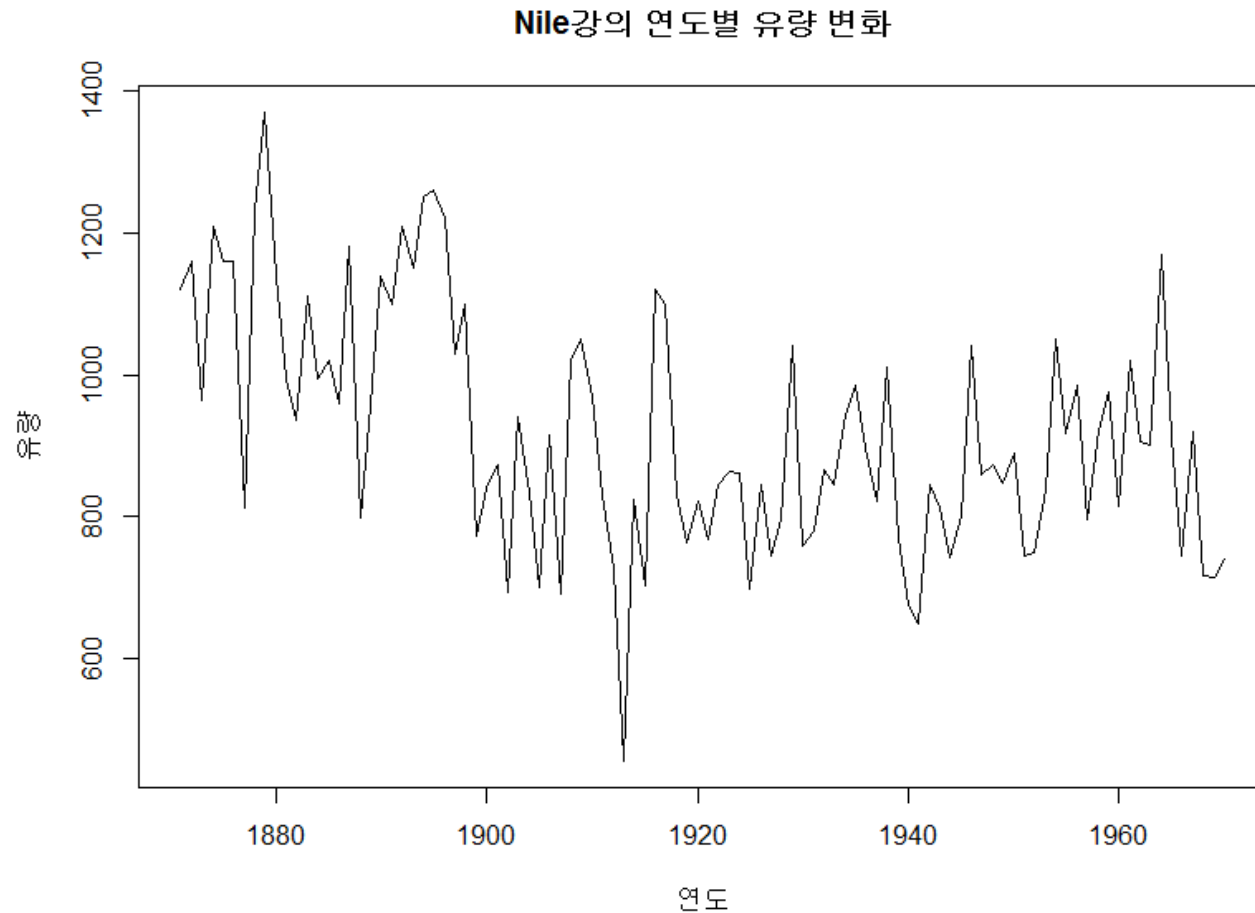
- xlab="연도", ylab="유량",

- x축의 제목으로 “연도”, y축의 제목으로 “유량”으로 합니다.

- 7줄의 type="p“

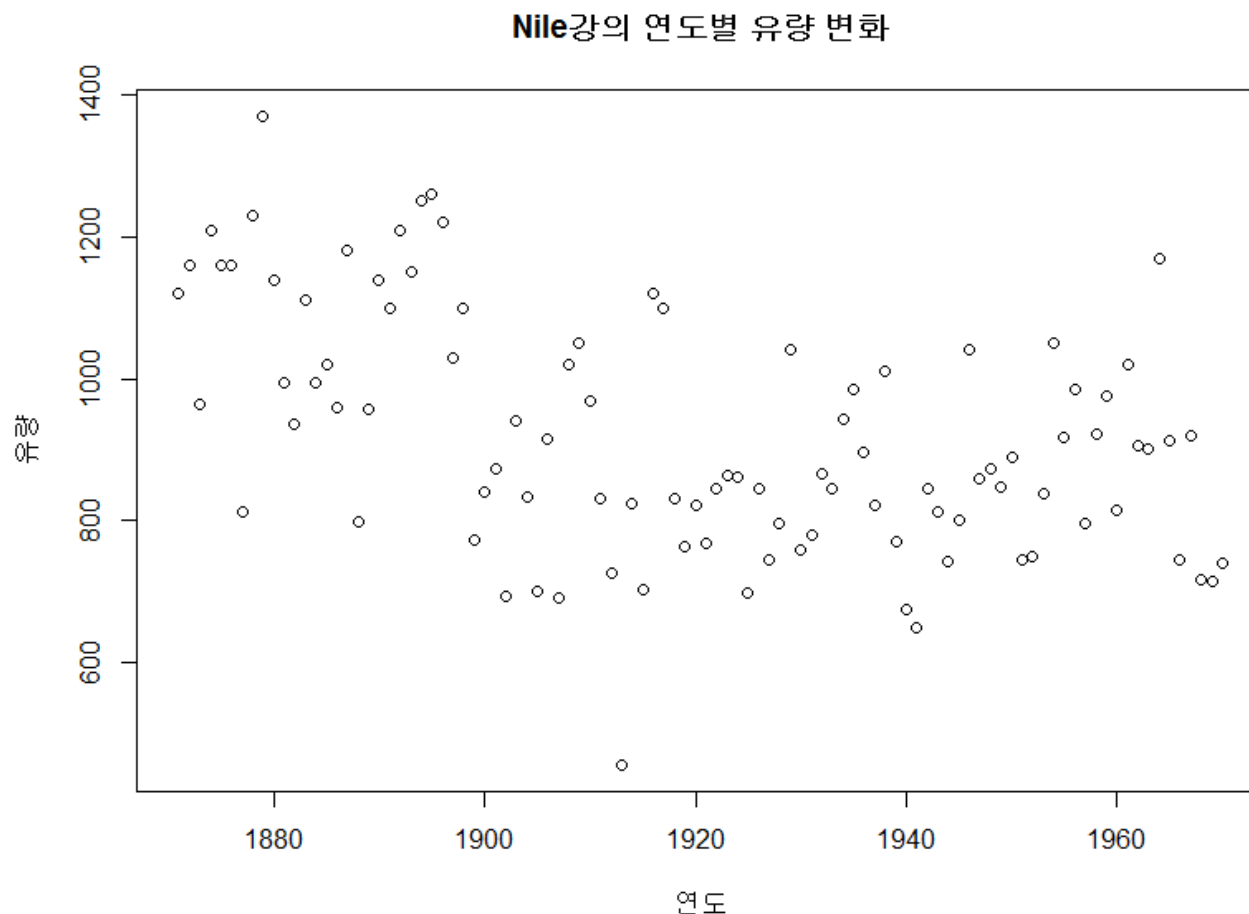
- 산점도로 표시하는 형태를 점(point)으로 합니다.
 - 6줄에서 type을 지정하지 않았을 때와 비교해 봅시다.

산점도



각 점들을 선으로 연결하여 변화되는 추이를 잘 파악할 수 있도록 합니다.

산점도



변화되는 추이를 한 눈에 파악하기 힘들고 전체 모양만 감소하는 것으로 확인할 수 있습니다. 시간에 따른 변화는 선으로 연결하는 것이 변화를 파악하는데 더 용이합니다.

막대그래프와 히스토그램

- 막대그래프와 히스토그램

- 두 그래프 모두 기둥의 형태지만, 각 그래프에서 의미를 찾는 것은 자료의 종류에 따라 조금 다릅니다.
- 막대그래프는 이산형 혹은 질적 자료의 개수를 나타내는 데 사용합니다.
 - 주 관심사는 기둥의 높이입니다.
 - R에서 막대그래프는 자료로부터 바로 구하지 않고 (일반적으로 `table()` 함수 등을 이용한) 요약한 자료를 사용합니다.
 - `table()` 함수는 이산형 자료등에서 각 값의 빈도를 계산하여 표의 형태로 반환해 줍니다. (자세한 내용은 7장의 '8장을 위한 자료준비'에서 학습합니다.)
- 히스토그램은 연속형 자료의 개수 혹은 비율을 나타내는 데 사용합니다.
 - 주 관심사는 기둥의 넓이입니다.

막대그래프와 히스토그램

예제 2-2 막대그래프와 히스토그램

준비파일 | 02.barplot_histogram.R

- 출생아 수 [표 1-1]

- (남자)출생아 수를 기록한 이산형 자료입니다.
- 출생아 수별 막대그래프를 작성해 봅시다.
 - 막대그래프 작성에 앞서 각 출생아 수별로 자료를 요약합니다. (table() 함수 이용)

```
1: load("data.rda")
2: tableV5 <- table(data$V5)
3: tableV5
4: barplot( tableV5, main="출생아(남자)별 빈도",
            xlab="출생아수", ylab="빈도" )
```

막대그래프와 히스토그램

- 코드 설명

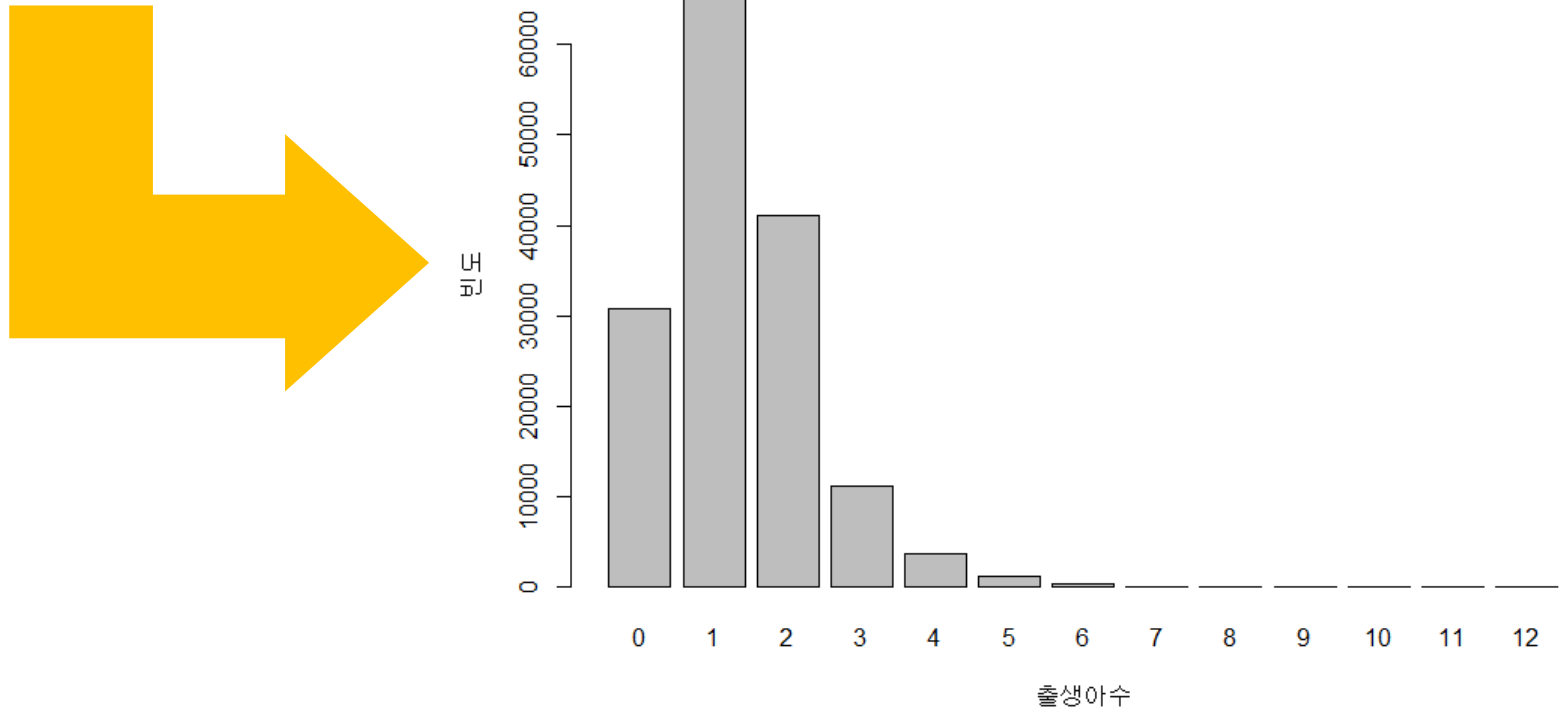
- 1줄 : `load("data.rda")`
 - 1장에서 저장한 자료를 불러옵니다.
- 2줄 : `tableV5 <- table(data$V5)`
 - 1장에서 자료 저장시 `data`란 이름으로 저장하였고, 출생아 수를 기록한 변수의 이름은 `V5`입니다.
 - `table()` 함수를 이용하여 출생아 수별 빈도 테이블을 만들어 변수 `tableV5`에 저장합니다.
- `barplot()`
 - `tableV5`
 - 2줄에서 만든 `tableV5`를 이용하여 x축에 각 출생아 수를 출생아 수별 빈도를 높이로 하는 막대그래프를 작성합니다.
 - `main`, `xlab`, `ylab` 등은 거의 모든 그래프에서 동일하게 사용합니다.

막대그래프와 히스토그램

> tableV5

0	1	2	3	4	5	6
30788	69624	41010	11165	3667	1228	346
7	8	9	10	11	12	
104	21	8	4	10	1	

출생아(남자)별 빈도



막대그래프와 히스토그램

• 히스토그램

- 연속형 자료인 나이를 히스토그램으로 그려봅시다.
- data 자료에서 나이는 V2 변수에 저장되어 있습니다.
- 히스토그램을 그릴 구간을 변경해 봅시다.

```
8: hist( data$V2, main="연령별 분포", xlab="연령", ylab="빈도" )
9: hist( data$V2, breaks=c(seq(0, 90, 10)), right=F,
      main="연령별 분포", xlab="연령", ylab="빈도")
추가 : hist( data$V2, probability=T,
      main="연령별 분포", xlab="연령", ylab="밀도" )
```

막대그래프와 히스토그램

- 코드 설명

- 6줄 : `data$V2`

- `barplot()`과 다르게 히스토그램을 작성할 자료를 직접 넣어줍니다.
- R이 자료로부터 각 막대를 그릴 계급구간을 결정하여 히스토그램을 그립니다.

- 7줄 : `breaks=c(seq(0, 90, 10))`

- `breaks` 전달인자에 사용자가 지정하는 구간을 넣어 구간을 결정합니다.
- 예에서는 0부터 90까지 10씩 증가하는 벡터로 구간을 넣어 0, 10, 20, ... 90이 계급구간이 되도록 하였습니다.

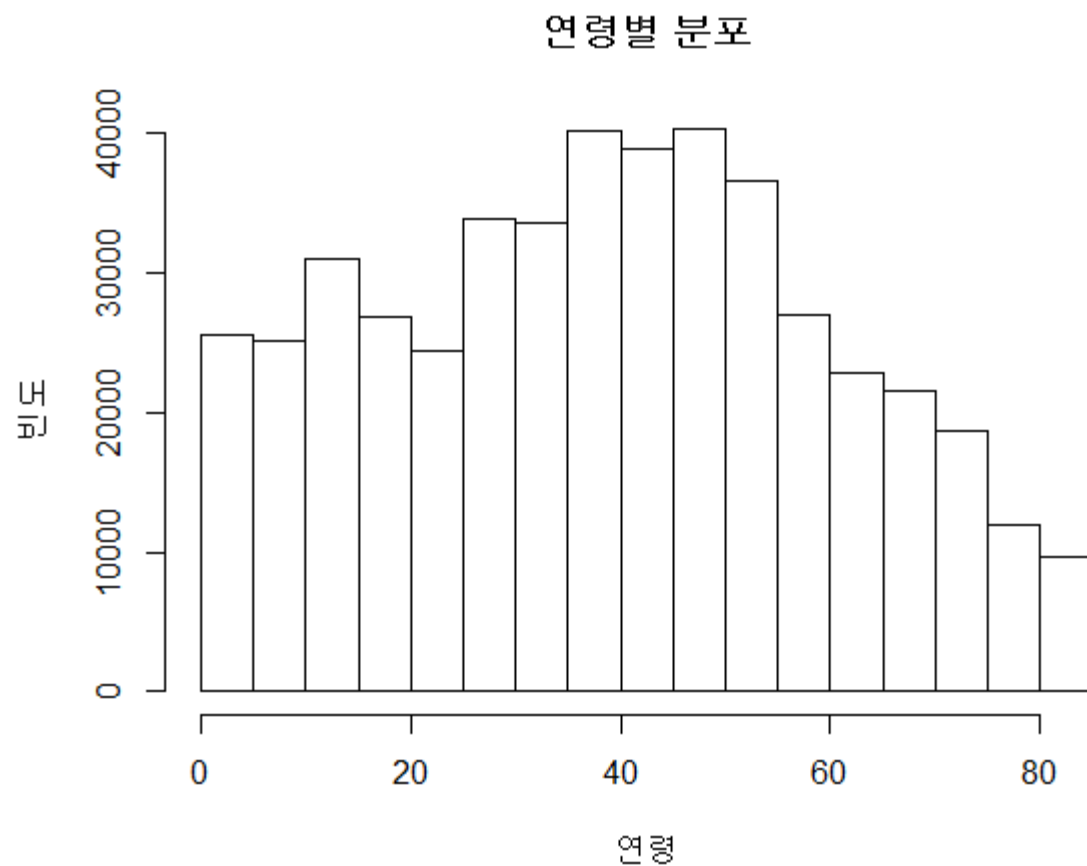
- 7줄 : `right=F`

- 각 계급의 구간을 계급의 시작점 이상, 끝점 미만이 되도록 합니다. 즉, 위의 `breaks`로 만든 구간을 $[0, 10)$, $[10, 20)$, ... $[80, 90)$ 와 같이 되도록 합니다.

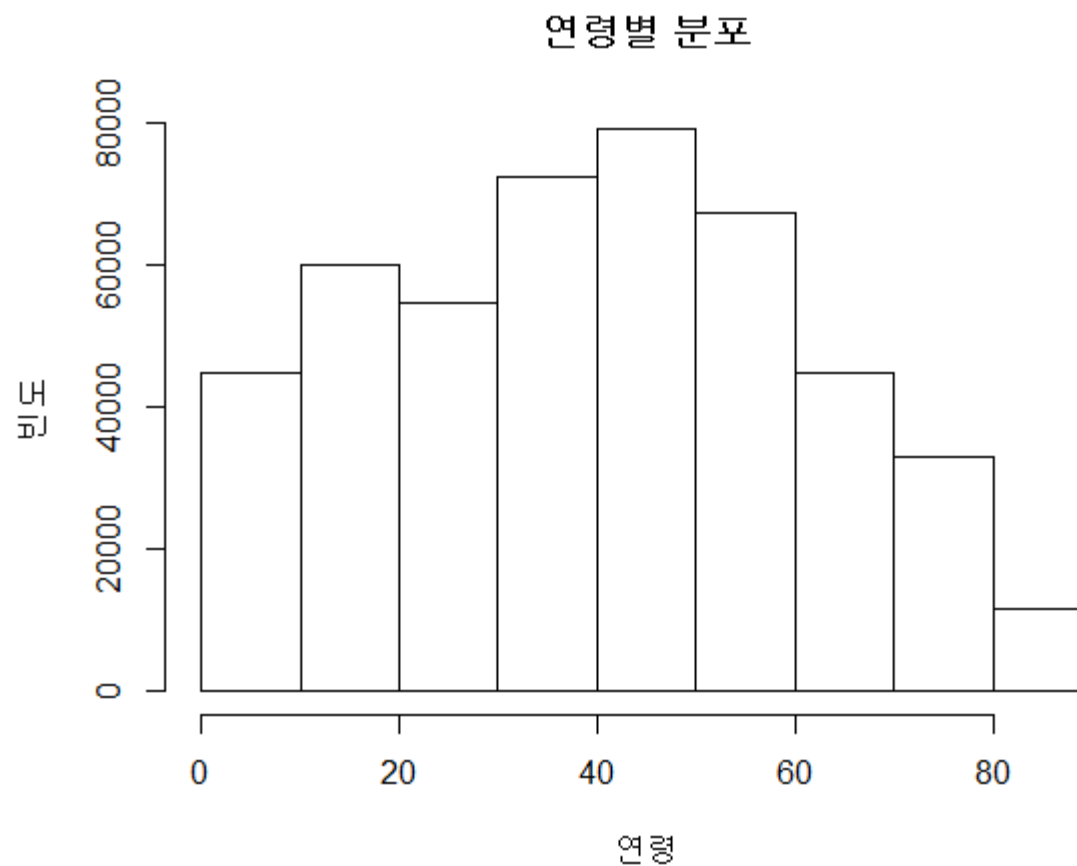
- 추가 : `probability=T`

- 히스토그램을 빈도가 아닌 전체에서의 비중(밀도)가 되도록 합니다. (상대도수)
- `freq = F` 와 동일합니다.

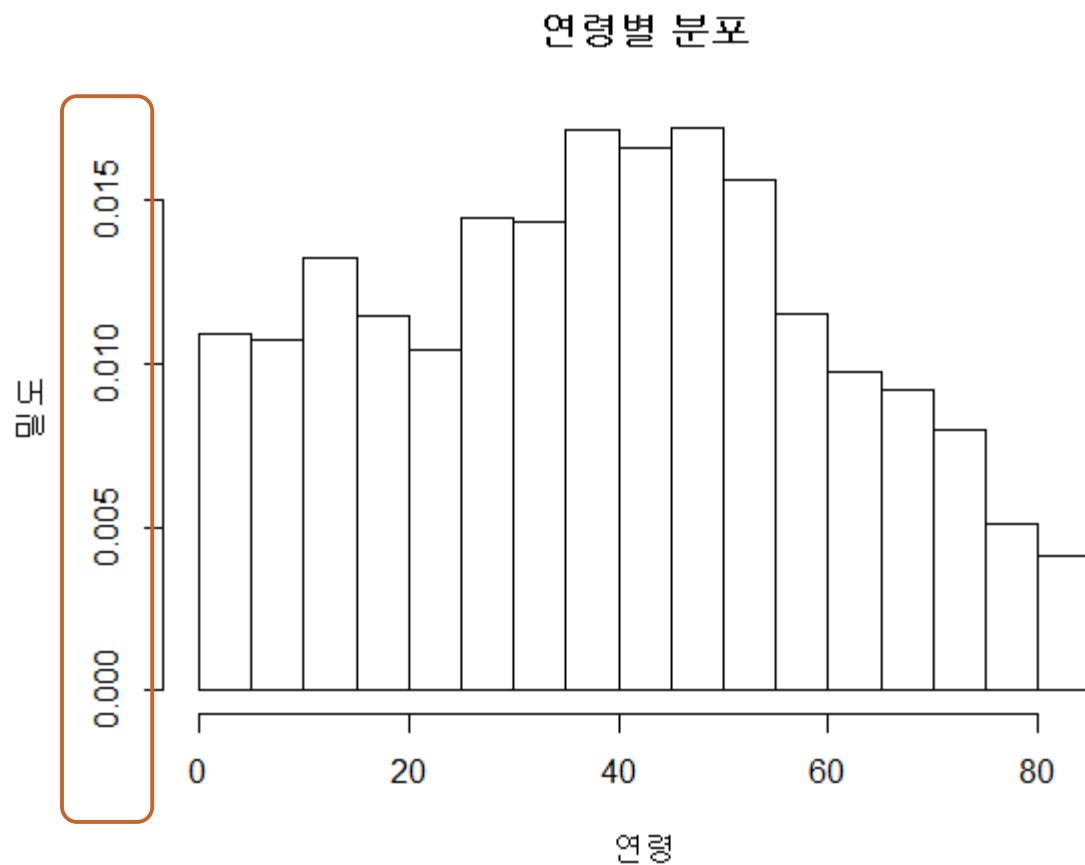
막대그래프와 히스토그램



막대그래프와 히스토그램



막대그래프와 히스토그램



원 도표

예제 2-3 원 도표

준비파일 | 04.pie.R

- **질적자료에서 각 범주가 차지하는 비중 비교**
 - 특히, 가장 큰 비중을 차지하는 조각이 눈에 잘 들어옵니다.
 - 피자를 생각해 보면, 8개로 나눈 조각 중 가장 큰 조각이 눈에 잘 들어옵니다.
 - 나이팅게일의 장미도표 또한 원도표의 일종이라고 볼 수 있습니다.
- **학력수준별 비중을 비교하는 원도표 작성**
 - 학력수준은 조사에서 8개로 되어 있으며, 명목형 자료입니다.
 - 각 수준별 크기를 비교하여 어떤 학력이 가장 많은 비중을 차지하는지 확인해 봅시다.
 - 막대그래프를 이용하여 크기 순으로 나열할 수 있지만, 이 때는 단순히 크기 비교용으로 전체에서 얼마큼을 차지하는지 한 눈에 알기 힘듭니다.

원 도표

```
2: table.V4 <- table(data$V4)
3: table.V4
4: pie( table.V4, main="학력수준별 비중", cex=0.8)
```

- 코드 설명

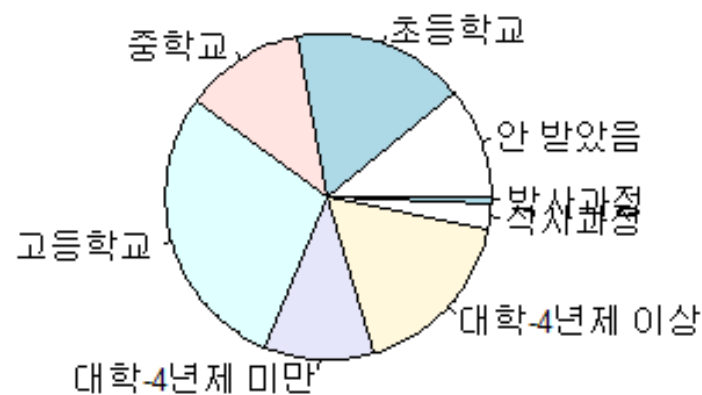
- 2줄 : barplot()과 마찬가지로 table() 함수를 이용하여 자료를 요약한 다음에 사용합니다.
- 4줄 : 2줄에서 만든 table.V4를 pie() 함수에 그래프로 그릴 자료로 전달합니다.
- 4줄 : cex=0.8 (PT 에서 추가)
 - 각 조각별 이름표의 크기를 기준 크기의 0.8배로 축소합니다.


원 도표

> table.V4

안 받았음	초등학교	중학교	고등학교
51085	80710	55704	134246
대학-4년제 미만	대학-4년제 이상	석사과정	박사과정
50753	81110	11741	2935

학력수준별 비중





02. 모수와 통계량

: 숫자를 이용한 자료의 특성 묘사

1. 숫자를 이용하여 자료를 요약하고, 요약된 정보를 바탕으로 자료의 모양을 유추하는 방법에 대해 학습한다.
2. 기본적인 자료의 특성을 이해하고, 각각 R을 통해 구해본다.

모수와 통계량 : 숫자를 이용한 자료의 요약(특성 묘사)

• 개요

- 그래프를 통해 전체 자료의 모양을 확인하고, 어떤 특성이 있는지 확인하였습니다.
- 그래프가 전체 모양을 살피는 과정이라면, 숫자를 이용하는 과정은 전체 자료 속에 숨겨져 있는 특성들을 숫자를 이용하여 밝히는 과정입니다.

• 예제

- 라니의 카페에서는 평소 하루에 50잔 정도의 커피 재료를 준비하였습니다.
- 개업을 하고 얼마간의 시간이 지나니 재고가 점점 쌓이는 것을 발견하였습니다.
- 커피 판매 자료를 통해 판매량의 특성들을 살펴봅시다.

모수와 통계량 : 숫자를 이용한 자료의 요약(특성 묘사)

• 커피 판매량 자료

[표 2-2] '라니의 카페' 개업 후 커피 판매량

41	33	34	27	20	23	32	31	30	27
30	27	26	24	18	22	21	28	23	31
29	48	25	31	25	35	33	35	16	24
20	11	21	8	8	4	4	3	5	6
4	13	4	16	14	10	11			

- ▣ 개업 후 47일간의 커피 판매량입니다.(자료의 크기 $n=47$)
- ▣ 실제로는 요일, 날씨 등 판매량에 영향을 끼치는 요인들이 많지만 이를 배제하고 판매량의 특징들을 살펴봅시다.
- ▣ 먼저, 막대그래프로 판매량 별 횟수를 그려보았습니다.

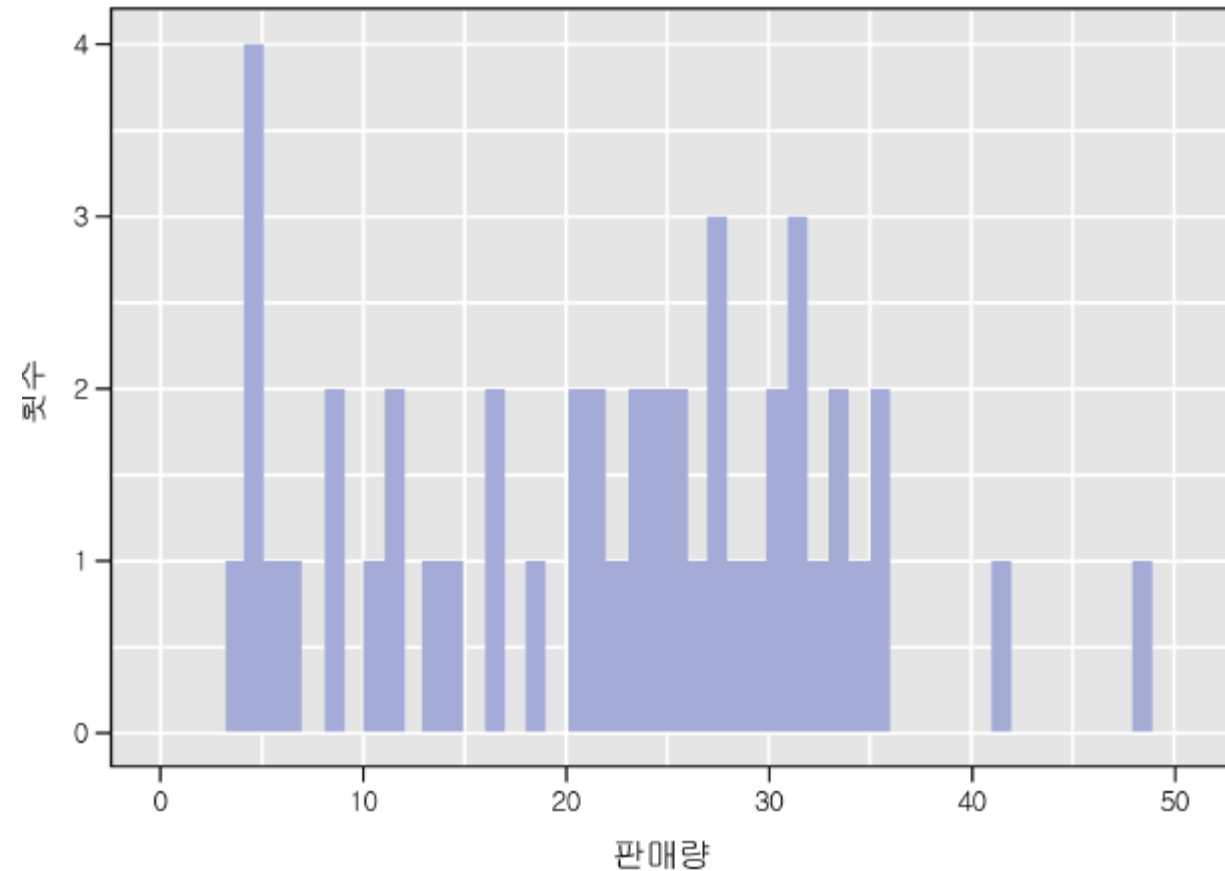
Dataset: "Student-run Cafe Business Data" submitted by Concetta A. DePaolo, Scott College of Business Indiana State University. Dataset obtained from the Journal of Statistics Education

(<http://www.amstat.org/publications/jse>). Accessed 2015-10-06,

관련 논문 : <http://www.amstat.org/publications/jse/v19n1/depaolo.pdf>

모수와 통계량 : 숫자를 이용한 자료의 요약(특성 묘사)

라니의 카페 커피 판매량



- 다음을 관찰해 봅시다
 - 10잔 이하로 판매된 날
 - 40잔 이상 판매된 날
 - 가운데 자료들이 많이 몰려있는 판매량 구간
- 눈으로 확인할 수 없는 자료가 갖고 있는 특성들을 찾아 봅시다.

모수와 통계량 : 숫자를 이용한 자료의 요약(특성 묘사)

• 찾고자 하는 자료들의 특성

- ▣ 자료 중 가장 큰 값과 가장 작은 값 : 최댓값과 최솟값
 - 이를 알면 최소 주문량과 최대 주문량이 얼마 정도일지 알 것 같습니다.
- ▣ 자료들이 가장 많이 관찰된 값 : 최빈값
 - 가장 많은 판매량을 기준으로 재료를 준비할 수도 있을 것입니다.
- ▣ 자료들의 중심 : 평균과 중앙값
 - 자료들은 값들이 모두 제 각각입니다. 이런 값들을 하나의 값으로 나타내기에는 자료들의 중심이 좋을 것 같습니다.
- ▣ 자료들의 퍼져 있는 정도 : 표준편차와 사분위수 범위
 - 위에서 자료들이 모두 제 각각이라고 하였는데, 이렇게 제 각각 흩어져 있는 정도를 나타내는 값이 있으면 변동이 얼마나 있을지 감을 잡을 수도 있습니다.

최댓값과 최솟값

예제 2-4 최댓값과 최솟값

준비파일 | 05_descstat.R

- **그간의 자료를 바탕으로 커피 주문량의 최댓값과 최솟값을 구합니다.**
 - 이를 안다면, 커피 주문량을 최솟값 이하로 혹은 최댓값 이상으로 하지는 않을 것입니다.
 - 또한 “최댓값 – 최솟값”을 통해 구하는 “범위”를 구할 수 있습니다.
 - 가장 기초적인 자료가 퍼져있는 정도로 범위를 사용할 수도 있습니다.
- **최댓값과 최솟값 구하기**
 - 최댓값과 최솟값을 자료를 순서대로 정렬하여 구합니다.
 - 자료가 적을 때는 큰 문제가 없지만, 자료가 많을 때는 컴퓨터의 도움없이 구하기 힘듭니다.
 - 최댓값과 최솟값은 수집된 자료 확인시 자료의 값이 원하는 구간 내에 있는지 확인하는 도구로 사용할 수 있습니다.

최댓값과 최솟값

```
16: sort( ranicafe$Coffees )
17: sort( ranicafe$Coffees )[1]
18: sort( ranicafe$Coffees, decreasing=TRUE )
19: sort( ranicafe$Coffees, decreasing=TRUE )[1]
```

▣ sort() 함수를 이용한 정렬

- 16줄 : sort 함수를 사용해 하루의 커피 판매량을 작은 값부터 큰 값의 순으로 정렬한 벡터를 구합니다.
- 17줄 : 16줄의 결과는 값이 작은 값부터 큰 값 순으로 정렬된 벡터이므로 이 중에서 첫 번째 원소([1])는 최솟값을 나타냅니다.
- 18줄 : sort 함수에 decreasing 전달인자로 TRUE를 전달하여 큰 값부터 작은 값 순으로 정렬한 벡터를 구합니다.
- 19줄 : 17줄과 마찬가지로 18줄과 같이 구한 결과는 값이 큰 값부터 작은 값순으로 정렬된 벡터이므로 이 중에서 첫 번째 원소는 최댓값을 나타냅니다.

최댓값과 최솟값

```
20: min( ranicafe$Coffees )
21: max( ranicafe$Coffees )
```

- R 함수 min()과 max()
 - 20줄 : 최솟값을 구하는 함수 min()을 사용하여 가장 작은 값을 가져옵니다.
 - 21줄 : 최댓값을 구하는 함수 max()를 사용하여 가장 큰 값을 가져옵니다.

[출력 2.12] min() 함수를 이용한 최솟값 출력

```
[1] 3
```

[출력 2.13] max() 함수를 이용한 최댓값 출력

```
[1] 48
```

- 커피 판매량의 최솟값은 3잔이고, 최댓값은 48잔 입니다.

최빈값

- **최빈값**

- 몇 잔의 커피가 가장 많이 판매되었는지 살펴봅시다.
- 통계에서 유용하게 사용하는 줄기-잎 그림과 함께 알아봅시다.

- **줄기-잎 그림**

- 줄기-잎 그림은 숫자로 나타내는 그래프로 자료값을 직접 써나가면서 자료의 분포를 살필 수 있습니다.
- 줄기 : 자료의 구분점으로 히스토그램에서 사용하는 구간과 같은 역할입니다.
- 잎 : 줄기로 표현되는 구간에 관찰한 각 자료들로 줄기 옆에 나열하여 표시합니다.
- 다음은 R이 자료를 보고 판단하여 줄기의 단위를 5로 한 줄기-잎 그림입니다.

최빈값

Console F:/Dropbox/StatwithR/BookSource/ ↗

```
> rc <- ranicafe$Coffees
> stem(rc)
```

The decimal point is 1 digit(s) to the right of the |

```
0 | 34444
0 | 5688
1 | 01134
1 | 668
2 | 001123344
2 | 55677789
3 | 001112334
3 | 55
4 | 1
4 | 8
```

stem() 함수를 이용하여 줄기-잎 그림 작성

• 줄기-잎 그림 읽기

- 줄기의 구간은 $[0, 5)$, $[5, 10)$, ..., $[40, 45]$, $[45, 50)$ 입니다.
- 각 줄기별로 관찰된 값을 나타냅니다.
- 줄기와 잎의 구분은 ‘|’가 합니다.
- 동일한 관찰값이 있을 시 중복하여 나타냅니다.
- 네 번째 줄기에서 줄기 값은 1이고, 잎으로 나타난 값은 668로 각각 16, 16, 18을 나타냅니다.
- 가장 많이 관찰된 값은 첫번째 줄기에 있는 4입니다.
- 줄기-잎 그림을 좌로 90도 회전하면 히스토그램과 유사한 형태로 기둥의 넓이만이 아닌 어떤 값들이 있는지 확인할 수 있습니다.

평균과 중앙값

• 평균과 중앙값

▣ 대표값

- 전체 자료를 대표하는 값입니다.
- 대표값은 중심 위치를 나타내는 특성을 사용합니다.
- 앞서 살핀 최빈값 또한 대표값 중 하나입니다.

▣ 자료의 중심은 두가지

- 무게 중심 : (산술)평균

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

- 순서상 중간 : 중앙값

$$\text{홀수일 때 : } x_{median} = x_{\left(\frac{n+1}{2}\right)}$$

- $x_{(i)}$: x 의 i 번째 순위값

$$\text{짝수일 때 : } x_{median} = \frac{x_{\left(\frac{n}{2}\right)} + x_{\left(\frac{n}{2}+1\right)}}{2}$$

평균과 중앙값

예제 2-5 평균

준비파일 | 05.descstat.R

- 커피 주문량의 평균을 R을 통해 구해봅시다.

```
32: rc <- ranicafe$Coffees
33: weight <- (1/length(rc))
34: sum( rc * weight )
```

- 평균 식을 이용한 계산

- 32줄 : 커피 판매량 자료의 복사본 rc를 만듭니다.
- 33줄 : 변수 weight에 '1/자료의 개수($1 / \text{length}(\text{rc})$)'를 저장합니다.
자료의 개수를 알고 있더라도 벡터의 원소의 개수를 구하는 `length()` 함수를 이용해서 자료의 개수를 직접 자료로부터 구합시다.
- 34줄 : rc 벡터의 각 값과 '1/n'을 곱한 결과를 모두 더해 평균을 구합니다.

평균과 중앙값

- R에서 평균을 구하는 함수 : `mean()`

```
35: mean( rc )
```

- R에서 평균을 구하는 함수는 `mean()`입니다. 여기서는 `rc`는 벡터이고 `mean()` 함수에 전달된 벡터 값의 평균을 구합니다. 평균 판매량은 21.51064 잔입니다.

```
38: rc <- c(rc, NA)
39: tail(rc, n=5)
40: mean( rc )
41: mean( rc, na.rm=TRUE )
```

- 결측(NA)이 있을 때의 `mean()`
 - 38줄 : `rc` 벡터 뒤에 새로운 원소 `NA`를 추가하고 이를 다시 `rc`로 저장합니다.
 - 39줄 : `rc` 벡터 뒤의 5개의 원소를 출력합니다. `NA`가 마지막에 추가된 것을 확인합니다
 - 40줄 : `mean()` 함수는 자료에 결측이 있으면 그 결과 역시 `NA`가 됩니다.
 - 41줄 : 결측값이 있을 경우 이를 어떻게 처리할지를 R에게 알려줘야 하는데, 결측값을 제거하는 방법을 지원합니다(`na.rm=TRUE`).

평균과 중앙값

예제 2-6 양 끝 값의 변화에 대한 평균의 변화

준비파일 | 05.descstat.R

- 양 끝 값의 변화에 민감한 평균 : 평균의 약점
 - 최대 판매량인 48잔이 480잔을 잘못 기록한 것이라고 가정하고 이 때의 평균의 변화를 살펴 봅시다.

```
51: rc <- ranicafe$Coffees
52: rc[rc == max( rc )] <- 480
53: mean( rc )
```

- 51줄 : 커피 판매량 자료의 복사본 rc를 만듭니다. (앞서 rc에 NA가 들어갔습니다.)
- 52줄 : 커피 판매량의 최댓값(rc == max(rc))을 480으로 변경합니다.
- 53줄 : 자료의 평균을 구합니다.
- 약 21.5잔에서 30.7잔으로 크게 변경되었습니다.

```
[1] 30.70213
```

평균과 중앙값

예제 2-7 중앙값

준비파일 | 05.descstat.R

- 커피 주문량의 중앙값을 R을 통해 구해봅시다.

```
58: ( median.idx <- ( length(rc)+1 ) / 2 )
59: ( rc.srt <- sort(rc) )
60: rc.srt[ median.idx ]
```

- 58줄 : 커피 판매량 자료는 47로 홀수에 해당하므로 자료의 개수(`length(rc)`)에 1을 더한 값을 2로 나누고 이를 `median.idx`에 저장합니다.
- 59줄 : 자료를 순서대로 정렬한 값을 변수 `rc.srt`에 저장합니다.
- 60줄 : 59줄에서 정렬한 자료 중 58줄에서 구한 순서에 해당하는 값을 출력합니다. 중앙값은 23임을 알 수 있습니다

```
61: median( rc )
```

- 61줄 : R에서 중앙값을 구하는 함수는 `median()` 입니다.

평균과 중앙값

예제 2-8 양 끝 값의 변화에 대한 중앙값의 변화

준비파일 | 05.descstat.R

- 양 끝 값의 변화에 중앙값은 어떻게 반응하는지 살펴봅시다.

```
65: rc <- ranicafe$Coffees
66: rc[rc == max( rc )] <- 480
67: ( median( rc ) )
```

- 65줄 : 커피 판매량 자료를 변수 rc에 저장합니다.
- 66줄 : 커피 판매량의 최댓값($rc == \max(rc)$)을 480으로 변경합니다.
- 67줄 : 자료의 중앙값을 구합니다.

```
[1] 23
```

- 중앙값은 양 끝 값의 변화에 둔감합니다. (강건하다)

평균과 중앙값

- **평균과 중앙값으로부터 자료의 모양 유추하기**

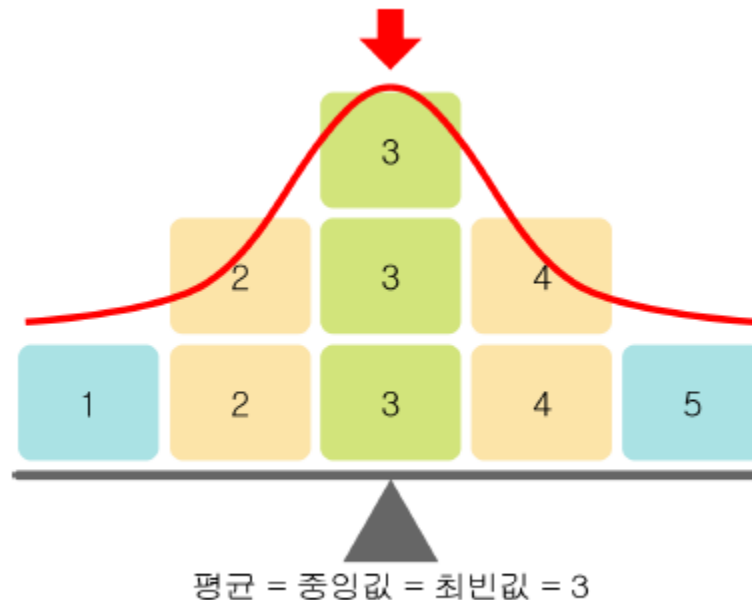
- 숫자를 이용하여 자료의 특성을 파악하면, 이렇게 구해진 특성만으로 대략의 자료의 모양을 추측할 수 있습니다.
- 다음의 자료를 통해 평균과 중앙값으로 대략의 자료의 모양을 유추해 봅시다.
 - 정확하지 않으므로 그래프를 그리는 과정을 거쳐야 하지만, 그래프를 그리기 전에 특성을 안다면 미리 어떤 형태가 될지 추측해 볼 수 있습니다.
 - 최빈값이 하나이며, 최빈값 이전과 이후로 점점 자료의 개수는 줄어드는 것으로 한정해 봅시다.

평균과 중앙값

- 평균, 중앙값, 최빈값이 동일할 때(예에서는 3으로 동일)

1 2 2 3 3 3 4 4 5

- 이와 같이 평균과 중앙값이 같거나 그 차이가 크지 않은 경우(최빈값과도) 평균(혹은 중앙값)을 중심으로 좌우대칭으로 유추해 볼 수 있습니다.



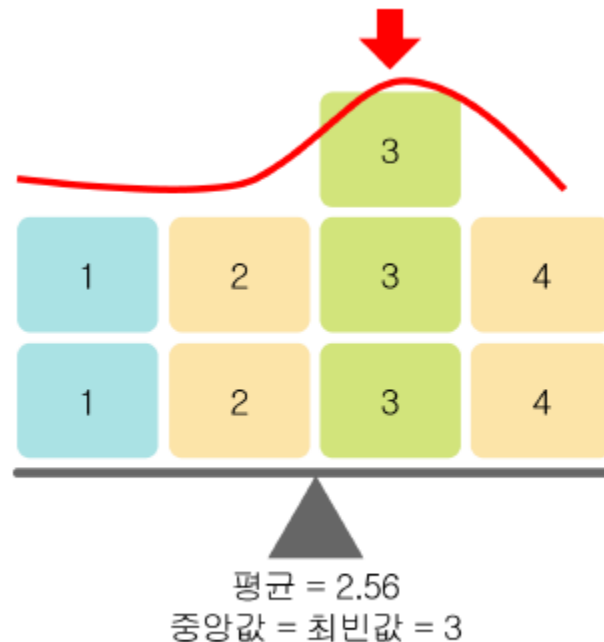
[그림 2-15] 평균과 중앙값이 비슷한 경우

평균과 중앙값

- 평균이 중앙값 보다 작을 때(중앙값은 최빈값보다 작거나 같음, 예에서는 동일)

1 1 2 2 3 3 3 4 4

- 평균이 중앙값보다 작을 경우 꼬리가 작은 쪽으로 길게 늘어집니다. (왼쪽으로 꼬리가 늘어진 자료)



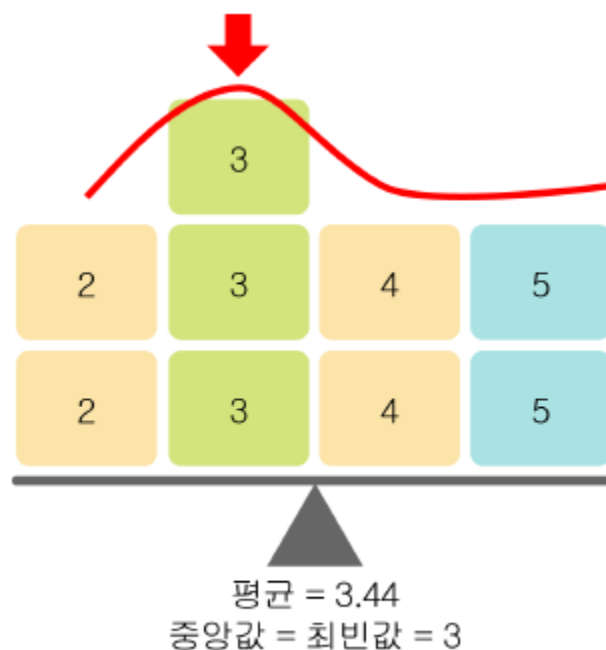
[그림 2-16] 평균이 중앙값보다 작은 경우

평균과 중앙값

- 평균이 중앙값보다 클 때 (중앙값은 최빈값보다 크거나 같음, 예에서는 동일)

2 2 3 3 3 4 4 5 5

- 평균이 중앙값보다 큰 경우 꼬리가 큰 쪽으로 길게 늘어집니다. (오른쪽으로 꼬리가 늘어난 자료)



[그림 2-17] 평균이 중앙값보다 큰 경우

평균과 중앙값

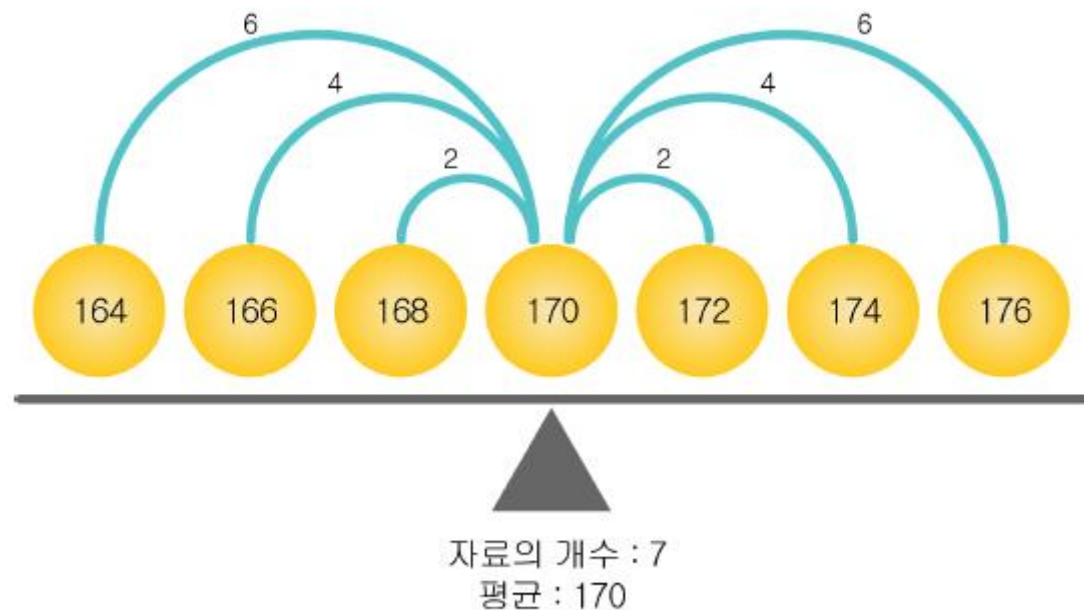
- 대표값이 지닌 약점

- 모든 데이터를 대표하는 값으로 각 자료가 갖고 있는 정보가 무시되고 대표값으로만 전체 자료의 정보가 대표값 하나만 남게 됩니다.
- 자료는 한 값만으로 이루어지지 않고 변동하고 있습니다.
 - 이런 자료의 특성을 나타내는 “자료의 퍼진 정도”를 구해 봅시다.
- 자료의 퍼진 정도로 대표값이 지닌 약점을 보완하여 전체 자료를 알아보는데 유용하게 사용해 봅시다.

표준편차와 사분위수 범위

표준편차

- “각 자료들이 평균에 대해서 평균적으로 얼마나 떨어져 있을까?”
- 다음의 자료로 중고등학교에서 배운 표준편차 계산하는 방법을 다시 한번 익혀 봅시다.



[그림 2-18] 평균이 170이고 자료의 개수는 7인 자료

표준편차와 사분위수 범위

예제 2-9 개별 관찰값과 평균과의 차이에 대한 평균

준비파일 | 05.descstat.R

- 표준편차 구하기 1단계

- 앞선 자료에서 자료의 개수는 7이고, 평균은 170입니다.
- 7개의 각 관찰값은 평균을 중심으로 흩어져 있습니다.
- 각 관찰값이 평균으로부터 평균적으로 얼마나 떨어져 있는지 구해 봅시다.
 - 이를 위해 각 ‘관찰값 – 평균’을 구합니다.

```
89: height <- c(164, 166, 168, 170, 172, 174, 176)
90: ( height.m <- mean( height ) )
91: ( height.dev <- height - height.m )
92: sum( height.dev )
```

표준편차와 사분위수 범위

• Code 설명

- ▣ 89줄 : 자료들을 height라는 이름의 벡터로 저장합니다.
- ▣ 90줄 : height의 평균을 구해 height.m에 저장하고, 그 값을 출력합니다. (평균은 170 입니다.)
- ▣ 91줄 : height 벡터의 개별값에서 평균인 height.m을 뺀 값들을 height.dev 벡터에 저장하고, 그 값을 출력합니다.
 - 여기서 구한 개별 관찰값과 평균과의 차이를 통계에 서는 **편차(deviation)**라고 합니다.
- ▣ 92줄 : height.dev의 합을 출력합니다.
 - 편차의 합은 항상 0입니다.
 - 평균과 개별 관찰값의 차이 즉, 편차는 합이 항상 0으로 평균이 0이 됩니다.
 - 처음에 시작한 관찰값과 평균과의 차이의 평균이 의미를 잃습니다.
 - 편차가 부호(+, -)를 갖고 각 값을 모두 더하면 상쇄되기 때문입니다. 부호를 없애기 위해 제곱을 사용합니다.

표준편차와 사분위수 범위

예제 2-10 편차 제공의 평균 구하기

준비파일 | 05.descstat.R

- 편차 제공의 평균을 구합니다.
 - 편차들을 제공들로 편차 제공의 평균을 구해봅시다.

```
95: ( height.dev2 <- height.dev ^ 2 )  
96: sum( height.dev ^ 2 )  
97: mean( height.dev ^ 2 )
```

- **Code 설명**
 - 95줄 : height.dev2에 편차 제공을 저장하고, 출력합니다.
 - 96줄 : 편차 제곱합을 구합니다.
 - 97줄 : 편차 제공의 평균을 구합니다. (값은 16)

표준편차와 사분위수 범위

- 분산

- 편차 제곱의 평균
- 키의 자료에서 평균은 170cm 이고 각 자료들은 평균 16cm² 떨어져 있습니다.
- 여기서 분산은 평균과 단위가 다릅니다.
 - 평균과 함께 사용하기 힘듭니다.
 - 이를 위해 평균과 단위를 맞춰주기 위해 분산의 제곱근을 구합니다. (cm² → cm)

- 표준편차

- 분산의 제곱근으로 구합니다.
- 평균과 동일한 단위로 평균과 함께 사용할 수 있습니다.
 - 평균 ± 표준편차

표준편차와 사분위수 범위

예제 2-11 표준편차 구하기

준비파일 | 05_descstat.R

- 분산에 제곱근을 취해 표준편차를 구합니다.

```
100: sqrt( mean( height.dev ^ 2 ) )
```

- 100줄 : 편차 제곱합의 평균, 즉 분산의 제곱근을 구합니다. (4입니다.)

예제 2-12 분산과 표준편차 구하기

준비파일 | 05_descstat.R

- R이 제공하는 함수를 이용하여 분산과 표준편차를 구해봅시다.

```
103: var( height )
104: sd( height )
```

- 103줄 : var() 함수를 이용하여 height의 분산을 구합니다.
- 102줄 : sd() 함수를 이용하여 height의 표준편차를 구합니다.

표준편차와 사분위수 범위

• 여기서 잠깐!

- 위에서 구한 분산과 표준편차가 앞서 구한 분산과 표준편차와 다릅니다.
 - 코드로 직접 구한 분산 : 16, 표준편차 : 4
 - R 함수로 구한 분산 : 18.66667, 표준편차 : 4.320494
- R 함수로 구한 분산과 표준편차는 표본의 분산과 표준편차입니다.
 - 코드로 직접 구한 분산과 표준편차는 모집단의 분산과 표준편차 입니다.
 - R 함수로 구한 분산과 표준편차는 표본의 분산과 표준편차 입니다.
 - 각각의 차이는 분산(혹은 표준편차)을 구하기 위해 사용하는 식에서 분모가 다르기 때문입니다. R 함수가 구한 분산(혹은 표준편차)은 분모가 표본의 개수(n)가 아닌 “표본의 개수 - 1”($n-1$) 입니다
 - 표본 분산을 구하기 위해 $n-1$ 로 나눈 이유는 5장 추정에서 학습합니다.

표준편차와 사분위수 범위

• 사분위수

- ▣ 사분위수는 전체 자료를 순서대로 나열한 후 4등분 한 각각의 위치에 해당하는 값으로 자료의 25%, 50%, 75%, 100%가 되는 값입니다.
- ▣ R에서는 `quantile()` 함수로 구할 수 있습니다.
 - 커피 판매량 자료(rc)를 `quantile()` 함수로 구해봅시다.

```
> quantile(rc)
 0%  25%  50%  75% 100%
 3   12   23   30   48
```

- ▣ 자료를 순서대로 나열해 놓아 다음을 나타냅니다. (0%는 최솟값)
 - 25%가 되는 값(제 1사분위수, Q_1) : 12
 - 50%가 되는 값(제 2사분위수, Q_2 = 중앙값) : 23,
 - 75%가 되는 값(제 3사분위수, Q_3) : 30,
 - 100%가 되는 값(제 4사분위수, Q_4) : 48

표준편차와 사분위수 범위

예제 2-15 사분위수 범위와 상자도표

준비파일 | 05.descstat.R

- 사분위수 범위

- 중앙값을 포함하는 영역인 제1사분위수에서 제3사분위수 사이의 길이

- 상자도표

- 사분위수를 이용하여 자료의 모양을 나타내는 도표
- 제 1사분위수와 제 3사분위수 사이의 영역을 상자로 표시하고 최솟값 및 제 4사분위수(최댓값)까지는 상자에서 뻗어 나온 수염으로 연결

```
152: ( qs <- quantile(rc) )  
153: print( qs[4] - qs[2] )  
154: IQR(rc)  
156: bp <- boxplot(rc, main="커피 판매량에 대한 상자도표", axes=F)
```

표준편차와 사분위수 범위

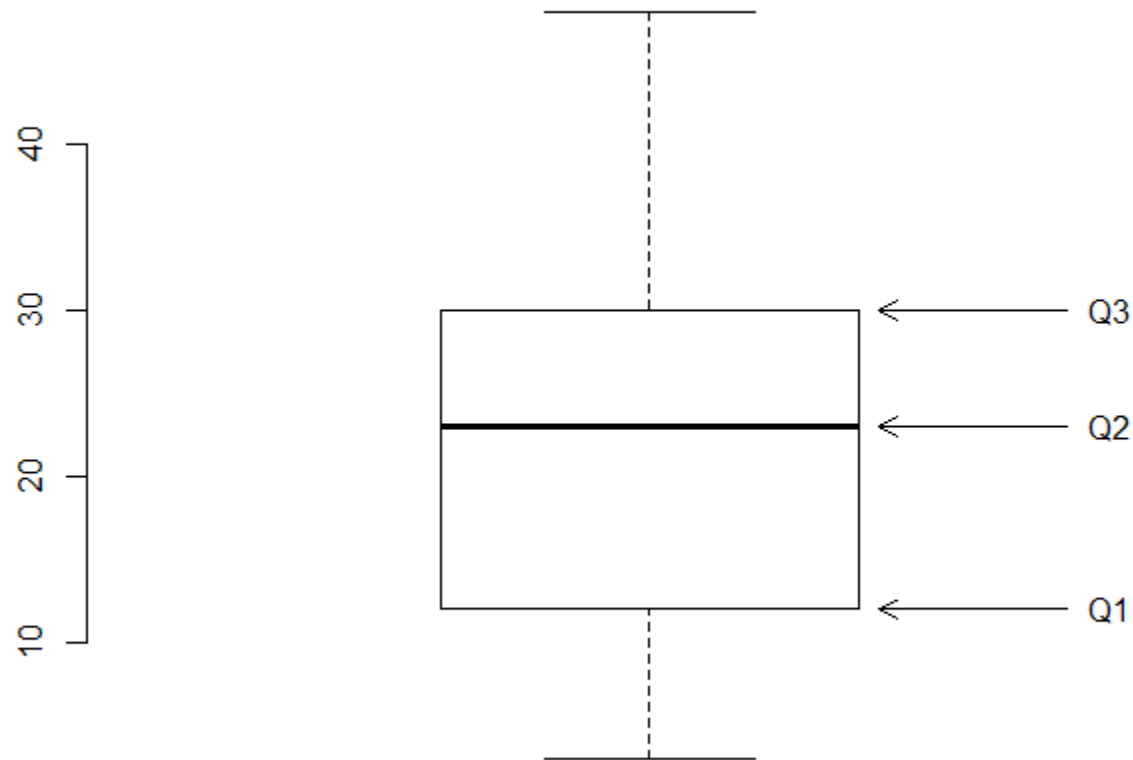
- **Code 설명**

- 152줄 : `quantile()` 함수를 이용하여 사분위수를 구해 그 값을 `qs`에 저장하고 값을 출력합니다.
- 153줄 : `qs`의 네 번째 자료는 제3사분위수를 갖고 있고, 두 번째 자료는 제1사분위수를 저장하고 있어, 제3사분위수에서 제1사분위수를 뺀 값인 사분위수 범위를 출력합니다.
 - 출력에서 `quantile()` 함수는 이름을 가진 벡터로 75%라는 이름이 나왔지만, 사분위수 범위인 18을 출력해줍니다.
- 154줄 : R의 내장함수 중 드물게 대문자 이름인 `IQR()`을 통해 사분위수 범위를 구합니다. (153줄과 같은 18)
- 155줄 : 상자도표(`boxplot`)를 출력합니다

표준편차와 사분위수 범위

▣ 상자도표

커피 판매량에 대한 상자도표



표준편차와 사분위수 범위

- 상자도표로 부터 알 수 있는 것

- 상자는 전체 자료 중 가운데에 위치하는 50%의 자료들을 나타냅니다.
- 상자도표의 각 선(사분위수를 나타내는 선) 사이의 길이가 짧으면 그 구간내의 자료들이 좀 더 촘촘히 몰려 있음을 나타냅니다.
- 이상치
 - 일반적인 경우 자료들은 중심위치에 많이 몰려 있습니다.
 - 하지만, 어떤 자료는 다른 자료들과 많이 떨어져 있을 수 있는데 그 정도가 심해 자료의 형태로 보았을 때 발생하기 어려운 양 끝 쪽(작은 쪽과 큰 양 쪽)의 값을 이상치라고 합니다.
 - 상자도표는 이상치를 판별하고 이를 도표상에 나타냅니다.
 - 이상치의 판별과 상자도표에서는 이를 어떻게 표현하는지 확인해 봅시다.

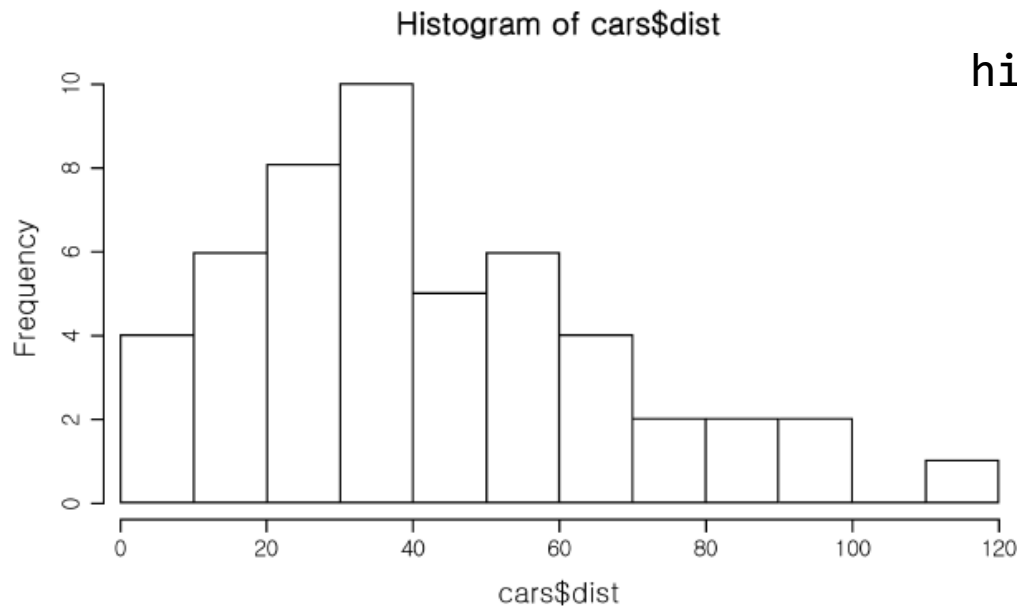
표준편차와 사분위수 범위

- 사용할 데이터 : R의 내장자료 cars에 있는 제동거리(dist)

[표 2-5] 자동차의 속도에 따른 제동거리

R 내장자료 | cars\$dist

2	4	10	10	14	16	17	18	20	20
22	24	26	26	26	26	28	28	32	32
32	34	34	34	36	36	40	40	42	46
46	48	50	52	54	54	56	56	60	64
66	68	70	76	80	84	85	92	93	120



```
hist(cars$dist,
     breaks = seq(0, 120, 10))
```

표준편차와 사분위수 범위

예제 2-16 이상치 판별

준비파일 | 06.outlier.R

- 일반적으로 다음의 값을 이상치로 판별합니다.
 - 제 3사분위수 + 1.5 x 사분위수 범위 보다 큰 값
 - 제 1사분위수 - 1.5 x 사분위수 범위 보다 작은 값
 - R을 이용해 이상치를 판별하고 상자도표에서는 어떻게 표현되는지 확인해 봅시다.

```

7: ( Q <- quantile(cars$dist) )
8: ( ll <- Q[2] - 1.5 * IQR(cars$dist) )
9: ( ul <- Q[4] + 1.5 * IQR(cars$dist) )

11: cars$dist[cars$dist < ll]
12: cars$dist[cars$dist > ul]

13: boxplot(cars$dist, main="Boxplot of Distance")

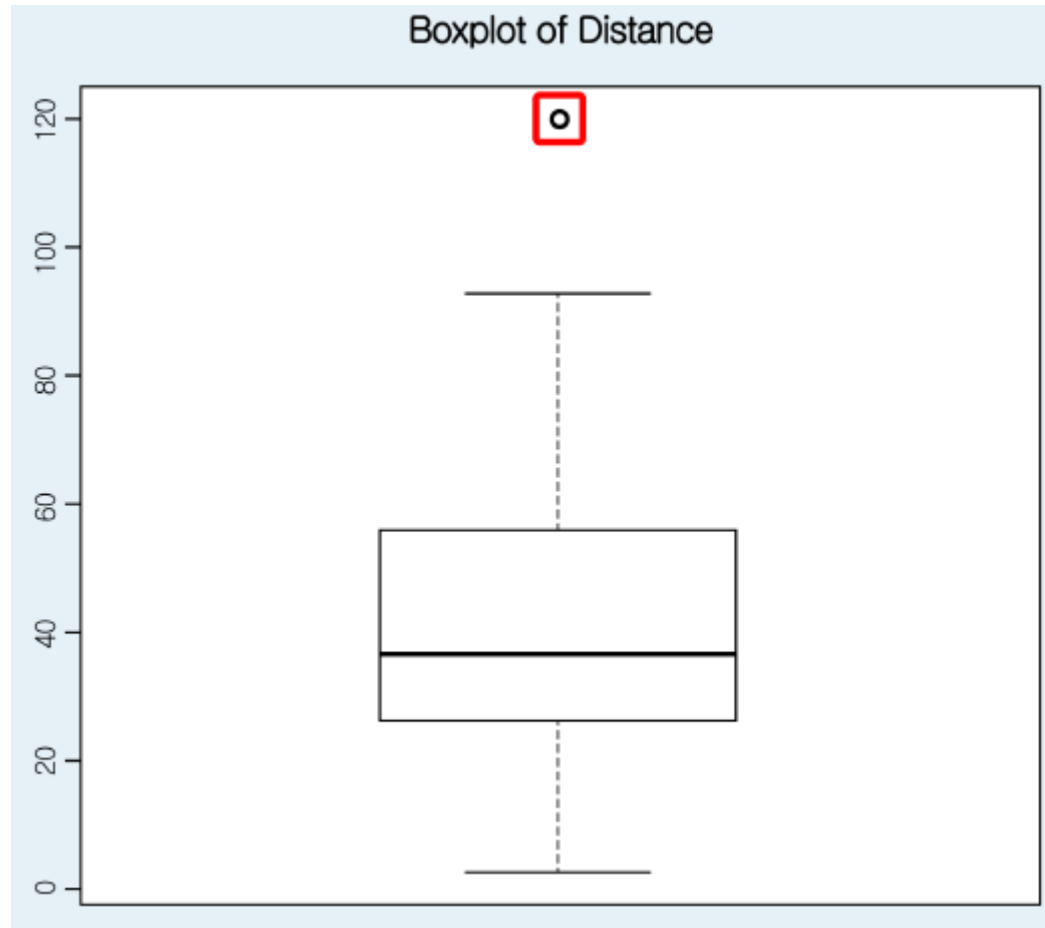
```


표준편차와 사분위수 범위

• Code 설명

- 7줄 : 제동거리 자료의 사분위수 값을 Q에 저장하고 출력합니다.
- 8줄 : $Q1 - 1.5 \times IQR$ 을 변수 ll에 저장하고 이보다 작은 값을 이상치로 판별합니다. 자료에서 구한 하한은 19로 이보다 작으면 이상치입니다
- 9줄 : $Q3 + 1.5 \times IQR$ 을 변수 ul에 저장하고 이보다 큰 값을 이상치로 판별합니다. 자료에서 구한 상한은 101로 이보다 크면 이상치입니다
- 11줄 : cars\$dist에서 그 값이 ll보다 작은 값을 출력합니다. 이에 맞는 값이 없어 numeric(0)을 출력하였으며, 이는 작은 쪽의 이상치는 없음을 의미합니다.
- 12줄 : cars\$dist에서 그 값이 ul보다 큰 값을 출력합니다. 120은 101보다 크므로 이 값이 출력되었으며, 이 값은 이상치입니다.
- 14줄 : 상자도표에서 이상치는 작은 원(그림에서 표시된 부분)으로 표시됩니다. 또한 이상치가 아닌 값들 중 가장 큰 값 혹은 가장 작은값까지 점선으로 연결한 후 닫습니다.

표준편차와 사분위수 범위



정리

- 정리 : 라니 카페의 커피 판매량의 특성을 다음과 같이 정리하였습니다.

[표 2-6] 자료의 특성과 커피 판매량의 특성값

	R 함수	값	설명
자료의 개수(n)	length()	47	자료의 개수
최솟값($\min(x), x_{(1)}$)	min()	3	자료 중 가장 작은 값
최댓값($\max(x), x_{(n)}$)	max()	48	자료 중 가장 큰 값
범위	range	(3, 48)	최댓값-최솟값으로 전체 자료가 분포하는 범위를 나타냄
최빈값	table()로 확인	4	자료 중 빈도수가 가장 많은 값 혹은 구간
평균	mean()	21.51	전체 자료의 무게중심이 되는 값으로 양 끝 값의 변화에 민감한 단점을 갖고 있음
중앙값	median()	23	자료를 순서대로 나열했을 경우 중앙이 되는 값으로 순위가 정해진 후에는 각 값이 갖고 있는 값은 무시됨
표준편차	sd()	11.08	평균을 중심으로 자료가 퍼진 정도를 나타내는 값
제1사분위수	quantile()	12	자료를 순서대로 나열했을 경우 25% 위치의 값
제3사분위수		30	자료를 순서대로 나열했을 경우 75% 위치의 값
사분위수 범위	IQR()	18	(제3사분위수 - 제1사분위수)

정리

- **모수와 통계량**

- 앞서 분산(혹은 표준편차)의 경우 모집단의 분산과 표본의 분산을 구하는 방법이 달랐습니다.
- 모집단의 특성을 모수라고 합니다.
 - 모집단의 특성은 일반적으로 알지 못할 뿐 정해져 있습니다.
- 표본의 특성을 통계량이라고 합니다.
 - 앞서 살펴본 평균, 분산, 중앙값 등등은 자주 사용하는 통계량입니다.
 - 표본에 따라 그 값이 달라집니다.
 - 추측통계학에서 표본의 특성으로 부터 모집단의 특성을 유추합니다.



3장을 위한 준비

: R의 패키지 관리

R에서 패키지 관리하기

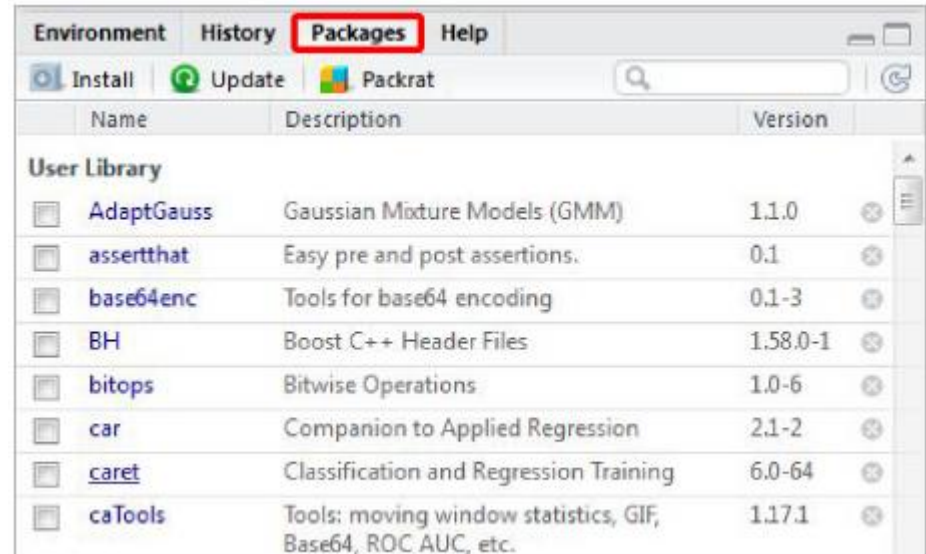
• R 패키지

- R은 기본 기능만으로도 많은 작업을 수행할 수 있지만, 좀 더 편리하게 자료의 정리, 분석, 그래프 작성, 각종 계산 등을 할 수 있도록 전 세계의 개발자들이 R 위에서 작동하는 프로그램을 개발하여 R Foundation에 등록하고, 사용자들이 자유롭게 다운로드 받아서 사용할 수 있는 패키지(package)를 제공하고 있습니다.
- 새로운 알고리즘과 기법 등을 개발하는 전 세계의 연구자에 의해 빠르게 증가하고 안정적으로 작동하고 있습니다.
- 우리는 이미 R 패키지를 사용하였습니다.
 - R로 그림을 그릴 때 사용한 각종 함수들도 R 설치 시 기본으로 설치되는 패키지인의 graphics 함수들을 사용하였습니다.
 - 평균과 분산 등을 계산할 때 사용한 각종 함수들은 R의 기본 패키지인 base 패키지의 함수들입니다.
- 기본 패키지 외에 유용한 다른 패키지를 설치하는 방법을 알아보시다.

R에서 패키지 관리하기

예제 2-17 R Studio에서의 패키지


- 실습 내용
 - R 패키지 중 하나인 'ggplot2' 패키지를 설치하고, 작업환경으로 가져오는 과정을 실습 해봅시다.
- 패키지 설치
 - RStudio 하단의 'Packages' 탭을 열어봅시다.
 - RStudio에서 간편하게 패키지를 관리하는 기능을 제공하며, 현재 사용자에게 설치된 패키지들을 보여줍니다.



The screenshot shows the RStudio interface with the 'Packages' tab selected. The 'User Library' section lists several installed packages with their names, descriptions, and versions. Each package has a checkbox on the left and a refresh icon on the right.

Name	Description	Version
<input type="checkbox"/> AdaptGauss	Gaussian Mixture Models (GMM)	1.1.0
<input type="checkbox"/> assertthat	Easy pre and post assertions.	0.1
<input type="checkbox"/> base64enc	Tools for base64 encoding	0.1-3
<input type="checkbox"/> BH	Boost C++ Header Files	1.58.0-1
<input type="checkbox"/> bitops	Bitwise Operations	1.0-6
<input type="checkbox"/> car	Companion to Applied Regression	2.1-2
<input type="checkbox"/> caret	Classification and Regression Training	6.0-64
<input type="checkbox"/> caTools	Tools: moving window statistics, GIF, Base64, ROC AUC, etc.	1.17.1

R에서 패키지 관리하기

- 패키지를 설치하기 위해  Install 을 클릭하면 그림과 같은 창이 나타납니다.
다음의 네 가지 항목을 확인합니다.

① 패키지를 외부에서 가져올 것인지 컴퓨터상의 파일에서 설치할 것인지

- 실습에서는 외부로부터 다운로드 받는 화면으로 진행 합니다.
(Repository)

② 설치할 패키지의 이름

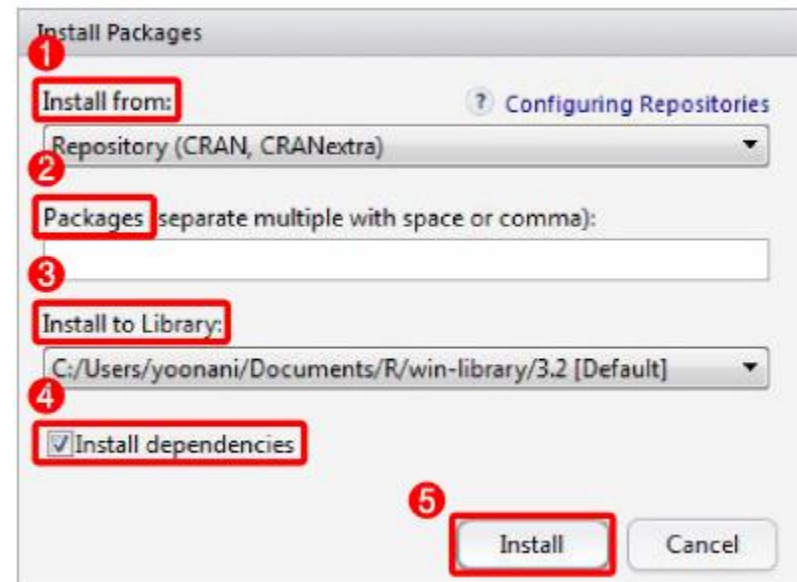
- 여러 개 설치 시 콤마(,)나 복수의 빈칸으로 구분

③ 어디에 설치할 것인지

- 어디에 설치하는지 경로를 확인 합니다.

④ 패키지 간의 관계를 고려하여 필요한 패키지를 설치할 것인지의 여부

- 실습에서는 의존 관계를 고려하여 설치합니다.

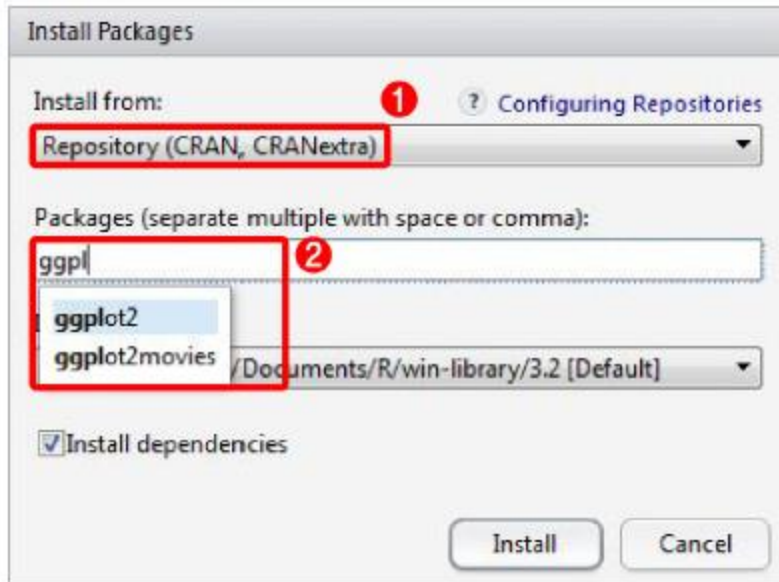


R에서 패키지 관리하기

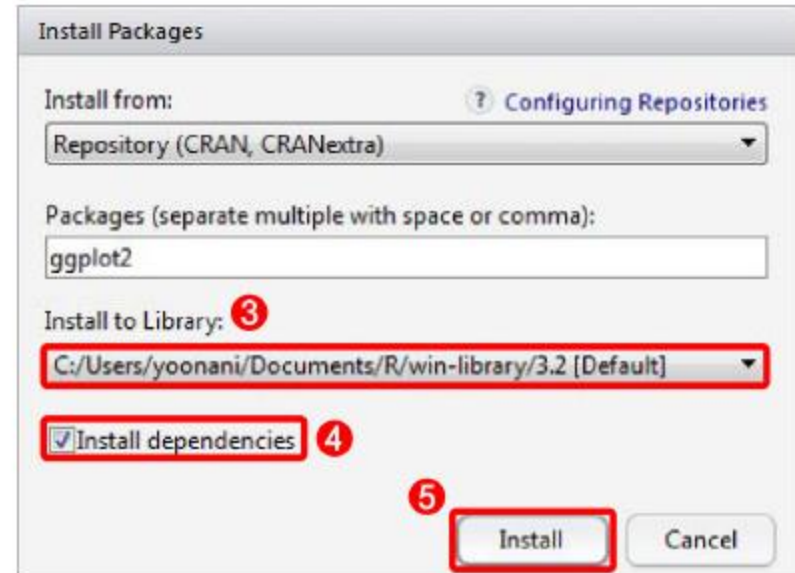
- **ggplot2 패키지 설치하기**

- ① 설치할 패키지의 위치는 기본값으로 설정되어 있는 CRAN에서 가져옵니다.
 - ‘Repository (CRAN, CRANextra)’
- ② 설치할 패키지의 이름은 ‘ggplot2’입니다. 입력하는 문자열을 R Studio가 받아들이며 동일한 문자열을 갖는 패키지를 [그림 2-32]처럼 추천해줍니다.
- ③ 설치 위치는 기본값으로 합니다
 - 설치하고자 하는 컴퓨터마다 위치가 다를 수 있으며, [Default]로 되어 있는 위치가 R의 라이브러리 경로로 기본값이 됩니다).
- ④ 아주 특별한 상황이 아니라면 ‘Install dependencies’는 선택합니다.
- ⑤ 모두 입력 후 ‘Install’을 눌러 설치합니다.

R에서 패키지 관리하기



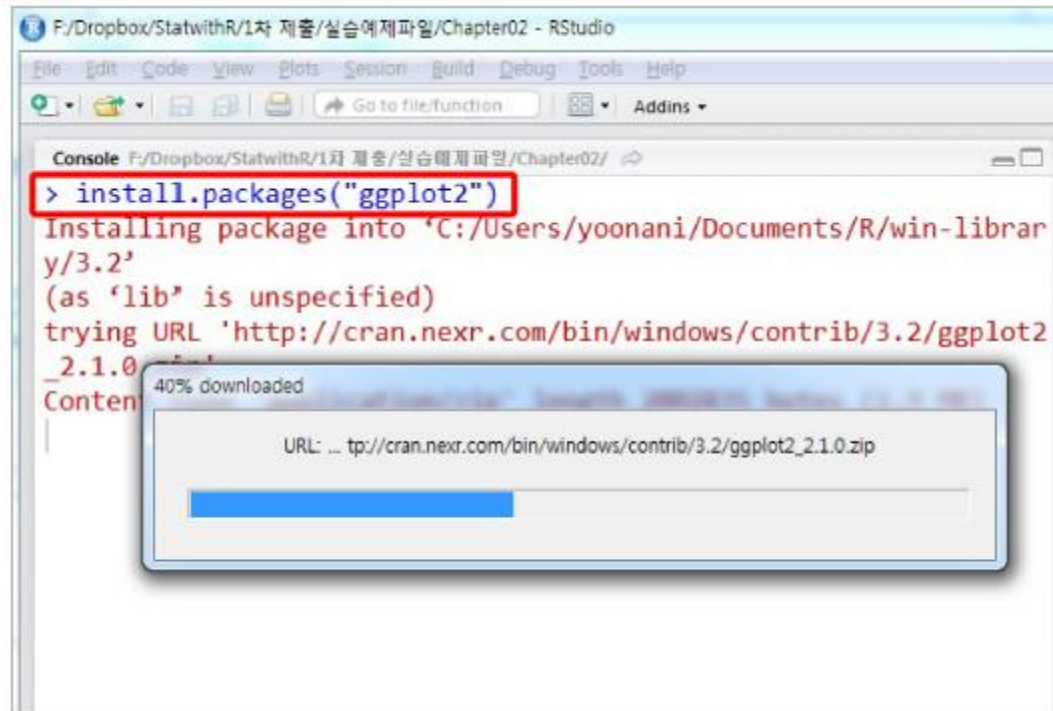
[그림 2-32] 패키지 설치 1



[그림 2-33] 패키지 설치 2

R에서 패키지 관리하기

▣ 패키지 설치 화면



▣ 패키지 설치 명령

- 다음과 같이 R 명령어를 직접 입력하여 패키지를 설치할 수 있습니다.

```
> install.packages("ggplot2")
```

R에서 패키지 관리하기

• 패키지 사용하기

- ▣ ggplot2 패키지를 설치하였습니다.
- ▣ 이제 ggplot2 패키지에 있는 ggplot() 이라는 함수를 실행해 봅시다.
 - 함수 사용을 위해 전달인자가 필요하지만, 일단 실행 가능한지 확인해 봅시다.
 - 아래 그림과 같이 ggplot 이라는 함수를 찾을 수 없다는 메시지가 나옵니다.



```
Console ~/StatWithR/Chapter02/ ↵  
> ggplot()  
Error: could not find function "ggplot"  
> |
```

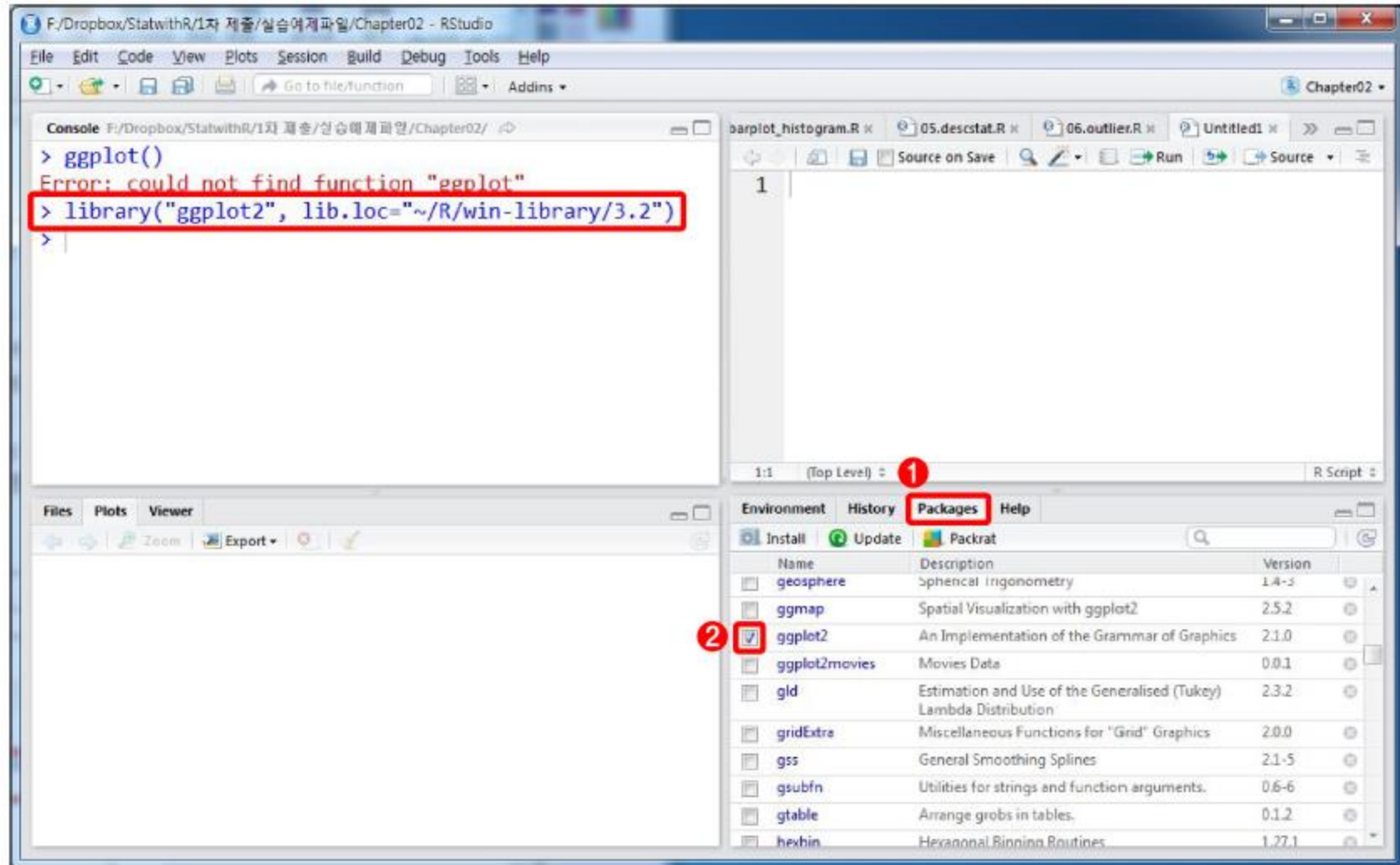
- ▣ ggplot2 패키지를 현재의 작업공간에서 사용하도록 해 봅시다.

R에서 패키지 관리하기

- 패키지 설치는 단순히 패키지 파일을 R이 접근할 수 있는 위치로 복사한 것이며, 작업 시에 이를 사용하기 위해서는 R에게 알려줘야 합니다.
- 이를 위해 [그림 2-36] 처럼 ❶ 'Packages' 탭에 설치된 목록에서 ❷ 'ggplot2'를 찾아 이름 앞의 체크박스를 클릭하면, R 콘솔에 다음과 같은 명령을 직접 입력한 것과 동일한 기능을 하면서 R에서 ggplot2를 쓸 수 있도록 합니다
 - lib.loc 이후에 나오는 위치는 기본 설치 위치로 생략 가능하며, 사용자마다 다를 수 있습니다. 설치 시 다른 곳에 설치할 경우에 해당 위치를 지정합니다).

```
> library("ggplot2", lib.loc="~/R/win-library/3.2")
```

R에서 패키지 관리하기

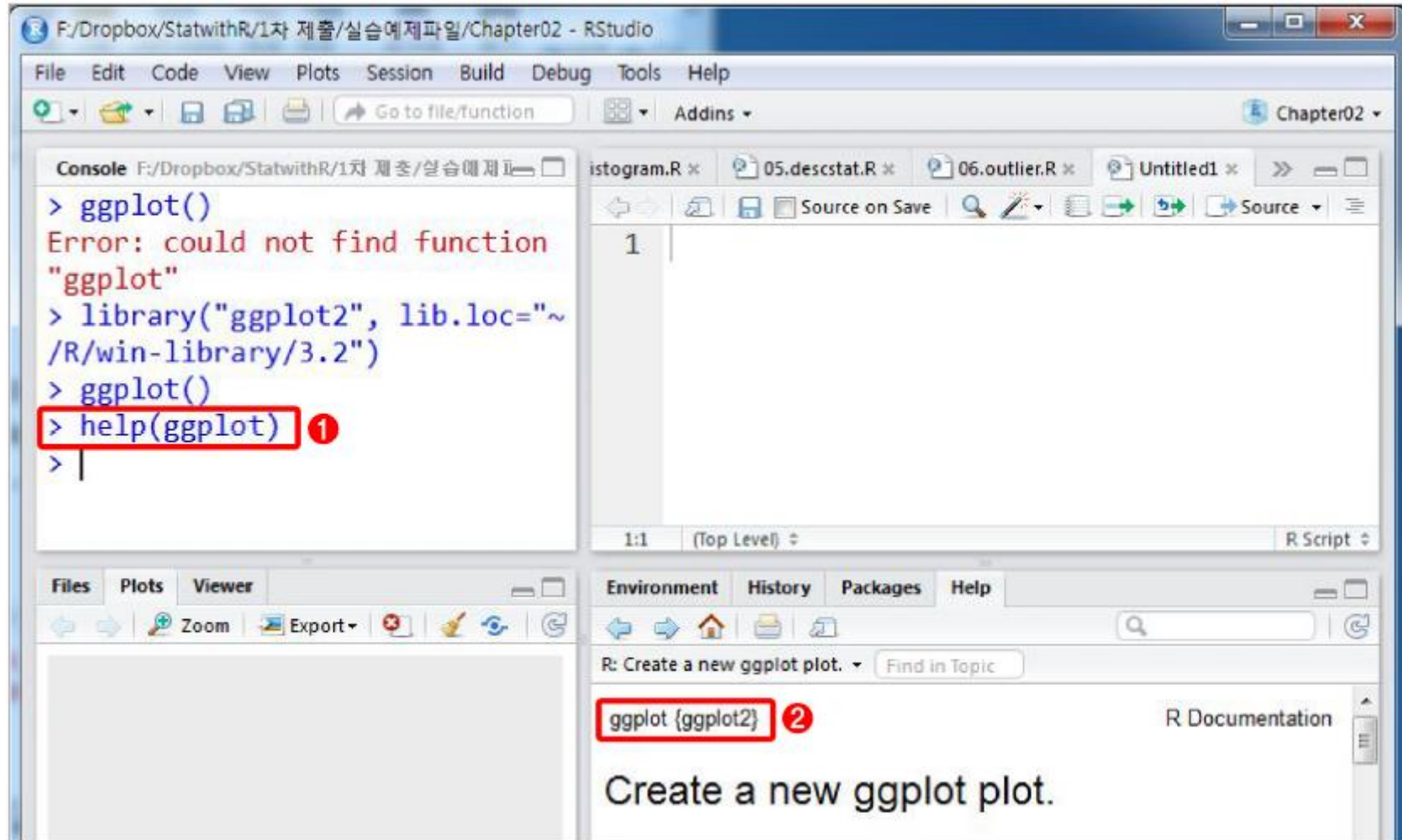


[그림 2-36] 패키지 사용을 R에게 알려줘야 함

R에서 패키지 관리하기

- 다음 장의 그림처럼 ❶ `help()` 함수를 이용하여 `ggplot()` 함수의 도움말을 얻어 봅시다.
- 도움말을 얻을 수 있다면 패키지를 사용할 준비가 된 것입니다.
 - ❷ 도움말보기화면의 상단에 '`ggplot {ggplot2}`'와 같이 출력됩니다.
 - 이는 `ggplot()` 함수가 `ggplot2` 패키지의 함수임을 알려줍니다.

R에서 패키지 관리하기



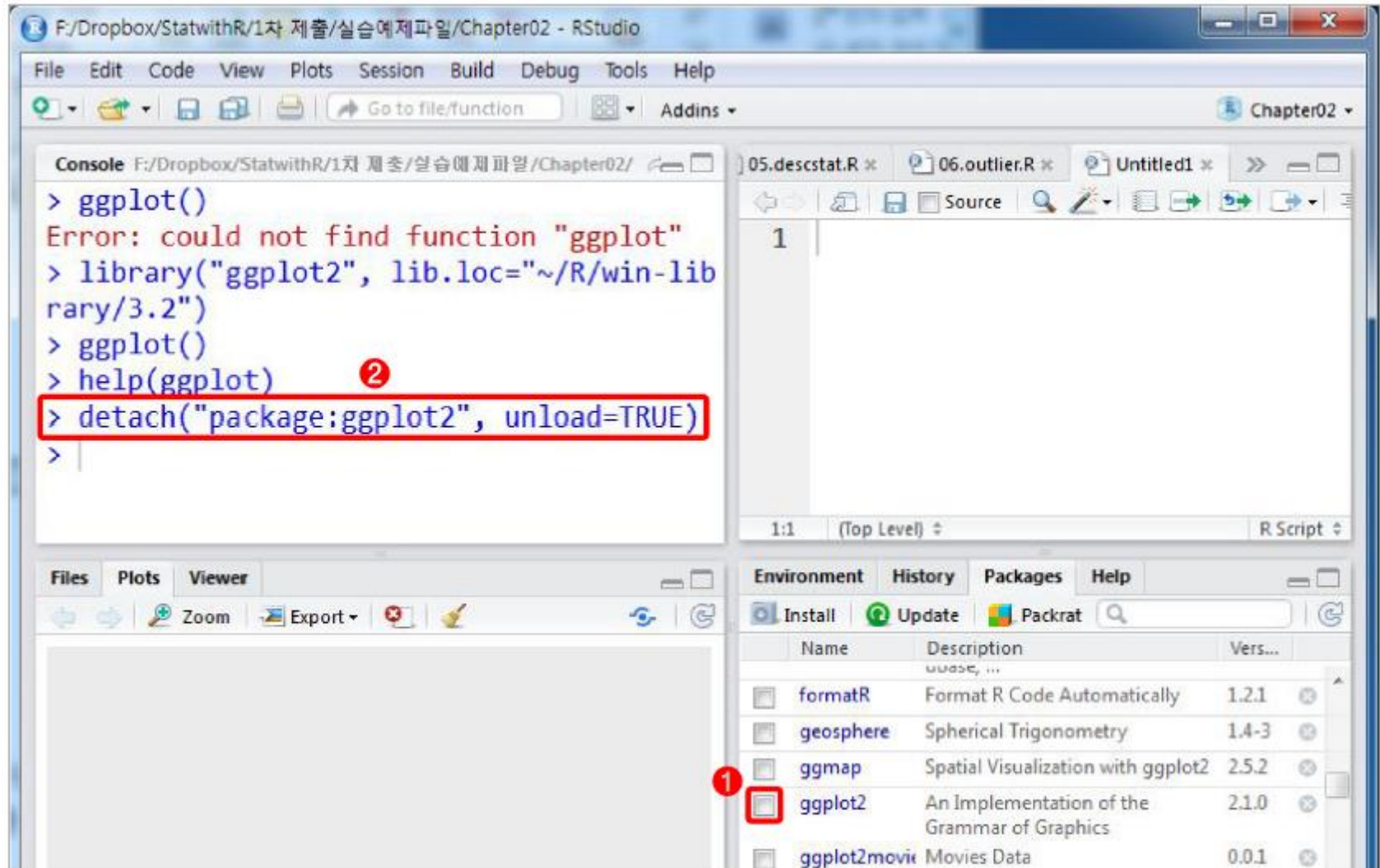
R에서 패키지 관리하기

• 패키지 사용 해제

- 패키지사용해제는 ❶ 패키지 목록에서 선택을 해제함으로써 ❷ detach() 명령을 실행하는 것으로 합니다.
- 패키지 사용해제는 현재 작업공간과의 연결을 끊는 것으로 삭제와는 다릅니다.
 - 즉, 패키지는 설치된 채로 남아 있습니다.
- 패키지를 다시 사용하려면 패키지 목록에서 활성화하거나 library() 함수를 실행하면 됩니다.
- R에서 직접 다음과 같이 명령을 내려 사용을 해제할 수 있습니다.
 - 이상의 과정을 살펴보면, Rstudio의 GUI 환경에서 내리는 명령은 모두 R의 명령을 대체하는 것임을 확인할 수 있습니다.
 - 다음 예제는 이렇게 직접 명령을 입력하여 패키지를 설치하고 사용하는 방법에 대해 안내하고 있습니다.

```
> detach("package:ggplot2", unload=TRUE)
```

R에서 패키지 관리하기



R에서 패키지 관리하기

예제 2-18 R 명령어를 통한 패키지 'prob' 설치

준비파일 | 07.packages,R

• 실습 내용

- R 명령어를 직접 입력하여 패키지를 설치할 수 있습니다.
- 3장에서 사용할 prob 패키지를 R 명령어를 통해 직접 설치하고 작업환경으로 가져오는 R 코드들을 익혀봅시다.
 - 실제 더 많이 사용하는 방법입니다.

```
1: install.packages("prob")
2: library("prob")
3: tosscoin(1)
4: detach("package:prob", unload=T)
```

R에서 패키지 관리하기

- **Code 설명**

- 1줄 : R Studio에 설정된 CRAN 위치로부터 기본 설치 위치에 prob 패키지를 설치합니다
- 2줄 : prob 패키지를 사용합니다.

- 출력화면

다음의 패키지를 부착합니다: 'prob'

The following objects are masked from 'package:base':

`intersect, setdiff, union`

- 패키지 사용시 서로 다른 패키지에서 동일한 함수명을 사용할 때 먼저 작업 공간으로 포함된 함수명이 덮어쓰여짐을 알리고 있습니다.
- 만일 기존 base 패키지의 `intersect`, `setdiff`, `union`을 사용하기 위해서는 'base::intersect', 'base::setdiff', 'base::union'과 같이 '패키지명::함수명'의 형태로 사용하면 됩니다

R에서 패키지 관리하기

- **Code 설명**

- 3줄 : prob 패키지의 tosscoin() 함수를 사용합니다.
- 4줄 : 현재의 작업 공간에서 prob 패키지 부착을 해제합니다.
 - 다시 사용하려면, library() 함수를 사용하면 됩니다.
 - 별다른 메시지가 없으면 성공적으로 해제된 것입니다,



Q & A



수고하셨습니다.