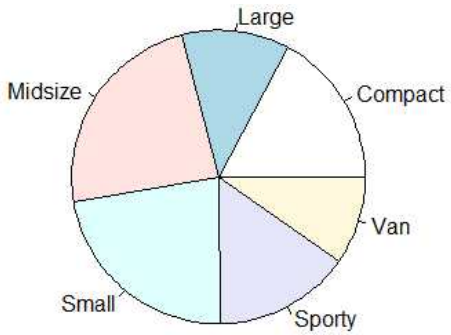
1. 원형그래프(Pie Chart)

원형그래프란 먼저 원을 그리고 이 원을 상대도수에 비례하여 중심각을 나누어 마치 Pie를 조각으로 나눈 것과 같은 형태의 그림이다. 즉, 원 안에 데이터값이 차지하는 비율을 넓이로 나타낸 그래프이다.

```
data(    )     # R에 내장되어 있는 데이터들의 목록을 확인하기
library(MASS)
data(Cars93)      # R에서 기본 제공하는 내장 데이터 가져오는 방법[1)]
str(Cars93)
'data.frame': 93 obs. of  27 variables
View(Cars93)
table(Cars93$Type)  # 93대 차종의 도수분포표
Compact   Large   Midsize   Small   Sporty   Van
   16       11       22       21       14       9
```

이 예시에 대한 Pie Chart는 pie( ) 함수를 사용한다.

```
pie(table(Cars93$Type))
```




```
table(Cars93$Type)
x <- table(Cars93$Type)
pie(x)
pie(x, main='차량유형', col=rainbow(length(x)),
    paste(round(x/sum(x)*100), '%'), radius=1.2 )[2)]
```

---

1) Cars93: 1993년 미국에서 판매된 93대 자동차에 대한 데이터
2) 퍼센트 같은 경우 글자를 pie 안에 입력하는 것이 좋지만 pie 함수 안에는 그런 기능이 없다. 원 안에 글자를 넣으

radius = 파이의 크기, 0.8(기본값)

paste( ) 함수 / paste0( ) 함수

(1) 묶이지 않은 원소

```
paste("paste","는","붙이는","함수","이다")
[1] "paste 는 붙이는 함수 이다"  # paste는 나열된 원소 사이에 공백을 두고 결과값을 출력한다.
```

(2) 묶인 원소

```
paste(c("paste","는","붙이는","함수","이다"))
[1] "paste" "는" "붙이는" "함수" "이다"
```

paste는 주로 vector안에 있는 값들을 하나로 합칠 때 많이 사용한다.

구분자는 collapse 옵션을 넣어 사용

```
paste(c("paste","는","붙이는","함수","이다"), collapse=" ")
paste0("paste","는","공백없이","붙이는","함수","이다")
[1] "paste는공백없이붙이는함수이다"
```

▣ 3차원 원그래프 작성하기

```
install.packages('plotrix')                    # 3차원 pie chart
library(plotrix)
pie3D(x, main='차량유형', explode=0.2)  # explode 속성으로 조각을 분리 가능 / 파이 간 간격

pie3D(x, main='차량유형', col=rainbow(length(x)),
     paste(round(x/sum(x)*100), '%'), radius=1.2, explode=0.2 )
pie3D(x, main='차량유형', col=rainbow(length(x)), labels=names(x),
       radius=1, explode=0.2 )  # 파이별 레이블 지정
```

려면 ggplot2을 사용한다.

2. 줄기-잎 그림(Stem-and-Leaf plot)

줄기-잎 그림은 개별적인 값과 빈도라는 두 가지 정보를 동시에 보여주는 그래프이다. 히스토그램과 유사하지만 개별 값들도 알 수 있다는 특징이 있다.

<그리는 순서>

① 관측 값을 줄기(stem)과 잎(leaf)으로 나눈다. 일반적으로 줄기는 두 자리 수 이상이 될 수 있지만, 잎은 한자리 수이여야 한다. 일반적으로 잎은 일의 자리 숫자로 하고, 줄기는 나머지 값으로 한다.

② 줄기를 수직으로 열거하고 오른쪽에 수직선을 그린다.

③ 각 관측값에 대응하는 줄기와 같은 열에 잎 부분에 해당되는 관측값을 기록한다.

④ 각 줄기에서 잎을 크기순으로 정렬한다.

줄기-잎 그림은 stem( ) 함수로 나타낼 수 있다.

[예제1] (교재, P81) 100명의 영어 성적을 나열한 자료이다. (자료: english.txt)

63	73	52	76	67	68	70	64	54	40	80	65	49	83	65	51	54	79	78	88
55	58	57	43	96	85	55	66	65	78	62	82	60	52	45	61	71	72	81	83
62	58	72	86	50	65	78	59	61	56	65	69	55	76	68	45	64	66	67	70
83	71	70	74	59	60	77	69	80	50	60	54	85	42	90	66	42	67	66	78
72	30	79	75	75	35	51	65	47	56	85	63	78	53	75	75	68	70	38	64

먼저, 데이터 파일(english.text)을 불러오자.

unlist( ) 함수

기본적인 통계 함수들은 벡터에는 작동하지만 리스트에는 작동하지 않아 벡터가 필요한 상황이 많다. unlist( ) 함수를 써서 리스트 구조를 없애 벡터로 만들어 작업한다.

```
x <- read.table("~/R-Programming/data/english.txt")
is.list(x)
is.data.frame(x)
x <- unlist(x)  # unlist(  )를 통해 리스트를 벡터로 변환한다.
stem(x)
```

```
  3 | 058
  4 | 02235579
  5 | 00112234445556678899
  6 | 000112233444555555666677788899
  7 | 000011222345555678888899
  8 | 001233355568
  9 | 06
```

[예제2] (교재, P79) 어느 해에 졸업한 대학생의 초임을 조사한 결과이다.(자료: salary.txt)

(단위 : 만원)

219	228	236	254	247	259	276	248	286	243
265	254	248	298	257	271	282	234	253	276
263	224	262	261	258	251	266	253	234	257
249	271	277	254	265	262	281	283	269	275
268	273	264	273	248	243	258	245	236	256

```
x <- read.table("~/R-Programming/data/salary.txt")
x <- unlist(x)
stem(x)
```

[예제3] 외부데이터

```
str(mtcars)
View(mtcars)
stem(mtcars$mpg)
```

R에 내장된 mtcars 데이터 중 **연비**를 나타내는 **mpg**(miles/gallon)변수를 stem 함수를 이용하여 구한 stem-and-leaf 차트는 10.4, 10.4가 두 종류가 존재함을 알 수 있다.

3. 상자그림(Box Plot)

상자그림은 자료의 특징을 잘 요약해주는 **사분위수**(제1사분위수 Q_1, 제3사분위수 Q_3), 중앙값, 최댓값, 최솟값인 5개의 자료요약을 그래프로 표현한 것이다.

(1) **중앙값**(median)은 중심위치에 대한 측도이고, 자료를 오름차순으로 정렬했을 때 정가운데에 위치하는 값을 말한다.

① n이 홀수이면, $\frac{n+1}{2}$ 번째 자료 값

② n이 짝수이면, $\frac{n}{2}$ 번째와 $\frac{n}{2}+1$ 번째 자료의 평균

(2) **사분위수**(quartile)

① 크기 순서에 따라 나열했을 경우 4등분되는 위치의 관측 값

② 중앙값을 기준으로 자료를 두 그룹으로 나눈다. 데이터 하단부의 중앙값이 제1사분위수이고, 데이터 상단부의 중앙값이 제3사분위수이다.

③ **사분위수의 범위**(IQR, inter-quartile range) : 제3사분위수와 제1사분위수의 차로서 자료의 퍼진 정도를 나타낸다.

$$IQR = Q_3 - Q_1$$

④ 이때 중앙값과 사분위수(Q_1, Q_3) 사이의 거리로 자료의 치우침을 알 수 있다.

(3) **이상점**(outlier)을 파악할 수 있도록 상자그림에 표시하면 자료를 이해하는데 매우 유용하다

이상점은 안쪽울타리의 바깥쪽에 있는 자료를 나타내며, 안쪽울타리는 제1사분위수와 제3사분위수에서 각각 사분위수 범위의 1.5배 한 값만큼 떨어진 값을 뜻한다.

$$Q_1 - 1.5 \times IQR \ , \ \ Q_3 + 1.5 \times IQR$$

[상자그림을 그리는 순서]

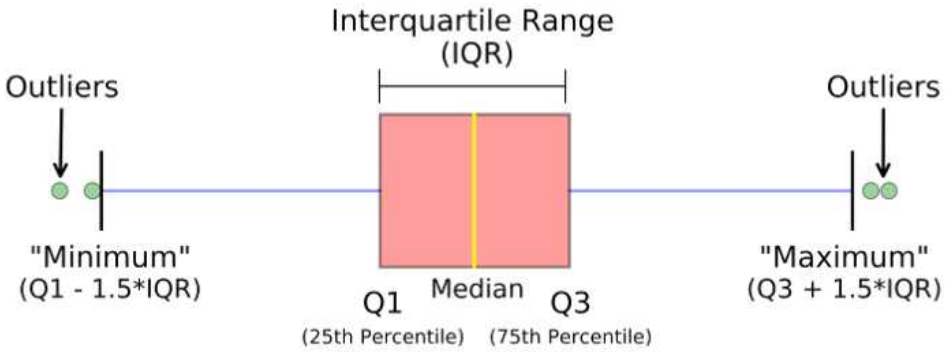
1. 사분위수(Q1, Q3)를 결정한다.
2. Q1과 Q3을 네모난 상자로 표시하고, 중앙값(Q_2)의 위치에 수직선을 긋는다.
3. 사분위수의 범위 $IQR = Q_3 - Q_1$ 을 계산한다.
4. 상자 양끝에서 $1.5 \times IQR$ 크기의 범위를 경계로 하여, 이 범위에 포함되는 최솟값과

최댓값을 Q1과 Q3으로부터 각각 선(수염, Whisker[3])으로 연결한다.

$$Q_1 - 1.5 \times IQR \ , \ \ Q_3 + 1.5 \times IQR$$

5. 양쪽 경계를 벗어나는 자료 값들을 * 로 표시하고, 이 점들을 **이상점**(outlier)이라고 한다.

6. 상자그림에 알맞은 제목을 붙인다.



[수평 상자그림][4]

[예제2] (교재 P77) 고혈압 환자 20명의 나이 자료

45, 37, 40, 41, 47, 41, 45, 51, 42, 44,
45, 32, 46, 42, 43, 47, 49, 39, 50, 41

```
sort(c(45, 37, 40, 41, 47, 41, 45, 51, 42, 44, 45, 32, 46, 42, 43, 47, 49, 39, 50, 41))
        # 데이터를 오름차순으로 정렬
order(c(45, 37, 40, 41, 47, 41, 45, 51, 42, 44, 45, 32, 46, 42, 43, 47, 49, 39, 50, 41))
        # 자료의 순위
x <- c(45, 37, 40, 41, 47, 41, 45, 51, 42, 44, 45, 32, 46, 42, 43, 47, 49, 39, 50, 41)
quantile(x)      # quantile의 사용법(사분위수를 계산하는 9가지 유형)
    0%    25%    50%    75%   100%
 32.00  41.00   43.50   46.25   51.00

? quantile      # help(quantile)
```

quantile(x, y) 함수

x는 자료, y는 0~1 사이의 값. 예를 들면, **quantile(data, 0.25)**는 오름차순으로 정렬

3) 수염은 상자의 양쪽에서 연결된다. 수염은 특이치를 제외하고, 데이터 값의 하위 25%와 상위 25%의 범위를 나타낸다.

4) 이미지 출처: https://leedakyeong.tistory.com/

한 데이터의 제1사분위수 Q_1 를 구한다. **quantile(data, 0.62)**는 100등분을 하는 62번째 값을 출력한다.

quantile(x, probs=0.2) # 제20백분위수, numeric vector of probabilities with values in [0,1].

quantile(x, 0.2)

quantile(x, 0.5) # 제50백분위수, 중앙값

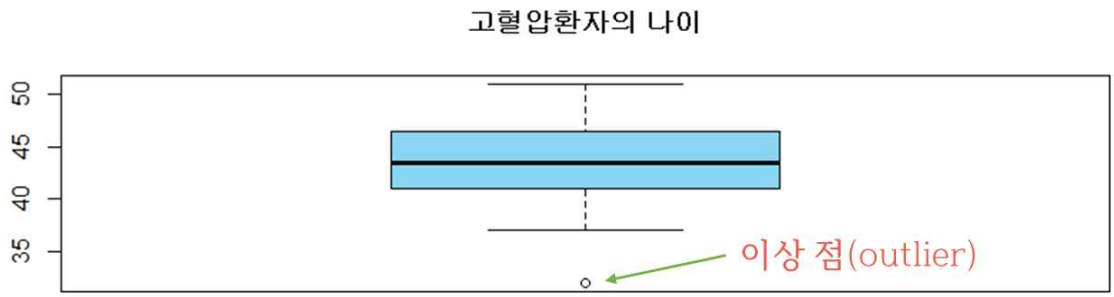
median(x)

IQR(x) # 사분위수의 범위

[1] 5.25 # $Q_3 - Q_1$ 의 값

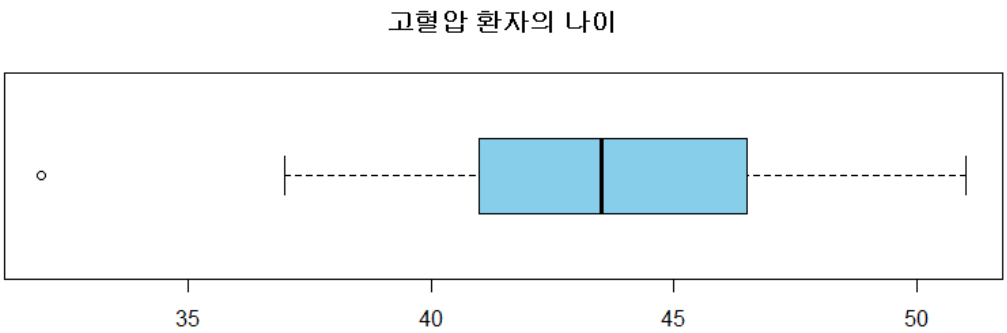
boxplot( ) 함수를 사용하여 상자그림으로 나타냄

boxplot(x, main="고혈압 환자의 나이", col="skyblue")



[수직 상자그림]

boxplot(x, horizontal=TRUE, main="고혈압 환자의 나이", col="skyblue")



[수평 상자그림]

[예제] R에 내장 데이터

```
str(mtcars)
View(mtcars)
boxplot(mtcars, main="연비", col="GreenYellow")
boxplot(mtcars$mpg, main="연비", col="GreenYellow")
```

▣ 여러 자료의 상자그림

[예제] 통계학을 수강한 48명 학생의 중간시험/기말시험 성적이다.

100	68	18	60	83	75	64	62	57	84
84	60	77	80	77	48	73	71	80	71
86	68	80	60	88	64	44	37	55	57
51	77	88	71	37	77	71	46	75	95
66	69	95	55	73	64	64	95		

73	68	12	91	84	75	68	72	57	83
81	58	80	60	90	64	44	43	54	57
51	77	88	71	37	77	71	46	75	95
84	60	77	80	77	48	73	71	80	21
65	64	95	53	53	69	64	95		

```
mid <- read.table("~/R-Programming/data/mid_exam.txt")
final <- read.table("~/R-Programming/data/final_exam.txt")
boxplot(c(mid, final), main="통계학 시험", names=c("중간", "기말"),
        col=c("GreenYellow","Rosy Brown") )
```