

전산통계학 실습

11. R 그래프

그래프

- 그래프

- 수집/처리된 데이터를 저장/분석 시 손쉽게 보여주는 방법
- 효율적인 결과 전달을 위하여 적절한 그래프를 선택해야 함
- 즉, 데이터와 미적 요소(aesthetic)의 결합으로 그래프가 도출됨

- 그래프의 종류

- 산점도
- 줄기-잎 그림
- 상자그림
- 히스토그램
- 바 차트
- 파이 차트 등

그래프

- 그래프의 종류

- 산점도

- 좌표계에 두 데이터를 점으로 찍어 관계(경향)를 파악할 수 있는 그래프

- 줄기-잎 그림

- 데이터의 분포를 확인할 수 있는 그래프
 - 줄기: 자료에서 공통되는 부분
 - 잎: 줄기 부분의 나머지 부분

- 상자그림

- 요약된 5가지의 통계 수치를 확인할 수 있는 그래프
 - 통계 수치: 최솟값, Q1(제1사분위수), 중앙값, Q3(제3사분위수), 최댓값

- 히스토그램/바 차트

- 데이터의 분포를 막대로 확인할 수 있는 그래프
 - 히스토그램: 연속적 변수 / 바 차트: 불연속적 변수

- 파이 차트

- 데이터의 분포를 원의 비율로 확인할 수 있는 그래프

데이터

- 데이터 확인하기

- > head(dataset)
- > str(dataset)

```
> head(mtcars)
      mpg  cyl  disp  hp drat   wt  qsec vs  am  gear  carb
Mazda RX4         21.0   6  160 110 3.90 2.620 16.46 0   1    4    4
Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02 0   1    4    4
Datsun 710        22.8   4  108  93 3.85 2.320 18.61 1   1    4    1
Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44 1   0    3    1
Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02 0   0    3    2
Valiant           18.1   6  225 105 2.76 3.460 20.22 1   0    3    1

> str(mtcars)
'data.frame':   32 obs. of  11 variables:
 $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
 $ cyl : num   6  6  4  6  8  6  8  4  4  6 ...
 $ disp: num  160 160 108 258 360 ...
 $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
 $ drat: num   3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
 $ wt  : num   2.62 2.88 2.32 3.21 3.44 ...
 $ qsec: num  16.5 17 18.6 19.4 17 ...
 $ vs  : num   0  0  1  1  0  1  0  1  1  1 ...
 $ am  : num   1  1  1  0  0  0  0  0  0  0 ...
 $ gear: num   4  4  4  3  3  3  3  4  4  4 ...
 $ carb: num   4  4  1  1  2  1  4  2  2  4 ...
```

데이터

- 데이터 확인하기
 - > ?dataset

Motor Trend Car Road Tests

Description

The data was extracted from the 1974 *Motor Trend* US magazine, and comprises fuel consumption and 10 aspects of automobile design and performance for 32 automobiles (1973–74 models).

Usage

```
mtcars
```

Format

A data frame with 32 observations on 11 (numeric) variables.

```
[, 1] mpg Miles/(US) gallon  
[, 2] cyl Number of cylinders  
[, 3] disp Displacement (cu.in.)  
[, 4] hp Gross horsepower  
[, 5] drat Rear axle ratio  
[, 6] wt Weight (1000 lbs)  
[, 7] qsec 1/4 mile time  
[, 8] vs Engine (0 = V-shaped, 1 = straight)  
[, 9] am Transmission (0 = automatic, 1 = manual)  
[,10] gear Number of forward gears  
[,11] carb Number of carburetors
```

데이터

- 데이터 고정하기

- > attach(dataset)

```
> mtcars$mpg
[1] 21.0 21.0 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 17.8 16.4 17.3 15.2 10.4 10.4 14.7
[18] 32.4 30.4 33.9 21.5 15.5 15.2 13.3 19.2 27.3 26.0 30.4 15.8 19.7 15.0 21.4
> attach(mtcars)
The following object is masked from package:ggplot2:

    mpg

> mpg
[1] 21.0 21.0 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 17.8 16.4 17.3 15.2 10.4 10.4 14.7
[18] 32.4 30.4 33.9 21.5 15.5 15.2 13.3 19.2 27.3 26.0 30.4 15.8 19.7 15.0 21.4
```

- 그래프 창 크기 설정 (새 윈도우)

- > win.graph(x, y)

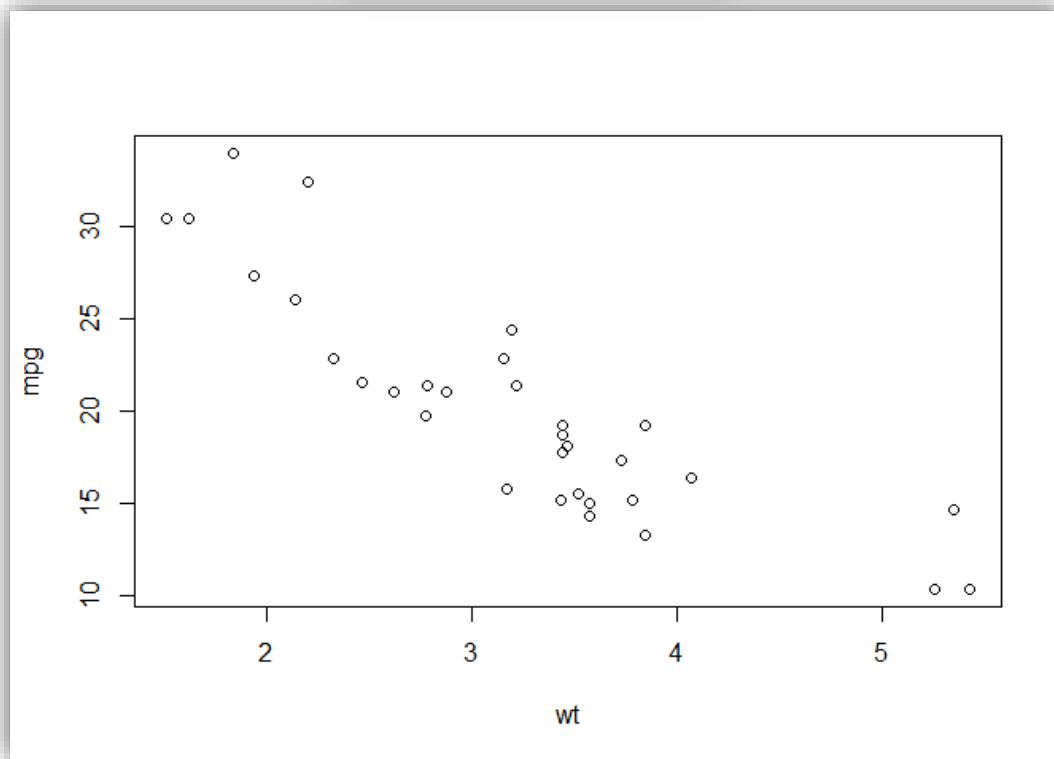
```
> win.graph(16,9)
```



산점도

- > plot(x, y, [options])
 - 두 변수를 각각 x, y 에 대입하고 여러 가지 options 들을 설정
 - 동일한 길이를 가진 두 벡터를 대입 (연관성이 있는 데이터)

```
> plot(x=wt, y=mpg)
```



그래프 옵션

그래프 옵션	설명
main="char"	그래프의 제목
xlab="char", ylab="char"	그래프 x축, y축 이름
xlim=vector, ylim=vector	x축, y축 범위 지정

그래프 출력 타입 옵션	설명
type="p"	점 모양 (기본값)
type="l"	선 모양 (꺼은선)
type="b"	점 + 선 모양
type="c"	"b" 옵션에서 점이 제거된 형태의 선 모양
type="n"	미출력

그래프 옵션



























그래프 모양 옵션	설명
<code>lty="blank"</code>	투명선
<code>lty="dashed"</code>	대쉬선
<code>lty="dotted"</code>	점선
<code>pch="char"</code>	출력 기호의 문양
<code>col="color"</code>	출력 기호의 색깔
<code>cex=numeric</code>	출력 기호의 크기 (기본값=1)
<code>lwd=numeric</code>	선의 굵기







• 색과 관련된 옵션

- `> par(bg="color")`
 - 배경색 지정은 팔레트 지정을 먼저 수행
 - 이후에 그래프 출력 코드를 사용하면 해당 배경색 위에 그래프가 그려짐
- `> colors()`
 - 벡터 내부에 657 가지의 이용 가능한 색깔이 미리 선언되어 있음
- `> rainbow(numeric)`
 - 원하는 개수에 맞는 무지개 색 추출 가능

그래프 옵션

- 점/선 모양 옵션 (pch/lty)

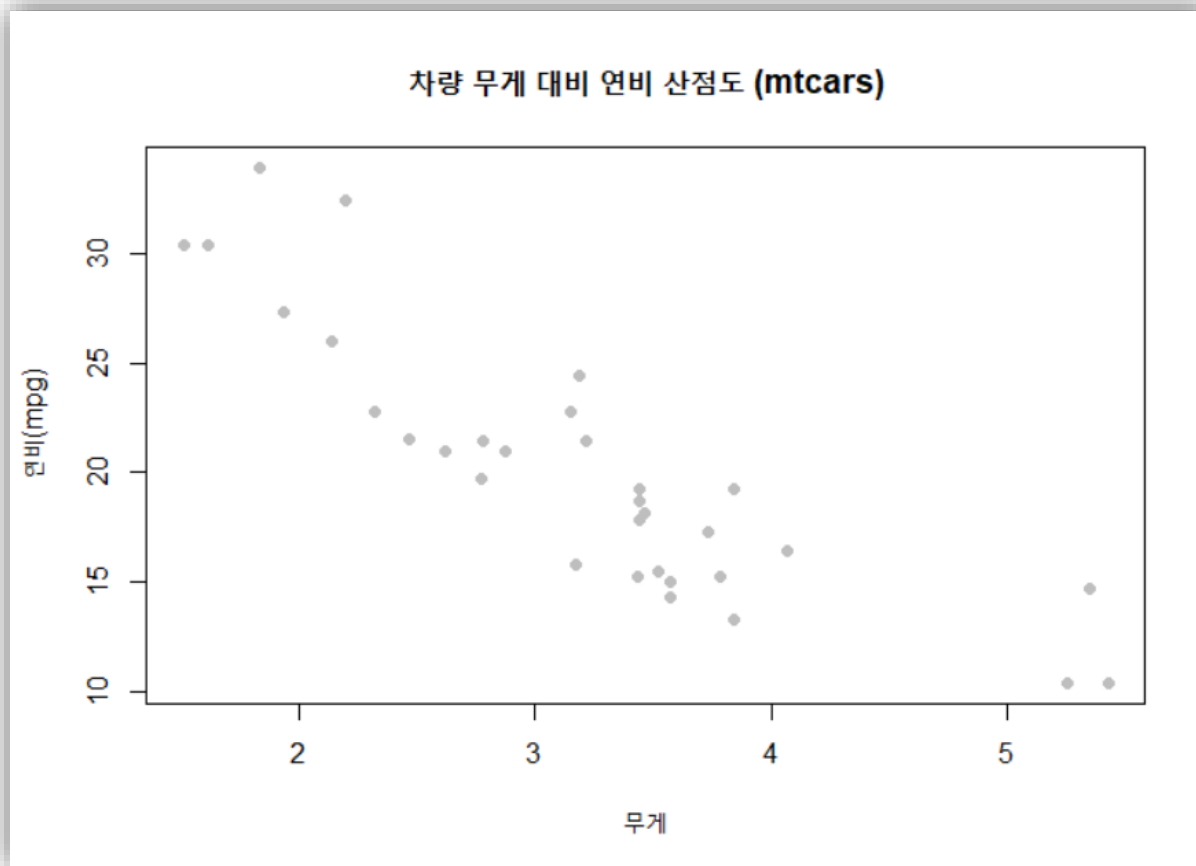
0	1	2	3	4	
					
5	6	7	8	9	
					
10	11	12	13	14	
					
15	16	17	18	19	
					
20	21	22	23	24	25
					

6.'twodash'	
5.'longdash'	
4.'dotdash'	
3.'dotted'	
2.'dashed'	
1.'solid'	
0.'blank'	

산점도

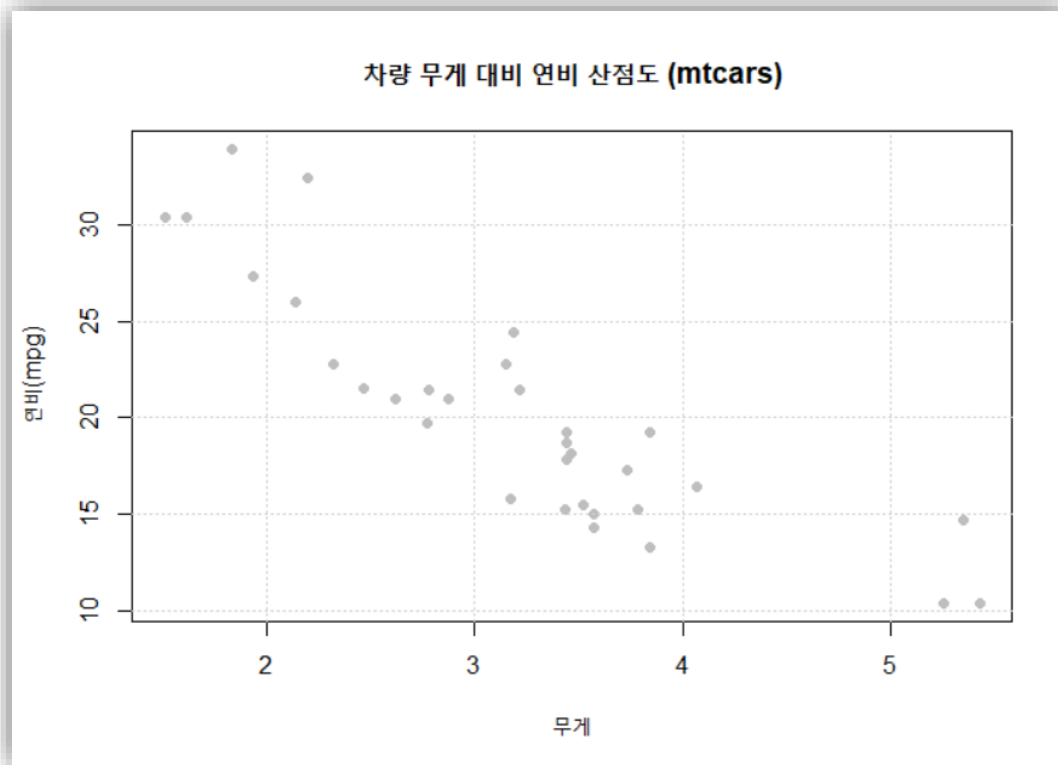
- 예시: 산점도 그리기

```
> plot(x=wt, y=mpg, main="차량 무게 대비 연비 산점도 (mtcars)", xlab="무게", ylab="연비(mpg)", pch=19, col="gray")
```



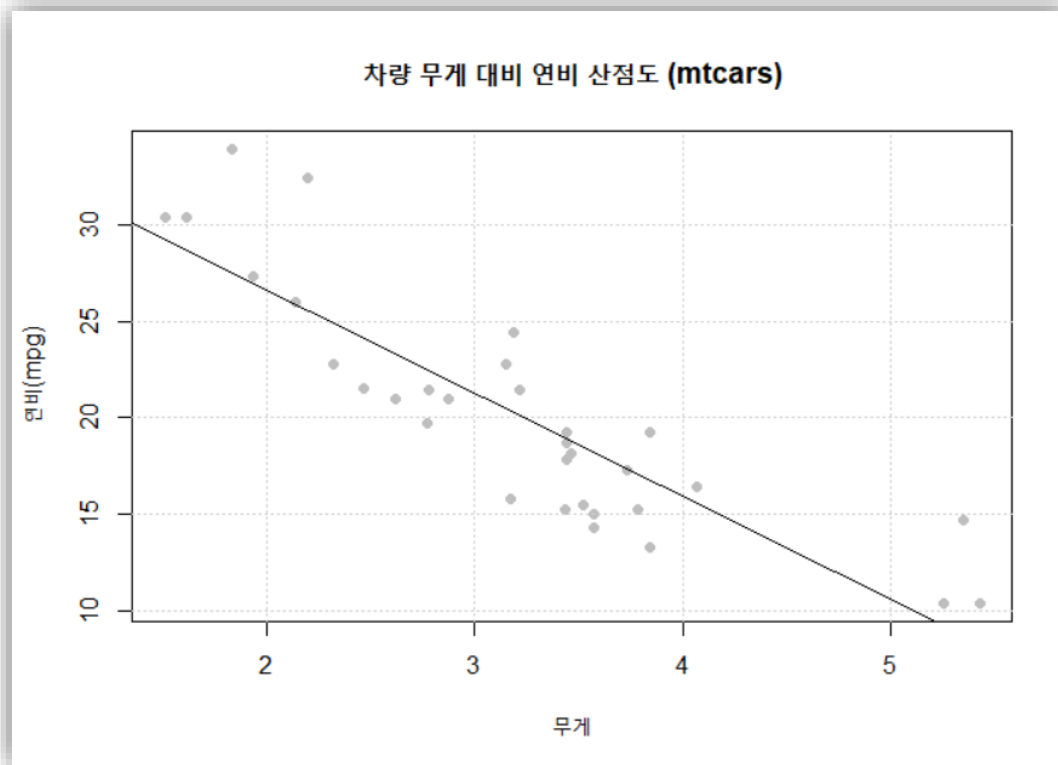
산점도

- `> grid([options])`
 - 그래프에 그리드(격자)를 추가
 - $nx / ny = x\text{축}/y\text{축의 격자 개수}$
 - `col`, `lty`, `lwd` 등의 옵션을 추가 가능



산점도

- `> abline(lm(y~x), [options])`
 - 그래프에 단순회귀선 추가
 - `lm` 함수를 이용하여 데이터를 단순회귀화
 - 선 그래프를 추가하는 것이므로 동일한 옵션들을 이용 가능



산점도

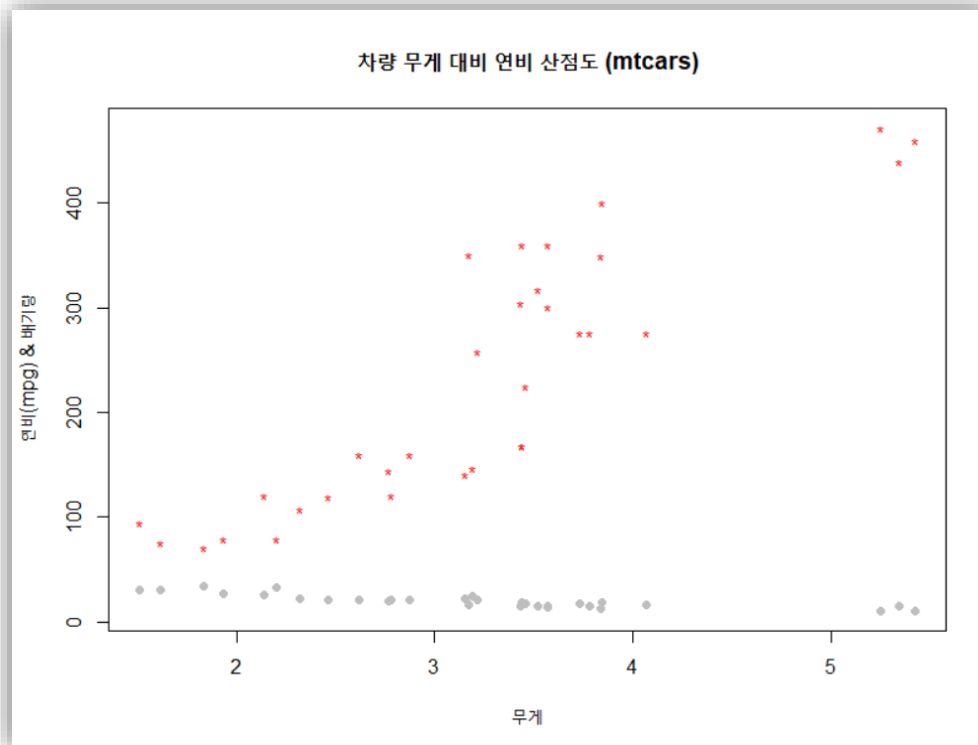
- 그래프 겹치기 (추가)
 - > points(x, y, [options])
 - 이미 그려진 그래프에 점 Points 그래프를 추가
 - 서로 다른 데이터를 올려 산점도 두 개를 한 좌표평면에 표현
 - > lines(x, y, [options])
 - 이미 그려진 그래프에 선 lines 그래프를 추가
 - 그래프 type 옵션의 'b'와 동일

산점도

- 예시: 산점도 그래프 겹치기 (추가)

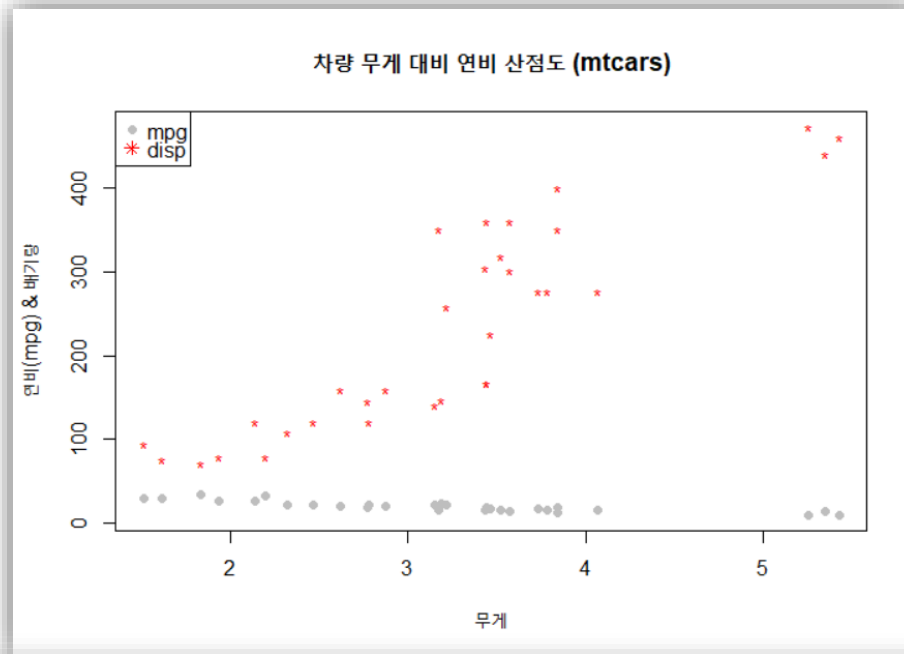
```
> min_combine <- min(c(mpg, disp))  
> max_combine <- max(c(mpg, disp))  
> plot(x=wt, y=mpg, main="차량 무게 대비 연비 산점도 (mtcars)", xlab="무게", ylab="연비(mpg) & 배기량", pch=19, col="gray", ylim=c(min_combine, max_combine))  
> points(x=wt, y=disp, pch="*", col="red")
```

```
> min(disp)  
[1] 71.1  
> max(disp)  
[1] 472  
> min(mpg)  
[1] 10.4  
> max(mpg)  
[1] 33.9
```



산점도

- `> legend(x, y, legend=vector, [options])`
 - x/y
 - 범례의 x/y 좌표 지정 (top, bottom, left, right, center 지정 가능)
 - legend
 - 범례 벡터 (옵션들은 해당 벡터 길이와 동일한 벡터여야 함)

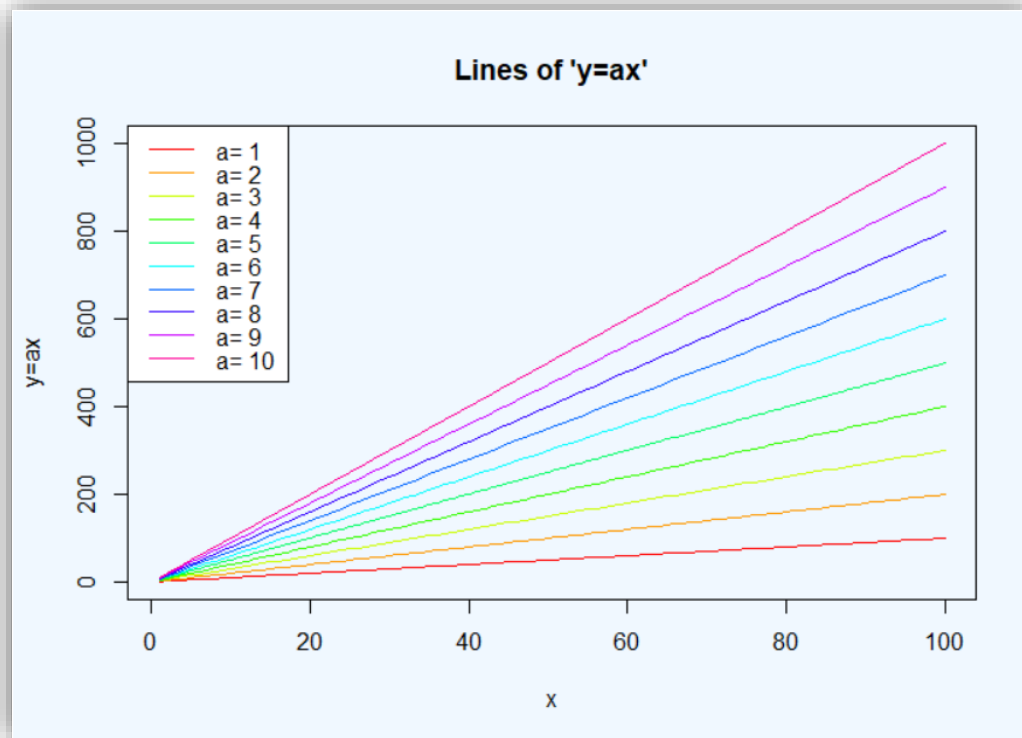


```
> legend(x="topleft", legend=c("mpg", "disp"), col=c("gray", "red"), pch=c(19, 8))
```


산점도

- 예시: 산점도 응용

$$y = ax$$
$$1 \leq a \leq 10$$
$$1 \leq x \leq 100$$
$$a, x \in \mathbb{N}$$



산점도

- 예시: 산점도 응용

```
> a <- 1:10
> x <- 1:100
> y <- data.frame(matrix(nrow=length(x), ncol=length(a)))
> rownames(y) <- x
> colnames(y) <- a
> for (i in 1:length(a)) {
+   y[i] <- a[i] * x
+ }
> head(y)
```

	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10
2	2	4	6	8	10	12	14	16	18	20
3	3	6	9	12	15	18	21	24	27	30
4	4	8	12	16	20	24	28	32	36	40
5	5	10	15	20	25	30	35	40	45	50
6	6	12	18	24	30	36	42	48	54	60

산점도

- 예시: 산점도 응용

```
par(bg="aliceblue")
plot(c(0,0), main="Lines of 'y=ax'", xlab="x", ylab="y=ax", xlim=c(min(x), max(x)),
     ylim=c(min(y), max(y)), type="n")

mycolor <- rainbow(length(a))
for (i in 1:length(a)) {
  lines(x, y[[i]], col=mycolor[i])
}

legend(x="topleft", legend=paste("a=", a), col=mycolor, lty=1, bg="white", cex=1.0)
```

줄기-잎 그림

- > stem(data, [options])
 - scale
 - 구간을 나누는 개수

```
> x
[1] 32 18 82 23 97 86 88 48 17  8 94 93 42 75 70 55 99 77 70 62 15 19 22 69 74 13 86 86 27 57 68
[32] 49 64 78 88 89 99 27 72 72 77 89 55 93 15 65 10 31 21 21 50 17  2 29 81 45 67 57 42 52 67 64
[63] 29 71 29 26 87  2 90 27 45  5 30 24 64 17 11 17 28 18 78 70 49 88 49 90 86  1 74 58 67 29 53
[94]  4 80 29  7 33 16 66
> stem(x)

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

0 | 1224578
1 | 0135567777889
2 | 112346777899999
3 | 0123
4 | 22558999
5 | 02355778
6 | 24445677789
7 | 0001224457788
8 | 0126666788899
9 | 00334799
```

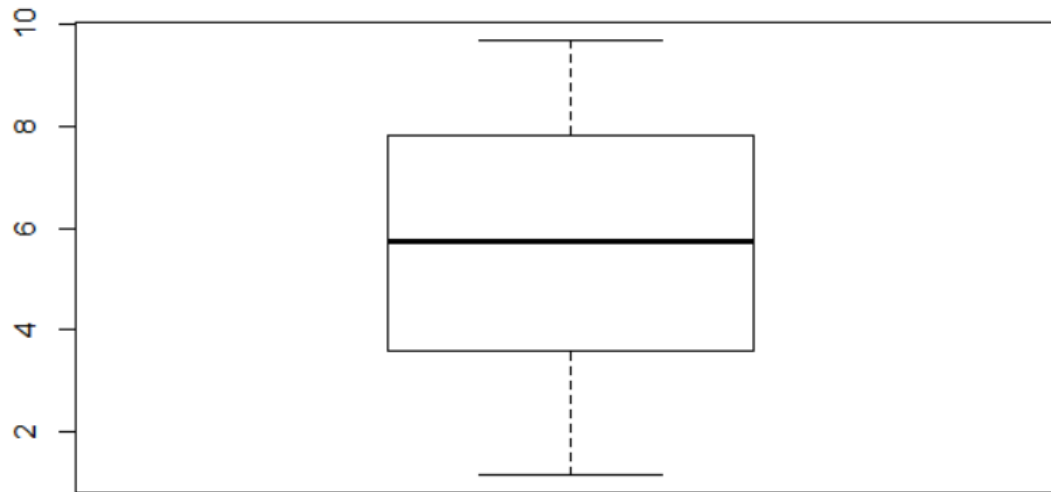
```
> stem(x, scale=2)

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

0 | 1224
0 | 578
1 | 013
1 | 5567777889
2 | 11234
2 | 6777899999
3 | 0123
3 |
4 | 22
4 | 558999
5 | 023
5 | 55778
6 | 2444
6 | 5677789
7 | 00012244
7 | 57788
8 | 012
8 | 6666788899
9 | 00334
9 | 799
```

상자그림

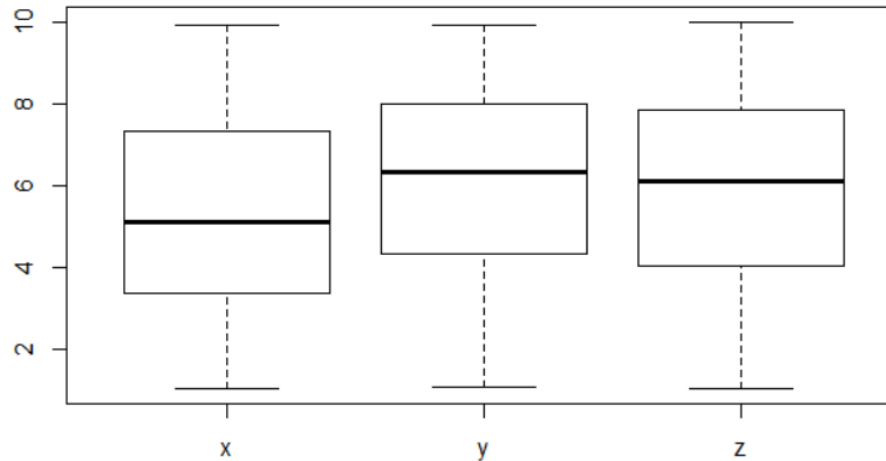
- `> boxplot(data, [options])`
 - `border`
 - 상자 선 색 지정



상자그림

- 예시: 다중 상자 그림 그리기

```
> x <- runif(100, 1, 10)
> y <- runif(100, 1, 10)
> z <- runif(100, 1, 10)
> x_ax <- as.factor(c(rep("x", 100), rep("y", 100), rep("z", 100)))
> y_ax <- c(x, y, z)
> boxplot(y~z)
> boxplot(y_ax~x_ax)
```



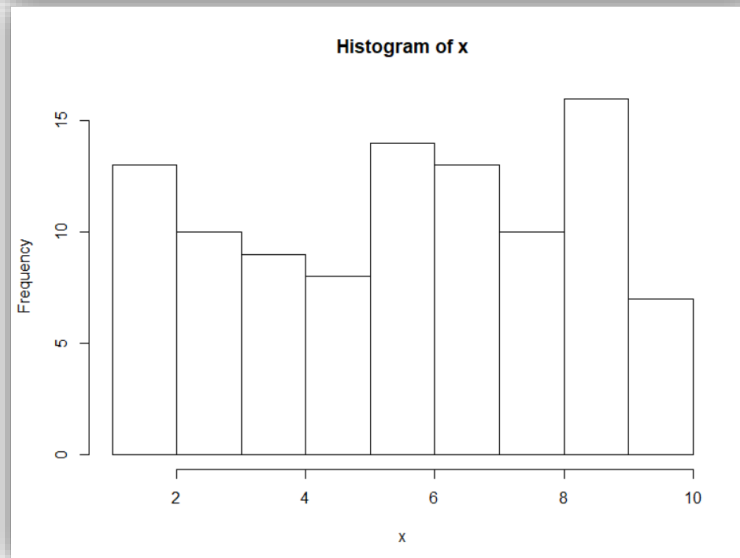
히스토그램

- `> hist(data, [options])`
 - `breaks` = 히스토그램 막대 수
 - `labels` = 히스토그램 막대 위에 도수 표시
 - `freq (prob)` = 빈도가 아닌 확률 밀도 함수 형태로 표시

히스토그램

- 예시 1: 히스토그램 그리기

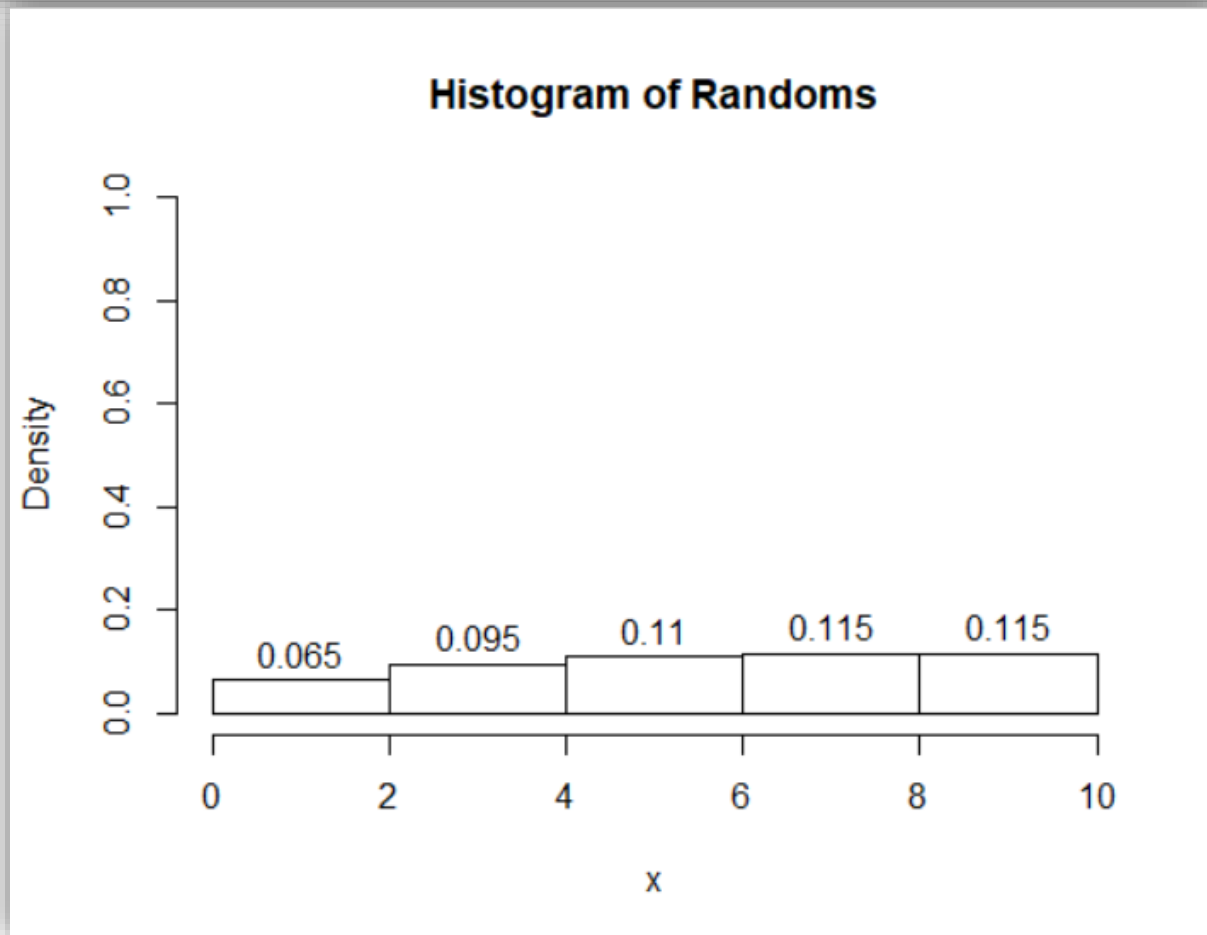
```
> x <- runif(100, 1, 10)
> x
 [1] 9.687878 5.361752 7.684970 5.890892 9.486441 8.990817 1.778546 2.556418 9.167090
[10] 6.918463 8.975320 5.088041 8.256633 3.521406 4.935289 8.534649 1.590649 2.597747
[19] 1.770116 4.426800 8.534667 2.300273 3.975043 8.910939 3.983841 7.917807 2.455266
[28] 6.951655 4.750135 1.664914 6.078144 5.149927 6.320336 4.526252 3.859522 8.954833
[37] 6.114400 5.015217 5.275733 7.389630 5.726237 3.693597 6.260783 7.762063 6.165961
[46] 5.596691 9.681056 3.012072 5.835603 7.033553 4.617885 7.616039 1.378892 8.772614
[55] 2.324877 6.051274 3.666828 8.929783 2.077721 2.977870 9.545429 5.629875 5.722350
[64] 8.185179 3.736131 6.001327 8.923592 6.380488 1.396490 4.518874 8.319436 6.229245
[73] 1.879125 3.842325 8.299026 8.066520 9.139971 2.278005 7.465290 1.506016 1.701886
[82] 2.092741 9.163889 1.666472 4.850785 8.367316 5.299230 5.504908 8.918442 6.878339
[91] 6.659174 5.728922 7.506349 7.877450 7.014684 1.911434 1.273684 4.960971 2.289566
[100] 1.163507
> hist(x)
```



히스토그램

- 예시 2: 히스토그램 그리기

```
> hist(x, ylim=c(0, 1), main="Histogram of Randoms", breaks=5, labels=TRUE, freq=FALSE)
```



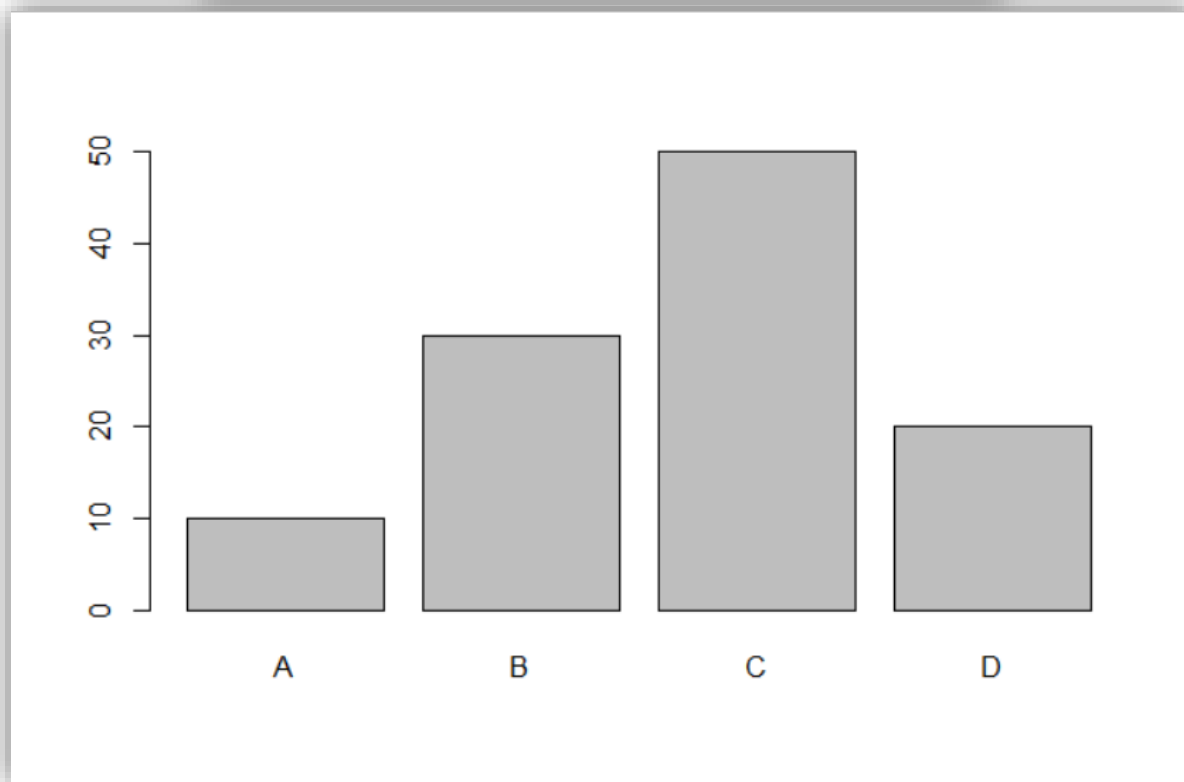
바 차트

- > barplot(data, [options])
 - names
 - 막대의 이름 지정 (분류)
 - horiz=T
 - 막대를 가로로 눕히기
 - beside=T
 - 여러 데이터를 막대로 그리기 위하여 행렬 데이터 출력 시 이용
 - 그룹별 막대를 중첩 누적하지 않고 여러 막대로 출력

바 차트

- 예시 1: 바 차트 그리기

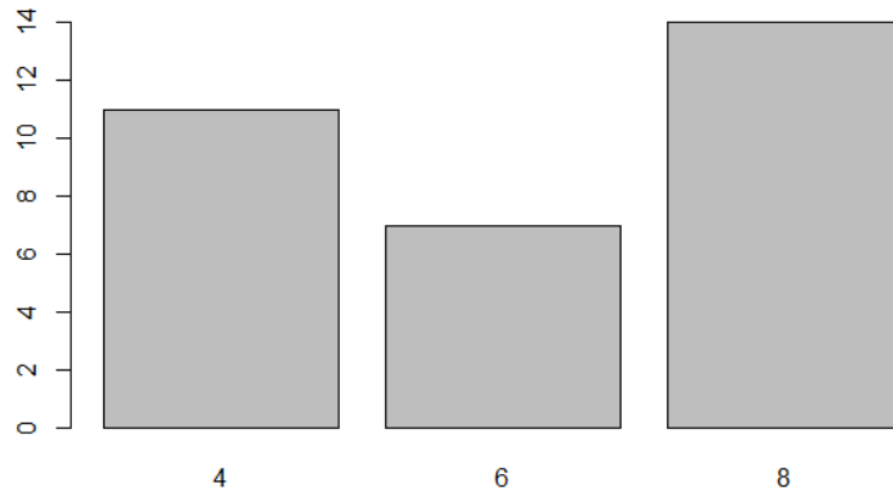
```
> y <- c(10, 30, 50, 20)  
> barplot(y, names=c("A", "B", "C", "D"))
```



바 차트

- 예시 2: 바 차트 그리기 (데이터)

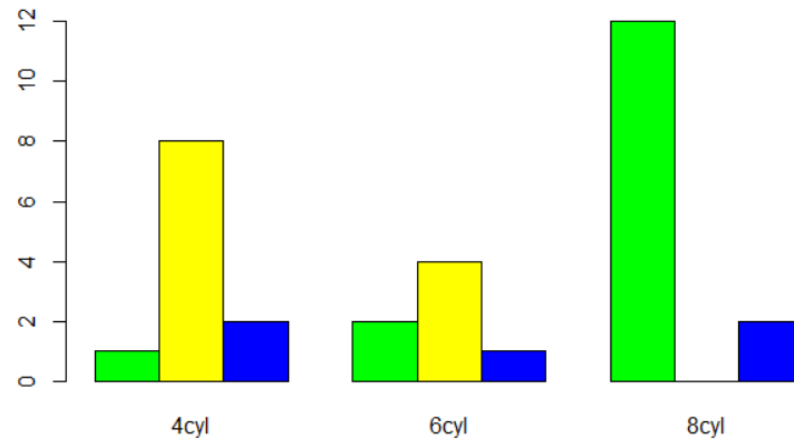
```
> cyl
[1] 6 6 4 6 8 6 8 4 4 6 6 8 8 8 8 8 8 4 4 4 4 8 8 8 8 4 4 4 8 6 8 4
> table(cyl)
cyl
 4  6  8
11  7 14
> barplot(table(cyl))
```



바 차트

- 예시 3: 중첩 바 차트

```
> m <- table(gear, cyl)
> m
      cyl
gear  4   6   8
  3    1   2  12
  4    8   4   0
  5    2   1   2
> barplot(m)
> barplot(m, names=c("4cyl", "6cyl", "8cyl"), col=c("green", "yellow", "blue"))
> barplot(m, beside=T, names=c("4cyl", "6cyl", "8cyl"), col=c("green", "yellow", "blue"))
```



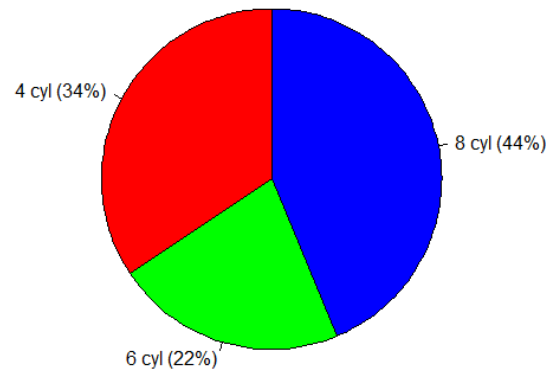
파이 차트

- > pie(data, [options])
 - label
 - 구성 데이터 이름 출력
 - init.angle
 - 파이 차트 출력 시작 각도
 - clockwise=T
 - 출력 시계/반시계 방향
 - 기본값 = 반시계 방향

파이 차트

- 예시: 파이 차트 그리기 (빈도 % 출력)

```
> data <- table(cyl)
> total <- sum(data)
> pct <- round(data/total*100)
> mylabel <- names(data)
> mylabel <- paste(mylabel, " cyl (", pct, "%)", sep="")
> mylabel
[1] "4 cyl (34%)" "6 cyl (22%)" "8 cyl (44%)"
> pie(data, init.angle=90, label=mylabel, col=rainbow(length(data)))
```



그래프 패키지

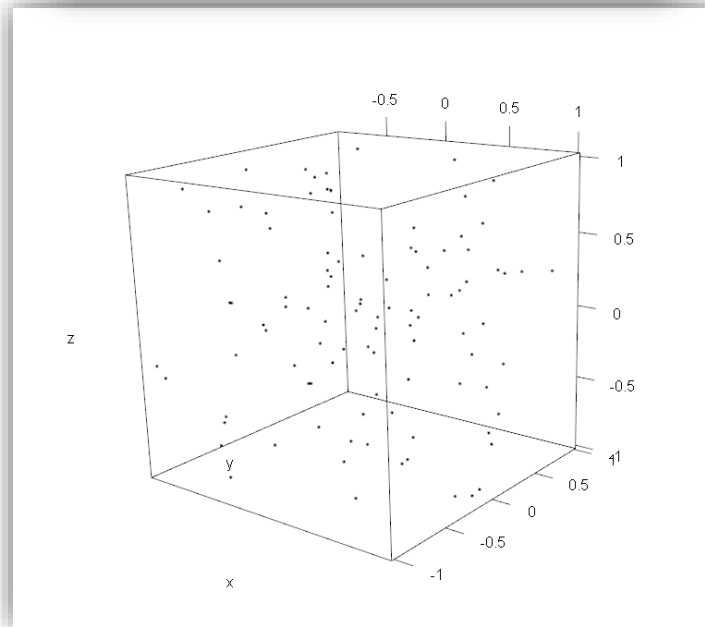
- 그래프 패키지
 - R의 강점 중 하나인 시각화 패키지들이 다수 존재
 - ggplot2
 - 대표적인 시각화 패키지
 - <https://ggplot2.tidyverse.org/reference/>
 - rgl
 - Interactive 3D 시각화 패키지

그래프 패키지

- 3D 산점도

- 'rgl' 패키지의 plot3d() 함수
- 3개의 변수에 대해 3차원 형태의 데이터 플로팅

```
> library(rgl)
경고메시지(들):
패키지 'rgl'는 R 버전 3.5.3에서 작성되었습니다
> x <- runif(100, -1, 1)
> y <- runif(100, -1, 1)
> z <- runif(100, -1, 1)
> plot3d(x, y, z)
```



그래프 패키지

- ggplot2 패키지 사용 예시

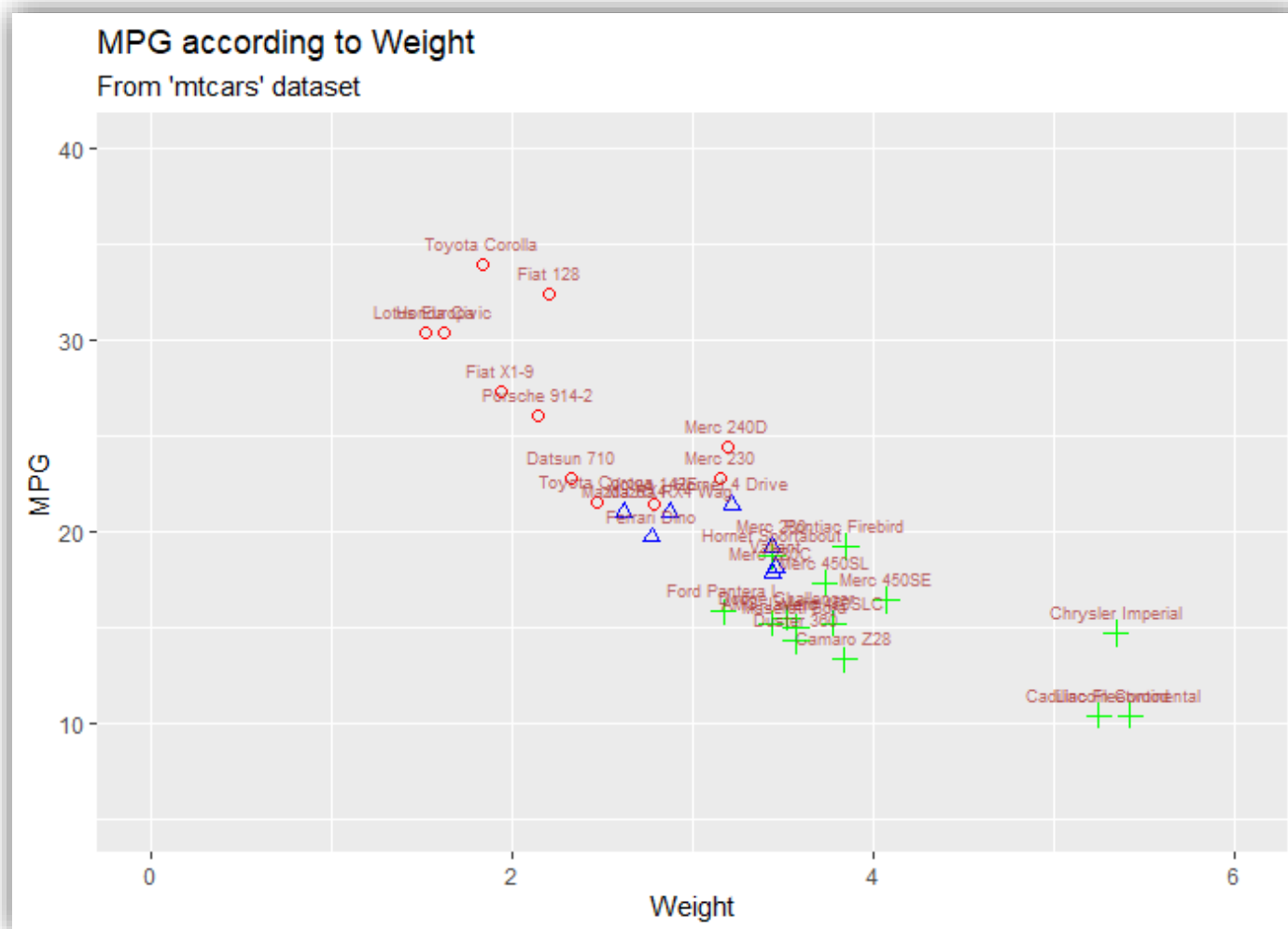
```
y <- ggplot(data=mtcars,
            mapping=aes(x=wt, y=mpg))
y <- y + geom_point(color=c("red", "blue", "green")[as.factor(cyl)],
                   pch=c(1, 2, 3)[as.factor(cyl)],
                   size=c(2, 2, 3)[as.factor(cyl)])
y <- y + coord_cartesian(xlim=c(0, 6),
                        ylim=c(5, 40))
y <- y + labs(title="MPG according to Weight",
             subtitle="From 'mtcars' dataset",
             x="Weight",
             y="MPG")
y <- y + annotate(geom="text",
                x=wt,
                y=mpg,
                label=rownames(mtcars),
                color="brown",
                alpha=0.7,
                size=2.5,
                hjust=0.5,
                vjust=-1)
```

y

1. Data 가져오기
2. 산점도 선택 (미적 요소)
3. 좌표 범위 설정
4. 레이블 및 제목 설정
5. 강조 표시 추가

그래프 패키지

- ggplot2 패키지 사용 예시



과제

- MASS 패키지의 Cats 데이터를 이용한다.
 - 고양이의 성별에 따른 개체 수를 확인하기 위한 막대 그래프 출력한다.
 - 함수 `summary()`를 이용하여 성별에 따른 개체의 수 내용을 파악한다.
 - 개체의 수를 막대 그래프로 표현하고, 적절한 축 제목 및 그래프 제목을 정한다.
 - 아래 사항을 준수하여 고양이의 몸무게에 따른 심장 무게를 확인하기 위한 그래프를 출력한다.
 - X축: Bwt (Body Weight), Y축: Hwt (Heart Weight)
 - X축 Label: "Body Weight (kg)", Y축 Label: "Heart Weight (g)"
 - X, Y축의 데이터 범위는 `summary()` 함수를 이용하여 최솟값보다 크지 않은 정수, 최댓값보다 작지 않은 정수로 설정한다.
 - 그래프 제목: "Heart Weight (g) by Body Weight (kg) of Cats"
 - 출력 기호: '#', 색상: RED

과제 제출

- 제출 목록
 - 작성한 코드 파일(.R)
 - 결과 출력 화면 (.PDF)
 - 터미널 캡처(이미지)를 Word 혹은 HWP 에 붙여넣어 PDF 로 변환
- 제출 방법
 - 위 목록의 파일들을 압축
 - 아래 서식으로 압축파일 이름 지정
 - 블랙보드를 통해 제출
- 제출 서식 (XX = 주차번호 ex. 01, 02, ...)
 - 파일 이름: 전산통계학_실습과제_XX주차_학번_이름.zip