8장. 데이터 시각화

시각화 단계

8. 시각화

- 데이터 가공
  - 데이터를 R Language에서 사용하기 쉬운 형태로 변환합니다.
  - 주로 base, stats, plyr, reshape, reshape2 등의 패키지를 사용합니다.
- 시각화
  - R Language를 사용하여 원 데이터를 시각화하거나 분석된 결과를 시각화합니다.
  - graphics, ggplot, ggplot2 등의 패키지를 기본적으로 사용합니다.
  - 생성된 시각화 이미지는 jpg, png, pdf 형태로 저장할 수 있습니다.
- 꾸미기
  - 시각화 결과를 보고 하거나 외부로 배포할 경우, 의미를 명확하게 전달하기 위해서 인포그래픽스 작업을 진행합니다.
  - 꾸미기 작업은 R Language가 아닌 외부 전문 이미지 편집 도구를 사용합니다.

시각화의 종류

종류	유사어	상세
산점도	점 그래프 Scatter plot	• 색 추가, 모양 추가, 크기 추가 • 산점도 행렬 (Scatter plot matrix)
선 그래프	Line graph	• line : 방향, path : 무방향 • Time series plot (시계열 그래프)
히스토그램	Histogram	• 연속형 변수의 빈도수 분포, 단일 변수 차트
밀도 그래프	Density	• 값이 아니라 밀도값으로 그린 선 그래프
막대 그래프	Bar chart	• 이산형 변수의 빈도수 분포표, 단일 변수 차트 • 도수분포표(Frequency table) 또는 누적 막대 그래프
박스 그래프	Boxplot	• 사분위수와 이상값 표시
모자이크 플롯	Mosaic plot	• 이산형 데이터의 다차원 도수 분포표
파이 차트	Pie chart	• 이산형 변수의 빈도수 분포, 단일 변수 차트

High level plOtting cOmmands

8. 시각화

고 수준 그래프 함수들은 그래프 영역에 그래프를 그릴 때 항상 새로운 그래프를 시작합니다. 그러므로 그래프 함수를 호출할 때마다 그림이 그려지는 영역을 초기화 하고 다시 그립니다.

고 수준 그래프 함수	설명
plot()	일반적인 기본 그래프 함수입니다.
barplot()	막대그래프 함수입니다.
boxplot()	박스플롯 그래프 함수입니다.
hist()	히스토그램 그래프 함수입니다.
curve()	수식을 그래프로 그립니다.
qqnorm()	분위수-분위수(Q-Q) 그래프 함수입니다.

Low level plotting commands

저 수준 그래프 함수는 새로운 그래프를 생성할 수는 없습니다. 그러므로 이미 그려진 그래

프에 점, 선, 텍스트 그리고 장식 등을 더하기 위해 사용합니다.

저 수준 그래프 함수	설명
points()	점을 추가하는 함수입니다.
lines()	선을 추가하는 함수입니다.
abline()	직선을 추가하는 함수입니다.
polygon()	닫힌 다각형을 추가하는 함수입니다.
text()	문자를 추가하는 함수입니다.
segments()	선분을 추가하는 함수입니다. segment(x1, y1, x2, y2)는 두 점 (x1, y1)과 (x2, y2)를 잇는 직선을 추가합니다.

그래프 파라미터

8. 시각화

파라미터	설명
ask	TRUE 이면 그래프를 여러 개 그릴 때 다음 그래프를 그리기 위해 사용자의 입력을 받습니다.
bg	배경색을 설정합니다. 기본배경색은 "white"입니다.
bty	그래프의 박스 타입(type of box)을 설정합니다. ("o", "L", "7", "c", "u", "]"), 문자 모양대로 테두리가 만들어짐, n은 박 스 테두리가 없습니다. o가 디폴트 이며, 박사의 상/하/좌/우 모두 테드리를 표시합니다.
cex	텍스트 또는 기호(symbol)의 크기를 설정합니다.
col	기호의 색을 설정합니다. 숫자도 가능합니다. (1:black, 2:red, 3:green, 4:blue, 5:cyan, 6:purple, 7:yellow, 8:grey)
font	텍스트의 폰트를 설정합니다. (1: normal, 2: italics, 3: bold, 4: bold italics, 5: expected)
las	axis label 스타일을 설정합니다. (0: axes와 평행, 1: 수평, 2: 축에 수직, 3: 수직)
lty	선의 유형을 설정합니다. (0=blank, 1=solid (default), 2=dashed, 3=dotted, 4=dotdash, 5=longdash, 6=twodash) 또 는 ("blank", "solid", "dashed", "dotted", "dotdash", "longdash", "twodash", "blank")
mar	그래프의 여백을 조정합니다. par(mar=c(bottom margin, left, top, right)) 형식으로 설정합니다.
mfcol	n개의 행으로 된 그래프 행렬을 만듭니다. c(nr,nc)를 matrix형태로 분할해서 그래프를 열 순서로 그립니다.
mfrow	n개의 열로 된 그래프 행렬을 만듭니다. c(nr,nc)를 matrix형태로 분할해서 그래프를 행 순서로 그립니다.
pch	점의 표시 유형을 설정합니다. pch="임의의문자"를 이용하면 임의의 문자가 점 대신 출력되며, pch=n를 이용하면, pch simbol 마크를 출력합니다.
ps	글꼴을 설정합니다. (1: normal, 2: italics, 3: bold, 4: bold italics, 5: expected)
new	기본값은 FALSE입니다. TRUE이면 고수준 그래프 함수를 이용하더라도 이전 그래프를 삭제하지 않고 그립니다.

par()

par() 함수는 그래프를 조정하거나 그래프창의 특성을 지정하기 위해서 사용하며, 선의 굵기와 종류, 문자의 크기와 글꼴, 색상 등 다양한 변경이 가능합니다. 전역변수를 수정하는 것이므로 모든 그래픽에 영향을 미칩니다. par()함수의 리턴 값은 파라미터 설정 전 객체가 반환됩니다. 이를 이용하면 그래프를 그리기 전 파라미터 상태를 저장해 둘 수 있습니다.

```
> oldPar <- par(bty="L")    #  파라미터 지정 전의 객체를 저장합니다.
> plot(cars)                # 변경된 파라미터대로 그래프가 그려집니다.
> par(oldPar)               # 원래 파라미터로 설정을 되돌린다.
> plot(cars)                # 원래 파라미터로 그래프가 그려집니다.
```

파라미터 변경은 par()함수를 이용하거나, 그래프 함수의 인자로 설정할 수 있습니다.

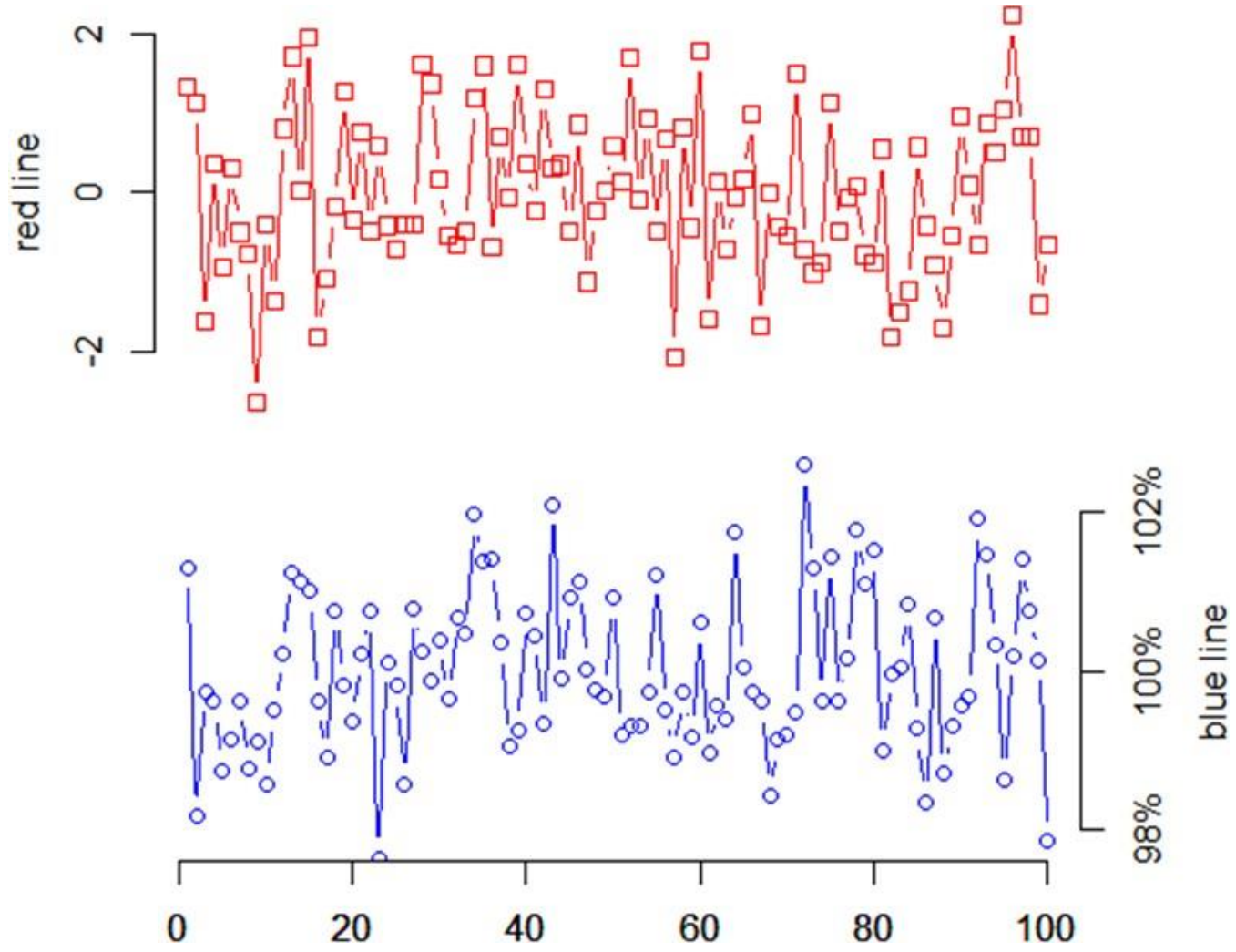
```
> plot(cars, bty="7")   #그래프를 그릴 때 만 파라미터가 설정됩니다.
> plot(cars)
```

par() 사용 예

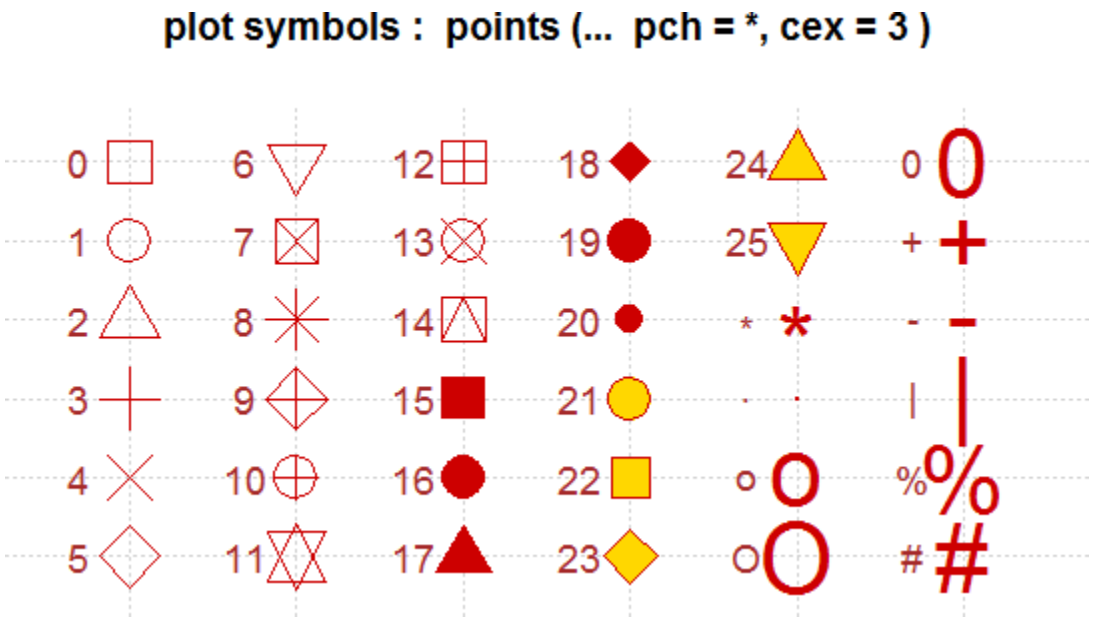
다음 코드는 파라미터를 사용한 예입니다.

```
> x <- 1:100
> y1 <- rnorm(100)   #평균 0, 표준편차 1인 정규분포를 따르는 데이터 100개
> y2 <- rnorm(100) + 100

> oldPar <- par(mar=c(5,5,5,5))
> # (1) draw red line
> plot(x, y1, pch=0, type="b", col=2, yaxt="n", ylim=c(-8,2), ylab="", bty="n")
> axis(side=2, at=c(-2,0,2))
> mtext("red line", side=2, line=2.5, at=0)

> par(new=TRUE)
> # (2) draw blue line
> plot(x, y2, pch=1, type="b", col="blue",
+      yaxt="n", ylim=c(98, 108), ylab="", bty="n")
> axis(side=4, at=c(98, 100, 102), label=c("98%", "100%", "102%"))
> mtext("blue line", side=4, line=2.5, at=100)

> par(oldPar)
```

par() 사용 예



pch symbol


plot symbols : points (... pch = *, cex = 3)
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
*
.
o
0
+
-
|
%
#

plot()

8. 시각화

```
plot(x, y, ...)
```

구문에서...

- x : 그래프의 x축 점 좌표들입니다. plot() 함수는 함수 또는 임의의 R 객체를 가질 수도 있습니다.
- y : 그래프의 y축 점 좌표들입니다.
- ... : 함수에 전달할 인수들입니다. 그래프 파라미터들이 포함될 수 있습니다. 앞의 그래프 파라미터에서 설명하지 않은 인수들에는 type, main, sub, xlab, ylab, asp 등이 있습니다.
  - type : 어떤 유형의 그래프가 그려질지를 지정합니다. p(points; 산점도 그래프), l(lines; 선 그래프), b(both; 점과 선을 잇는 그래프), c(b에서 p를 뺀 그래프), o(overplotted; 점과 선이 중첩된 그래프), h(histogram; 히스토그램 그래프), s(stair steps; 계단 그래프), S(stair steps; 거꾸로 그린 계단 그래프), n(그래프 표시 없음) 등이 있습니다. 기본값은 산점도 그래프를 그리는 p입니다.
  - main : 그래프의 제목을 지정합니다.
  - sub : 그래프의 부제목을 지정합니다.
  - xlab : x 축의 제목을 지정합니다.
  - ylab : y 축의 제목을 지정합니다.
  - asp : y/x 종횡비를 지정합니다.

plot() 예

다음 코드는 cars 데이터셋의 산점도 그래프를 그립니다. 그래프의 제목과, x 축, y 축의 제목, 그리고 축 눈금의 레이블은 수평으로 나타냅니다.

```
> plot(cars, main="Speed and Stopping Distances of Cars",
+      xlab="Speed (mph)", ylab="Stopping distance (ft)", las=1)
```

다음 그림은 cars 데이터를 이용하여 산점도 그래프를 그린 결과입니다.

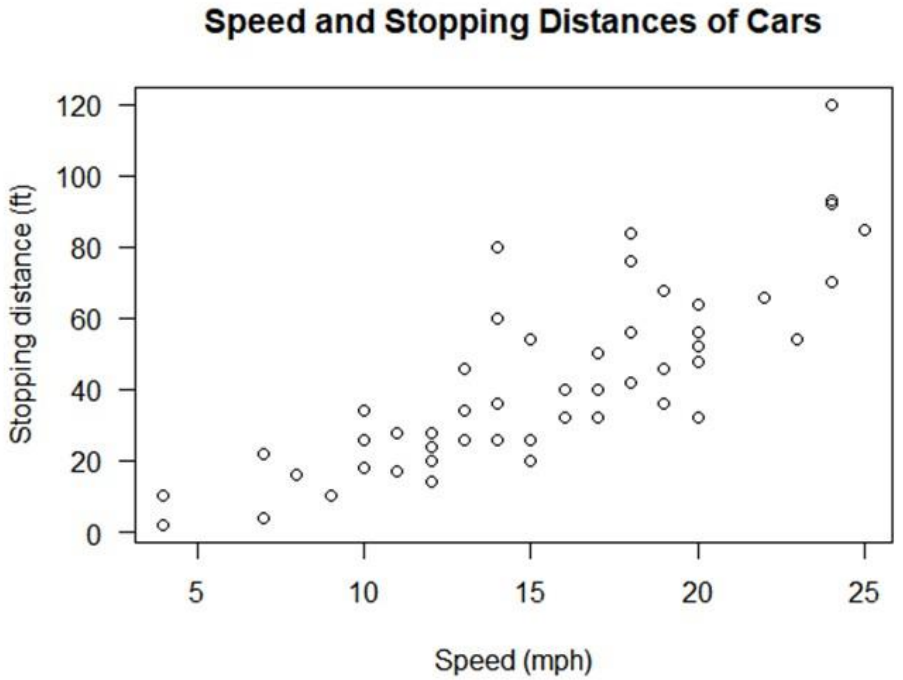
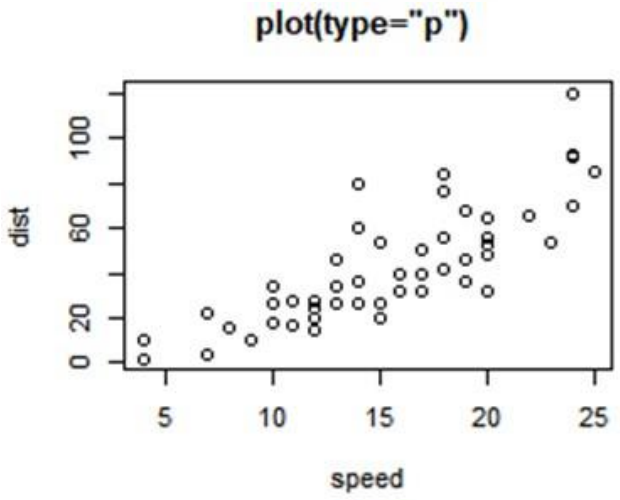
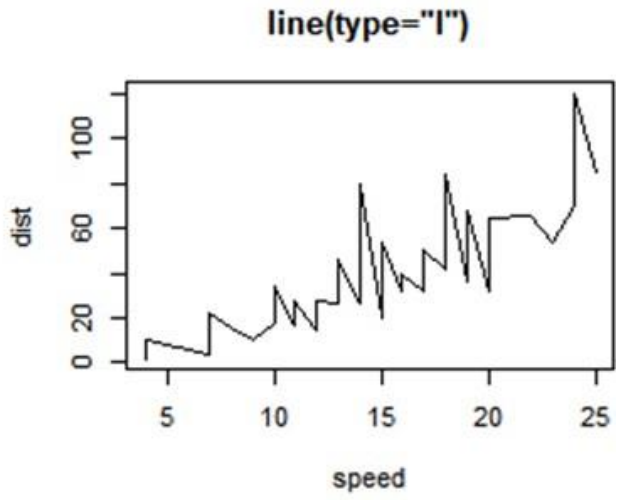
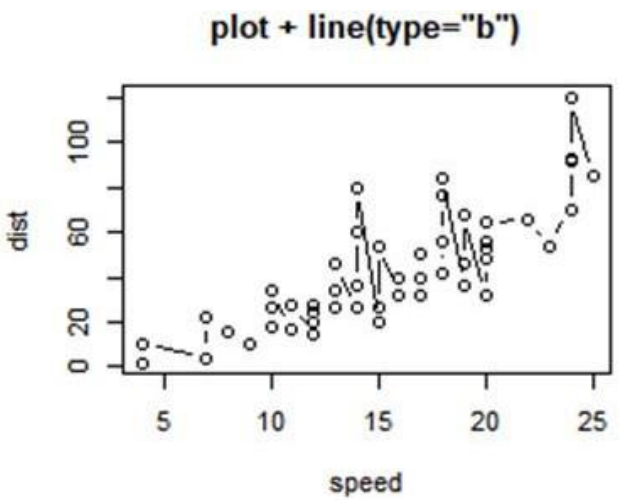


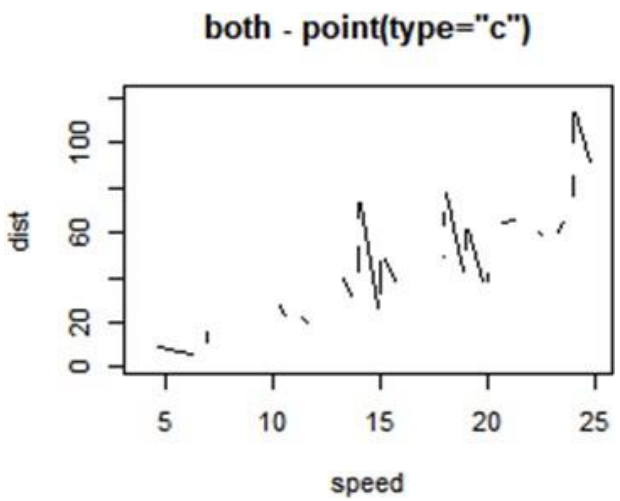
그림 3. cars 산점도 그래프

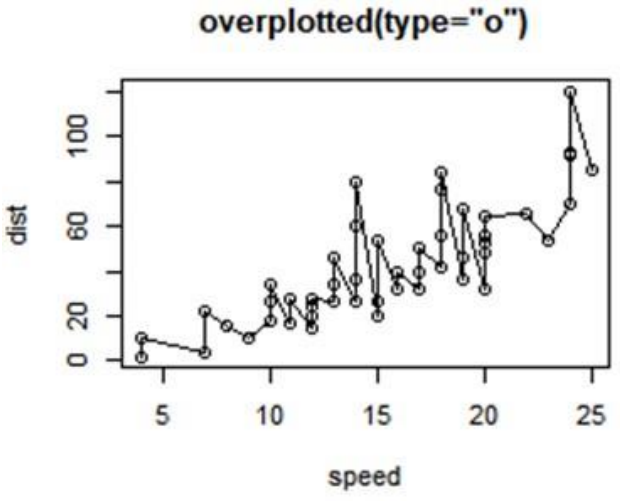
plot() 함수의 type 파라미터

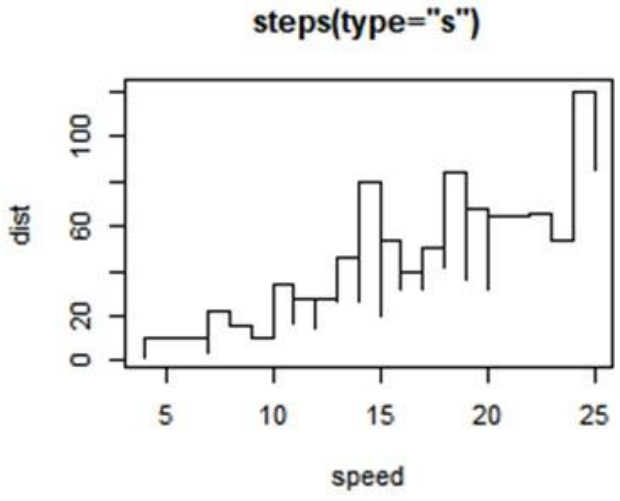












barplot()

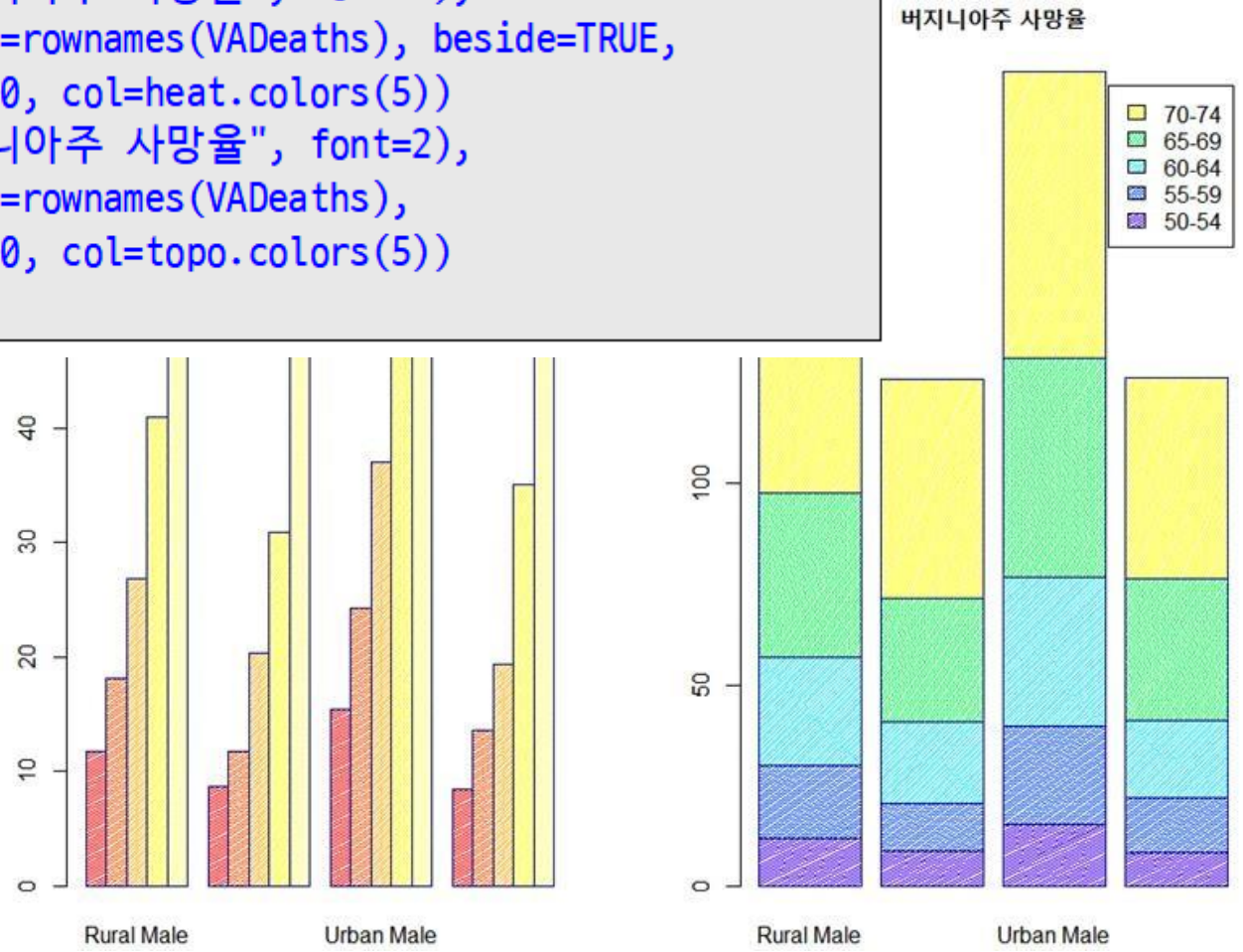

```
barplot(height, ...)
```

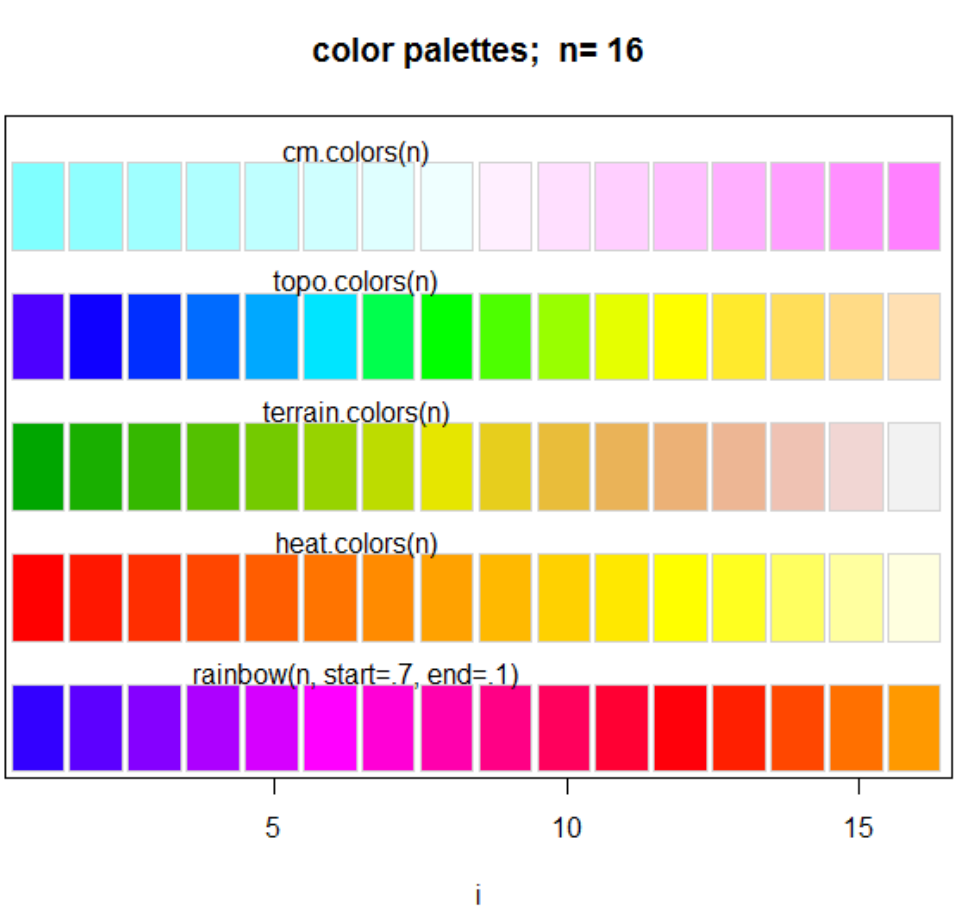
구문에서...

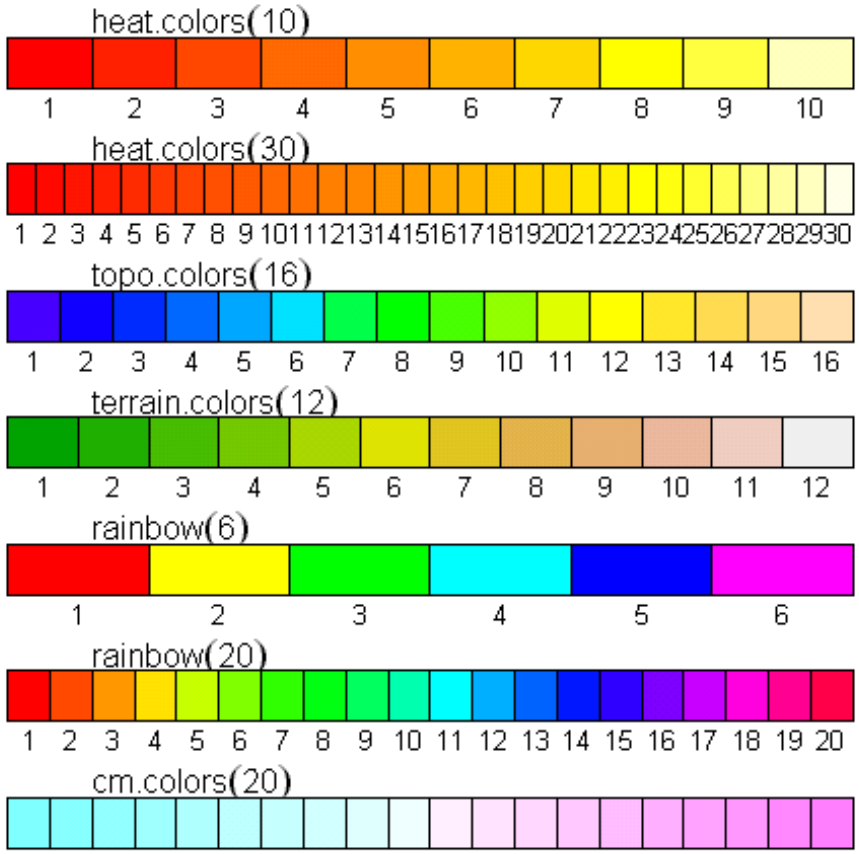
- height : 그래프를 구성하는 막대를 설명하는 값의 벡터 또는 행렬입니다. 높이가 벡터 인 경우 그래프는 벡터의 값으로 지정된 높이의 사각형 막대들로 구성됩니다. 높이가 행렬이고 beside=FALSE이면 그래프의 각 막대는 높이 열에 해당하며 열의 값은 막대를 구성하는 누적 된 막대의 높이를 나타냅니다. 높이가 행렬이고 besides=TRUE 인 경우 각 열의 값은 누적되지 않고 나란히 배치됩니다.

barplot() 예


```
> op <- par(mfrow=c(1,2))
> barplot(VADeaths, main=list("버지니아주 사망율", font=2),
+         border="dark blue", legend=rownames(VADeaths), beside=TRUE,
+         angle=15+10*1:5, density=50, col=heat.colors(5))
> barplot(VADeaths, main=list("버지니아주 사망율", font=2),
+         border="dark blue", legend=rownames(VADeaths),
+         angle=15+10*1:5, density=50, col=topo.colors(5))
> par(op)
```




color palettes; n= 16
cm.colors(n)
topo.colors(n)
terrain.colors(n)
heat.colors(n)
rainbow(n, start=.7, end=.1)
5
10
15
i


heat.colors(10)
1 2 3 4 5 6 7 8 9 10
heat.colors(30)
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 26 27 28 29 30
topo.colors(16)
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
terrain.colors(12)
1 2 3 4 5 6 7 8 9 10 11 12
rainbow(6)
1 2 3 4 5 6
rainbow(20)
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
cm.colors(20)

boxplot()

주어진 값을 이용해 box-and-whisker(사분위수) 그래프를 생성합니다.

```
boxplot(formula, data=NULL, ..., subset, na.action=NULL)
```

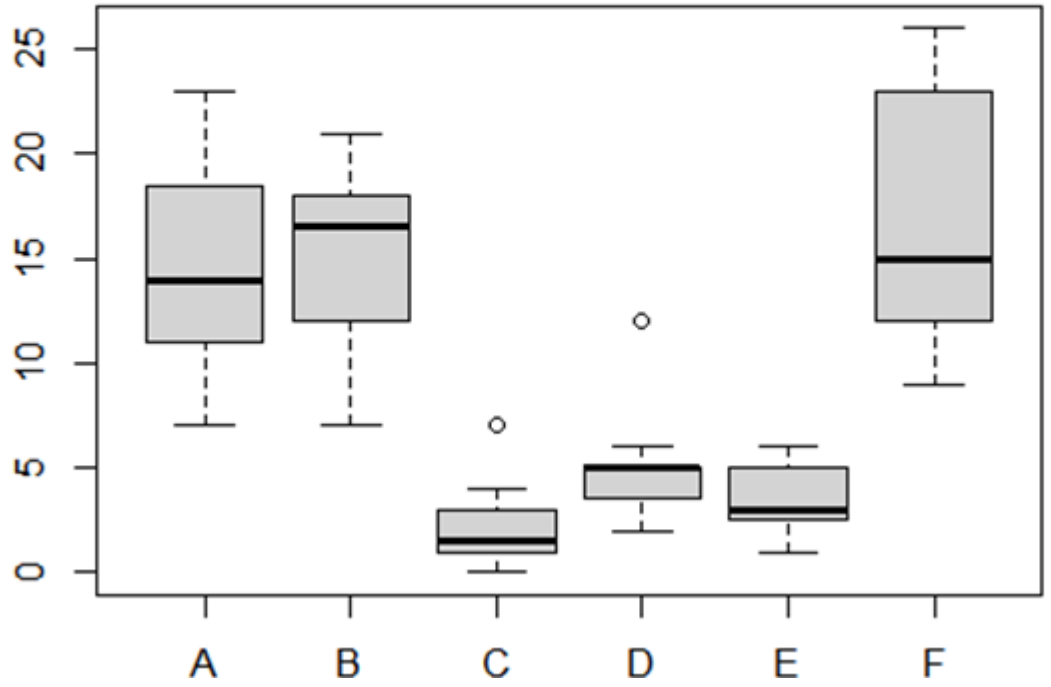
구문에서...

- formula : y~grp 형식의 포뮬라입니다. y는 그룹화 변수 grp(일반적으로 팩터)에 따라 그룹으로 나눌 데이터의 숫자 벡터입니다.
- data : 포뮬라에서 변수를 취해야 하는 데이터 프레임 또는 리스트입니다.
- subset : 그래프를 그리기 위해 사용되는 관측치의 서브 세트를 지정하는 벡터입니다. 이 인수는 선택사항입니다.
- na.action : 데이터에 NA가 포함될 때 실행 할 함수입니다. 기본값(NULL)은 응답 또는 그룹에서 누락 된 값을 무시하는 것입니다.

boxplot() 예

다음 코드는 InsectSparys 데이터셋을 이용해 사분위수 그래프를 그립니다.

```
> boxplot(count ~ spray, data=InsectSprays, col="lightgray")
```



hist()

hist() 함수는 주어진 데이터 값의 히스토그램을 계산합니다. plot=TRUE 인 경우 결과가 반환되기 전에 plot.histogram에 의해 "histogram"클래스의 결과 객체가 그려집니다.

```
hist(x, breaks="Sturges", ...)
```

구문에서...

- x : 히스토그램이 필요한 벡터입니다.
- breaks : 히스토그램 셀을 나누기 위한 벡터 또는 셀의 수를 지정합니다. 또는 셀의 수를 계산하기 위한 함수 또는 알고리즘을 지정할 수 있습니다. Sturges는 데이터의 범위를 이용해 크기를 나누는 것을 의미합니다.

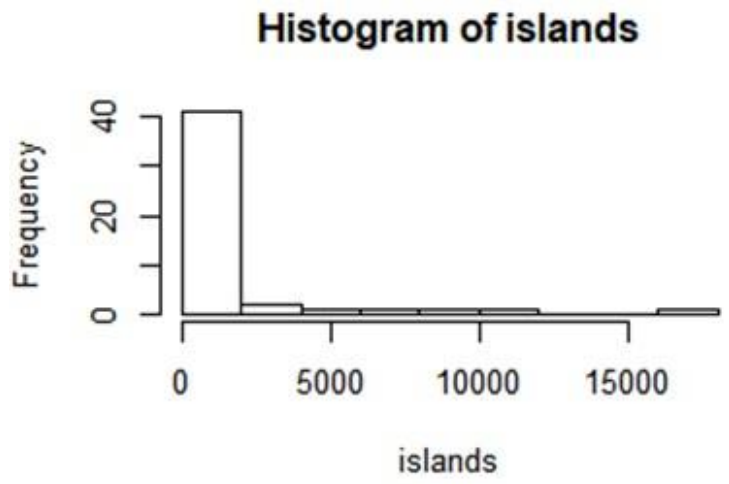
실습을 위해 islands 데이터셋을 사용합니다. islands 데이터셋은 1만 평방마일을 초과하는 주요 대륙의 넓이 정보입니다.

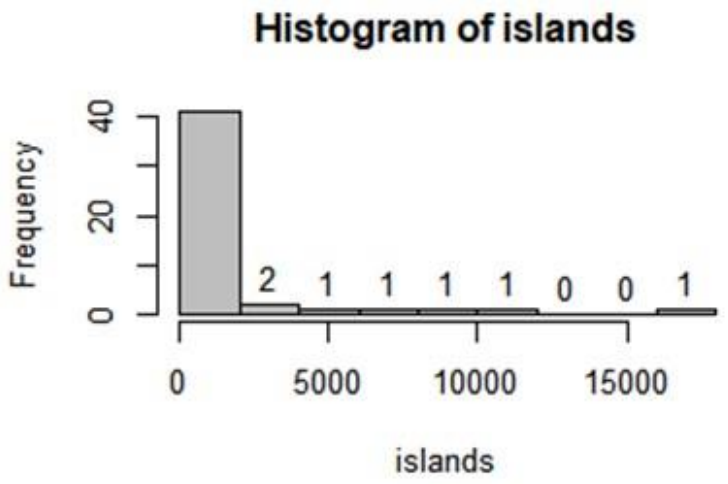
hist() 예

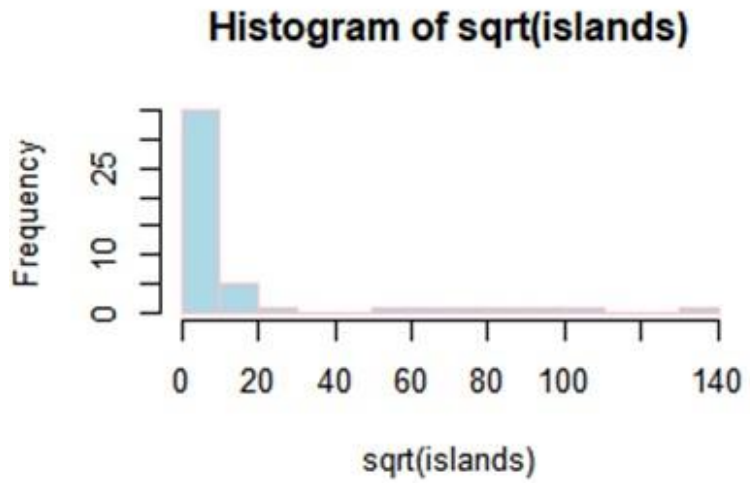
다음 코드는 islands 데이터셋을 이용하여 히스토그램을 그립니다.

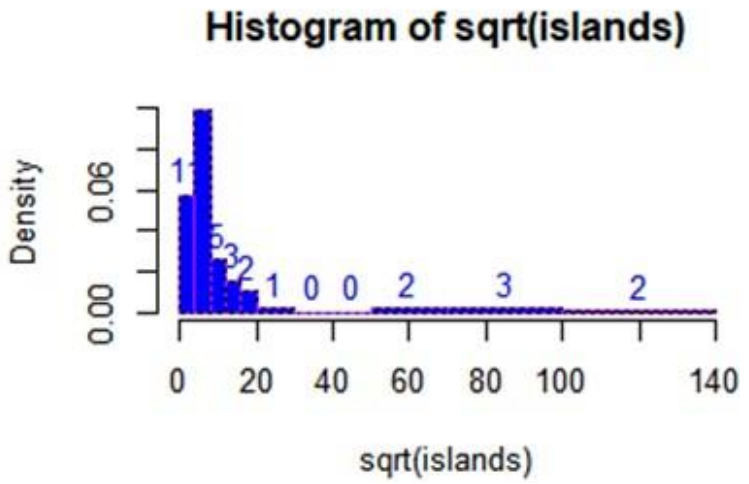
```
> op <- par(mfrow=c(2, 2))
> hist(islands)
> utils::str(hist(islands, col="gray", labels=TRUE))
List of 6
 $ breaks  : num [1:10] 0 2000 4000 6000 8000 10000 12000 14000 16000 18000
 $ counts  : int [1:9] 41 2 1 1 1 1 0 0 1
 $ density : num [1:9] 4.27e-04 2.08e-05 1.04e-05 1.04e-05 1.04e-05 ...
 $ mids    : num [1:9] 1000 3000 5000 7000 9000 11000 13000 15000 17000
 $ xname   : chr "islands"
 $ equidist: logi TRUE
 - attr(*, "class")= chr "histogram"
> hist(sqrt(islands), breaks=12, col="lightblue", border="pink")
> r <- hist(sqrt(islands), breaks=c(4*0:5, 10*3:5, 70, 100, 140),
+           col="blue1")
> text(r$mids, r$density, r$counts, adj=c(.5, -.5), col="blue3")
> sapply(r[2:3], sum)
   counts   density
48.000000  0.215625
> sum(r$density * diff(r$breaks))
[1] 1
> lines(r, lty=3, border="purple") # -> lines.histogram(*)
> par(op)
```

hist() 예


Histogram of islands
Frequency
0
20
40
0
5000
10000
15000
islands


Histogram of islands
Frequency
0
20
40
2
1
1
1
1
0
0
1
0
5000
10000
15000
islands


Histogram of sqrt(islands)
Frequency
0
10
25
0
20
40
60
80
100
140
sqrt(islands)


Histogram of sqrt(islands)
Density
0.00
0.06
1
1
3
2
1
0
0
2
3
2
0
20
40
60
80
100
140
sqrt(islands)

pie()

8. 시각화

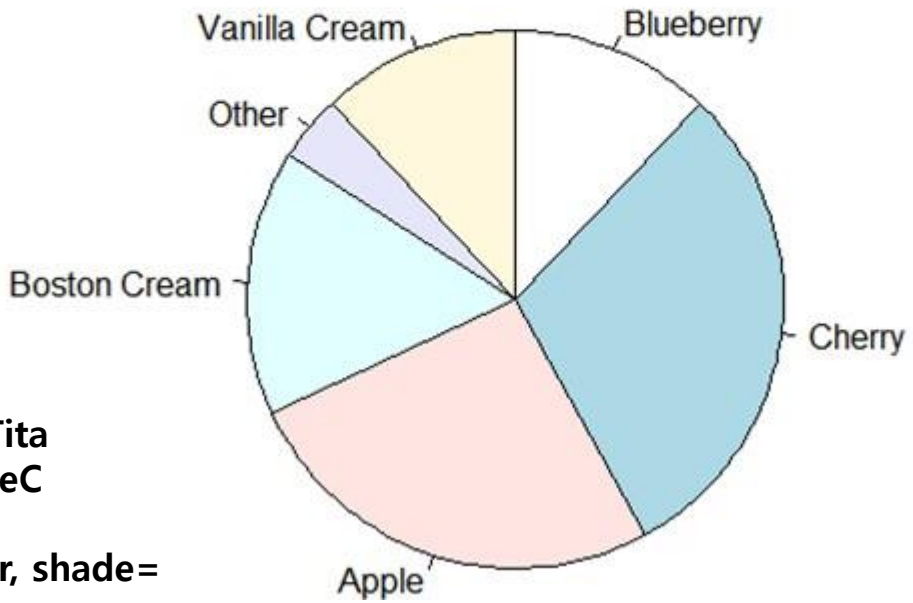
```
pie(x, labels=names(x), edges=200, radius=0.8,
    clockwise=FALSE, init.angle=if(clockwise) 90 else 0,
    density=NULL, angle=45, col=NULL, border=NULL,
    lty=NULL, main=NULL, ...)
```

구문에서...

- x : 음수가 아닌 양의 벡터입니다. x의 값은 파이 조각의 영역으로 표시됩니다.
- labels : 슬라이스에 이름을 지정하는 하나 이상의 표현식 또는 문자열입니다. 다른 객체는 as.graphicsAnnot에 의해 강제로 형변환 됩니다. 비어 있거나 NA(강제 변환 이후) 레이블의 경우 레이블이나 지시선이 그려지지 않습니다.
- edges : 파이의 원형 윤곽은 이 수만큼 모서리를 가진 다각형으로 근사됩니다.
- radius : 파이는 측면이 -1에서 1 사이 인 정사각형 상자의 가운데에 그려집니다. 조각을 표시하는 문자열이 길면 작은 반경을 사용해야 할 수도 있습니다.
- clockwise : 슬라이스가 시계 방향(TRUE) 또는 반 시계 방향(FALSE, 수학적으로 양의 방향)으로 그려져야 하는 것을 논리값으로 표시합니다. 반 시계 방향이 기본값입니다.
- init.angle : 슬라이스의 시작 각도 (도)를 지정하는 숫자입니다. 기본값은 clockwise가 FALSE(시계방향) 이면 0(3시 방향)이며, clockwise가 TRUE 이면 9(12시 방향)입니다.
- density : 1 인치 당 선의 음영 선의 밀도입니다. 기본값 NULL은 음영 라인이 그려지지 않은 것을 의미합니다. 양의 값이 아닌 경우 음영 선을 그릴 수도 없습니다.
- angle : 음영 선의 기울기를 각도로 표시(시계 반대 방향) 합니다.
- col : 조각을 채우거나 음영 처리 할 때 사용할 색상 벡터입니다. 누락 된 경우 par("fg")가 사용될 때 밀도가 지정되지 않은 한 6 개의 파스텔 색상 세트가 사용됩니다.
- border, lty : 각 슬라이스를 그리는 다각형에 전달 된 인수입니다. 경계선과 선의 타입을 지정합니다.
- main : 그래프의 주 제목을 지정합니다.

pie() 예

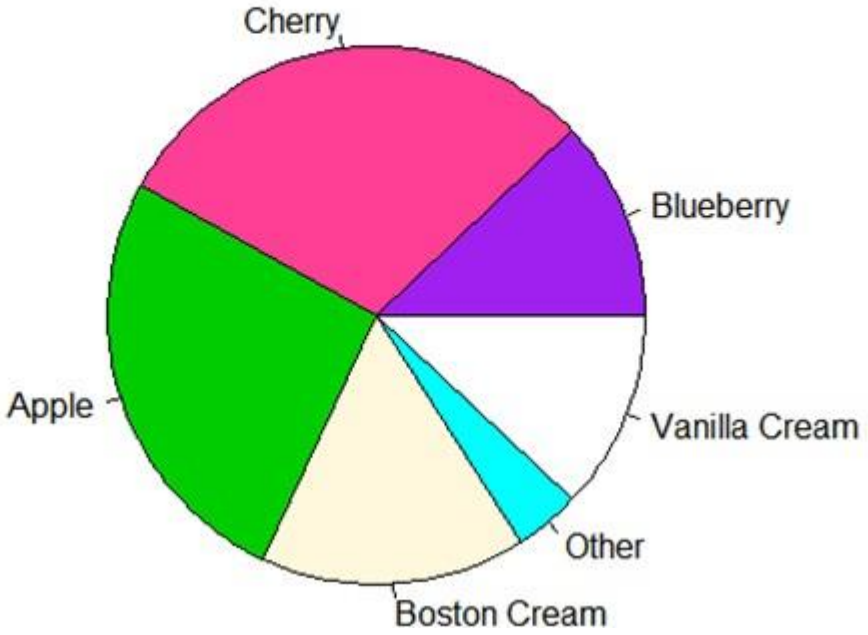

```
> pie.sales <- c(0.12, 0.3, 0.26, 0.16, 0.04, 0.12)
> names(pie.sales) <- c("Blueberry", "Cherry",  "Apple", "Boston Cream",
"Other", "Vanilla Cream")
> pie(pie.sales, clockwise=TRUE) # 기본색상
```



str(Titanic) mosaicplot(Titanic, color=T) str(HairEyeColor)
mosaicplot(HairEyeColor, shade=T)

pie() 예


```
> pie(pie.sales,
      col=c("purple", "violetred1", "green3", "cornsilk", "cyan", "white"))
```



mosaicplot

8. 시각화

```
mosaicplot(x, main=deparse(substitute(x)),
           sub=NULL, xlab=NULL, ylab=NULL,
           sort=NULL, off=NULL, dir=NULL,
           color=NULL, shade=FALSE, margin=NULL,
           cex.axis=0.66, las=par("las"), border=NULL,
           type=c("pearson", "deviance", "FT"), ...)
```

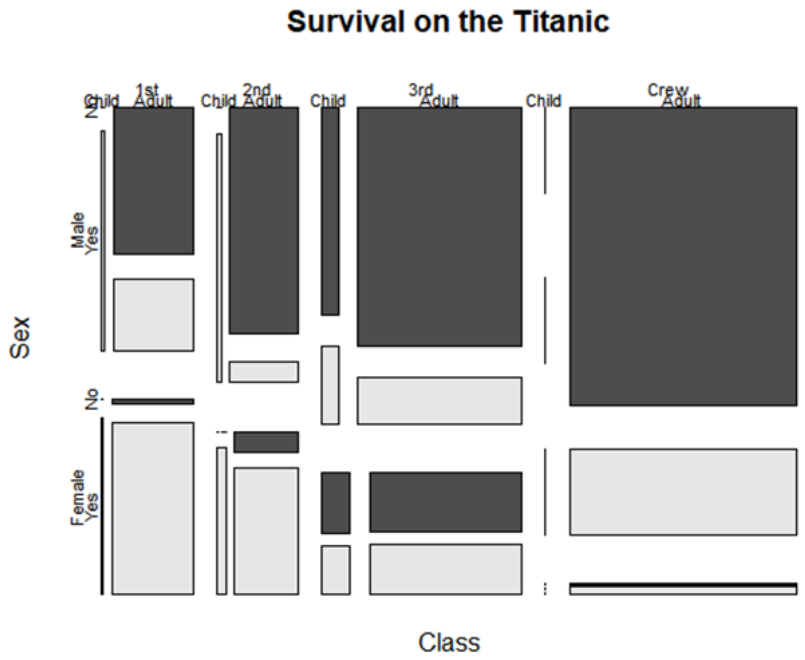
구문에서...

- x : dimnames(x) 속성에 카테고리 레이블이 지정된 배열 형식의 표입니다.
- main : 모자이크 플롯의 제목을 지정합니다.
- sub : 모자이크 플롯의 부 제목을 지정합니다.
- xlab, ylab : 플롯에 사용되는 x축 및 y축 레이블입니다.
- sort : 변수의 순서를 지정합니다. 이것은 정수 1부터 length(dim(x)) 사이의 정수 순열을 포함합니다.
- off : 모자이크의 각 레벨에서 백분율 간격을 결정하는 오프셋 벡터(적절한 값은 0과 20 사이이고 기본값은 2차원 테이블의 분할 수의 20배, 그렇지 않으면 10)입니다. 최대 50으로 축소되고 만일 필요한 경우에 재사용됩니다.
- dir : 모자이크의 각 레벨에 대한 분할 방향 벡터(수직의 경우 "v", 수평의 경우 "h")입니다. 테이블의 각 차원에 대한 한 방향입니다. 기본값은 수직 분할로 시작하는 교차 방향으로 구성됩니다.
- color : 색상 쉐이딩을 위한 색상의 논리 또는 벡터입니다. 음영이 거짓이거나 NULL(기본값) 인 경우에만 사용됩니다. 기본적으로 회색 상자가 그려집니다. color=TRUE는 감마 보정 된 회색 팔레트를 사용합니다. color=FALSE는 음영이 없는 빈 상자를 제공합니다.
- shade : 확장 된 모자이크 플롯을 생성할지 여부를 나타내는 논리 또는 잔차에 대한 절단점의 절대 값을 제공하는 최대 5 개의 양수 인 숫자 벡터입니다. 기본적으로 shade=FALSE이며 간단한 모자이크가 만들어집니다. shade=TRUE를 사용하면 2와 4의 절대 값으로 자릅니다.
- margin : 로그 선형 모델에 적합해야하는 벡터의 목록입니다.
- cex.axis : 축 주석에 사용되는 배율입니다. "cex"의 배수를 의미합니다.
- las : 축 레이블의 스타일을 숫자로 지정합니다. 일반적인 그래프 파라미터의 las와 같습니다. (0: axes와 평행, 1: 수평, 2: 축에 수직, 3: 수직)
- border : 테두리 색을 지정합니다.

mosaicplot 예

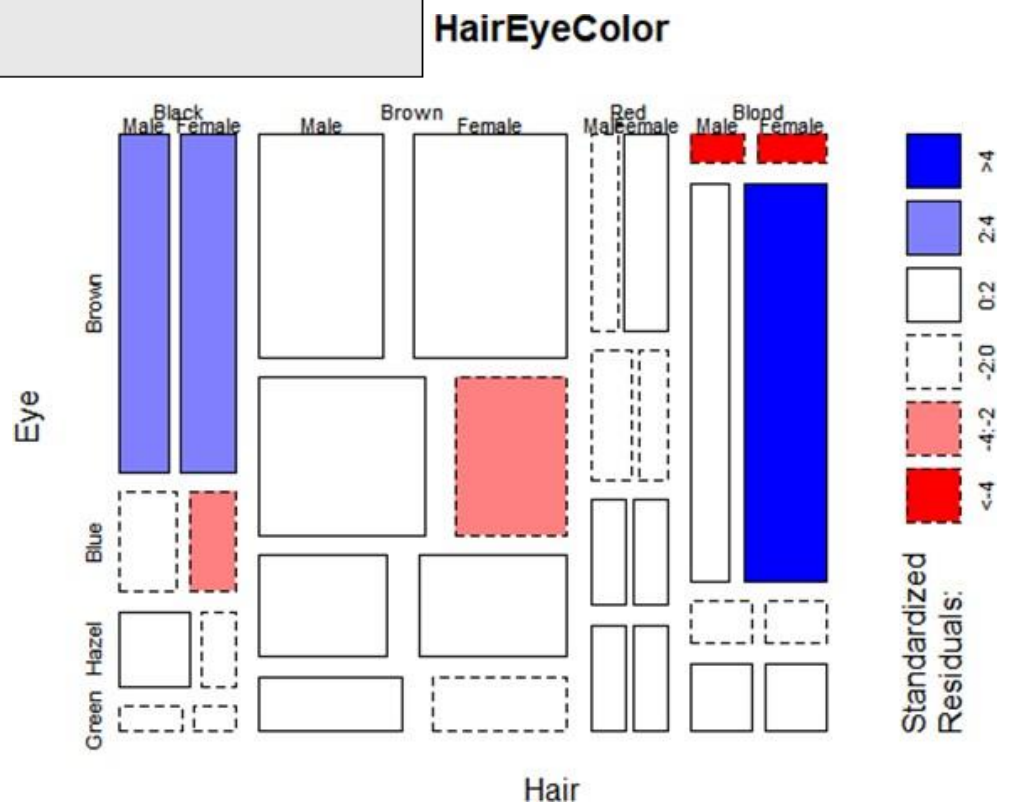
다음 코드는 타이타닉(Titanic)[39] 데이터셋을 이용해 모자이크 플롯을 출력하는 예입니다.

```
> str(Titanic)
 table [1:4, 1:2, 1:2, 1:2] 0 0 35 0 0 0 17 0 118 154 ...
 - attr(*, "dimnames")=List of 4
  ..$ Class   : chr [1:4] "1st" "2nd" "3rd" "Crew"
  ..$ Sex     : chr [1:2] "Male" "Female"
  ..$ Age     : chr [1:2] "Child" "Adult"
  ..$ Survived: chr [1:2] "No" "Yes"
> mosaicplot(Titanic, main="Survival on the Titanic", color=TRUE)
```



mosaicplot 예


```
> str(HairEyeColor)
 table [1:4, 1:4, 1:2] 32 53 10 3 11 50 10 30 10 25 ...
 - attr(*, "dimnames")=List of 3
  ..$ Hair: chr [1:4] "Black" "Brown" "Red" "Blond"
  ..$ Eye : chr [1:4] "Brown" "Blue" "Hazel" "Green"
  ..$ Sex : chr [1:2] "Male" "Female"
> mosaicplot(HairEyeColor, shade=TRUE)
```



- shade=TRUE이기 때문에 잔차를 2와 4의 절대값으로 자른 후 이를 색으로 표시해 줍니다.
- 이 모자이크 플롯은 머리카락과 눈 색깔과 성별의 독 립 모델에 있어서 독립의 경우에 예상보다 푸른 눈동 자 금발 여성이 더 많고 (잔차가 4보다 큼) 갈색 눈동자 여성의 수가 너무 적음(-4보다 작음)을 나타냅니다.

pOints()

points() 함수는 지정된 좌표에 일련의 점을 그리는 일반적인 함수입니다. 지정된 문자가 좌표에 가운데에 표시됩니다.

```
points(x, y=NULL, type="p", ...)
```

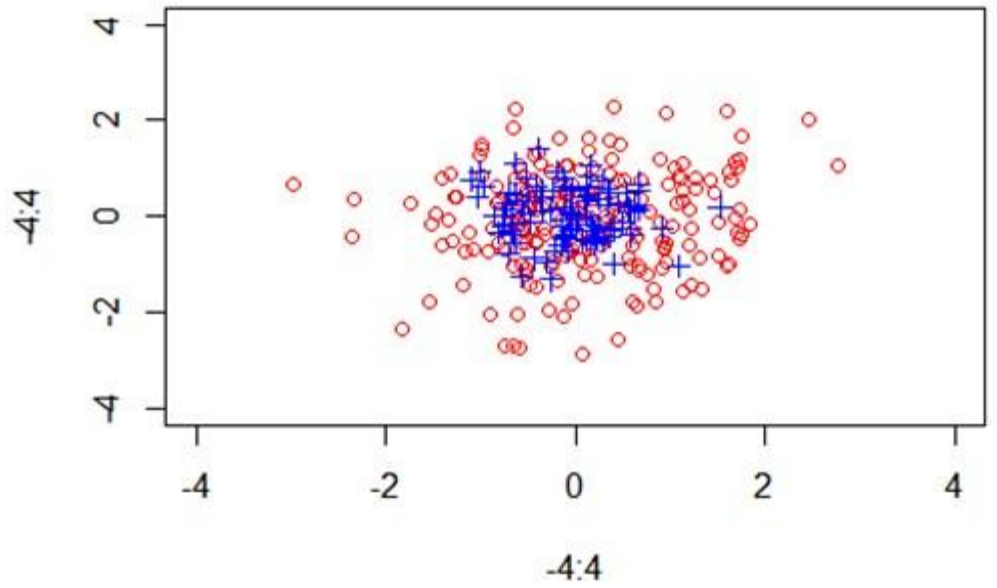
구문에서...

- x, y : 그림을 그릴 점의 좌표 벡터입니다.
- type : 어떤 유형의 그래프가 그려질지를 지정합니다. 기본값은 p입니다. c(b에서 p를 뺀 그래프), o(overplotted; 점과 선이 중첩된 그래프), h(histogram; 히스토그램 그래프), s(stair steps; 계단 그래프), S(stair steps; 거꾸로 그린 계단 그래프), n(그래프 표시 없음) 등이 있습니다.

points() 예

다음 코드는 x축과 y축이 각각 -4~4까지인 빈 그래프를 그린 후 점들을 그리는 예입니다.

```
> plot(-4:4, -4:4, type="n")
> points(rnorm(200), rnorm(200), col="red")
> points(rnorm(100)/2, rnorm(100)/2, col="blue", pch=3)
```



lines()

lines() 함수는 좌표를 다양한 방법으로 가져 와서 해당 점을 선분으로 결합합니다.

```
lines(x, y=NULL, type="l", ...)
```

구문에서...

- x, y : 그림을 그릴 선의 좌표 벡터입니다.
- type : 선의 타입입니다. 숫자 또는 문자로 설정 가능합니다. 숫자는 0부터 6까지 사용할 수 있으며(0=blank, 1=solid(default), 2=dashed, 3=dotted, 4=dotdash, 5=longdash, 6=twodash) 숫자 외에 문자("blank", "solid", "dashed", "dotted", "dotdash", "longdash", "twodash"")로 지정할 수 있습니다.

lines() 예

다음 코드는 cars 데이터셋을 이용하여 산점도를 그린 후 그 위에 가중 선형 회귀곡선을 그립니다.

```
> plot(cars, main="Stopping Distance versus Speed")
> lines(stats::lowess(cars))
```

- lowess()[39] 함수는 이변량 자료를 작은 윈도우로 나누어 각각의 구간에서 가중 선형 회귀를 하여 곡선을 구합니다. lowess(y~x, f) 형식으로 지정하며, f(smoother span, 평활기너비)의 디폴트는 2/3이고, f가 크면 회귀함수가 직선에 가까운 밋밋한 곡선이 되고, f가 작으면 회귀함수가 흰 정도가 큰 곡선이 됩니다. 너무 밋밋한 곡선이 되면 과소적합(under-fitting), 흰 정도가 큰 곡선이 되면 과다적합((over-fitting)이라 합니다.

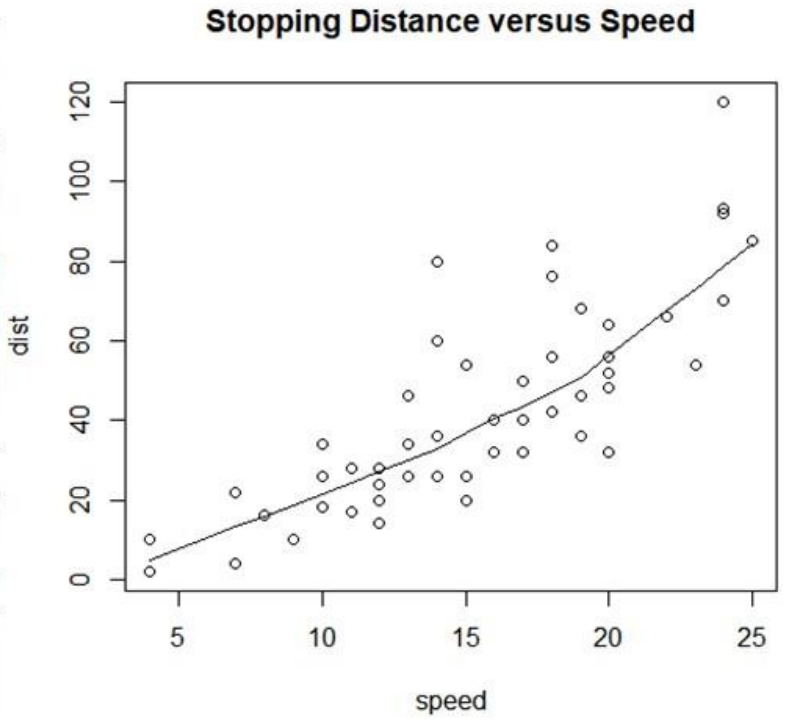


그림 16. lines() 예

abline()

abline()은 기존 그래프 위에 직선을 추가하는 함수입니다.

```
abline(a=NULL, b=NULL, h=NULL, v=NULL, reg=NULL, coef=NULL,
       ...)
```

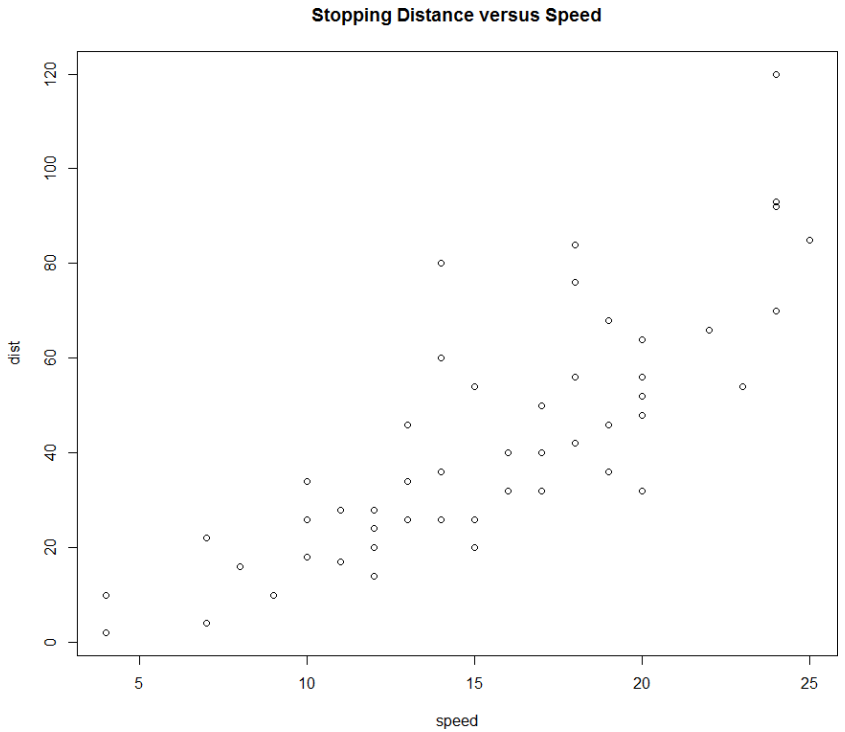
구문에서...

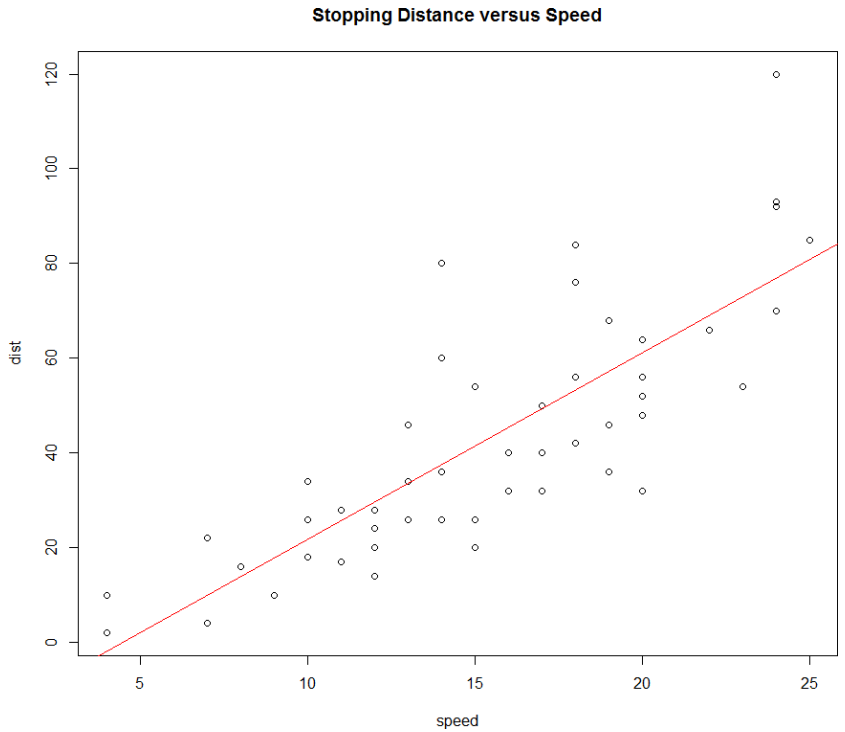
- a, b : 단일 값이어야 하며, 절편(intercept)과 기울기(slope)입니다.
- h : 수평선을 그리기 위한 y 값입니다.
- v : 수직선을 그리기 위한 x 값입니다.
- coef : 절편과 기울기를 주는 길이 2의 벡터입니다.
- reg : coef 메소드가 있는 객체입니다.

abline()

8. 시각화

- 기존 그래프 위에 직선을 추가하는 함수
- z <- lm(dist ~ speed, data=cars)
- plot(cars, main="Stopping Distance versus Speed")
- abline(z, col="red")

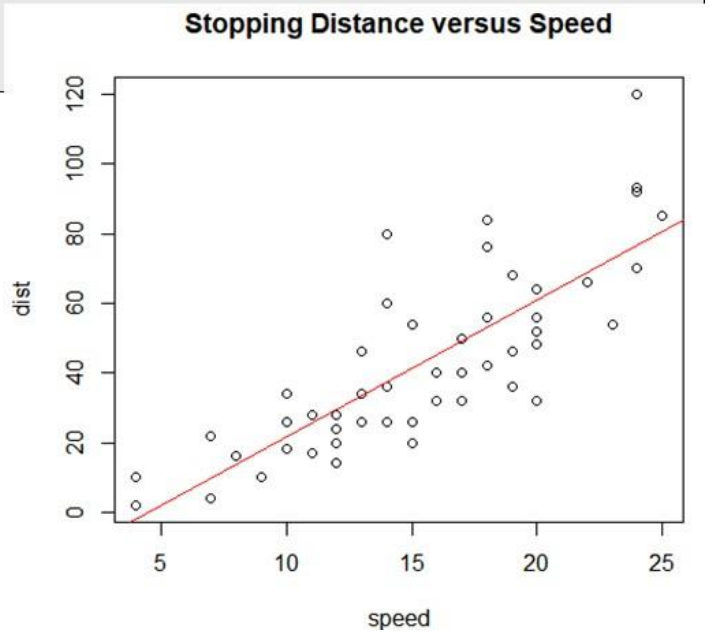




abline() 예

다음 코드는 cars 데이터셋을 이용해 산점도를 그린 후 회귀직선을 그리는 예입니다.

```
> plot(cars, main="Stopping Distance versus Speed")
> z <- lm(dist ~ speed, data=cars)
> z

Call:
lm(formula = dist ~ speed, data = cars)

Coefficients:
(Intercept)        speed
    -17.579        3.932

> abline(z, col="red")
```



- lm()함수(Linear Models)는 선형 모델을 계산합니다. 이것은 회귀 분석을 수행하는 데 사용될 수 있습니다. 위의 코드는 산점도의 선형회귀식을 이용하여 직선을 그립니다.

text()


```
text(x, y=NULL, labels=seq_along(x$x), adj=NULL, pos=NULL,
     offset=0.5, vfont=NULL, cex=1, col=NULL, font=NULL,
     ...)
```

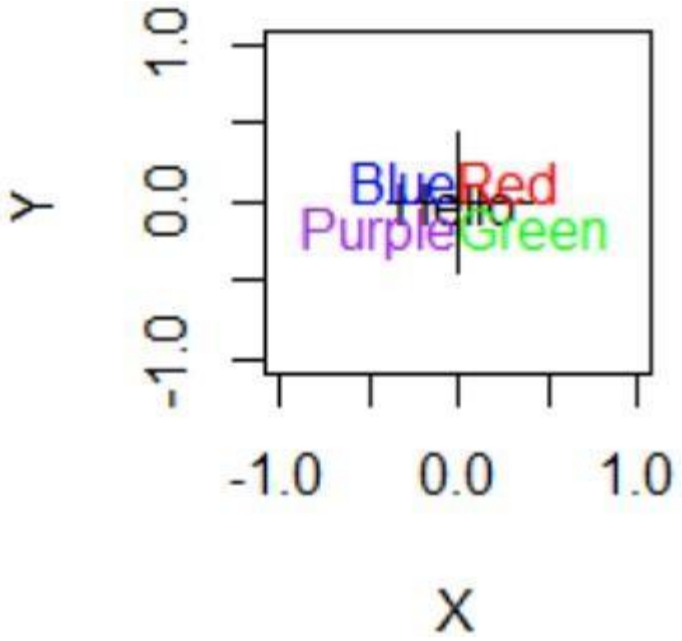
구문에서...

- x, y : 텍스트 레이블을 써야하는 좌표의 숫자 벡터입니다. x와 y의 길이가 다른 경우, 더 짧은 것이 재활용됩니다.
- labels : 기입해지는 텍스트를 지정하는 캐릭터 벡터 또는 수식 표현(expression)입니다. 레이블이 x와 y보다 긴 경우 좌표는 레이블 길이만큼 재활용됩니다. expression을 이용하면 수식을 표현할 수 있습니다.
- adj : 레이블의 x(그리고 선택적으로 y) 위치를 조정을 지정하는 0~1 사이의 하나 또는 두 개의 값입니다. adj=(0,0)이면 텍스트의 왼쪽 아래가 기준 위치가 되도록 합니다. adj=(1,0)이면 텍스트의 오른쪽 아래가 기준 위치가 됩니다. 0.5는 중간입니다.
- pos : 텍스트의 위치 지시자입니다. 이 값을 지정하면 지정된 adj 값을 무시합니다. 값 1, 2, 3 및 4는 각각 지정된 좌표의 아래, 왼쪽, 위 및 오른쪽의 위치를 나타냅니다.
- offset : pos가 지정되면 이 값은 지정된 좌표로부터의 label의 오프셋(offset)을 문자 폭의 분수로 나타냅니다.
- vfont : 현재 글꼴 군의 경우 NULL이거나 허시(hershey)[40] 벡터 글꼴의 경우 길이가 2 인 문자 벡터입니다. 벡터의 첫 번째 요소는 서체를 선택하고 두 번째 요소는 스타일을 선택합니다. 레이블이 표현식이면 무시됩니다.
- cex : 숫자 문자 확장 요소입니다. par("cex")를 곱하면 최종 문자 크기가 됩니다. NULL 및 NA는 1.0과 같습니다.
- col, font : 색상과 글꼴(vfont=NULL 인 경우)을 사용합니다. par()에 있는 전역 그래픽 매개 변수 값의 기본값입니다.

text() 예

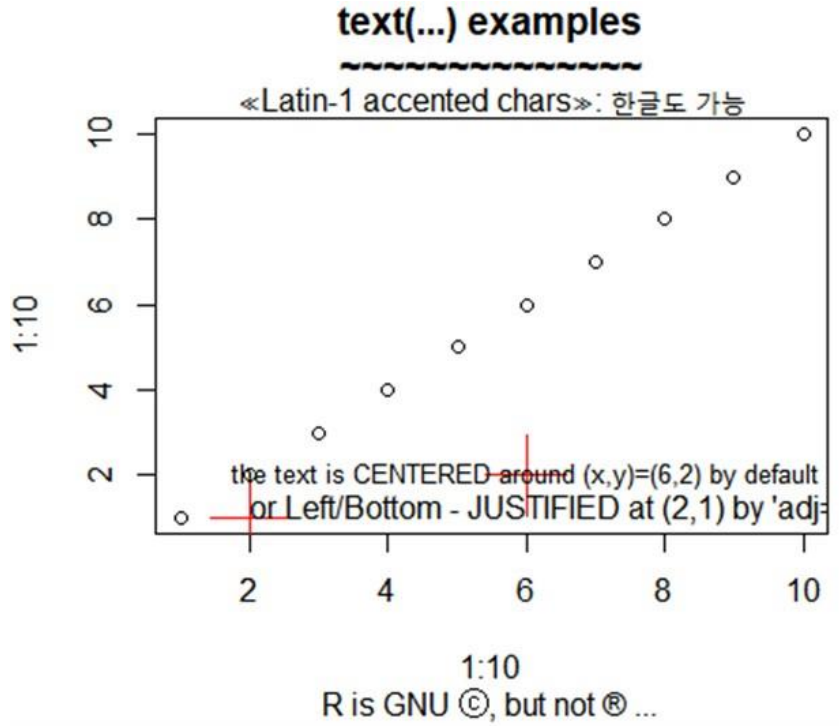
다음 코드는 adj 파라미터를 테스트하기 위한 예입니다.

```
> plot(-1:1, -1:1, type="n", xlab="X", ylab="Y")
> points(0,0, cex=4, pch=3)
> text(0,0, "Hello")
> text(0,0, "Red", adj=c(0,0), col="red")
> text(0,0, "Blue", adj=c(1,0), col="blue")
> text(0,0, "Green", adj=c(0,1), col="Green")
> text(0,0, "Purple", adj=c(1,1), col="purple")
```



text() 예


```
> plot(1:10, 1:10, main="text(...) examples\n~~~~~~~~~~~~~~~",
+      sub="R is GNU ©, but not ® ...")
> mtext("«Latin-1 accented chars»: 한글도 가능", side=3)
> points(c(6,2), c(2,1), pch=3, cex=4, col="red")
> text(6, 2, "the text is CENTERED around (x,y)=(6,2) by default", cex=.8)
> text(2, 1, "or Left/Bottom - JUSTIFIED at (2,1) by 'adj=c(0,0)'", adj=c(0,0))
```

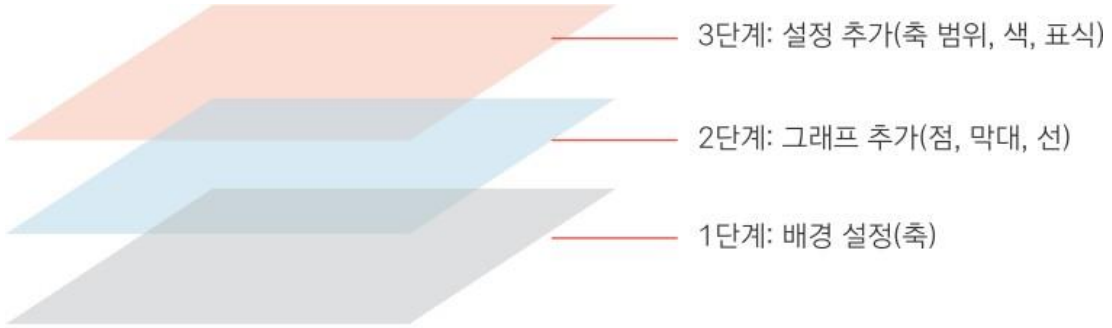


ggplot2

- ggplot2
  - Hadley Wichham 교수가 만든 데이터를 이해하는데 좋은 시각화 도구로 데이터와 시각화 요구를 객체화 시켜 시각화를 구현

- ggplot2이 주요 시각화 함수
  - Qplot() : 손쉽고 빠르게 시각화하기 위한 도구
  - ggplot() : 데이터와 시각화 요소를 분리하여 시각화하는 도구

- ggplot2의 시각화 레이어 구조 이해하기



ggplot2 레이어 구조

ggplot2

- 기하객체 (Geometric object, geom)
  - geom_point() : 산점도
  - geom_smooth() : 회귀선
  - geom_line() : 선 그래프 (방향 : 왼쪽→ 오른쪽)
  - geom_histogram() : 히스토그램
  - geom_density() : 밀도 그래프
  - geom_bar() : 막대 그래프, 도수분포표
  - geom_pointrange() : 값과 범위 표시
  - geom_hline() : 가로선

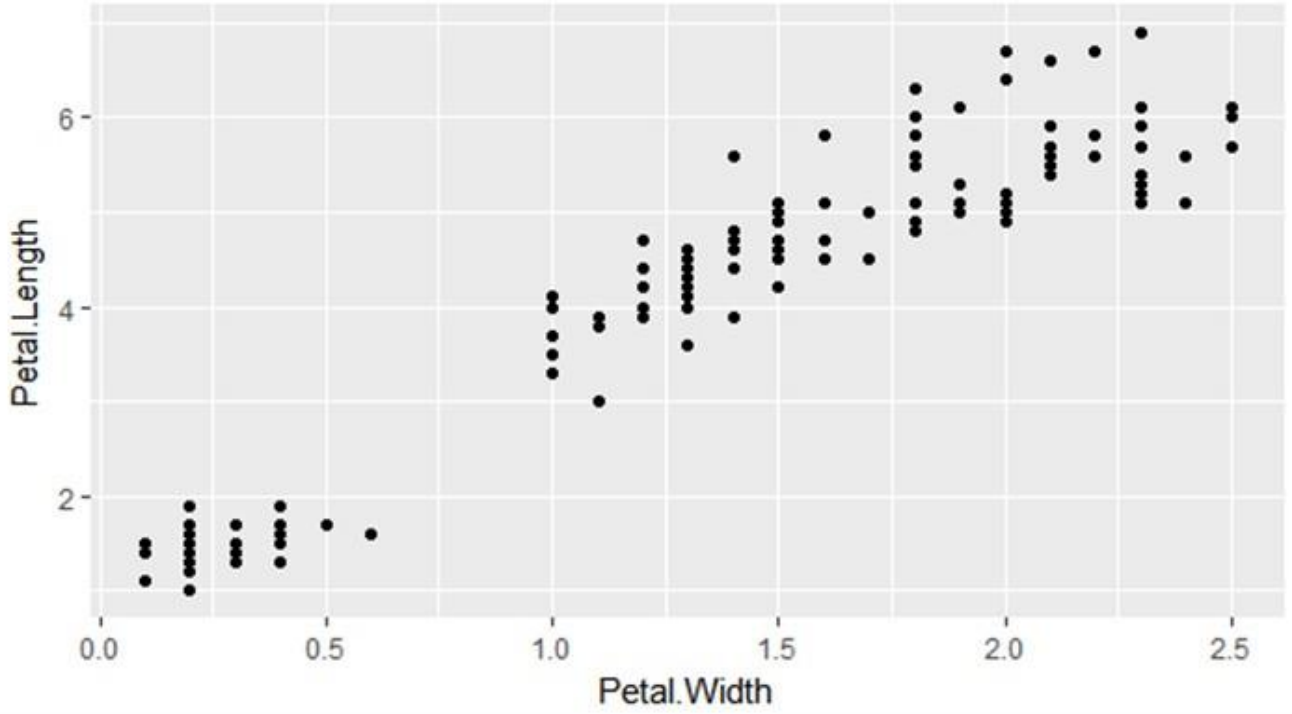
- 미적 속성 (Aesthetic attributes, aes)
  - x, y : x, y 좌표의 값
  - color : 색상 분류 기준 (선/점 색)
  - shape : 점의 모양 분류 기준, NA. 표시하지 않음
  - size : 점의 크기, 선의 굵기 (1. default)
  - alpha : 투명도, 작을수록 투명함
  - fill : 색상 분류 기준 (채워 넣는 색)

- 미적 속성은 적용하는 기하객체로 상속됩니다.

x, y 축 설정

다음 코드는 iris 데이터를 이용해 Petal.Width와 Petal.Length의 산점도를 그립니다. ggplot() 함수의 aes()로 x와 y축 데이터를 지정했습니다.

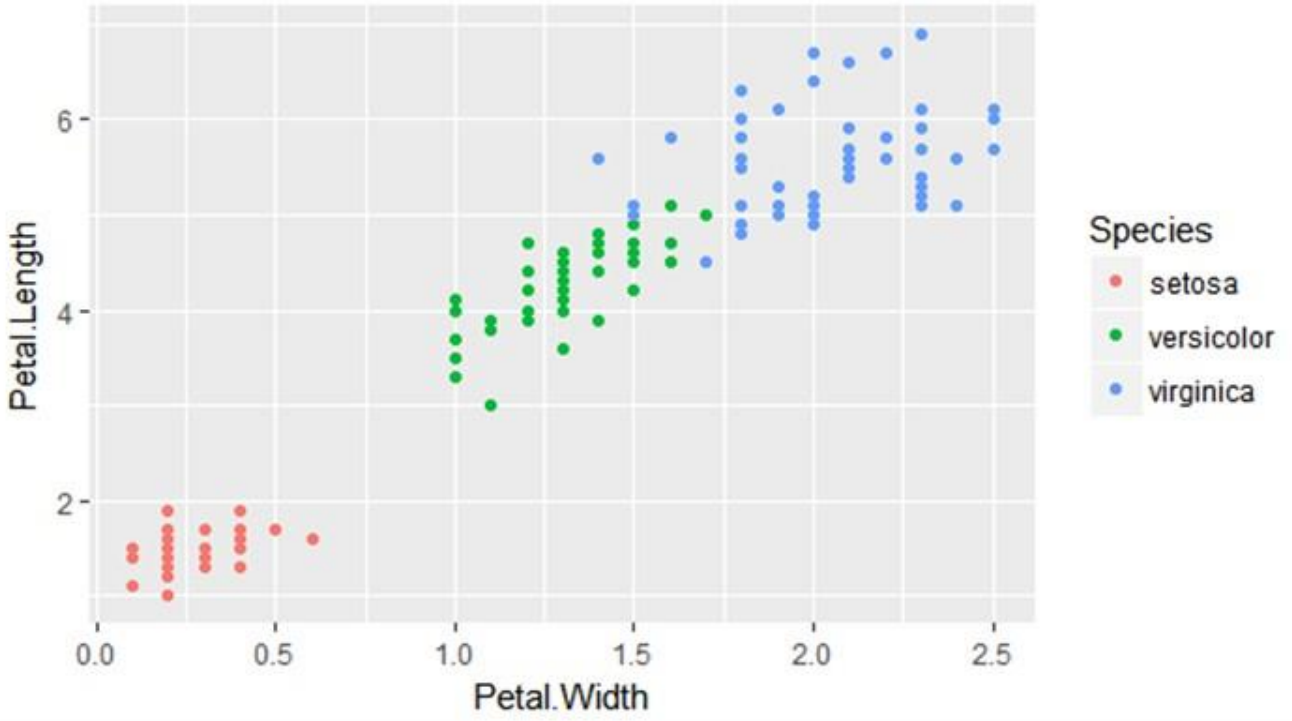
```
> ggplot(iris, aes(x=Petal.Width, y=Petal.Length)) +
+   geom_point()
```



데이터를 이용한 미적속성 설정

geom_point() 함수에서 aes()로 산점도의 색상을 Species 열을 이용해 지정했습니다. ggplot2는 aes() 함수로 미적 속성을 지정할 때 데이터의 변수를 이용해 지정할 수 있습니다.

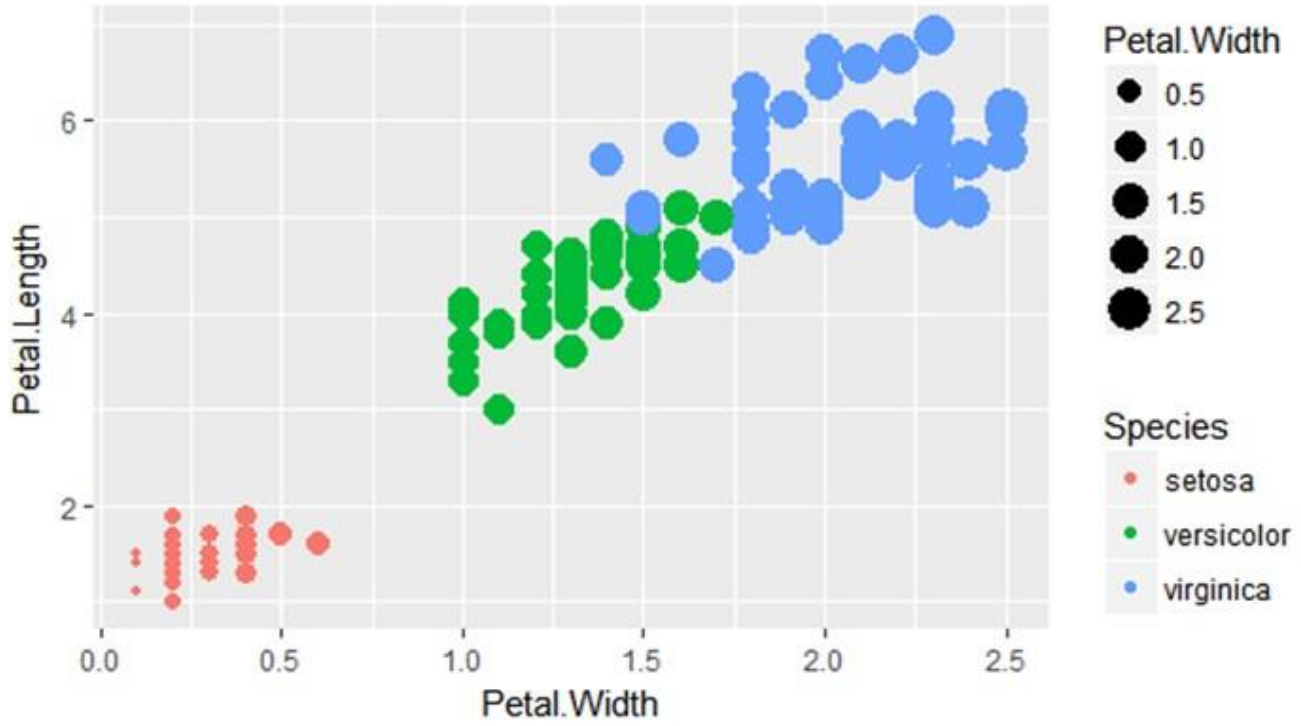
```
> ggplot(iris, aes(x=Petal.Width, y=Petal.Length)) +
+   geom_point(aes(color=Species))
```



데이터를 이용한 미적속성 설정

다음 코드는 점의 크기를 Petal.Width를 이용해 그립니다. 이렇게 하면 Petal.Width 값을 좀 더 잘 표현할 수 있습니다.

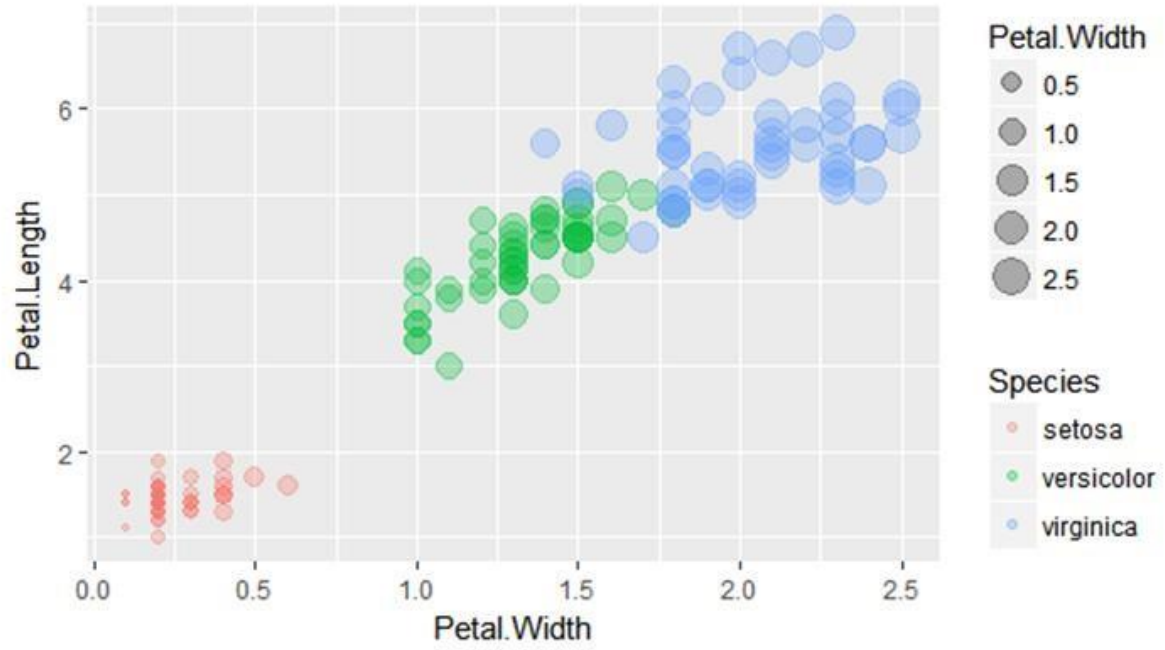
```
> ggplot(iris, aes(x=Petal.Width, y=Petal.Length)) +
+   geom_point(aes(color=Species, size=Petal.Width))
```



알파값 지정

위의 예처럼 점들이 같은 위치 또는 비슷한 위치에 있을 경우 분포를 확인하기 어려운 단점이 있습니다. 다음 코드는 그것을 해결하기 위해 alpha값을 지정한 예입니다. alpha 값은 투명도를 지정하며 0(투명)~1(불투명)까지 값을 가질 수 있습니다. 이렇게 하면 같은 위치에 점들이 표시될 때 더 진하게 표시되므로 데이터의 분포를 더 잘 확인할 수 있습니다.

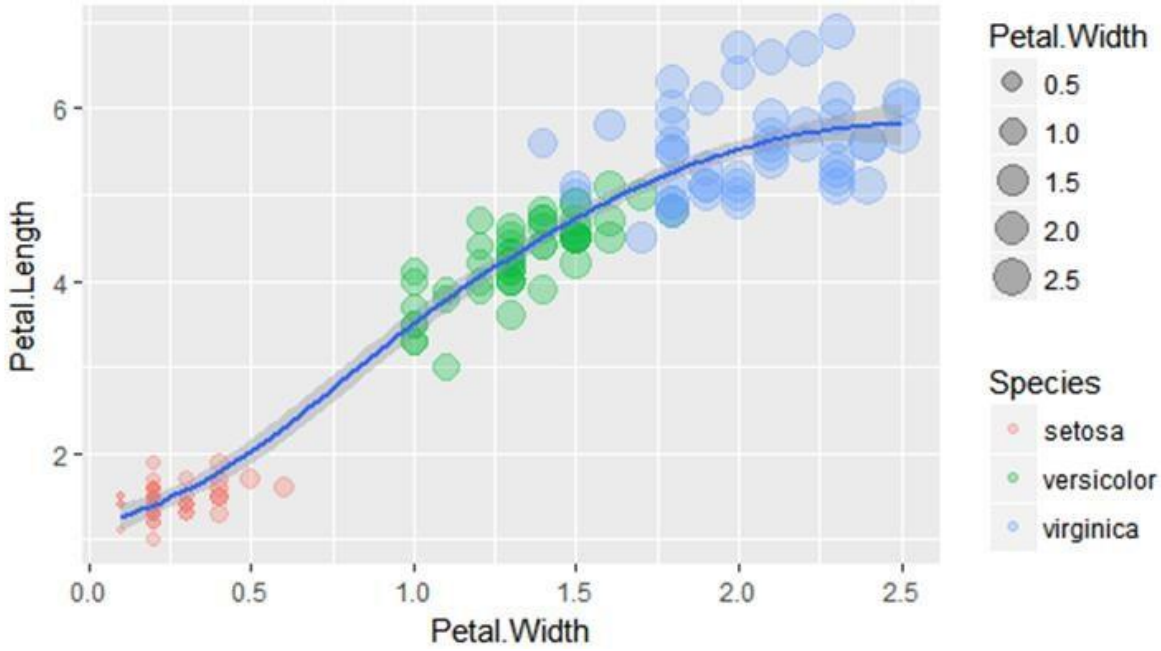
```
> ggplot(iris, aes(x=Petal.Width, y=Petal.Length)) +
+   geom_point(aes(color=Species, size=Petal.Width), alpha=0.3)
```



추세선 추가

다음 코드는 추세선[68]을 산점도와 함께 표현한 그래프입니다. 추세선을 직선으로 그리고 싶다면 method="lm"으로 설정하면 됩니다. 추세선에 신뢰 구간을 표시하지 않으려면 geom_smooth 함수의 se 파라미터를 FALSE로 설정하면 됩니다.

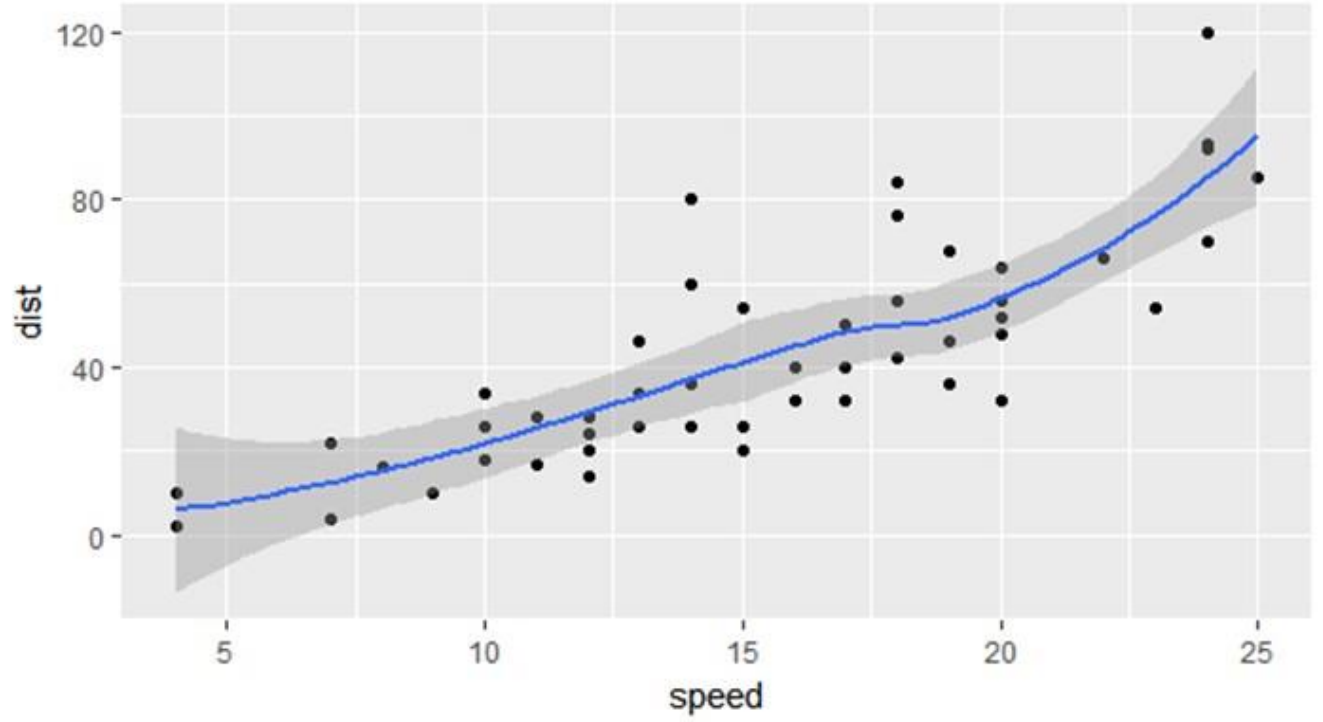
```
> ggplot(iris, aes(x=Petal.Width, y=Petal.Length)) +
+   geom_point(aes(color=Species, size=Petal.Width), alpha=0.3) +
+   geom_smooth(method="loess", color="blue")
```



신뢰구간

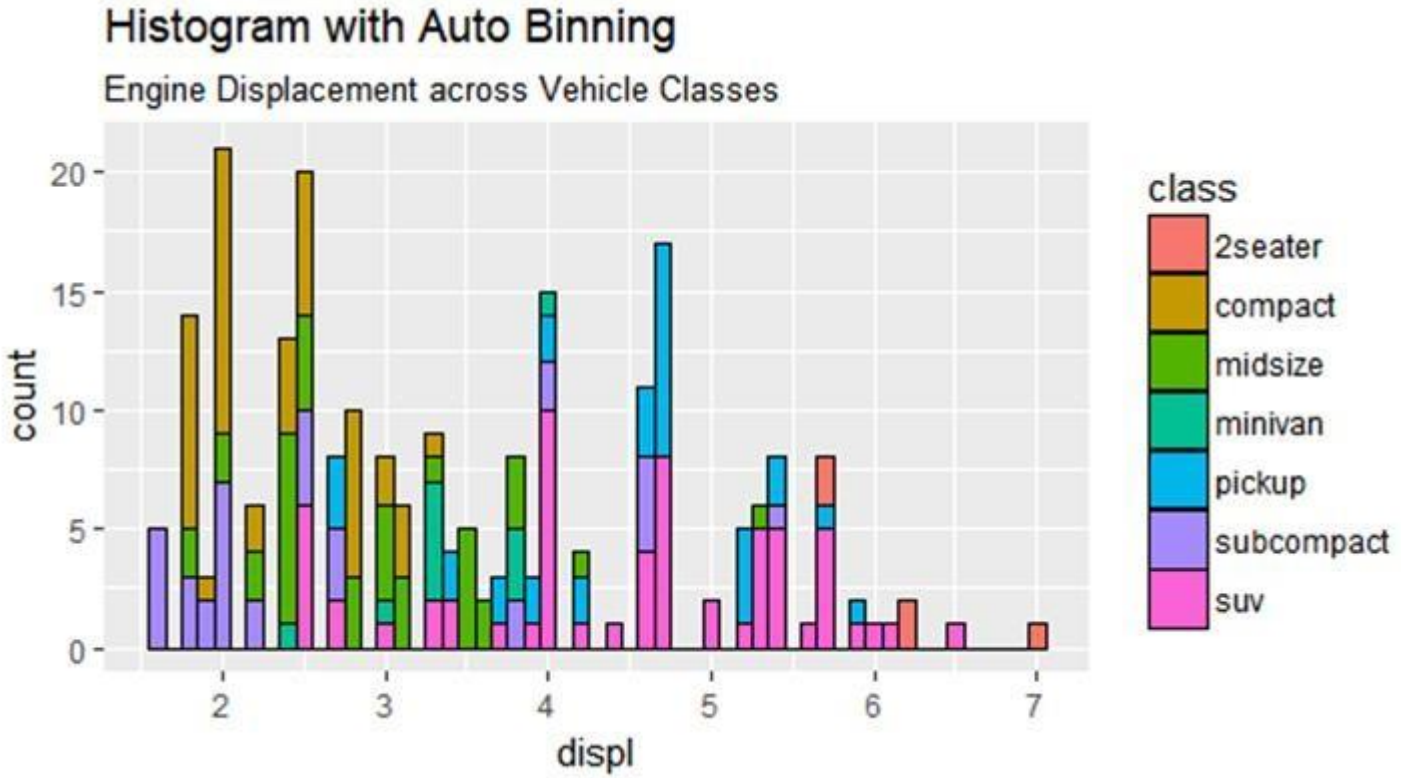
다음은 cars 데이터셋을 이용해 산점도, 추세선, 신뢰구간을 그리는 예입니다.

```
> ggplot(cars, aes(speed, dist)) +
+   geom_point() +
+   geom_smooth(method="loess")
```



히스토그램

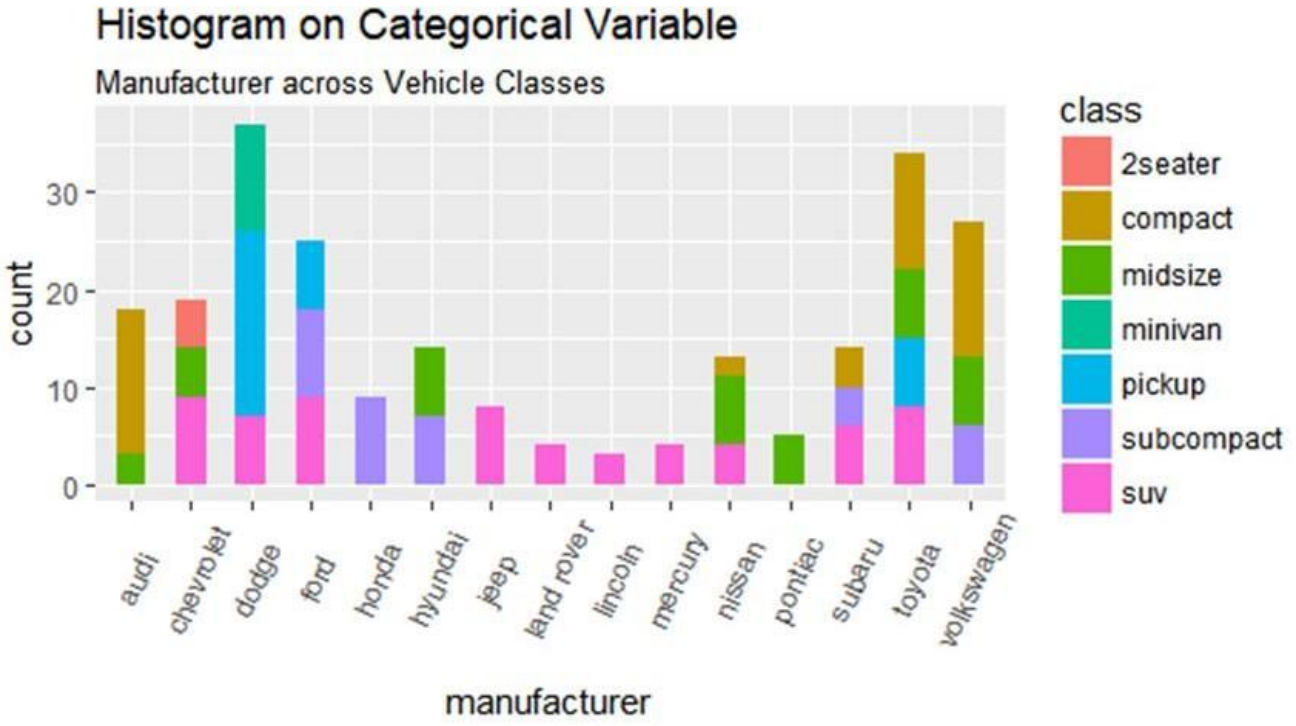

```
> g1 <- ggplot(mpg, aes(displ))
> g1 + geom_histogram(aes(fill=class),
+                     binwidth=.1,
+                     col="black", size=.1) +
+      labs(title="Histogram with Auto Binning",
+           subtitle="Engine Displacement across Vehicle Classes")
```



막대그래프

8. 시각화

```
> g3 <- ggplot(mpg, aes(manufacturer))
> g3 + geom_bar(aes(fill=class), width = 0.5) +
+      theme(axis.text.x = element_text(angle=65, vjust=0.6)) +
+      labs(title="Histogram on Categorical Variable",
+           subtitle="Manufacturer across Vehicle Classes")
```



밀도그래프

8. 시각화

```
> g4 <- ggplot(mpg, aes(cty))
> g4 + geom_density(aes(fill=factor(cyl)), alpha=0.8) +
+      labs(title="Density plot",
+           subtitle="City Mileage Grouped by Number of cylinders",
+           caption="Source: mpg",
+           x="City Mileage",
+           fill="# Cylinders")
```

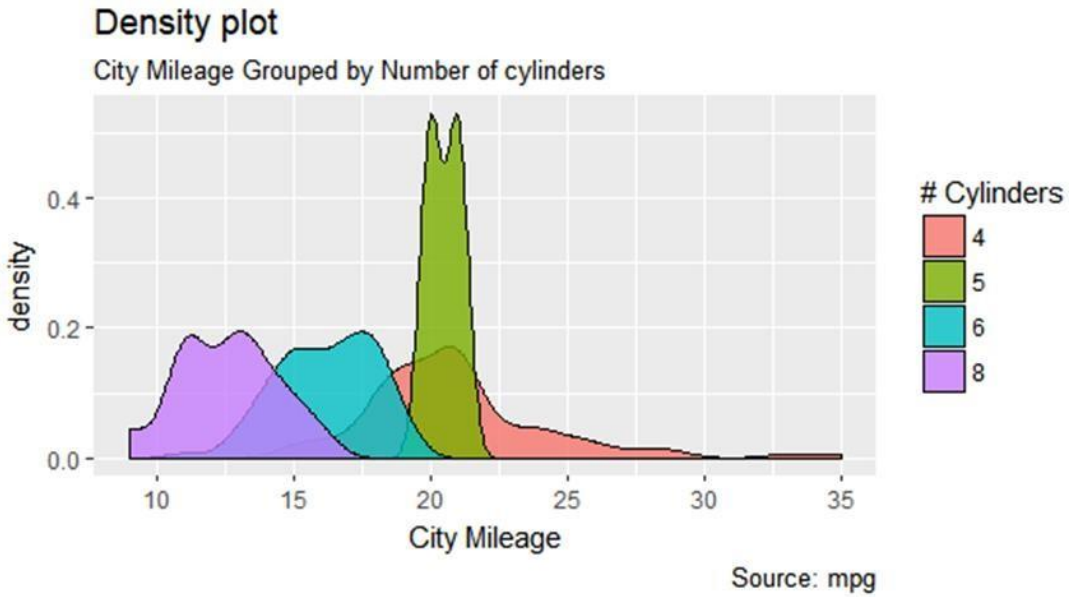
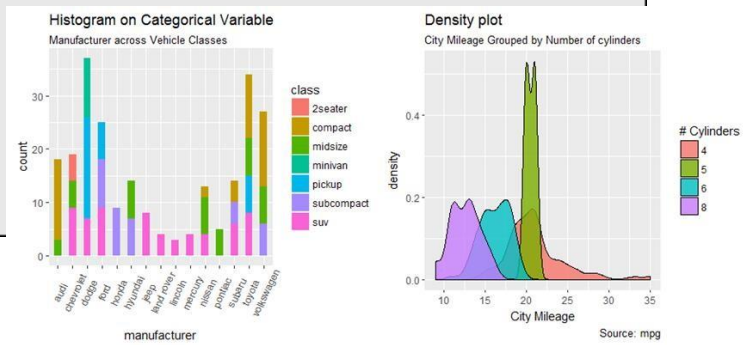


차트 분할 출력

8. 시각화

```
> install.packages("gridExtra")
> library(gridExtra)
```

```
> g3 <- ggplot(mpg, aes(manufacturer))
> g3 <- g3 + geom_bar(aes(fill=class), width = 0.5) +
+            theme(axis.text.x = element_text(angle=65, vjust=0.6)) +
+            labs(title="Histogram on Categorical Variable",
+                 subtitle="Manufacturer across Vehicle Classes")
> g4 <- ggplot(mpg, aes(cty))
> g4 <- g4 + geom_density(aes(fill=factor(cyl)), alpha=0.8) +
+            labs(title="Density plot",
+                 subtitle="City Mileage Grouped by Number of cylinders",
+                 caption="Source: mpg",
+                 x="City Mileage",
+                 fill="# Cylinders")
> grid.arrange(g3, g4, ncol=2)
```

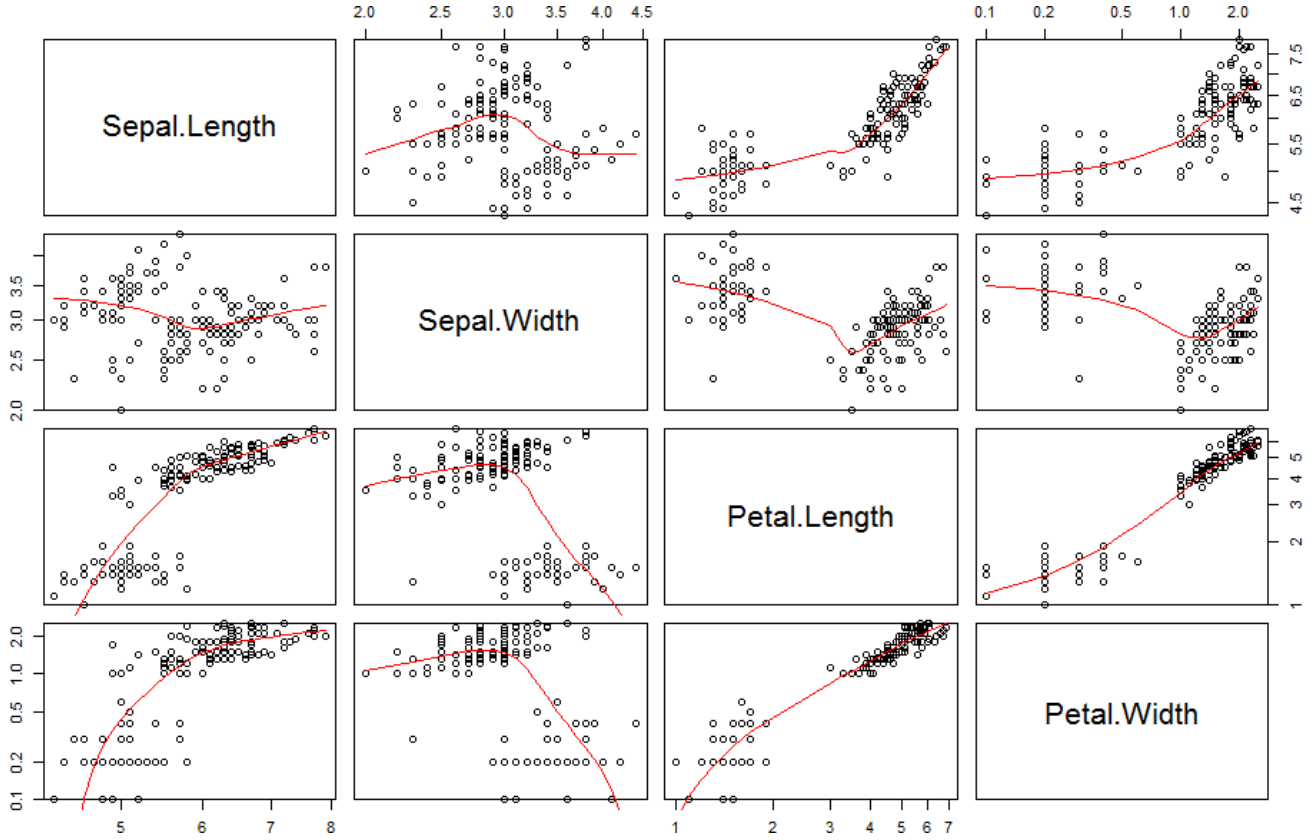


산점도 행렬(pairs)

8. 시각화

- 점 그래프로 주로 시간과 관련되지 않은 데이터를 그래프로 표현

- pairs(iris[-5],panel=panel.smooth)
- plot(iris[-5])



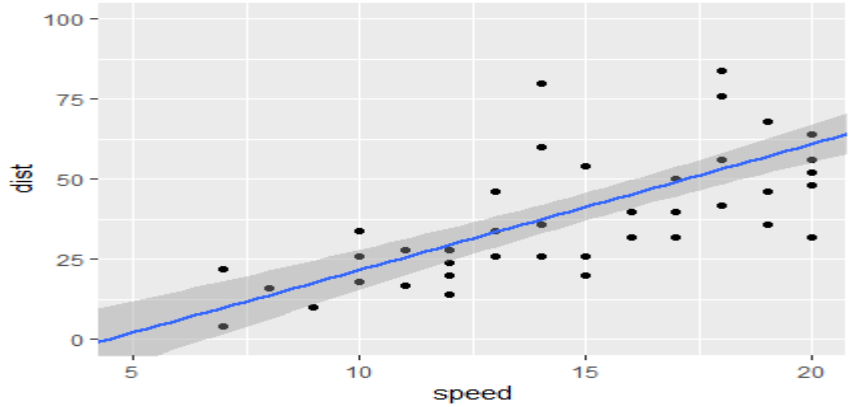
문제 1~10번까지 해결

본인이름.R
ex01.png ~ ex10.png
본인이름.html(마크다운으로 생성된 html)
본인이름.Rmd

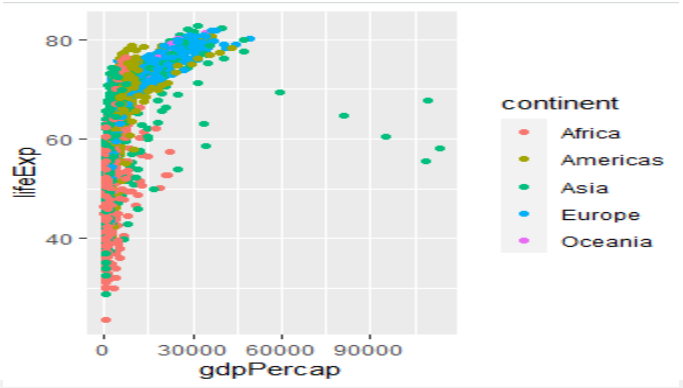
위의 13개 파일을 압축해서 제출합니다

연습문제 8. 시각화

1. datasets::cars데이터 셋을 이용하여 속도에 대한 제동거리의 산점도와 적합도(신뢰구간 그래프)를 나타내시오(단, 속도가 5부터 20까지 제동거리 0부터 100까지만 그래프에 나타냄).

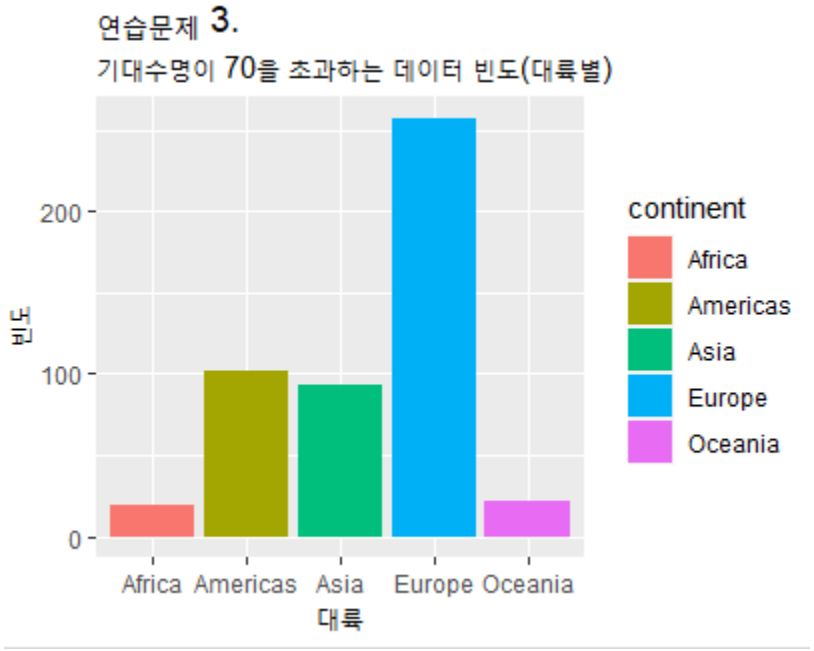


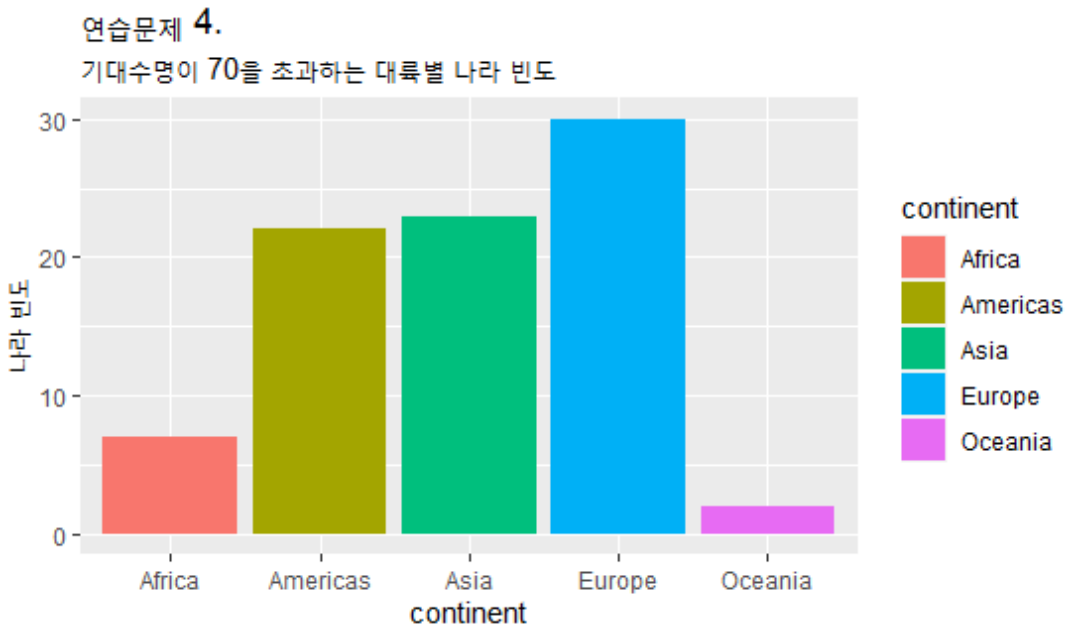
2. gapminder::gapminder 데이터 셋을 이용하여 1인당국내총생산에 대한 기대수명의 산점도를 대륙별 다른 색으로 나타내시오.



연습문제 8. 시각화

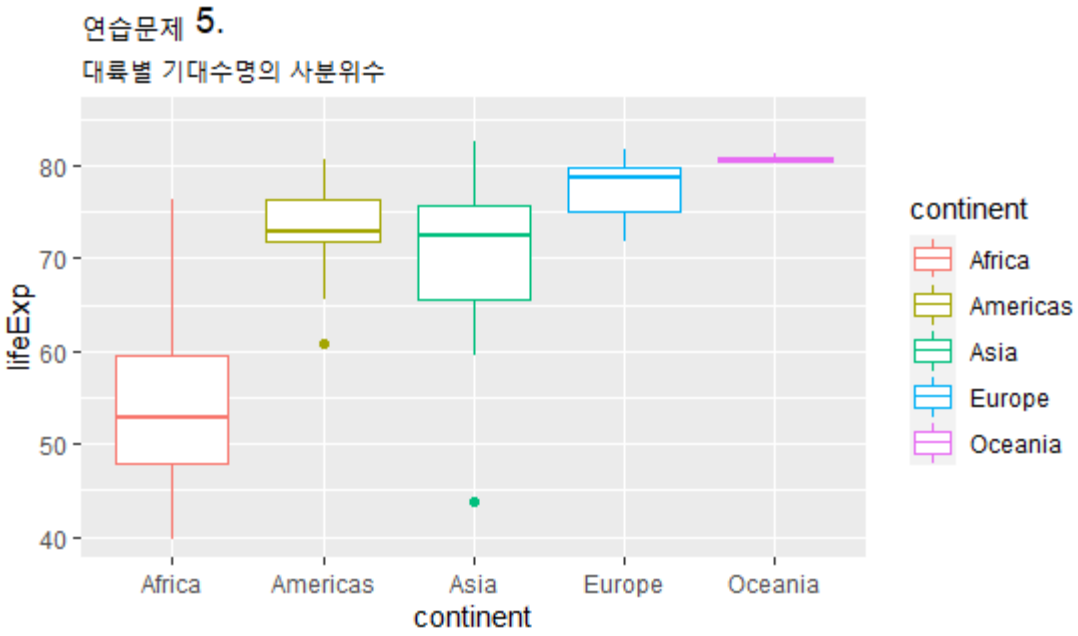
3. gapminder::gapminder 데이터 셋을 이용하여 개대 수명이 70을 초과하는 데이터에 대해 대륙별 데이터 갯수
4. gapminder::gapminder 데이터 셋을 이용하여 기대수명이 70을 초과하는 데이터에 대해 대륙별 나라 갯수.

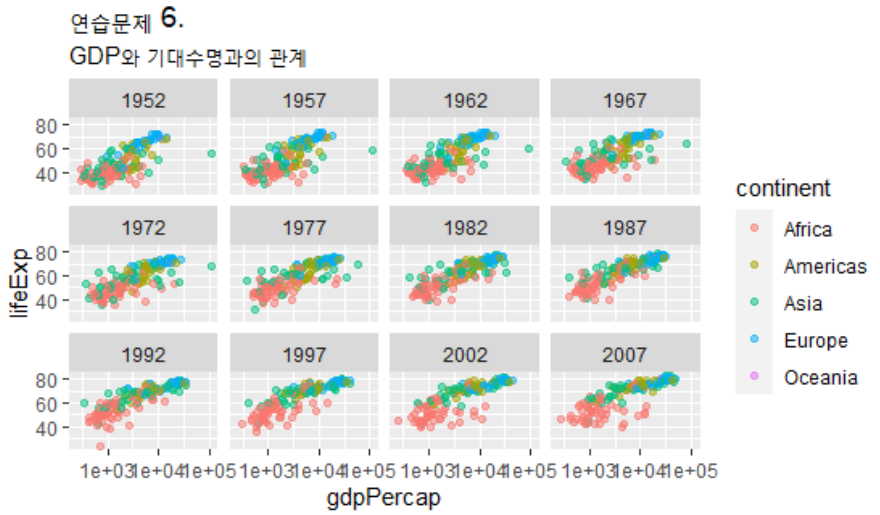




연습문제 8. 시각화

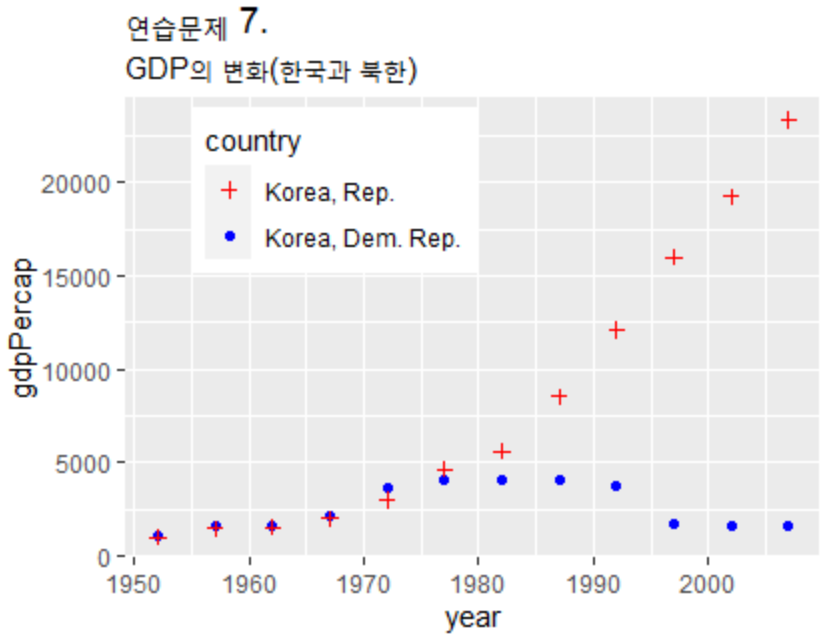
5. gapminder::gapminder 데이터 셋을 이용하여 대륙별 기대수명의 사분위수를 시각화
6. gapminder::gapminder 데이터 셋을 이용하여 년도별로 gdp와 기대수명과의 관계를 산점도를 그리고 대륙별 점의 색상을 달리 시각화

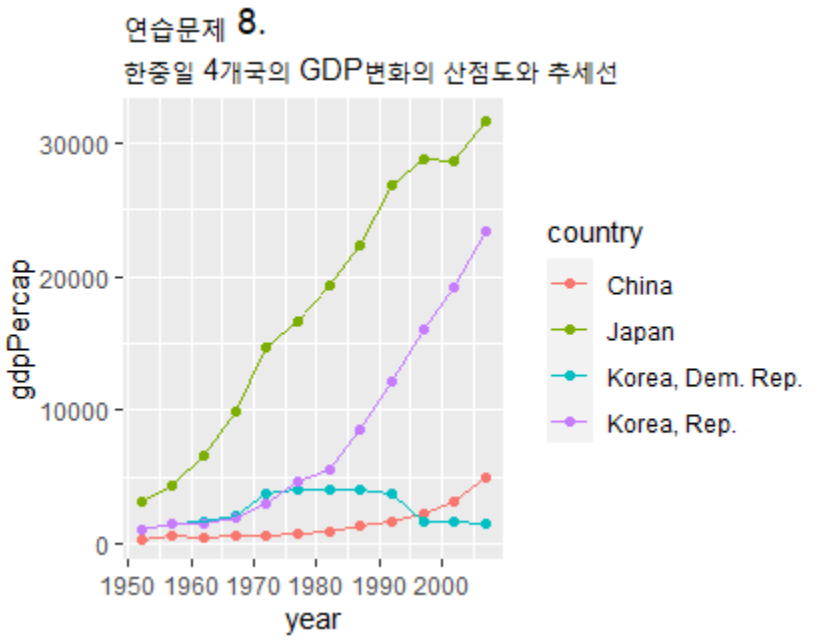




연습문제 8. 시각화

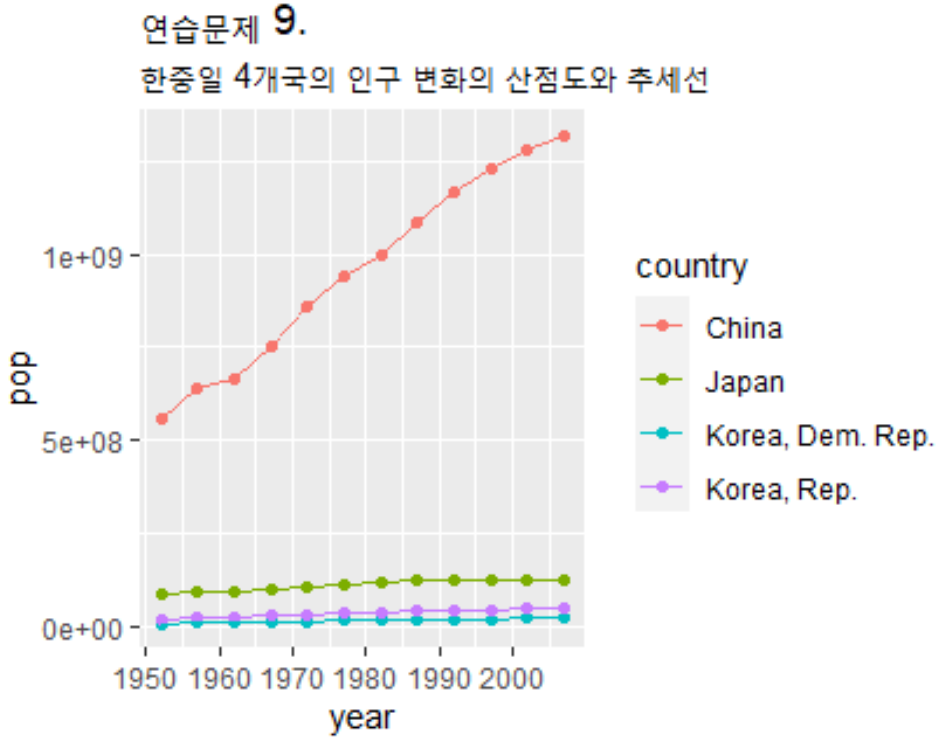
7. gapminder::gapminder 데이터 셋에서 북한과 한국의 년도별 GDP 변화를 산점도로 시각화하시오(북한:Korea, Dem. Rep. 한국:Korea, Rep. substr(str, start, end)함수 이용)
8. gapminder::gapminder 데이터 셋을 이용하여 한중일 4개국별 GDP 변화를 산점도와 추세선으로 시각화 하시오.

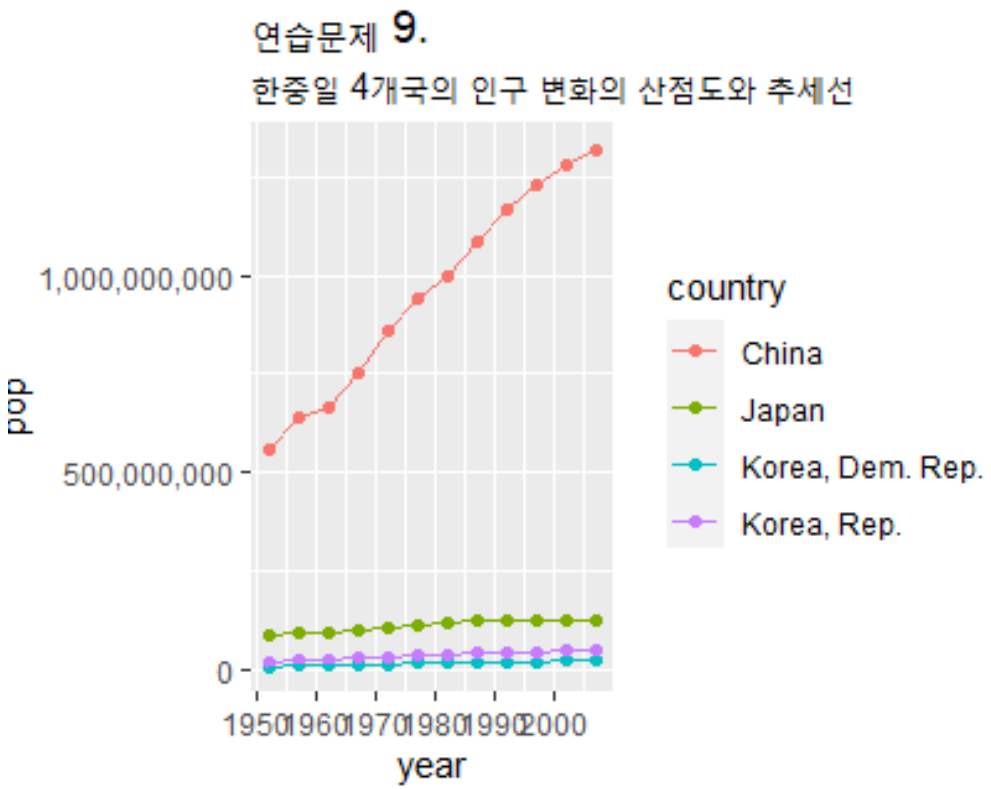




연습문제 8. 시각화

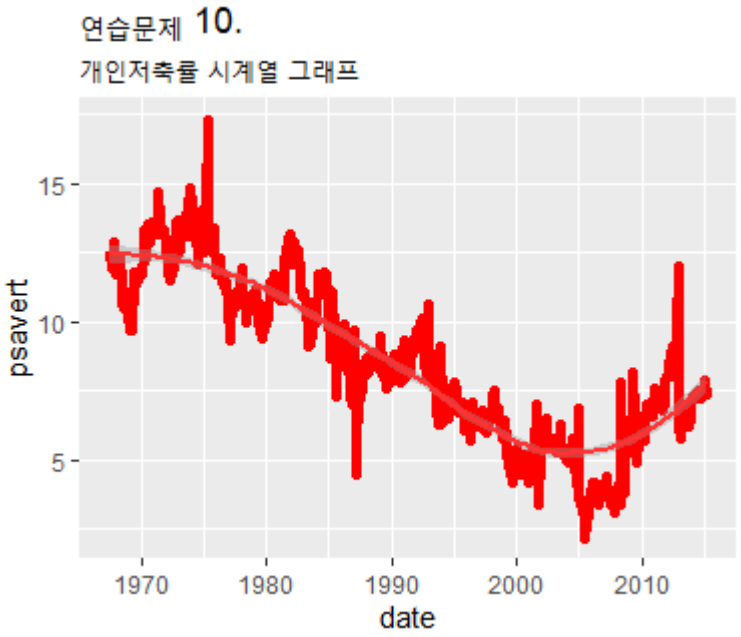
9. gapminder::gapminder 데이터 셋에서 한중일 4개국별 인구변화 변화를 산점도와 추세선으로 시각화 하시오(scale_y_continuous(labels = scales::comma)추가시 우측처럼)

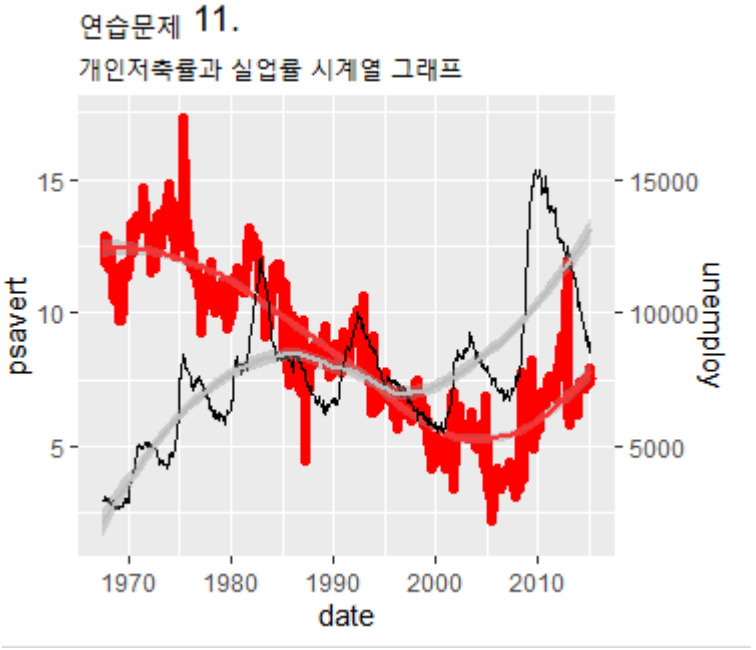




연습문제 8. 시각화

10. Ggplot2::economic 데이터 셋의 개인 저축률(psavert)가 시간에 따라 어떻게 변해 왔는지 알아보려 한다. 시간에 따른 개인 저축률의 변화를 나타낸 시계열 그래프와 추세선을 시각화하시오
11. Ggplot2::economic 데이터 셋의 개인 저축률(psavert)과 실업의이 시간에 따라 서로 어떻게 변해 왔는지 알아보려 한다. 시간에 따른 개인 저축률과 실업률의 변화를 한 그래프에 중첩하여 시각화하시오





RColorBrewer 패키지를 설치하면 사용 가능한 칼라 팔렛트